

7-26-2016

# Categories of Conceptions of Proofs by Students of Computer Science

Thérèse Smith

*University of Connecticut - Storrs*, [therese.smith@uconn.edu](mailto:therese.smith@uconn.edu)

Follow this and additional works at: <https://opencommons.uconn.edu/dissertations>

---

## Recommended Citation

Smith, Thérèse, "Categories of Conceptions of Proofs by Students of Computer Science" (2016). *Doctoral Dissertations*. 1188.  
<https://opencommons.uconn.edu/dissertations/1188>

# Categories of Conceptions of Proofs by Students of Computer Science

Thérèse Mary Smith, Ph.D.

University of Connecticut, 2016

## ABSTRACT

Guided by constructivism, which posits that students assimilate new knowledge into what has made sense to them previously, we researched student conceptualizations of proof. We used the qualitative research methods of thematic analysis and phenomenography to learn and categorize student conceptualizations of proof, and of mathematization more generally. Our published work exhibited an explanatory connection between publications of others in the mathematics education community and the computer science education community. The connection we found is lack of understanding of proof by mathematic induction as an argument, as occurs among students of mathematics and of computer science, explains lack of understanding of how recursive algorithms work. We used these phenomenographic categories to intuit ideas whose emphasis might be helpful for students' development of deeper understanding of proof.

# Categories of Conceptions of Proofs by Students of Computer Science

Thérèse Mary Smith

M.S. Computer Science and Engineering, University of Connecticut, Storrs,  
Connecticut, USA, 2014

M.S. Biomedical Engineering, Iowa State University of Science and Technology,  
Ames, Iowa, USA, 1982

S.B. Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts,  
USA, 1975

A Dissertation

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

at the

University of Connecticut

2016

Copyright by

Thérèse Mary Smith

2016

## APPROVAL PAGE

Doctor of Philosophy Dissertation

# Categories of Conceptions of Proofs by Students of Computer Science

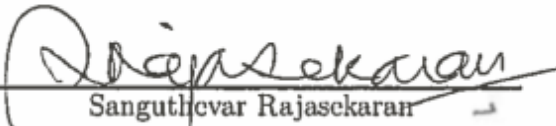
Presented by

Thérèse Mary Smith, S.B. Physics, M.S. BME, M.S. CSE

Major Advisor

  
Robert McCartney

Associate Advisor

  
Sanguthavar Rajasekaran

Associate Advisor

  
Jinbo Bi

University of Connecticut

2016

## ACKNOWLEDGMENTS

Thank you to all of the members of the department, each of whom has helped me to some degree. Your daily example means a great deal. Scholarship, graciousness, patience and compassion are a few of the qualities with which you inspire us, the students. Thank you to Professor Schwartzmann for inviting me to the department and giving me the courage to apply. Thank you to Professor Donald Sheehy for helpful suggestions and exemplary teaching. Thank you to Professor Rajasekaran, especially for helping me with United Technologies. Thank you to Professor Huang for conversations about computational biology. Thank you to Professor Mandoiu for conversations about architectural support for computational biology. Thank you to Professor Peters for conversations about computation biology on IBM BlueGene, and knots in computational biology. Thank you to Professors Gokhale and Ammar for shared papers, support and kindness. Thank you to Professors Russell and Michel for inspirational research and teaching. Thank you to Rebecca Randazzo and Joy Billion for exemplary competence and unfailing assistance. Thank you to Professor McCartney for support, example, and astonishing patience. Thank you to Jack, *semper providente prospectu*.

# Contents

|   |           |
|---|-----------|
| <b>Ch. 1. Introduction</b>  | <b>1</b>  |
| 1.1 Research Questions . . . . .                                      | 3         |
| 1.2 Qualitative Research . . . . .                                    | 5         |
| 1.3 Phenomenography with Variation Theory . . . . .                   | 6         |
| 1.3.1 Conceptualizations . . . . .                                    | 8         |
| 1.3.2 Relevance Structure . . . . .                                   | 9         |
| 1.4 Phenomenography / Variation Theory for these Research Questions . | 12        |
| 1.5 Overview . . . . .  | 12        |
| <b>Ch. 2. Research Perspective and Epistemological Framework</b>      | <b>13</b> |
| 2.1 Phenomenography and Variation Theory . . . . .                    | 16        |
| 2.1.1 Variation Theory . . . . .                                      | 17        |
| 2.1.2 Application to Proof in Computer Science . . . . .              | 19        |
| 2.1.3 An example of applying variation theory . . . . .               | 20        |
| 2.1.4 Variation Theory and Conjunctions . . . . .                     | 21        |
| 2.2 Constructivism . . . . .  | 22        |
| 2.2.1 Piagetian Constructivism . . . . .                              | 23        |
| 2.2.2 Social Constructivism . . . . .                                 | 25        |
| 2.3 Mathematics Education . . . . .                                   | 28        |
| <b>Ch. 3. Methodology</b>   | <b>31</b> |
| 3.1 Design of the Study . . . . .                                     | 32        |
| 3.2 Parts of the Study . . . . .                                      | 33        |
| 3.3 Population Studied . . . . .                                      | 34        |
| 3.3.1 Proofs by Mathematic Induction . . . . .                        | 36        |
| 3.3.2 Purpose of Proof . . . . .                                      | 37        |
| 3.3.3 Proofs Using the Pumping Lemma for Regular Languages . .        | 37        |

|               |  |           |
|---------------|--|-----------|
| 3.3.4         | Student Use of Proof for Applicability of Algorithms . . . . .                                     | 38        |
| 3.4           | Data Collection . . . . .  | 38        |
| 3.4.1         | Interviews . . . . .   | 39        |
| 3.4.2         | Documents . . . . .  | 39        |
| 3.4.3         | Observations from Tutoring and Help Sessions . . . . .   | 40        |
| 3.4.4         | Other Sources . . . . .  | 40        |
| 3.5           | Method of Analysis . . . . .   | 40        |
| 3.5.1         | Analysis of Interviews . . . . .   | 48        |
| 3.5.2         | Analysis of Help Session and Tutoring . . . . .  | 51        |
| 3.5.3         | Example: Application of Phenomenographic Analysis to What<br>Students Think Proof is For . . . . . | 53        |
| 3.6           | Method of Creating Summary Report (for Member Checking) . . . . .                                  | 53        |
| 3.7           | Method of Addressing Validation . . . . .  | 54        |
| 3.8           | Method of Presentation of Results . . . . .  | 54        |
| <b>Ch. 4.</b> | <b>Phenomenographic Analysis</b>   | <b>56</b> |
| 4.1           | Phenomenographic Analysis of What Students Think Proof Is . . . . .                                | 58        |
| 4.1.1         | Categories . . . . .   | 58        |
| 4.1.2         | Illustrative Quotations for Categories . . . . .   | 60        |
| 4.1.3         | Relations . . . . .  | 62        |
| 4.1.4         | Critical Factors . . . . .   | 63        |
| 4.1.5         | Dimensions of Variation . . . . .  | 65        |
| 4.1.6         | Validation . . . . .   | 71        |
| 4.1.7         | Outcome Space . . . . .  | 72        |
| 4.2           | Phenomenographic Analysis of How Students Attempt to Understand<br>Proofs . . . . .                | 76        |
| 4.2.1         | Categories . . . . .   | 76        |
| 4.2.2         | Illustrative Quotations for Categories . . . . .   | 78        |
| 4.2.3         | Relations . . . . .  | 83        |
| 4.2.4         | Critical Factors . . . . .   | 84        |
| 4.2.5         | Dimensions of Variation . . . . .  | 87        |
| 4.2.6         | Validation . . . . .   | 87        |
| 4.2.7         | Outcome Space . . . . .  | 88        |
| 4.3           | Phenomenographic Analysis of What Students Think Proof is For . . . . .                            | 91        |
| 4.3.1         | Categories . . . . .   | 91        |
| 4.3.2         | Illustrative Quotations for Categories . . . . .   | 94        |
| 4.3.3         | Relations . . . . .  | 95        |
| 4.3.4         | Critical Factors . . . . .   | 96        |
| 4.3.5         | Dimensions of Variation . . . . .  | 97        |



|       |  |     |
|-------|--|-----|
| 4.3.6 | Validation . . . . .   | 98  |
| 4.3.7 | Outcome Space . . . . .  | 98  |
| 4.4   | Phenomenographic Analysis of How Students Attempt to Apply Proofs<br>(When Assigned) . . . . .                                       | 100 |
| 4.4.1 | Categories . . . . .   | 100 |
| 4.4.2 | Illustrative Quotations for Categories . . . . .   | 100 |
| 4.4.3 | Relations . . . . .  | 102 |
| 4.4.4 | Critical Factors . . . . .   | 103 |
| 4.4.5 | Dimensions of Variation . . . . .  | 104 |
| 4.4.6 | Validation . . . . .   | 104 |
| 4.4.7 | Outcome Space . . . . .  | 105 |
| 4.5   | Phenomenographic Analysis of How Students Attempt to Apply Proofs<br>(When Not Assigned) . . . . .                                   | 107 |
| 4.5.1 | Categories . . . . .   | 107 |
| 4.5.2 | Illustrative Quotations for Categories . . . . .   | 107 |
| 4.6   | Phenomenographic Analysis of Whether Students Exhibit Consequences<br>of Inability With Proof (such as avoiding recursion) . . . . . | 108 |
| 4.6.1 | Categories . . . . .   | 108 |
| 4.6.2 | Illustrative Quotations for Categories . . . . .   | 108 |
| 4.6.3 | Relations . . . . .  | 109 |
| 4.6.4 | Critical Factors . . . . .   | 109 |
| 4.6.5 | Dimensions of Variation . . . . .  | 109 |
| 4.6.6 | Validation . . . . .   | 109 |
| 4.7   | Phenomenographic Analysis of Student Familiarity with Specific Proof<br>Techniques . . . . .   | 110 |
| 4.7.1 | Categories . . . . .   | 110 |
| 4.7.2 | Illustrative Quotations for Categories . . . . .   | 111 |
| 4.7.3 | Relations . . . . .  | 113 |
| 4.7.4 | Critical Factors . . . . .   | 113 |
| 4.7.5 | Dimension of Variation . . . . .   | 114 |
| 4.7.6 | Outcome Space . . . . .  | 115 |
| 4.8   | Phenomenographic Analysis of How Students Use Structure in Proof . . . . .   | 117 |
| 4.8.1 | Categories . . . . .   | 117 |
| 4.8.2 | Illustrative Quotations for Categories . . . . .   | 117 |
| 4.8.3 | Relations . . . . .  | 119 |
| 4.8.4 | Critical Factors . . . . .   | 119 |
| 4.8.5 | Dimensions of Variation . . . . .  | 120 |
| 4.8.6 | Validation . . . . .   | 121 |
| 4.8.7 | Outcome Space . . . . .  | 121 |

|               |  |            |
|---------------|--|------------|
| 4.9           | Phenomenographic Analysis of What Students Think Makes a Proof Valid . . . . .   | 124        |
| 4.9.1         | Categories . . . . .   | 124        |
| 4.9.2         | Illustrative Quotations for Categories . . . . .   | 124        |
| 4.9.3         | Relations . . . . .  | 126        |
| 4.9.4         | Critical Factors . . . . .   | 126        |
| 4.9.5         | Dimensions of Variation . . . . .  | 127        |
| 4.9.6         | Validation . . . . .   | 128        |
| 4.9.7         | Outcome Space . . . . .  | 129        |
| 4.10          | Phenomenographic Analysis of Combined Data . . . . .   | 131        |
| 4.10.1        | Categories . . . . .   | 131        |
| 4.10.2        | Illustrative Quotations for Categories . . . . .   | 133        |
| 4.10.3        | Relations . . . . .  | 138        |
| 4.10.4        | Critical Factors . . . . .   | 139        |
| 4.10.5        | Dimensions of Variation . . . . .  | 140        |
| 4.10.6        | Validation . . . . .   | 140        |
| <b>Ch. 5.</b> | <b>Interpretation/Discussion</b>   | <b>141</b> |
| 5.1           | Interpretation of What Students Think Proof Is . . . . .   | 143        |
| 5.2           | Interpretation of How Students Attempt to Understand Proofs . . . .  | 146        |
| 5.3           | Interpretation of What Students Think Proof is For . . . . .   | 148        |
| 5.4           | Interpretation of How Students Attempt to Apply Proofs (When Assigned) . . . . .                                       | 150        |
| 5.5           | Interpretation of How Students Attempt to Apply Proofs (When Not Assigned) . . . . .                                   | 152        |
| 5.6           | Interpretation of Whether Students Exhibit Consequences of Inability With Proof (such as avoiding recursion) . . . . . | 153        |
| 5.7           | Interpretation of Student Familiarity with Specific Proof Techniques .   | 153        |
| 5.7.1         | Processes and Conviction . . . . .   | 154        |
| 5.7.2         | Alternate Dimensions of Variation . . . . .  | 155        |
| 5.7.3         | Alternate Critical Factors . . . . .   | 155        |
| 5.8           | Interpretation of How Students Use Structure in Proof . . . . .  | 156        |
| 5.9           | Interpretation of What Students Think Makes a Proof Valid . . . . .  | 158        |
| 5.10          | Interpretation of Combined Data . . . . .  | 159        |
| 5.10.1        | Interpretation of Analysis from Section 4.10 . . . . .   | 160        |
| 5.10.2        | Combined Interpretations . . . . .   | 164        |
| 5.10.3        | Comparison of Combination Before vs. After Interpretation .  | 165        |
| 5.11          | Report: Navigating Through Space of Conceptualization of Proof . .   | 166        |

|               |   |            |
|---------------|---|------------|
| 5.11.1        | Context . . . . .   | 166        |
| 5.11.2        | Conjectures and Conceptualizations . . . . .  | 168        |
| 5.11.3        | Report . . . . .  | 172        |
| <b>Ch. 6.</b> | <b>Validity and Reliability</b>   | <b>175</b> |
| 6.1           | Introduction . . . . .  | 175        |
| 6.2           | Validity and Reliability Goals in the Social Constructivist Paradigm .                        | 176        |
| 6.2.1         | Trustworthiness . . . . .   | 177        |
| 6.2.2         | Credibility . . . . .   | 177        |
| 6.2.3         | Transferability . . . . .   | 178        |
| 6.2.4         | Confirmability . . . . .  | 179        |
| 6.3           | Approaches . . . . .  | 179        |
| 6.3.1         | Purposefully Seeking Diversity . . . . .  | 180        |
| 6.3.2         | Diversity in Participants . . . . .   | 181        |
| 6.3.3         | Diversity in Sources . . . . .  | 182        |
| 6.3.4         | Alternative Explanations . . . . .  | 184        |
| 6.3.5         | Consistency Checking . . . . .  | 185        |
| 6.4           | Specific Papers and Manuscripts . . . . .   | 187        |
| 6.4.1         | Validity and Reliability in Proofs Using the Pumping Lemma<br>for Regular Languages . . . . . | 187        |
| 6.4.2         | Proof by Induction . . . . .  | 189        |
| 6.5           | Statement of Researcher Bias . . . . .  | 190        |
| <b>Ch. 7.</b> | <b>Related Work</b>   | <b>191</b> |
| 7.1           | Summary . . . . .   | 196        |
| 7.2           | Research Questions . . . . .  | 197        |
| 7.3           | Constructivism . . . . .  | 198        |
| 7.3.1         | Social Constructivism . . . . .   | 208        |
| 7.3.2         | Radical Constructivism . . . . .  | 218        |
| 7.3.3         | Neo-Piagetian Theory of Cognitive Development . . . . .                                       | 225        |
| 7.3.4         | Definitions . . . . .   | 226        |
| 7.3.5         | APOS Theory . . . . .   | 227        |
| 7.4           | Social Science Research . . . . .   | 228        |
| 7.4.1         | Psychology of Decision Making . . . . .   | 228        |
| 7.4.2         | Qualitative Research . . . . .  | 230        |
| 7.5           | Mathematics Education Related to Proof . . . . .  | 238        |
| 7.5.1         | Mathematical Problem Solving . . . . .  | 246        |
| 7.5.2         | Proof . . . . .   | 248        |
| 7.6           | Computer Science Education on Proof . . . . .   | 257        |

|               |  |     |
|---------------|--|-----|
| 7.6.1         | Logic . . . . .  | 257 |
| 7.6.2         | Abstraction and Representation . . . . .   | 261 |
| 7.6.3         | Quantifiers . . . . .  | 264 |
| 7.6.4         | Symbols . . . . .  | 264 |
| 7.6.5         | Mathematic Induction and Recursion . . . . .   | 266 |
| 7.6.6         | Teaching Pumping Lemmas . . . . .  | 266 |
| <b>Ch. 8.</b> | <b>Conclusion</b>  | 268 |
| 8.1           | What Students Think Proofs Are . . . . .   | 269 |
| 8.2           | How Students Attempt to Understand Proofs . . . . .                                    | 270 |
| 8.3           | What Students Think Proof is For . . . . .   | 270 |
| 8.4           | How Students Attempt to Apply Proof, When Assigned, and When<br>Not Assigned . . . . . | 271 |
| 8.5           | Whether Students Exhibit Consequences of Inability With Proof . . .                    | 271 |
| 8.6           | Student Familiarity with Specific Proof Techniques . . . . .                           | 272 |
| 8.7           | How Students Use Structure in Proof . . . . .  | 272 |
| 8.8           | What Students Think Makes a Proof Valid . . . . .                                      | 273 |
| 8.9           | Conclusions from the Combined Data . . . . .   | 273 |
| 8.10          | Recognizing an Endpoint . . . . .  | 276 |
| 8.11          | Application of Findings / Implications for Pedagogy . . . . .                          | 277 |
| <b>Ch. 9.</b> | <b>Future Work</b>   | 278 |
| 9.1           | Pursuit of Conclusions . . . . .   | 278 |
| 9.2           | Helping Students Discern Derivation for Proof of Correctness . . . . .                 | 280 |
|               | <b>Bibliography</b>  | 281 |
|               | <b>Appendices</b>  | 317 |
| <b>Ch. A.</b> | <b>Brochure for Critical Aspects Integrated into Curriculum</b>                        | 318 |
| A.1           | Motivation . . . . .   | 320 |
| A.2           | Step of Abstraction . . . . .  | 320 |
| A.3           | Language of Abstraction . . . . .  | 322 |
| A.4           | Logical Deduction, from Definitions, without Anxiety . . . . .                         | 323 |
| A.5           | Representations . . . . .  | 323 |
| A.6           | Transformation Trajectories . . . . .  | 324 |
| A.7           | Recognizing When Techniques Apply . . . . .  | 325 |
| A.8           | Proofs from Algorithms, from Theory of Computation . . . . .                           | 325 |

|   |            |
|---|------------|
| <b>Ch. B. Examination of Multiple Representations</b> | <b>326</b> |
| B.1 . . . . .   | 327        |
| B.2 . . . . .   | 329        |
| B.3 . . . . .   | 330        |
| B.4 . . . . .   | 332        |
| B.5 . . . . .   | 333        |
| B.6 . . . . .   | 335        |
| B.7 . . . . .   | 336        |
| B.8 . . . . .   | 337        |
| B.9 . . . . .   | 338        |
| B.10 . . . . .  | 339        |
| B.11 . . . . .  | 340        |
| B.12 . . . . .  | 341        |
| <b>Ch. C. Interview Protocol</b>                      | <b>342</b> |

## List of Figures

|   |     |
|---|-----|
| 4.1.1 Outcome space from What Proof Is. The outcome space consists of categories of understandings, shown as ovals, and critical factors, given in italic text. Critical factors are thoughts that are necessary[177] for students, in order to develop a deeper or more inclusive understanding. | 59  |
| 4.2.1 Outcome space from How Students Approach Comprehending Proof .  | 77  |
| 4.3.1 Initial Categories from What students think a proof is for . . . . .  | 91  |
| 4.3.2 Outcome space from What students think a proof is for . . . . .   | 93  |
| 4.4.1 Outcome space from How Students Attempt to Apply Proofs When Assigned . . . . .   | 101 |
| 4.7.1 Outcome space from How Students Use Structure in Proof . . . . .  | 110 |
| 4.8.1 Outcome space from What Structural Elements Students Find in Proof  | 118 |
| 4.9.1 Outcome space from What Students Think Makes a Proof Valid . . .  | 125 |
| 4.10.1 Categories from Combined Data . . . . .  | 131 |
| 7.5.1 How proof develops, Tall Mejia-Ramos . . . . .  | 240 |
| B.1.1 Equation (1) . . . . .  | 327 |
| B.1.2 Equation (2) . . . . .  | 327 |
| B.2.1 Three part diagram . . . . .  | 329 |
| B.2.2 Four part diagram . . . . .   | 329 |
| B.3.1 First . . . . .   | 330 |
| B.3.2 Second . . . . .  | 330 |
| B.6.1 Puzzle pieces . . . . .   | 335 |
| B.8.1 Counting by expanding boundary . . . . .  | 337 |

# List of Tables

|   |     |
|---|-----|
| 1.3.1 Distinct Ways of Experiencing Learning, from Marton and Booth [176].                          | 8   |
| 3.2.1 Parts of the Study . . . . .  | 34  |
| 3.5.1 Phenomenographic Analysis of Reasons for Teaching Proof . . . . .                             | 41  |
| 4.1.1 Categories for Student Conceptualizations of What Proof Is . . . . .                          | 60  |
| 4.1.2 Illustrative Quotations for Student Conceptualizations of What Proof Is                       | 61  |
| 4.1.3 Relationships for Student Conceptualizations of What Proof Is . . . . .                       | 64  |
| 4.1.4 Critical Factors for What Proof Is . . . . .  | 66  |
| 4.2.1 Categories for Student Conceptualizations of How to Understand Proofs                         | 76  |
| 4.2.2 Illustrative Quotations for How Students Approach Comprehending<br>Proof . . . . .            | 81  |
| 4.2.3 Relationships for Student Conceptualizations of How to Understand<br>Proofs . . . . .         | 85  |
| 4.2.4 Critical Factors for How Students Approach Comprehending Proof . .                            | 86  |
| 4.3.1 Categories for Student Conceptualizations of Reasons for Teaching Proof                       | 92  |
| 4.3.2 Grouped Categories for Student Conceptualizations of Reasons for<br>Teaching Proof . . . . .  | 93  |
| 4.3.3 Illustrative Quotations for What Students Think Proof is For . . . . .                        | 95  |
| 4.3.4 Relationships for Student Conceptualizations of Reasons for Teaching<br>Proof . . . . .       | 96  |
| 4.3.5 Critical Factors for What Students Think Proof is For . . . . .                               | 97  |
| 4.4.1 Categories for How Students Attempt to Apply Proofs (When Assigned)                           | 100 |
| 4.4.2 Illustrative Quotations for How Students Attempt to Apply Proofs<br>(When Assigned) . . . . . | 102 |
| 4.4.3 Relations for How Students Attempt to Apply Proofs (When Assigned)                            | 103 |
| 4.4.4 Critical Factors for How Students Attempt to Apply Proofs (When<br>Assigned) . . . . .        | 104 |

|   |     |
|---|-----|
| 4.5.1 Illustrative Quotations for How Students Attempt to Apply Proofs<br>(When Not Assigned) . . . . .                                   | 107 |
| 4.6.1 Illustrative Quotations for Whether Students Exhibit Consequences of<br>Inability With Proof (such as avoiding recursion) . . . . . | 108 |
| 4.7.1 Categories for Student Conceptualizations of Specific Proof Techniques  | 111 |
| 4.7.2 Illustrative Quotations for Student Familiarity with Specific Proof Tech-<br>niques . . . . .                                       | 111 |
| 4.7.3 Relationships for Student Conceptualizations of Familiarity with Spe-<br>cific Proof Techniques . . . . .                           | 113 |
| 4.7.4 Critical Factors for Student Familiarity with Specific Proof Techniques   | 114 |
| 4.8.1 Categories for Student Conceptualizations of What Structural Ele-<br>ments are Found in Proofs . . . . .                            | 117 |
| 4.8.2 Illustrative Quotations for How Students Use Structure in Proof . . .   | 118 |
| 4.8.3 Relationships for Student Conceptualizations of What Structure Stu-<br>dents Notice in Proofs . . . . .                             | 119 |
| 4.8.4 Critical Factors for What Structure Students Notice in Proofs . . . .   | 120 |
| 4.9.1 Categories from What Students Think Makes a Proof Valid . . . . .   | 124 |
| 4.9.2 Illustrative Quotations for What Students Think Makes a Proof Valid   | 125 |
| 4.9.3 Relationships from What Students Think Makes a Proof Valid . . . .  | 127 |
| 4.9.4 Critical Factors from What Students Think Makes a Proof Valid . . .   | 128 |
| 4.10.1 Categories for Combined Data . . . . .   | 132 |
| 4.10.2 Illustrative Quotations for Combined Data . . . . .  | 134 |
| 4.10.3 Relationships for Combined Data . . . . .  | 138 |
| 4.10.4 Critical Factors from Combined Data . . . . .  | 139 |



# Chapter 1

## Introduction

Proofs are taught in the computer science curriculum, with the intent that students will be able to use them. Students should use them to ascertain whether the algorithms they have learned are applicable to the problems they wish to solve. Moreover, it is desirable that students will be able to show that the resource consumption of their implementation is suitable for the circumstances of their implementations. Beyond correctness and resource utilization, students will use proof to know that many problems are undecidable, and that complexity classes can be defined.

It can be the case that not every student gains the whole of the capabilities that we might wish. Insight into the partial understandings students do achieve might direct our attention to those elements of the related courses, further emphasis upon which might aid student learning.

The population we study is students of computer science, or of computer science and engineering (henceforth, computer science).

The purpose of the research is to discover the understandings of proof we find in

these students, so as to provide information that might be helpful for teaching. It is a qualitative study, because we seek the variety in the nature of the various understandings. We do not attempt to establish the relative frequencies with which these conceptualizations occur.

The analytic lens of the study is phenomenography, including variation theory[172, 253, 176, 179]. This lens is appropriate for discovering ideas critical for advancing between stages of understanding.

In this chapter we list our research questions addressing conceptions of proof in students of computer science. We briefly summarize what distinguishes qualitative research from other research. We provide a short description of phenomenography as extended to variation theory. We explain how phenomenography and variation theory are suited to investigate the questions we have pursued.

It is important for students of computer science to comprehend, apply, and synthesize proofs. These skills are needed because proofs are used to demonstrate the resource needs and performance effects of algorithms, as well as for safety, liveness, and correctness/accuracy. We claim herein that some students, having learned an algorithm, are not certain of the problem environment in which this kind of algorithm is effective, and as a result are reluctant to apply the algorithm. It is desirable for students to be able to ascertain that an algorithm is a good match for a problem, which can sometimes be proved, otherwise their knowledge of the algorithm is less useful.

## 1.1 Research Questions

In order to address the students effectively, it can help to know their preparation and their approach to gaining new knowledge. This preparation may include useful ideas, and also may include unhelpful conceptions. Their approach to learning might not yet include the degree of attentiveness to precision and thoroughness that is appropriate for deductive logic. Besides their current knowledge and learning approach, their opinion of the structure of relevancy is of interest. We have identified facets of student conceptions surrounding the idea of proof, which led to specific questions.

We propose to research these questions:

- What do students think a proof is?
- How do students attempt to understand proofs?
- What do students think a proof is for?
- What do students attempt to apply proof, when assigned and when not assigned?
- Do students exhibit any consequence of inability in using proof (i.e., applying proof techniques they have studied, or noticing that proof is an appropriate technique when it is), such as, avoiding using recursion?
- How do students use structure in proof?
- How familiar and/or comfortable are students with different (specific) proof techniques: induction, construction, contradiction?
- What do students think makes a proof valid?

These questions are interesting because, with the curriculum, we are trying to build capabilities into the students, that will enable them to tackle various problems they may encounter. Moreover, we wish the students to develop the ability to have, in the terminology of Harel and Sowder[118], conviction with an internal source, and to be correct in their convictions. As new situations emerge, and as students who have graduated find the occasion to modify an algorithm to a new situation, we want these individuals to be able to know that their modified algorithms are appropriate. It is important that they understand this algorithm-applicability purpose of proof, so that they can judge applicability for themselves, and it is important to know what hindrances they are experiencing, so that we can help the students overcome them. It is important that they recognize that there is structure in proofs, and that they can construct their own proofs, because we cannot foresee every situation our students may experience.

Because we are greatly concerned that students should apply their knowledge of proof to algorithm-related contexts they may subsequently encounter, the split between what students perform for assessment, and what students perform for their own purposes is significant to us. Thus it may be helpful to supplement what assessments tell us, about the extent to which the students are absorbing the knowledge about proof we are trying to impart, with information from interviews. Interviews impose a burden of analysis which is usually too extreme for a lecture class. Progression across multiple courses is beyond the scope of a lecture class.

Phenomenographic research yields critical factors, which are ideas whose emphasis is thought to be particularly helpful in deepening student understanding. Thus the relevance of this research to the curriculum is that the work will generate suggestions about points to emphasize.

## 1.2 Qualitative Research

“When instructors understand what students know and how they think — and then use that knowledge to make more effective instructional decisions — significant increases in student learning occur” Black and Williams[30]

How students think, for example, how they approach the study of proof, is a qualitative question.

One product of qualitative analysis is definition of categories that describe some structure of the collection of understandings in the student population. These can help phrase questions for quantitative studies.

Qualitative analysis has types; some of these were created to address specific domains of research. These types include basic qualitative research, phenomenology, ethnography, grounded theory, and narrative analysis, each of which is interpretive [190]. Critical research, while qualitative, goes beyond interpretation, as it intends to reform the object of its attention. Each of the above is a type of qualitative research associated with, and characterized by, a qualitative research methodology. One type of research, which differs from those above in that it delimits its scope to the description of ways of experiencing, by a student, the communication from a source of instruction, is phenomenography[253]. While the formerly mentioned types of qualitative research are characterized by a qualitative research methodology, phenomenography is characterized by an aim, namely to describe people’s conceptions. [253]. All the methods just mentioned produce descriptions. In phenomenography, it is necessary that the descriptions be subject to comparison and to systematization. As for the methods associated with phenomenography, Svensson states [253, p. 161–162] “development of methods was included as a main aim of the research and the devel-

opment of methods was an integrated part of the tradition. However, methods have been dealt with in a problem-solving attitude rather than by giving prescriptions. The methods have been considered to have been derived from the basis of the general aim, the general character of the phenomena of conceptions and also from the basis of the specifics of the phenomena and the situation under investigation. Therefore phenomenographic research represents a research approach in a double sense. It means an emphasis on approaching the research objects in the sense of creating methods adapted to the objects. . . . The approach to describing conceptions is closely related to the view of the research objects and is not a system of generally defined methods. The most significant characteristics of the approach are the aiming at categories of description, the open explorative form of data collection and the interpretive character of the analysis of data.”

Svensson [253] reports that phenomenography was extended to include variation theory. Variation is key to the phenomenography/variation theory qualitative research approach. Variation, by the instructor, of the communicated information is deemed necessary for discernment by the student of what is being described by the instructor[177].

### **1.3 Phenomenography with Variation Theory**

Phenomenography is an empirical research approach. [253] Svensson states [253, p. 169–171] that phenomenography’s “methodological assumptions also tend to have generality. The most central characteristics are the explorative character of the data collection and the contextual analytic character of the treatment of data. . . . The

analysis, then must not only mean an aggregation of specific data within generally given interpretations, but a delimitation of specific data related to each other as referring to parts of the same phenomena... The content is, then, not primarily considered in terms of meaning of linguistic units, but from the point of view of expressing a relation to parts of the world. . . . scientific knowledge about conceptions is based on differentiation, abstraction, reduction and comparison of meaning”.

Booth [36] states “When I say ‘methodology’ I mean that phenomenographic studies are based on certain principles but that the actual methods used vary according to the specific question being addressed; it is not prescriptive. Phenomenography is not an experimental methodology – phenomenographers do not set up controlled trials and attempt to measure the results of change; but it is relatively naturalistic, researchers engage with the learners themselves, in close proximity (both spatial and conceptual) to the learning situations they find themselves in. Nor is it a quantitative methodology – the results are descriptive and lie at a collective level, in the sense that individuals are seen as contributing fragments of data that together constitute a whole and collective experience, which can be subjected to research analysis.”

Marton and Säljö [180] performed an experiment which showed that students’ approaches to learning have been predictive of their learning outcome, reiterated in Marton and Booth. [176, p. 22] In looking at, and developing categories for, students’ ways of experiencing their learning, we may obtain insight into the students’ approaches, and can hope to improve their outcomes.

Marton[176, p. 36] has defined that one conception (of a thing,  $x$ ) differs from another, for the purposes of phenomenography, by the existence of a distinct manner in which participants were found to voice the way they thought about  $x$ . The categories of conceptions (also, conceptualizations) include two overriding categories,[176,

p. 35] the first being “a learning task, some facts to memorize”, and the second having as objective “a way to change oneself, to see things in a new light, to relate to earlier learning, and to relate to a (changed) world.” At the next level of drawing distinctions, Säljö [225] has found five qualitatively distinct conceptualizations, and Marton [176] has found six distinct conceptualizations falling into the two overriding, task and objective. (See Table 1.3.1.)

TABLE 1.3.1: Distinct Ways of Experiencing Learning, from Marton and Booth [176].

1. learn as increase knowledge
2. learn as increase and be able to reproduce knowledge
3. be able to apply new knowledge
4. acquiring new meaning, multiple ways of thinking about things, changed perspective, improved understanding, thinking more logically
5. modified perspective, multiple perspectives, dynamic perspective
6. changing the person

Marton and Booth[176, p. 78] observe that successive understandings increase in completeness as they move toward a theoretical understanding.

### 1.3.1 Conceptualizations

Selden and Selden[142] include, in their questions regarding teaching and learning mathematics, that instructors aim for their students to “achieve the kind of organizing and integrated use of language” used in the mathematics community.

Dörfler[69, p. 122] complements the idea of concept, saying “ ‘ What is the concept  $xy$ ? ’ should be substituted, or at least complemented, by such questions as ‘Which



actions can be recorded and/or guided by the concept  $xy$ ?' . . . Learners must indulge in the discourse . . . mathematical objects . . . are discursive objects. This means they come into existence exclusively by and within the discourse, even if this discourse ascribes to them existence an properties of an objective and independent character."

Wittgenstein said [301, p. 19–20] "To understand a phrase, we might say, is to understand its use. . . . Similarly, you only understand an expression when you know how to use it".

So, we are inquiring into student conceptualizations, as shown by the students' use of their concepts, and by the students' reflections (in interviews) upon their concepts.

### 1.3.2 Relevance Structure

Relevance structure is a part of phenomenography: Marton and Booth state [176, p. 143] "Each situation, whether we consider it a learning situation or a situation in which one is applying something learned, has a certain *relevance structure*: the person's experience of what the situation calls for, what it demands. It is a sense of aim, of direction, in relation to which different aspects of the situation appear more or less relevant. It is the way the learner experiences the situation as a whole, . . . that renders the perspective in its component parts."

Knowledge may alter the perspective with which students view new material. For example, students who know they are about to work on projects, in which they are expected to apply new material they are studying, are enabled to receive and organize, mentally, that new material, assisted more or less by its relation to their anticipated use of it.

Marton and Booth suggest that relevance structure may be "the driving force

of learning”[176, p. 145] and assert that learning’s “chief mechanism is variation”. Changes in the curiosity of the learner, as may be brought about by the presentation of problems, can change the student’s capability of experiencing something, and thereby change the student’s way of experiencing it, which is learning. Sudden insight is an example of a change in a person’s way of experiencing something.

Booth has written on relevance structure in learning computer science. [36] She writes “In pedagogical terms, the course aims to provide the heterogeneous group of new students with a relevance structure . . . a whole sense of the programme, what it is aimed at, what it demands and where it will lead – not only by bringing about genuine needs to use the practical tools they are learning, but also by raising questions that relate to theoretical studies to come and to social issues” [36, p. 176]

There is a connection between the ideas relevance structure and motivation, as illustrated in a remark of Booth’s “to create a relevance structure, or motivation if preferred, for the coming education by offering the students from the start a whole picture – however vague and incomplete at the outset – that encompasses the overall goals of the programme” [36, p. 178]

Thota[265] refers to the relevance structure when considering “pedagogic principles . . . that consider how to tie students’ experiences to the course goals (relevance structure) . . . The two principles of teaching that are espoused in phenomenographic pedagogy are (a) the relevance structure – making the learner personally experience the relevance of the learning situation, and (b) variation theory – emphasizing the critical aspects that lead to a change in the learner’s understanding of a learning concept.” Thota claims that “The learning that occurs is dependent on how the learner personally experiences the relevance of the learning situation”. [265, p. 127]

Thota, citing Bowden and Marton[38] observes “exposure to the critical aspects

of professional situations, likely to be encountered in the future, enables students to develop holistic capabilities that link disciplinary knowledge and professional skills.” Thota relates that a relevance structure can be built, in part, by relating and contextualizing learning.[265, p. 128]

Thota reports[265, p. 131], of students “they become more engaged in learning when they see the relevance of programming to their chosen major and future career needs.”

Thota summarizes[265, p. 131] “The building of the structure of relevance can be achieved by: (a) awareness of pedagogical content knowledge as it should be understood by learners; (b) ensuring learners reveal their experience of ways of learning (including the what and how of learning); (c) relating learning to the learner’s own personal interests and motivations and contextualizing the set assignments to create authentic situations.”

Relevance structure plays a role in motivating student learning. Relevance structure can be present in a course at different levels of detail. Present not only at the level of “this test driven development practice might aid your employability”, relevance can be shown at the level of an individual algorithm. For example, the purpose of learning to prove whether a problem has optimal substructure can be to determine whether a dynamic programming approach is suitable for a solution. Students who do not know that optimal substructure is important for the use of a dynamic programming approach might not make mental connections between these ideas; it might not occur to them to make use of a dynamic programming approach in a situation when that is warranted. It is claimed herein that relevance structure provides another avenue for retrieval of knowledge, lessening the risk of knowledge being inert (in the sense of Whitehead[300]).

## 1.4 Phenomenography / Variation Theory for these Research Questions

Conceptualizations, as we have seen above, are important for our research questions, and are a central object of attention for phenomenography.

Phenomenography and variation theory (henceforth “phenomenography”) address mental concepts without reference to any sensory modality through which they may have been acquired. [176, p. 160] As a deductive, logical argument, a proof is a mental concept that can exist without reference to any sensory modality.

Phenomenography concerns itself with students’ approaches to learning. This allows us to hope to effect a change in approach, and thereby effect an improvement in students’ outcomes.

## 1.5 Overview

Chapter 2 discusses the phenomenographic research perspective, and the epistemological framework. Chapter 3 discusses the methodology applied in the study, including sections on sample selection, data collection, and analysis. Chapter 4 shows the analysis and describes the results. Chapter 5 provides interpretation and discussion. Chapter 6 discusses validation and reliability. Chapter 7 discusses some related work. Chapter 8 concludes the description of completed work. Chapter 9 describes some possible future work.

## Chapter 2

# Research Perspective and Epistemological Framework

This work is a phenomenographic study, adopting the epistemological framework of social constructivism.

A qualitative study ought, for transparency, to contain a description of the author's viewpoint, or at least, the viewpoint that was exercised in the study. Ernest[86, p. x], crediting von Glasersfeld[278, p. 41], notes:

To introduce epistemological considerations into a discussion of education has always been dynamite. Socrates did it, and he was promptly given hemlock. Giambattista Vico did it in the 18th century, and the philosophical establishment could not bury him fast enough.

The work of others has contributed to both the research perspective and the epistemological framework<sup>1</sup>.

---

<sup>1</sup>In the psychological sense, rather than that of Brouwer on the foundations of mathematics

- Marton developed the phenomenographic research perspective, which is broadly applicable to education.
- Many researchers have contributed to the literature on educating students in the use of proof.
- Significant work has been done on teaching computer science students about mathematical proof.

Vygotsky pointed out the “zone of proximal development”, contrasting what students could achieve on their own with that which students could achieve with the assistance of others. He wrote:[283, p. 85]

In studies of children’s mental development it is generally assumed that only those things that children can do on their own are indicative of mental abilities. We give children a battery of tests or a variety of tasks of varying degrees of difficulty, and we judge the extent of their mental development on the basis of how they solve them and at what level of difficulty. On the other hand, if we offer leading questions or show how the problem is to be solved and the child then solves it, or if the teacher initiates the solution and the child completes it or solves it in collaboration with other children – in short, if the child barely misses an independent solution of the problem – the solution is not regarded as indicative of his mental development. This “truth” was familiar and reinforced by common sense. Over a decade even the profoundest thinkers never questioned the assumption; they never entertained the notion that what children can do with the assistance of others might be in some sense even more indicative

of their mental development than what they can do alone.

He wrote[283, p. 86]

When it was first shown that the capability of children with equal levels of mental development to learn under a teacher's guidance varied to a high degree, it became apparent that those children were not mentally the same age and that the subsequent course of their learning would obviously be different. This difference . . . is what we call the zone of proximal development. It is the distance between the actual developmental level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or in collaboration with more capable peers.

By choosing a version of social constructivism that directs attention to finding that zone in which a teacher, sometimes employing collaboration with peers, may most effectively advance the student's development, we inherit the situation described by Ben-Ari[26]: "The task of the teacher is significantly more difficult than in the classical paradigm, because the guidance must be based on the understanding of each student's currently existing cognitive structures."

This choice about social constructivism affects our perspective by establishing a scope, namely, addressing specifically how a teacher may assist a student to learn, and how a teacher may make use of collaboration among peers to assist a student to learn. Using Vygotsky's definition of the zone of proximal development affects our perspective by focusing our efforts on matching teaching interventions with students' readiness to benefit from those interventions. By choosing as a basis, this social constructivism, the perspective is directed towards curriculum design, so that

the students' zones of proximal development may be encouraged to advance along a trajectory supported by sequenced teaching activities.

## 2.1 Phenomenography and Variation Theory

Phenomenography, a research orientation created and developed by Ference Marton and colleagues[176, 253], has as its purpose the discovery and elucidation of categories of conceptions, mainly in the context of education. Its subject of investigation is conceptions present in a group of students. These *conceptions*, also *conceptualizations*, are of the meaning of some thing, in particular of some thing which is the object of the instructor's exposition, called, the learning objective. To describe partial or imperfect learning, it is helpful to separate the meaning understood by the instructor from the meanings developed by the students. The nature of the conceptualizations, which are collectively the object of the description, is the meaning that something has to the individual, i.e., the individual's understanding.

To understand in some detail the degree of completeness of a conceptualization, the degree of superficiality or depth of a conceptualization, description of conceptualizations are needed. Note that the students' conceptualizations are not limited to a subset of the instructor's meaning, as students may bring other data, including incorrect understandings, from other sources. The descriptions are categorized, producing categories of description. The categories of description, to be useful, must be sufficiently comparable to be seen in relation to one another. When categories of description are related, the distinguishing factors between categories make manifest the concepts that are necessary for students to discern, to advance to an adjacent, better,



conceptualization. These are the *critical factors*. In the context of conceptualizations differing in completeness, these are also called *critical aspects*.

Thus the process of obtaining the descriptions and their categories is of some interest. The process uses an open, exploratory form of data collection. For raw data, researcher with with interview transcripts and other verbal productions be learners, extracting text fragments conveying meaning, and exploring possible categorizations of these excerpts.

Together, the categories and relationships between them are called the *outcome space*.

It can be that categories differ in completeness, and that more than two degrees of completeness are found in the conceptualizations of the students. It can be that categories differ in depth of understanding, and that more than two degrees of depth are found. These are only two examples of dimensions of variation. From the outcome space, researchers infer *dimensions of variation*. In turn, dimensions of variation provide a framework by which researchers can organize the information garnered from student verbal productions. Variation theory applies the idea of a range of values in multiple ways. One way is in making ideas discernible. Pang and Marton[177] describe that a child knowing of only one spoken language conflates the ideas of natural language and that one language, but a multilingual child knows that natural language is the more general concept.

### 2.1.1 Variation Theory

Dahlin[61, p. 328] recounts that variation theory was a development upon phenomenography that brought in dynamic elements to the description of conceptu-

alizations. “The concepts of discernment, variation and simultaneity are the core of variation theory. In order for learning to take place, the learner has to discern a critical aspect or dimension of variation in the phenomenon; she has to see how this aspect can vary; and she has to become simultaneously aware of the possible “values” along this dimension of variation in order to compare them.”

Variation is seen as occurring among conceptualizations, and as occurring during the learning. That is, the teacher may emphasize variation of an aspect of the material being taught, and may emphasize that values taken on along this dimension of variation are significant for the material being taught. This emphasis serves to help students discern not only the dimension of variation, but the factor that is changing; change of the factor calls attention to the factor. Were that factor constant, it might not be noticed. Variation among the categories of description extends the outcome space, such that more distinct conceptualizations are found.[176, p. 124–125].

Variation theory[178] suggests that critical aspects, which are particular ideas, are necessary[177] for meaning-making (understanding), to progress from one level of conceptualization to a more advanced level.

The research approach associated with variation theory has the goal of identifying these specific ideas, which, on the basis of an identification of the conceptualizations present in a student population, empirically are seen to differentiate one level of conceptualization from another.

Variation theory uses these so-called critical factors. They are emphasized in teaching, specifically by varying them, and considering the consequences. For example, we may highlight the significance of climate zones by considering the variation in annual rainfall from one zone to another. We may make salient the distinction between the ideas of language as contrasted with speech, the difference between speaking

(in general), and speaking in a specific language (in particular) by acquainting children with the existence of a second language. (In the context of only one language, the distinction still exists, but might not be so readily described or learned.)

Marton and Pang[177] identify “some necessary conditions of learning. To learn something, the learner must discern what is to be learned (the object of learning). Discerning the object of learning amounts to discerning its critical aspects. To discern an aspect, the learner must experience potential alternatives, that is, variation in a dimension corresponding to that aspect, against the background of invariance in other aspects of the same object of learning. (One could not discern the color of things, for instance, if there was[sic] only one color.) The study results illustrate that what students learn in a sequence of lessons is indeed a function of the pattern of variation and invariance constituted in that sequence. All teachers make use of variation and invariance in their teaching, but this study shows that teachers informed by a systematic framework do it more systematically, with striking effects on their students’ learning.”

### **2.1.2 Application to Proof in Computer Science**

Phenomenography, and its extension variation theory, are appropriate tools for comprehending student conceptualizations about proof in computer science, because the subject is the understandings, including those obtained after reflection, of proof present in the student population. These understandings have been seen to vary in completeness and in depth. Comparison of categorizations has suggested items to clarify for students, and students have responded positively to these suggested ideas.

Marton and Booth[176] extended phenomenography and variation theory to in-

clude the idea of structural relevance. Using this idea, we can hope to help students discern an aspect of a learning objective, by pointing out a connection between this relatively new aspect and a pre-existing notion that provides, for example, motivation to take note of this aspect. With proofs we may bring to students' attention that these will be used in future course work to understand algorithm's properties, such as computational complexity.

By attending to the construction of structural relevance as well as to the construction of the curriculum's learning objectives, we can provide help to students, in the form of motivation.

### **2.1.3 An example of applying variation theory**

A commonly used example of a proof utilizing one application of modus ponens is:

All men are mortal.

Socrates is a man.

Socrates is mortal.

Variation theory tells us we must vary critical factors, for students to discern them. Some examples of variation are:

Some men are mortal.

Socrates is a man.

Socrates is mortal, maybe, but not necessarily.

The quantifier "All" matters. We don't get the desired result when we use some.

When we have removed insignificant items from our proof, what's left matters. It's easy to find elements to vary that will affect the outcome.

All men are mortal.  
 John Doe is a man.  
 Socrates is mortal.

What's different here is, we have lost the warrant for Socrates being mortal. Without that, we cannot know for sure that Socrates is mortal.

All men are mortal.  
 Socrates is a person.  
 Socrates is mortal.

This time we've kept our attention on Socrates, but we have lost the warrant. To have a warrant, we must remain within the domain granted by the axiom.

All men are mortal.  
 Socrates is a man.  
 Socrates is an orator.

The final statement, though possibly true, is not justified by any warrant.

### **2.1.4 Variation Theory and Conjunctions**

Marton and Booth[176] have observed that increased differentiation, i.e., specialization, and also integration in the ways in which we experience the world are the results of learning.

The mind quickly learns certain specializations. This was predicted by Valiant[270]. This was verified experimentally by Fried, using single neuron experiments[93]. Cognitive neuroscience predicts that specializations that are conjunctions of positive literals of existing concepts are easy to learn and that conjunctions containing literals that are not existing concepts, but are negations of existing concepts may not be[270]. By examining the conceptualizations present in the population of learners, we can hope to find clusters from which we can learn features whose values differentiate the clusters. It is these features, called in variation theory “critical factors”, which instructors should emphasize, showing in their positive and negative form.

## 2.2 Constructivism

Constructivism entails the idea that students learn by aggregating new information onto their present conceptions.

Whilst part of what we perceive comes through our senses from the object before us, another part (and it may be the larger part) always comes out of our own head

—William James (Quoted in [273])

Brooks and Brooks[43, p. 4] observe: “Each of us makes sense of our world by synthesizing new experiences into what we have previously come to understand.”

There are many specializations of constructivism.[87] These range over the radical constructivism of von Glasersfeld[280] and social constructivism.[283] Some debate has gone on, in the mathematics education literature as to which type of constructivism is most suitable for mathematics education.[159, 26]

### 2.2.1 Piagetian Constructivism

Piaget[203] studied learning, and proposed the idea that learning was a form of adaptation to the environment. He suggested that learning took place as development, by additions to and modifications of what was already present. This has been called constructivism. Others have been influenced by constructivist ideas.

McGowen and Tall[186] suggest that “it is even more important to take into account the particular mental structures available to the individual that have been built from experience that the individual has ‘met-before’.” They say [p. 170] “New experiences that build on prior experiences are much better remembered and what does not fit into prior experience is either not learned or learned temporarily and easily forgotten.”

Thompson [264] states “...an instructor who fails to understand how his/her students are thinking about a situation will probably speak past their difficulties. Any symbolic talk that assumes students have an image like that of the instructor will not communicate. Students need a different kind of remediation, a remediation that orients them to construct the situation in a mathematically more appropriate way.”

It appears that within constructivism there are degrees to which the teachers regard their role as facilitative.

Wenger[299, p. 279–280] notes “Constructivist theories focus on the processes by which learners build their own mental structures when interacting with an environment. Their pedagogical focus is task-oriented. They favor hands-on, self-directed activities oriented toward design and discovery. They are useful for structuring learning environments, such as simulated worlds, so as to afford the construction of certain

conceptual structures through engagement in self-directed tasks.”

Brooks and Brooks[43] enumerate, in their description of constructivism applied in the classroom, five overarching principles:

- Teachers seek and value their students’ points of view.
- Classroom activities challenge students’ suppositions.
- Teachers pose problems of emerging relevance.
- Teachers build lessons around primary concepts and “big” ideas.
- Teachers assess student learning in the context of daily teaching.

They[43, p. *x*] go on to say “Engagement in meaningful work, initiated and mediated by skillful teachers, is the only high road to real thinking and learning.” They illustrate the student’s viewpoint of such interactions: “teachers ...made difficult concepts accessible by seeking to understand what [the student] knew at the time ...these remarkable teachers mattered so much because they were less concerned about covering material than they were about helping students connect their current ideas with new ones.”

Ben-Ari[26] articulated a slightly different version of constructivism:

“Passive learning will likely fail, because each student brings a different knowledge framework to the classroom, and will construct new knowledge in a different manner. Learning must be active: the student ‘must construct knowledge assisted by guidance from’ the teacher and feedback from other students.”

The value of social interaction is shown clearly by an experiment by Bausell et al.[25], carefully comparing tutoring with classroom instruction, in which tutoring produced significantly greater achievement.



Marton contrasts what he calls individual constructivism with his idea of social constructivism. [176]

### 2.2.2 Social Constructivism

There are multiple perspectives on social constructivism.

Wenger[299, p. 4] states: “My assumptions as to what matters about learning and as to the nature of knowledge, knowing, and knowers can be succinctly summarized as . . .

1. We are social beings. Far from being trivially true, this fact is a central aspect of learning.
2. Knowledge is a matter of competence with respect to valued enterprises – such as singing in tune, discovering scientific facts, fixing machines, writing poetry, being convivial, growing up as a boy or a girl, and so forth.
3. Knowing is a matter of participating in the pursuit of such enterprises, that is, of active engagement in the world.
4. Meaning – our ability to experience the world and our engagement with it as meaningful – is ultimately what learning is to produce.

As a reflection of these assumptions, the primary focus of this theory is on learning as social participation. . . .being active participants in the *practices* of social communities and constructing *identities* in relation to these communities.”

Ernest[86, p. 65] reports “social constructivism is used to refer to widely divergent positions. What they share is the notion that the social domain impacts on

the developing individual in some formative way, and that the individual constructs (or appropriates) his or her meanings in response to his or her experiences in social contexts.” He[86, p. 66] says, “In simplified terms, the key distinction among social constructivist theories of learning mathematics is that between individualistic and cognitively based theories (e.g., Piagetian or radical constructivist theories), on the one hand, and socially based theories (e.g., Vygotskian theories of learning mathematics), on the other.”

Lev Vygotsky founded the idea of social constructivism, which can be summarized as learning is facilitated by interactions in a group.

According to Cole and Scribner[283, p. 1], Vygotsky “and his colleagues sought to develop a Marxist theory of human intellectual functioning”. They say[p. 5–6] that “What Vygotsky sought was a comprehensive approach that would make possible description *and* explanation of higher psychological functions in terms acceptable to natural science. To Vygotsky, explanation meant a great deal. It included identification of the brain mechanisms underlying a particular function; it included a detailed explication of their developmental history to establish the relation between simple and complex forms of what appeared to be the same behavior; and, importantly, it included specification of the societal context in which the behavior developed.”

According to Cole and Scribner[283, p. 6], “In stressing the social origins of language and thinking, Vygotsky was following the lead of influential French sociologists, but to our knowledge he was the first modern psychologist to suggest the mechanisms by which culture becomes a part of each person’s nature. Insisting that psychological functions are a product of the brain’s activity, he became an early advocate of combining experimental cognitive psychology with neurology and physiology. Finally, by claiming that all of these should be understood in terms of a Marxist theory of the

history of human society, he laid the foundation for a unified behavioral science.”

According to Cole and Scribner[283, p. 7], “Vygotsky believed that the internalization of culturally produced sign systems brings about behavioral transformations and forms the bridge between early and later forms of individual development. Thus for Vygotsky, in the tradition of Marx and Engels, the mechanism of individual developmental change is rooted in society and culture.”

Marton and Booth[176, p. 11] “prefer to use ‘social constructivism’ as an umbrella term for a rather diverse set of research orientations that have in common an emphasis on what surrounds the individual, focusing on relations between individuals, groups, communities, situations, practices, language, culture and society.” They give further examples[176, p. 201] “an emphasis on cultural, linguistic, social, historical situations”. Of social constructivism, they[176, p. 12] say “Individual constructivism is a form of cognitivism in the sense that it regards the outer (act, behavior) as being in need of explanation and the inner (mental acts) as explanatory, whereas, as we have pointed out, the reverse is true of *social constructivism*.” This understanding must be taken in context, because Marton and Booth go on to say[176, p. 12] that “in this book the dividing line between ‘the outer’ and ‘the inner’ disappears. . . . The world . . . is *constituted* as an internal relation between them.”

It can also be observed that a person’s inner life can be influenced by the external world, and that the inner life can in turn motivate an individual’s behavior, and that the social surround of a person may react to that behavior, with the reaction impacting the individual. Thus, a cyclic feedback situation may result. The idea of a person embedded in a feedback situation seems to encompass more possibilities than either of the ideas Marton places in contrast, “individual constructivism” or “social constructivism”. Moreover, feedback loops, which seem a feasible model for human

interaction with society, enjoy more complex dynamics than open loop systems, which may endow them with greater explanatory power.

From this range of inquiry about and understanding of social constructivism, we extract a domain over which we can imagine the style of thinking our students are using. Some students may wish to think individually, developing an opinion in their own way, exploring relevant materials and activities on their own, before engaging socially in discussion on a topic. Other students might find exploration of possibilities in a social context helpful, as they are in the process of deciding how they are experiencing and integrating new concepts with what they already know.

In this study we employ a social constructivism into our theoretical and epistemological framework: We are aware the students can vary as to their ways of acquiring, consolidating, relabilizing and reconsolidating meaning.

## 2.3 Mathematics Education

Part of the research perspective is formed by the goals for what students learning proof should know: according to Ball et al.[170, p, 32 – 34] “These activities – mathematical representation, attentive use of mathematical language and definitions, articulated and reasoned claims, rationally negotiated disagreement, generalizing ideas, and recognizing patterns – are examples of what we mean by *mathematical practices*. . . . These practices and others are essential for anyone learning and doing mathematics proficiently. . . . investing in understanding these ‘process’ dimensions of mathematics could have a high payoff for improving the ability of the nations’ schools to help all students develop mathematical proficiency”. Ball goes on to say[170, p. 37] “Another

critical practice – the fluent use of symbolic notations – is included in the domain of representational practice. Mathematics employs a unique and highly developed symbolic language upon which many forms of mathematical work and thinking depend. Symbolic notation allows for precision in expression. It is also efficient – it compresses complex ideas into a form that makes them easier to comprehend and manipulate. Mathematics learning and use is critically dependent upon one’s being able to fluently and flexibly encode ideas and relationships. Equally important is the ability to accurately decode what others have written.”

Even more tightly focused on proofs, Ball continues [170, p. 37–38] “A second core mathematical practice for which we recommend research and development is the work of justifying claims, solutions, and methods. *Justification* centers on how mathematical knowledge is certified and established as ‘knowledge’. Understanding a mathematical idea means both knowing it and also knowing why it is true. For example, knowing that rolling a 7 with two dice is more likely than rolling a 12 is different from being able to explain why this is so. Although ‘understanding’ is part of contemporary education reform rhetoric, the reasoning of justification, upon which understanding critically depends, is largely missing in much mathematics teaching and learning. Many students, even those at university level, lack not only the capacity to construct proofs – the mathematician’s form of justification – but even lack an appreciation of what a mathematical proof is.”

Rota[219] points out that proofs that are perceived by mathematicians as beautiful are easier to remember.

Of course there is more to the use of proof than learning some proofs, and learning some proof techniques. There is a creative, problem solving, component, that appears in creating proofs, and also sometimes in applying proofs (for example, in Introduction

to the Theory of Computation).

Mathematician Alan Schoenfeld, who has had noteworthy success in teaching students to be creative mathematicians and problem solvers, states [142, p. 91] “develop knowledge and skills, pursue connection, extensions, generalizations to know how to make good conjectures and know how to prove them, have a sense of what it means to understand mathematics and good judgment about when they do. Have the tools that will enable them to do so. That means having a rich knowledge base, a wide range of problem solving strategies and good meta-cognitive behavior”. He had, earlier on the same page, described meta-cognitive behavior as reflecting and acting on what you know.

## Chapter 3

# Methodology

Knowledge about how students conceptualize has a qualitative nature. For qualitative research, methodology varies, but has standard parts: design of the study, sources and their selection, data, the process of analysis, the interpretation, and the approach to validation. Sample selection is recorded and reported so that others may judge transferability to their own context. The kinds of data in a qualitative study include interviews and documents. Interviews are the principle technique used by phenomenographical research[190, p. 86]. Normal conduct of teaching can also provide data that can be used, if in an anonymous, aggregate form. Both deductive and inductive analysis can be carried out on these data. The analysis produces a description of the situation under study. This description may include a narrative, often called a thick and rich description, and also specific attributes, such as categories of findings and relationships among these categories.

### 3.1 Design of the Study

Information learned in tutoring and lecturing undergraduates inspired the research questions. More specifically, questions asked by the students suggested that they were not learning enough about proof techniques to understand material that appeared later in the curriculum. So, it seemed useful to discover what their ideas were, about proofs. We used Bloom's taxonomy of the cognitive domain[32] to subdivide the domain in which we hope to find student ideas. Correspondingly we created parts of the study: recognition and comprehension were grouped together into one part, application was a part of the study, and the third part included analysis, synthesis, and evaluation.

We chose a qualitative approach because we seek to be able to describe the nature of the various understandings achieved by the students, rather than the relative frequency with which any particular understanding is obtained. We chose a phenomenographic approach because it is aimed at identifying and expressing student understandings in a way that transforms these understandings into suggestions how to help them advance their learning.

We collected data about recognition and comprehension in interviews, and in group help sessions and during tutoring, and also with a written list of questions, and by incorporating observations of computer science classes. We collected data about application on homeworks, and on practice and actual examinations. We collected data about analysis in interviews. We collected data about synthesis on homeworks, and on practice and actual examinations.

We conducted over 30 interviews. The conceptualizations of undergraduate computing students were sought. We incorporated into our design, a method of validation,



called triangulation. To check our findings, we also interviewed faculty and graduate students who had provided courses related to proof. Our interview participants were sampled from a large public research-oriented university in the northeastern United States.

Consistent with a grounded theory approach, we used interviews conducted early in the study to explore students' notions of proof, adapting to the student preference for proof by mathematical induction and incorporating the use of recursive algorithms.

We used interviews conducted later in the study to investigate questions that developed from analysis of earlier interviews.

We used exams to study errors in application of the pumping lemma for regular languages.

We used homework to observe student attempts at proofs, and to observe student familiarity/facility with different (specific) proof techniques: induction, construction, contradiction, and what students think it takes to make an argument valid. We used yet later interviews to discover whether students used proof techniques on their own, and how students ascertained whether circumstances were appropriate for the application of algorithms they knew, and how students ascertained certain properties of algorithms.

## 3.2 Parts of the Study

The parts of the study reflect the several research questions. The parts of the study are organized taking inspiration from Bloom's Taxonomy of Cognitive Domain.[32] The first part of the study was about recognition: what undergraduates think proof

TABLE 3.2.1: Parts of the Study

| Part  | Purposes   |
|---|--|
| recognition,<br>comprehension and<br>structural relevance | what proof is<br>how students approach understanding proofs<br>what proof is for   |
| application   | how students apply proofs they have been taught  |
| analysis,<br><br>synthesis and<br><br>evaluation          | use of structure<br>how is validity attained<br>comfort level with proof<br>student use of proof<br>consequences of not applying proof |

is, and comprehension: how they go about understanding them, and about structural relevance (part of phenomenography[176]): why students think proof is taught. The second part was about application: How students attempt to apply proof. The third part was about analysis, synthesis and evaluation: what students do when a situation might be well-addressed by proof. This part informed us about any structure students used as they pursued proof-related activities, and about what students thought was required for an attempted proof to be valid. This part also informed us about the comfort level students have about the use of proof, and the consequences students experience, as a result of their choices about application of proof.

These parts are summarized in Table 3.2.1.

### 3.3 Population Studied

The population of interest is undergraduate students in the computing disciplines. In a phenomenographic study, it is desirable to sample widely to obtain as broad as possible a view of the multiple ways of experiencing a phenomenon within the

population of interest, according to Marton,[176], who in turn cites Glaser and Strauss 1967[103]. We studied undergraduate students who have taken computer science courses involving proof. Typically but not always, these are students majoring in computer science. Some of these undergraduate students are dual majors, in computer science and mathematics. We interviewed graduate students emphasizing those who have been teaching assistants for courses involving proofs. We interviewed faculty who have taught courses involving proof. We have interviewed former students who have graduated from the department. We have interviewed undergraduates who transferred out of the department.

The demographics of the interviewed students is somewhat representative of the demographics of the Computer Science & Engineering department. The demographic categories used in our university are:

- male/female
- full time / part time
- non-resident alien / resident
- Hispanic Latinx / American Indian or Alaska native / Asian / Black or African American / Hawaiian or other Pacific Islander / White / Two or more races

The demographic categories of (volunteer) participants in our study were:

- male/female: 85% male
- full time / part time not requested
- non-resident alien / resident 12% non-resident alien

- The resident population, adding to 88%:
  - 2% 2 or more races, which included people with Latino, as well as African American heritage
  - 0% American Indian or Alaska natives
  - 12% Asian
  - 2% Black or African American
  - 0% Hawaiian or other Pacific Islander
  - 72% White

Every student who signed a consent form was requested to schedule an interview, from an interval including 8AM to 9PM. Every student who scheduled an interview was interviewed.

All participants were volunteers. Volunteers were sought in all computer science undergraduate classes involving proofs, and also some that did not involve proofs, so that we could sample students at different stages in their undergraduate careers.

Graduate student volunteers were also sought. Most of the graduate student interviews were among teaching assistants in courses that taught and/or used proofs. We also included faculty of courses that involved proofs. Graduate student and faculty provided another perspective that was used as triangulation, a validation method in qualitative research.

### **3.3.1 Proofs by Mathematic Induction**

This part of the study contributes to recognition and comprehension of proof, and also to synthesis, yielding insight into consequences of student use (or not) of proof

when the situation warrants.

The participants for the study of proof by mathematic induction were taking, or had recently taken, a course on Discrete Systems required of all computer science, and computer science and engineering students. Volunteers were solicited from all students attending the Discrete Systems courses. Interviews of eleven students were transcribed for this study. Participants included 2 women and 9 men. Two were international students, a third was a recent immigrant.

### **3.3.2 Purpose of Proof**

This part of the study contributes to the first part; it is about structural relevance.

Undergraduate students were sought for this study, because we wanted to know what students thought the purpose was while they were taking the undergraduate subjects.

### **3.3.3 Proofs Using the Pumping Lemma for Regular Languages**

This part of the study contributes the second part; it is about how students apply proofs they have been taught.

The participants for the study of proofs using the pumping lemma for regular languages were forty-two students, of whom thirty-four were men and eight women, forty-one traditional aged, there were three students having Latin-heritage surnames, 1/4 of the students had Asian heritage, 2 had African heritage, and 8 were international students. Each student individually took the final exam. A choice among five questions was part of the final exam; one required applying the pumping lemma.

Half the students (21/42) selected this problem. These were 17 men and 4 women. Three quarters of those (15/42) selecting the pumping lemma got it wrong. These students, who chose the pumping lemma problem and subsequently erred on it, form the population of our study.

### **3.3.4 Student Use of Proof for Applicability of Algorithms**

This part of the study contributes the third part, about student use of proof.

The students participating in this part were mainly those having internships or summer jobs. This changed the ratio of domestic to international students, such that a greater proportion were domestic students. Also, the ratio of women to men students was affected, such that a greater proportion were male students.

## **3.4 Data Collection**

Our corpus includes interview transcripts, homework, practice and real tests, and observations from individual tutoring sessions, and group help sessions. Homework, and practice and real tests, from several different classes were analyzed for proof attempts. (Incidentally, data from multiple instructors was combined, and no use of information about any specific instructor was used.) Data from individual tutoring sessions and group help sessions were also informative. Aggregations of anonymous data were used. Towards the end of data collection, the creation of new codes and categories became very slow. This expected behavior suggests an endpoint to data collection, and is called saturation.

### **3.4.1 Interviews**

An application to the Institutional Review Board was approved, for the conduct of the interviews. The protocol numbers include H13-065, H14-112 and H15-022.

The audio portion of all interviews was collected by electronic recorder and subsequently transferred to a password protected computer. From here the interviews were transcribed, and names were redacted.

#### **Student Conceptions of What Proof Is**

Interviews were solicited in class by general announcement, and by email. Interviews were conducted in person, using a voice recorder. The interview protocol is in appendix C.

Almost every student introduced and described proof by mathematic induction as experienced in their current or recent class.

### **3.4.2 Documents**

#### **Proofs Using the Pumping Lemma for Regular Languages**

The study was carried out on both real and practice exam documents. The interpretation was informed by the events that occurred in the natural conduct of lectures, help sessions and tutoring. One method of assessing whether students understood the ease of application of the pumping lemma to a language to be proved not regular was offering a choice between using the Myhill-Nerode theorem with a strong hint or using the pumping lemma. The pumping lemma problem, which could very easily have been solved by application of the Myhill-Nerode theorem, especially with the

supplied hint, was designed, when tackled with the pumping lemma, to require, for each possible segmentation, a different value of  $i$  (the number of repetitions) that would create a string outside of the language. The intent was to separate students who understood the meaning of the equation's symbols, and the equation itself, from those students engaged in a manipulation with at most superficial understanding.

### **3.4.3 Observations from Tutoring and Help Sessions**

We also compared the results from this with information obtained in tutoring and larger help sessions. These were noted down, at the conclusion of the help session or tutoring session, for incorporation into manuscripts under preparation at the time.

### **3.4.4 Other Sources**

We consulted faculty, who had experience with teaching this material, and who had experience with students who were supposed to have learned this material in prerequisites.

## **3.5 Method of Analysis**

The phenomenographic approach to analysis has been written about by Marton and Booth[176]. This method works on interview and other data, and aims to produce a set of categories with relationships among them. Moreover, these categories and relations are used to infer critical aspects, which are ideas that are critical for developing to a more advanced conceptualization from a less advanced conceptualization.



The process by which this transformation of data occurs has been further clarified by Marton and Booth[176, p.103], who have written that an analyst should apply “the principle of focusing on one aspect of the object and seeking its dimension of variation while holding other aspects frozen” is helpful.

One example of applying this principle is the analysis directed to the question of what students think about why proofs are taught in the curriculum. Using the terminology of Marton and Booth[176], “structural relevance”, we consider structural relevance to be an aspect of proof in the curriculum. Students should learn about proof for reasons that are connected with other material in the curriculum. For example, proof by mathematic induction is relevant for understanding the explanation of why context free grammars generate the languages accepted by non-deterministic pushdown automata. We focus on the idea of the students’ conceptions of why proof is taught. We look for a dimension of variation: some of the students’ ideas about why proof is taught will contain more of the reason underlying the presence of proof in the curriculum. Using this single dimension we can sequence excerpts of student interview transcripts, student utterances, according to how little or much of this reason they recognize. This exercise is provided as an example in Table 3.5.1.

Excerpts of student transcripts were selected on the basis of being related to this question. A dimension of variation emerged from the data, such that the excerpts seemed readily organized along this dimension.

---

TABLE 3.5.1: Phenomenographic Analysis of Reasons for Teaching Proof

| Category               | Representative |
|------------------------|----------------|
| Continued on next page |                |

Table 3.5.1 – continued from previous page

| Category   | Representative  |
|--|---|
| some students do not see any point to proof  | <p>They teach it to us because they were mathematicians and they like it.</p> <p>we didn't see ok why do i really have to know the proof of the theorem to do that right? We didn't see the point, because no one taught us the point, so, that's a very important part that was missing.</p> |
| some students think that it satisfies the curriculum goals, to be able to reproduce a previously taught proof, or follow a procedure to generate a proof, without being personally convinced | I was able to get a full score, but I don't understand why a proof by induction is convincing   |
| Some students do not see a relationship between a problem and approach   | When I have to prove anything, I always start with proof by mathematic induction, that was the one they taught the most.  |
| Some students are surprised to discover that there is a relation between proof by induction and recursion  | I never noticed that before, but now that you mention it, I see that they are isomorphic.   |
| Continued on next page   |   |

Table 3.5.1 – continued from previous page

| Category   | Representative  |
|--|---|
| Some students see the relationship but do not use it   | Professor (redacted) would be really proud of me that I learned to understand proof by induction quite well. ... I understand how recursion matches induction, there's a base case, there's a way of proceeding. ... I just couldn't figure out how to program the merge-sort algorithm.  |
| some students do not generalize reasons for studying proof beyond what they are shown in class | <p>I would never consider writing a proof except on an assignment.</p> <p>I understand the proof of the lower bound on comparison sort. ... I understand the proof of the upper bound on searching in a binary search tree. ... If I had to prove something about termination on a search tree, I don't know how I would do that.</p> <p>I know that recursion has the same structure as proof by mathematical induction. ... If I had an algorithm with a recursive data structure like a tree, and I had to prove something like termination about it, I'm not sure what approach I would use, it would depend.</p> |
| Continued on next page   |   |

Table 3.5.1 – continued from previous page

| Category   | Representative  |
|--|---|
| Some students see that they could employ proof to explore whether an algorithm can be expected to solve a problem in a given context that includes bounds upon resources that are available for consumption. | mostly design the algorithm first, we had some expectation of what that complexity results would be and then we try to find an approach to prove. |

Marton and Booth[176, p. 133] note that the phenomenographic method of analysis includes viewing excerpts of student utterances in specific perspectives. They advise “establish a perspective with boundaries, within which one seeks variation”, and to remember to apply perspectives “that pertaining to the individual and that pertaining to the collective”. So, when we establish a perspective with boundaries, we set a scope, allowing us to admit student text fragments relevant to that scope, filtering out other remarks. When we sequence or categorize the selected utterances, during which we will be comparing data from difference individuals, we must evaluate the utterances within the context of the interview from which they were obtained. For example, one student might be more prone to exaggeration than another. Also, one student may have more mathematical background than another.

Marton and Booth regard the learning objective as a collection of related aspects, with their relationships; we can observe that a component hierarchy can represent the aspects. Marton and Booth discuss the depth of understanding; we can observe that one consequence of depth of understanding is the development of a generalization/specialization hierarchy. Marton and Booth contrast situations with phenomena,

such that phenomena are understandings and situations serve as relatively concrete examples of phenomena, as used in instruction and assessment.

We may search for evidence of recognition of aspects; they might be mentioned by learners. Marton and Booth have observed that in different context, different aspects shift between foreground (consciousness) and background. Marton and Booth advise us to “assume that what people say is logical from their point of view” [176, p. 134], citing Smedlund [242].

Marton and Booth [176, p. 133] write that completion may be recognized by the achievement of a result, specifically the ability to identify a number of qualitatively different ways in which phenomenon has been experienced.

One approach we have taken, besides the single aspect oriented approach exemplified in Table 3.5.1, is to apply basic inductive analysis and deductive qualitative analysis, including axial coding (described later), with the phenomenographic paradigm in mind.

More specifically, when processing interview data, we transcribe the data, we transfer the transcribed and redacted<sup>1</sup> data to a website-based tool named Saturate<sup>2</sup>.

We use the Saturate application to select contiguous fragments of text the capture meaning in our judgment. Each selected fragment is labeled. These labels, which are also called codes, can be reused, thus collecting together multiple fragments, as synonymous. A process, called constant comparison, begins at this level of aggregation of the data. A code, representing the synonymous fragments, is chosen, either from among the fragments or not. The fragments sharing a code are compared with one another, to ascertain whether the group with that code is internally cohesive, to such

---

<sup>1</sup>Names of people are removed.

<sup>2</sup>The former version of what is now found at <http://www.saturateapp.com/>.

a degree that fragments in any one group are relatively distinct from fragments in other groups. A summary description (called a memo) of each code is written, and fragments are checked for compatibility with the code's description.

Data were analyzed using a modified version of thematic analysis, which is in turn a form of basic inductive analysis.[189, 191, 42, 89, 40] Using thematic analysis, we read texts, including transcripts, looked for “units of meaning”, and extracted these phrases. Deductive categorization began with defined categories, and sorted data into them. Inductive categorization “inferred” the categories, learned them, in the sense of machine learning, which is to say, the categories were determined from the data, as features and relationships found among the data suggested more and less closely related elements of the data. A check on the development of categories compared the categories with the collection of units of meaning. Each category was named by either an actual unit of meaning (obtained during open coding<sup>3</sup>) or a synonym (developed to capture the essence of the category). A memo was written to capture the summary meaning of the category.

Then, with a set of codes, we again perform grouping. This time we group codes into categories. Each category is reviewed to check whether the codes contained are relatively cohesive within a category, and relatively distinct from codes in other categories. A memo is written for each category.

Categories at this point in the analysis are also called initial themes. Next a process called axial coding, found in the literature on grounded theory, [250, 143, 101] was applied. This process considered each category in turn as a central hub; attention focused on pairwise relations between that central category with each of the others.

---

<sup>3</sup>Open coding is so-called because it occurs at a time when the analyst is the most open-minded about what the meanings being found in the data might be. [190, p. 178]

The strength and character of the posited relationship between each pair of categories was assessed. On the basis of the relationships characterized in this exercise, the categories with the strongest interesting relationships were promoted to main themes.

Attending to the phenomenographic paradigm, we seek dimensions of variation. These are delineated by the appearance of multiple categories that are usually related to each other by including more aspects (component parts) of an idea that is a learning objective. The basic inductive analysis with axial coding described above should make more evident relationships in the data that are of the nature of a dimension of variation. Thus we see that basic inductive analysis with axial coding is compatible with phenomenographic analysis, in that it can be directed towards achieving the goals of phenomenographic analysis insofar as one or more dimensions of variation can emerge.

Phenomenographic analysis proceeds beyond the identification of categories and relationships to infer critical aspects. These are differences between related categories, such that discernment by students of the ideas differentiating those two related categories are thought necessary for the students' depth of understanding to develop into the more inclusive category.

A diagram showing the main themes and their relationships, qualified by the other, subsidiary themes and the relationships between the subsidiary and main themes was prepared to present the findings. Using the process of constant comparison, the structure of these relationships was reviewed in the light of the meanings of the categories. A memo was written about each relationship in the diagram, referring to the meaning of the categories and declaring the meaning of the relationship. A narrative was written to capture the content of the diagram. Using the process of constant comparison, the narrative was reviewed to see whether it captured the sense

of the diagram. Units of meaning were compared with the narrative and their original context, to see whether the narrative seemed to capture the meaning. The products of the analysis were the narrative and the diagram.

The diagram and the memos are used to write the report, the description of context in which, should be detailed enough (thick and rich enough) that a reader can decide whether results obtained in that context are applicable to the reader's own context.

Member checking (i.e., asking for feedback on the report, from the population the report is allegedly about) of the summary report is used to estimate validity.

### **3.5.1 Analysis of Interviews**

The redacted interview data was analyzed and managed within Saturate[235]. Though a newer version than the one we use exists, the ease with which individual codes can be assigned, and collected into categories, in the former version, won our preference. The support for memos at multiple levels in Saturate is very useful. Not only do memos document the generalization, by holding the (inferred) definition of the category, they also support comparison between that definition and the member codes of the category, but there is yet more. Memos capture the audit trail, a means of supporting a claim of validity according to Merriam and Richards [190, 218]. We wove together contemporaneously collected interview data on the several research questions, and considered the relevance of material primarily related to one research question, for other research questions.

We used constant comparison, that is, we looked to higher levels of generalization, as in the process of creating a code from meanings and checked a category from codes,



and checking whether specializations of these generalizations were consistent with our data and our general sense of the students' conceptualizations. Constant comparison helped build our sense of the plausibility of our interpretation. It is a means by which we derive validity from the consistency of our interpretation with our data (see[190]).

The analysis took multiple perspectives: data from the students' points of view generated some speculative categorizations, and data from the instructors' points of view was compared with these speculations. Related literature was examined, and it also provided illumination and cross-checking of our inductions. The component model (described by Marton and Booth [176]) of the learning objectives, the lesson to be taught, was built up from student conceptualizations. This inductive approach was beneficial because, when we subsequently apply a deductive approach based upon what textbooks attempt to convey for the same lesson, the parts of the intended lesson, not found in student conceptualizations, are more obvious. Had we not bracketed off the deductive perspective, we might have been searching for, and extracted from random, insignificant words, material for populating those deductively obtained categories.

Items excerpted from interviews for analysis should be analyzed in the context of the specific interview and also in the context of the ensemble.[176].

Data were analyzed multiple ways. Both an orthodox phenomenographic analysis, and a modified thematic analysis were carried out.

### **Traditional Phenomenographic Analysis**

In the traditional phenomenographic analysis of interviews, the transcriptions are printed, and text fragments corresponding to units of meaning are cut out (as, with

scissors). These pieces are then grouped (making copies if necessary) according to a sense of similarity. During a stage in the process, categories are learned, as researchers sense of features that distinguish categories evolves. During this stage, text fragments are moved from one category to another. After this category development phase, researchers, look into each category, to recognize and describe each category. Subsequently the perspective is shifted so that relations between categories are sought. Thus the categories are arranged relative to one another, and pairwise relations, where they exist, are identified and described. This produces a graph. From the graph, critical features of the learning objective are inferred.

### **Modified Thematic Analysis**

Data were analyzed using a modified version of thematic analysis, which is in turn a form of basic inductive analysis.[189, 191, 42, 89, 40] Using thematic analysis, we read texts, including transcripts, looked for units of meaning, and extracted these phrases. Deductive categorization began with defined categories, and sorted data into them. Inductive categorization learned the categories, in the sense of machine learning, which is to say, the categories were determined from the data, as features and relationships found among the data suggested more and less closely related elements of the data. A check on the development of categories compared the categories with the collection of units of meaning. Each category was named by either an actual unit of meaning (obtained during open coding) or a synonym (developed to capture the essence of the category). A memo was written to capture the summary meaning of the category. Next we performed axial coding. A diagram showing the main themes and their relationships, qualified by the other, subsidiary themes and the relationships

between the subsidiary and main themes was prepared to present the findings. Using the process of constant comparison, the structure of these relationships was reviewed in the light of the meanings of the categories. A memo was written about each relationship in the diagram, referring to the meaning of the categories and declaring the meaning of the relationship. A narrative was written to capture the content of the diagram. Using the process of constant comparison, the narrative was reviewed to see whether it captured the sense of the diagram. Units of meaning were compared with the narrative and their original context, to see whether the narrative seemed to capture the meaning. The products of the analysis were the narrative and the diagram.

### **3.5.2 Analysis of Help Session and Tutoring**

Help sessions for Introduction to the Theory of Computation were scheduled weekly; attendance was optional. Typically six to twelve students would participate. Originally these were called help sessions, but the demographics of the attendees did not represent the enrolled students. Subsequently the name was changed to consultation sessions. This change had the desired effect, that the population attending better reflected the enrolled students. At these sessions, students would raise topics about which they had questions. Frequently the student would be requested to work at the white board, and leading questions were asked, and problems of very small size were posed, to urge the student along the right path of development of a solution. Occasionally these suggested paths were met with resistance from the students, which is to say, misunderstandings were encountered and discussed. Ideas mentioned in these discussions that were relevant to manuscripts in process at the time were noted,

anonymized, into the manuscripts.

Due to attention being focused on interacting with students in the normal course of teaching, these field notes are incomplete.

One use of such data is that they can give evidence that categories of conceptualization of proof already created in the mathematics literature can be found also in computer science students. This is similar to a deductive rather than inductive process, in that we are aware of the categories created by Harel and Sowder[118] and by Tall[254] and student utterances that seem well matched to those categories draw our attention to those categories, validating them for students of computer science.

Help session data were more complete in a sense than interview data, because help sessions normally involved a successful improvement in experience that student had of the meaning of a concept. This in turn offered validation of the newly discerned concept as a critical factor, at least for that student.

Taking the researcher only activity, dealing with transcripts, conjecturing categories, and inferring critical factors as a base for comparison, analysis of help session data analysis involves conjecturing (based on dynamic student utterances) an experience of meaning, inquiring to discover about partial and/or superficial knowledge, seeking incorrect ideas, forming a conjecture about a critical factor and posing a question to the student that calls upon, calling attention to, that missing resource. Encouraged into taking a helpful perspective, students often become aware of a gap in their knowledge, and are then ready to learn. This, when it occurs, confirms the utility of the proposed critical factor. Thus the main differences between analysis of interview transcripts and help session data is the number of participants, and availability of rapid feedback about utility.

### **3.5.3 Example: Application of Phenomenographic Analysis to What Students Think Proof is For**

The analysis for the research question “What do students think proof is for?”, which was approached as ‘Why do you think we teach proof?’ exemplifies a phenomenographic approach. One aspect of the phenomenon of proof is its utility. We set the scope of our perspective to be specific to usefulness. We selected student verbal productions related to the use of proof. We considered them in the context of their own interview, and we compared them to data from other interviews on the same theme.

We applied phenomenographic analysis by focusing on the aspect of relevance of proof for learning computer science and practicing as a software developer. In this case we had already identified the dimension of variation to be the depth of understanding of why we teach proof. Thus we could select fragments of student utterances and rank them according to depth of understanding. We then presented them in a sequence by rank.

## **3.6 Method of Creating Summary Report (for Member Checking)**

All participant data were considered. Using Marton and Booth’s[176] component model of the learning objective, we take note of what components of the idea of proof are seen to be missing in some of the conceptualizations of the students. Then we infer a conjecture about what student perspective might result in this conceptualization. Then we test the conjecture to see what results we might expect from that perspective. If the perspective predicts the student utterances we have heard, or the

student behaviors we have seen (e.g., incorrect negations), that lends confidence to our conjecture, and text to our thick and rich report.

### 3.7 Method of Addressing Validation

Our methods of addressing validation are itemized in Chapter 6, which refer back to specific sections in this chapter.

Triangulation is a technique for increasing the confidence that the results of analysis are reliable.

### 3.8 Method of Presentation of Results

The product of analysis in a phenomenographic study is a set of categories, and relationships *among* them. These categories and relationships are often depicted in a graph. This product may be accompanied by a “thick and rich” narrative description of the categories and relationships. This narrative must be consistent with the individual text fragments, excerpts from transcriptions, field notes or documents obtained for the study.

Marton and Booth[176, p. 135] state “in the late stages of analysis, our researcher [has] a sharply structured object of research, with clearly related faces, rich in meaning. She is able to bring into focus now one aspect, now another; she is able to see how they fit together like pieces of a multidimensional jigsaw puzzle; she is able to turn it around and see it against the background of the different situations that it now transcends.”

This tells us that the narrative should describe the categories of composition hierarchies found in the students' understandings. The faces or facets of the learning object have their importance and relationships as envisioned by the teacher. The students' conceptualizations may be less complete, contain superfluous items, and differ as to the relationships of the parts, especially by lacking profundity in understanding of relationships.

## Chapter 4

# Phenomenographic Analysis

In a phenomenographic study, an important purpose of analysis is to discover which points we might want to emphasize during instruction. Thus we infer categories from our data (frequently quotations from interviews), and infer the relationships between some pairs of categories; these relationships suggest elements of subject matter that when perceived by students, enable the students to deepen or otherwise extend, their understanding. These enabling ideas are called critical factors (and sometimes, critical aspects). Different research strands use these terms differently. For example, Pang and Ki[198] state:

Phenomenographic studies have brought critical aspects to prominence as specific and important features of the outcome space that distinguish qualitatively different ways of experiencing or seeing a phenomenon (Collier-Reed & Ingerman, 2013). Instead of being conceptualized as a theoretical concept, the notion of critical aspects has become more of a pragmatic construct to structure and order the outcome space. In the



variation theory of learning, in contrast, a critical aspect is more theoretically grounded, and taken as synonymous with a dimension of variation, whilst “critical features” are seen as the target values of critical aspects or dimensions of variation. It has been posited that variation in ways of experiencing a phenomenon can be accounted for by the different (critical) aspects or features to be discerned and focused upon by the experiencers (Marton & Booth, 1997). Some studies (e.g. Booth, 1997; Cope, 2000) have used the term “educationally critical aspects,” which are to do with what the teacher wants the students to learn in terms of what and how. In a learning study, however, a critical aspect is conceived of as an aspect that is critical to learners’ appropriation of the object of learning (i.e. what learners are expected to learn) (e.g. Lo, 2012). In retrospect, it seems that the term critical aspect was not clearly defined in the early theoretical papers referring to it, being used instead simply to denote an aspect (of something) that is critical (for some purpose).

This greater understanding is specifically another, better, conceptualization of the material being learned, another conceptualization found among the community of learners.

The combination of categories of conceptualizations and relationships between pairs of categories of conceptualizations is called the *outcome space*.

In this chapter we show diagrammatically, and describe, the outcome space for each research question. As the less sophisticated conceptualizations serve as a foundation for more elaborate ones, the outcome space shows these simpler conceptualizations at the bottom of the diagram.

We applied phenomenographic analysis to transcripts, field notes and documents. We addressed several research questions. The analyses are organized herein by the question addressed. There is a section for each research question. After each of the individual research questions' data are analyzed, a section is devoted to analysis of the combined data, in which they are analyzed as a single collection. Each section is organized into Categories, Illustrative Quotes for Categories, Relationships, Critical Factors, Dimension of Variation and Validation.

The analysis for the research question “What do students think proof is for?”, which was approached as “Why do you think we teach proof?” exemplifies a phenomenographic approach. One aspect of the phenomenon of proof is its utility. We selected student verbal productions related to the use of proof. We considered them in the context of their own interview, and we compared them to data from other interviews on the same theme.

The questions are ordered guided by the 1956 version of Bloom's Taxonomy, namely, recognition, comprehension, application, analysis, synthesis, evaluation.[32]

## **4.1 Phenomenographic Analysis of What Students Think Proof Is**

### **4.1.1 Categories**

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.1.1 and depicted in Figure 4.1.1.

We also used data from homework in Discrete Structures. Here we categorized incorrect proof attempts. These showed that students' conceptualizations included that

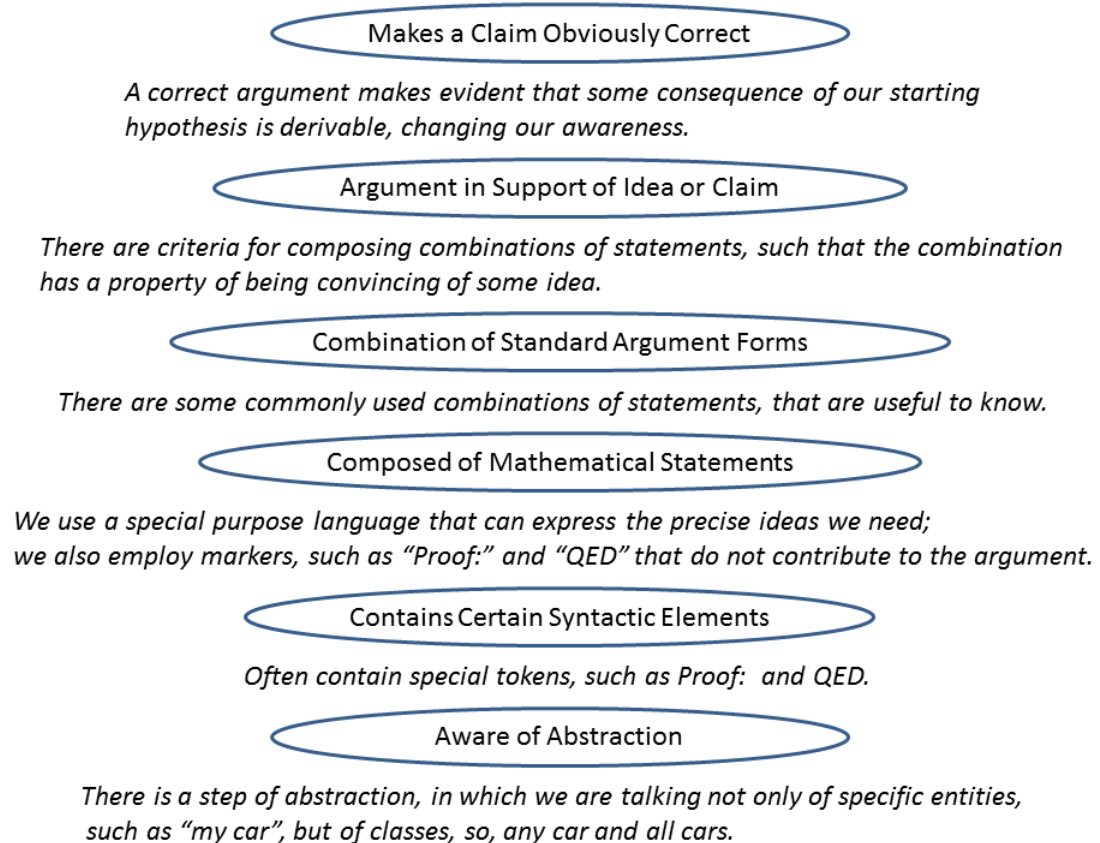


FIGURE 4.1.1: Outcome space from What Proof Is. The outcome space consists of categories of understandings, shown as ovals, and critical factors, given in italic text. Critical factors are thoughts that are necessary[177] for students, in order to develop a deeper or more inclusive understanding.

TABLE 4.1.1: Categories for Student Conceptualizations of What Proof Is

| Category                               | Description   |
|--|---|
| Makes a claim obviously correct        | That arguments can convince is recognized.  |
| Argument in support of idea or claim   | That argumentation is, or can be, expressed, is recognized.   |
| Combination of Standard Argument Forms | The existence and identity of patterns in this language are recognized.   |
| Composed of Mathematical Statements    | Productions of this grammar are considered.   |
| Contains Certain Syntactic Elements    | The existence of a specialized language and grammar for the discourse is recognized.  |
| Aware of Abstraction                   | Advancing beyond the domain of concrete objects, this conceptualization includes understanding that abstract concepts are useful. |

mathematically formulated statements be written in a sequence. However, students encountered difficulties in formulating such statements, and in creating sequences that formed a logical argument.

We then organized these data according to completeness and depth of understanding.

We formed categories for our data (see Table 4.1.1), and we inferred relationships between categories (see Table 4.1.3).

### 4.1.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.1.2.

TABLE 4.1.2: Illustrative Quotations for Student Conceptualizations of What Proof Is

| Category                                | Quote   |
|---|---|
| Makes claims                            | a logical method for determining whether something is true or not   |
| obviously correct                       | like everything, uh, like everything that there is to prove, it already exists. So proof is just like a way to get there.   |
| Argument in support of an idea or claim | like my (debating) points need to be clear and concise and they need to be connected one to the next. it is very much related to proofs   |
|   | Q: Were we studying proofs today? A: No. Q: Were we discussing certain contexts, and why certain ideas will always be true in those contexts? A: Yes. Q: Doesn't that seem like proof, then? A: Yes |
| Combinations of Standard Argument       | There are the ones that use process steps, and logic proofs, that use rules of inference.   |
| Forms                                   | Q: what made it difficult? A: probably not sufficiently understanding how the logic worked i guess, for certain techniques of proofs  |
| Continued on next page                  |   |

Table 4.1.2 – continued from previous page

| Category  | Quote   |
|---|---|
| Composed<br>of Mathe-<br>matical<br>Statements  | My favorite one is proof by mathematic induction. It has a base case and an induction step.   |
|   | i'm not too fond of induction, for whatever reason, i don't know why i think that one made the least sense when i was learning you could just say there's a base case i increment once and i guess abstracting from that, and it's true for everything it seems i don't know, it seems kind of weird, sometimes when you think about it |
| Contains<br>Certain<br>Syntactic<br>Elements    | begins with Proof: and ends with QED or $\square$   |
| Element<br>of Domain<br>of Mental<br>Constructs | I can understand it when it is about concrete entities like people I know, or models of cars, but when it is about variables, whether the names are long or short, I don't understand.  |

### 4.1.3 Relations

The relations are shown in Table 4.1.3.

Following the traditional phenomenographic method, we examine pairs of categories. We choose categories that appear to be adjacent in the space of features with which categories are distinguished from one another. This calls attention to features

whose values differ. These differing values are candidates for critical factors. We can consider whether a particular difference in feature value is important in distinguishing one category of conceptualization from another. The confidence, with which we hold that difference in feature value to be important, is the confidence, with which we feel that that difference is a critical factor.

For example, there is a conceptualization, found in the cohort of students, that in a spoken proof attempt, will produce the phrase “You know what I mean.” The aspect of argumentation that certain forms, such as mathematical formulation, can be suitable for proof, and other forms, such as “You know what I mean.” are not suitable, seems very important. We propose that “express ideas with logical statements, including mathematical formulation” is a critical factor differentiating the category “Domain of Mental Constructs” from “Composed of Statements”.

We used the relations to estimate critical factors (see Table 4.1.4).

#### 4.1.4 Critical Factors

The most fundamental critical factor discovered for this question is that there is a step of abstraction, in which we are talking not only of specific, concrete entities, such as “my car”, but instead of classes, such as all cars. Some students are working with conceptualizations in which the value of abstraction has not been recognized.

Another naive conceptualization is that proofs can be recognized by having syntactic elements including “Proof” and “QED”. Critical to advancing beyond this conceptualization is understanding that a special purpose language, mathematical formulation, is used to express with precision, properties of and relationships among abstract entities.

TABLE 4.1.3: Relationships for Student Conceptualizations of What Proof Is

| Category                                | Relationship  |
|---|---|
| Makes a a Claim Obviously Correct       | Combinations of statements are transforming one expression into another.                    |
| Argument in Support of an Idea or Claim |   |
| Argument in Support of an Idea or Claim | Combinations of statements are for legitimate reasons.                                      |
| Combination of Standard Argument Forms  |   |
| Combination of Standard Argument Forms  | Statements work together.   |
| Composed of Mathematical Statements     |   |
| Composed of Mathematical Statements     | Not all parts of the standard form are equally meaningful or useful for establishing proof. |
| Contains Certain Syntactic Elements     |   |
| Contains Certain Syntactic Elements     | A mental construct can get expressed in an appropriate form.                                |
| Aware of Abstraction                    |   |



With this language, mathematical statements can be composed, as can sequences of mathematical statements. We find in our population a conceptualization in which there is no requirement for justification of statements in a proof. This conceptualization, in which a warrant to proceed with a step that transforms the representation used in a mathematical statement is unnecessary, is found in our population.

Proofs that proceed, in this conceptualization, by following a set of instructions, can be correct. Patterns of mathematical statements are useful for building proofs. The intended result of ascertaining or convincing is not always achieved for students operating with this conceptualization.

A critical factor, that there are criteria useful combinations of mathematical statements, and insight into the nature of these criteria, has been identified. Hindering the understanding of warranting of statements is the low level of appreciation for definitions. Note the data showing that when asked what proof is, some students respond with examples rather than attempt a definition.

Once an understanding encompassing that an argument can be assembled from warranted transformation steps, the appreciation that claims can be shown to be correct can deepen.

Critical factors are listed in Table 4.1.4.

### **4.1.5 Dimensions of Variation**

Marton and Booth[176] describe a family of conceptualizations differing in degree of completeness. Conceptualizations differ according to which components of a complete understanding are included. For our standard of completeness we use Carnap's description[48], reprinted below for the convenience of the reader.

TABLE 4.1.4: Critical Factors for What Proof Is

| Critical Factor  |
|--|
| A correct argument makes evident that some consequence of our starting hypothesis is derivable, changing our awareness.  |
| There are criteria for composing combinations of statements, such that the combination has a property of being convincing of some idea.                                  |
| There are some commonly used combinations of statements, that are useful to know.  |
| We use a special purpose language that can express the precise ideas we need; we also employ markers, such as “Proof:” and “QED” that do not contribute to the argument. |
| Often contain special tokens, such as “Proof:” and “QED” .   |
| There is a step of abstraction, in which we are talking not only of specific entities, such as “my car”, but of classes, so, any car and all cars.                       |

The essential character of logical deduction, i.e. concluding from a sentence  $\mathfrak{S}_i$  a sentence  $\mathfrak{S}_j$  that is L-implied by it, consists in the fact that the content of  $\mathfrak{S}_j$  is contained in the content of  $\mathfrak{S}_i$  (because the range of  $\mathfrak{S}_i$  is contained in that of  $\mathfrak{S}_j$ ). We see thereby that logical deduction can never provide us with new knowledge about the world. In every deduction the range either enlarges or remains the same, which is to say the content either diminishes or remains the case *Content can never be increased by a purely logical procedure*. To gain factual knowledge, therefore, a non-logical procedure is always necessary. ... Though logic cannot lead us to anything new in the logical sense, it may well lead to something new in the psychological sense. Because of limitations on man’s psychological abilities, the discovery of a sentence that is L-true or of a relation of L-implication is often an important cognition.

But this cognition is not a factual one, and is not an insight into the

state of the world; rather it is a clarification of logical relations subsisting between concepts., i.e. a clarification of relations between meanings. Suppose someone knows  $\mathfrak{S}_i$  to begin with; and suppose that thereafter, by a laborious logical procedure, he finds that  $\mathfrak{S}_j$  is L-implied by  $\mathfrak{S}_i$ . Our subject may now regard  $\mathfrak{S}_j$  as known, but he may not count it as logically new: for the content of  $\mathfrak{S}_j$ , even though initially concealed, was from the beginning part of the content of  $\mathfrak{S}_i$ . Thus logical procedure, by disclosing  $\mathfrak{S}_j$  and making it known, enables practical activities to be based on  $\mathfrak{S}_j$ . Again, two L-equivalent sentences have the same range and hence the same content; consequently they are different formulations of this common logical content. However, the psychological content (the totality of associations) of one of these sentences may be entirely different from that of the other. Carnap, [48, p. 21-22]

### **Proof Viewed as a Unit**

Some students exhibit an understanding of proof at the “black box” level, i.e., there is understanding of the role of proof, without considering any internal structure. When a proof exists, we can know that the thing the proof proves is true, in the context that applies. We can “use that theorem”. Other students, though, do not have this idea consolidated yet. For example, if we consider proof by exhaustion applied to a finite set of cardinality one, we can associate to it, the idea of a test. Students, assigned to test an algorithm for approximating the sine function, knew to invoke their implementation with the value to be tested, but did not check their result, either against the range of the sine function, or by comparison with the provided sine

implementation. An example is presenting values for sine, that are over 480 million.

### **Proof Viewed as Components**

Moving to the “white box” level, we find a spectrum of variation in student understanding. The most opaque end of this spectrum has been called “magic incantation”. In this conceptualization we find those ideas of statements that are not clear, and use of mathematical symbols that is not understood.

Ellenberg[81, page 409] reports that some mathematicians regard axioms as strings of symbols without meaning, and that this quite formal conceptualization can be contrasted with another conceptualization that axioms are true statements about logical facts. He talks of these conceptualizations being taken by the same individual at different times.

In contrast, our participants seemed to regard axioms as strings of symbols that do mean something, though that meaning the participant ascribed might not be correct (especially as participants did not always know definitions of the entities being related), or the participant might feel that the meaning eludes her or him. We did find participants who appreciated the significance of definitions. Those we found were dual majors in math and computer science.

### **Proof Construction Process**

The source of such sequences of symbols served as a dimension of variation among the concepts we found in our students. Some students stressed the role of a procedure in synthesizing proofs. Proof by mathematic induction was considered preferable; a synthesis by procedure property was assigned to it. By contrast, proofs involving

sequences of statements warranted by rules of inference, but otherwise unconstrained as to form, were considered less desirable.

Some students do not see the sequence of statements as carrying out a transformation process on a representation. That is, students may see a proof as the output of an algorithm (“First produce the base case and then ...”). Students may or may not see a proof as carrying out a transformation.

## **Warrants**

Another waystation on this dimension of variation is “sequence of statements”. A more elaborate idea is “sequence of statements where each next statement is justified by what went before”.

Absence of attention to warrants has been reported by Alcock and Weber [7]. Some of our students have commented on this issue, saying they are not noticing how a pair of statements warrants a conclusion, saying that the proof steps taken by others always seem to be tricks, and not being convinced by the sequence of statements.

By contrast, there are students who clearly appreciate warrants.

Some students recognized patterns in sequences of statements. Contrapositive, contradiction, categorization into cases, proof by mathematic induction have been seen as patterns, consisting of steps that can be followed. These are contrasted with what were called “logic proofs”.

It could be difficult, for someone grading submitted work, to distinguish a correct succession of logical steps from the premise(s) to the desired consequence(s) that “reaches a psychologically useful revised formulation” from one that merely “carries out a pattern”. Indeed, the objectives of the course teaching proof may be met, while

the preparation for the course using proof to explain the nature of, say, complexity classes, might not.

A yet more complete concept is “finite sequence of statements, starting with the premise and ending with what we want to prove, and justified in each step.” A more profound conceptualization was found “finite sequence of statements, starting with axioms and premises, proceeding by logical deduction using (valid) rules of inference to what we wanted to prove, that shows us a consequence of the definitions with which we began, an exploration in which we discover the truth value of what we wanted to show, serving after its creation as an explanation of why the theorem is true”.

### **Exploration of Consequences of Definition**

Some students have a concept of the exploration purpose of proof.

A few categories, such as those above, serve to identify a dimension of variation. When our purposes include discovering which points we may want to emphasize, we can examine the categories seeking to identify how they are related and how they differ.

It can certainly be that having more categories provides more critical factors. For example, Harel and Sowder[118] offered extrinsic vs. intrinsic conviction, empirical proof schemes and their most advanced deductive proof schemes as broader categories, and seven useful subcategories of these, yielding six critical factors that suggest what teachers could usefully vary, to help learners discern items that would advance their understanding.

### 4.1.6 Validation

Triangulation is a technique useful in providing validity. In this study we applied triangulation (viewing the population from several perspectives) in several ways. We interviewed faculty teaching the courses involving proofs. We interviewed TAs assisting in the courses involving proofs. These give other views of the population, blending together the time these students are present on our campus. The students in the courses taught by these faculty and TAs are from our same population. Additional perspective on our population is gained through considering how students develop, as is done in studies making use of biographies. We added triangulation by looking at a similar population at an earlier stage of development. To get an idea of the background preparation of our students, we substitute-taught geometry and algebra II classes in a high school. The high school population was quite similar to our university population. It differed by consisting almost entirely of domestic students, studying in their first language, and by having a larger percentage of women students, and of declared transgender students. Though the community served by this high school is diverse over socio-economic status, this component of diversity is probably greater in our university population. It was helpful to see in the high school, a presentation of a proof technique that described production of this proof as a process (“First negate the consequence, then use the negated consequence as a premise ...”).

Consistency with the work of other researchers is another check on the validity of an analysis.

In this study we compared our results with those achieved by some other researchers in computer science education and also by some researchers in the mathe-

mathematics education community.

Checking possible interpretations is a technique that is thought to increase confidence in validity (see Chapter 6).

We prepared a list of questions that was addressed by several faculty and several students, that began an examination of the role of specific representation styles (mathematical notation, figures and pseudocode) for proof related problem statements.

Member checking is a technique recommended for obtaining validity. We used member checking of the summary report for this purpose.

#### **4.1.7 Outcome Space**

The outcome space from the research question “What do students think proof is?” has as its foundation conceptualization that mental constructs are the domain of proof. Because some students founder at the transition between thinking about specific concrete entities and about classes of entities, it is worthwhile to include this conceptualization as a milestone. The step from concrete entities to mental constructs, such as abstractions, and definitions, recurs in the several research questions, as does the type of memory consolidation that is generalization. The critical factor needed by students so that they may enjoy the more advanced conceptualizations is that abstraction is useful.

The conceptualization of “aware of abstraction” appears in our student population. The benefit of a specialized language for expressing ideas about abstractions is not seen by all of our students, even though the benefit of specialized languages for algorithms and programs is well-known. Even though we use pseudo-code, as well as languages with compilers or interpreters, not all students organize these ideas into



a mental hierarchy of meta-languages. Not all students generalize from the idea of specialized languages for computing to the notion of specialized languages for dealing with concepts. Not all students populate the general category of specialized languages for concepts with a specific member of mathematical formulation. Similarly, although students are familiar with styles of providing computer code with comments, there is found a conceptualization that tokens such as “Proof” and “QED” actively participate in the meaning of the proof. This conceptualization is named “contains certain syntactic elements”.

A critical factor has been identified, to help students transition from using the “contains certain syntactic elements” conceptualization to using the “composed of mathematical statements” conceptualization. This critical factor is that we use a special purpose language that can express the ideas (which are precise) that we need; we also employ the markers such as “Proof” and “QED” that help people see the parts of the proof, and do not contribute to the argument the proof is making.

The conceptualization of “composed of mathematical statements” exists in our student population, as students have taken courses in mathematics, including discrete structures, calculus and algebra. However the conceptualizations of mathematical statements include some understandings that are collections of examples that serve for recognition and some understandings that support comprehension, and some understandings that provide for synthesis. Implications are a type of statement that illustrate the conceptualization: algorithms include conditional statements. Students are familiar with conditional statements from algorithms. Some students sometimes operate with a conceptualization in which implications, as if they were program constructs, are always true. In this conceptualization, disproving an implication is a confusing idea.

There is a critical factor making available the conceptualization “combination of standard argument forms”, namely that there are some commonly used combinations of statements, that are useful to know.

The conceptualization of “combinations of standard argument forms” is present in our student population. This is analogous to programming; we have seen that analogies are not always noted and exploited by the students. This conceptualization, which includes that proof can be obtained by following process steps, such as “First construct a base case and prove it, then construct an induction step and prove it.”, is found in our population. Such constructions do not always convince the person who has learned the process steps.

The conceptualization that an argument is what is being constructed, an “argument in support of an idea or claim”, with these compositions of standard forms, is present in our population. Some students are dismayed because they know a process for constructing a type of proof, and they believe correctly that the argument thus constructed should be convincing, but they do not follow the reasoning, and they do not find the construction persuasive.

Appreciating the subtlety of warranted combinations of statements forming convincing arguments is a more profound conceptualization. The critical factor that enables students to develop this deeper understanding is that there are criteria for composing combinations of statements, such that the combination has a property of being convincing of some idea.

Helping students discern what it is that makes a combination of statements compel belief in something is valuable. Helping students see that it is logical deduction that compels belief, and that there are rules for proceeding logically, and there are techniques for suggesting which rules of inference may be helpful for progress is part

of this critical factor.

This idea for this critical factor can also be seen as raising the consciousness of the students that when they combine argument forms, they are constructing an argument using reasoning; they must justify the use of a statement on the basis of previous statements and axioms and definitions. From this conceptualization, which some students in our population have, an instructor can emphasize that students engaged in proof are building an unequivocal demonstration that something (the conclusion) is derivable from something else (the premises), from which the conclusion is indisputably derivable.

When they have this idea, they are operating with the conceptualization that a proof “makes a claim obviously correct”.

TABLE 4.2.1: Categories for Student Conceptualizations of How to Understand Proofs

| Category  | Description   |
|---|---|
| Appreciate generalization / specialization hierarchy of definitions | see the relationships among a collection of related definitions, as also seen in object-oriented languages, in an inheritance hierarchy |
| Look up the definitions and use them (Math major)                   | adopt (warranted) logical deduction based on definitions  |
| Use different representations                                       | comprehend idea in a different representation   |
| Use dynamics  | pay attention to change and its effect  |
| Use visualization   | engage visual processing, make analogy with spatial relationships   |
| Go over all axioms and theorems from class                          | infer from the examples, build on examples  |
| Study Proof Patterns from Class                                     | seen mathematic induction most often, so try that   |
| Study the Proof from Class  | superficial, pattern matching   |
| Emotional rather than intellectual response                         | Anxiety or other negative emotion   |

## 4.2 Phenomenographic Analysis of How Students Attempt to Understand Proofs

### 4.2.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.2.1 and depicted in Figure 4.2.1.

The arrangement of the categories follows that of the table, and is shown in Figure 4.2.1.

The first category of approach to comprehending a proof is to check whether it is one they have already examined in class. The next category is to check whether the proof follows a pattern that has been treated in class. The most important difference

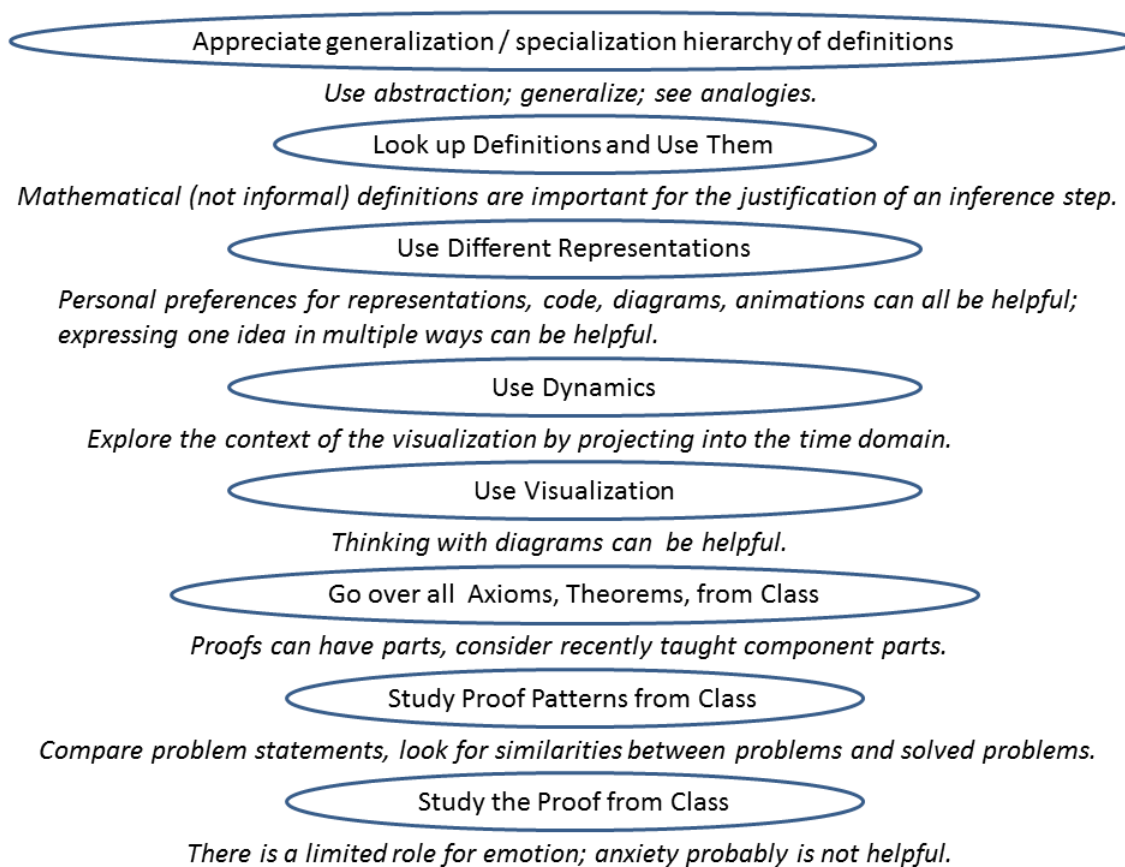


FIGURE 4.2.1: Outcome space from How Students Approach Comprehending Proof

seems to be that generalization from an instance to a pattern occurs.

The next category after a pattern that has been discussed in class seems to be to engage the visual domain. It is not clear that students view the argument as a process by which one representation of a truth gets transformed into another representation, that renders the claim obvious. Thus, it is not clear what the students are attempting to visualize.

Ideas that would have been welcome but did not appear include transformation of statements, lemmas, the question “Does problem statement suggest anything over which we might induct?”.

### 4.2.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.2.2. It could be that some students are not attempting to understand proofs. Students can experience anxiety about mathematical notation. Some students are attempting to understand proofs while not recognizing that they are studying a proof. Some students read proofs. Some students look up the definitions of terms used in the proofs and some do not. Some students think (or hope) they can solve problems involving producing proofs, without knowing the definitions of the terms used to pose the problem. Some students are aware that definitions are given, but “zone out” until examples are given. Some students think that the reading of proofs is normally conducted at the same speed as other reading, such as informal sources of information.

Of students who read proofs attentively, some try to determine what rule of inference was used in moving from one statement to the next, and some do not.

Some students experience transient understanding of proof techniques.

Some students notice that lemmas can be proved and then used as building blocks in a larger argument, and some do not.

Some students can identify the forms of proof learned in discrete systems, when they see them employed in proofs, such as the combination of arguing by contrapositive and modus ponens. Some cannot.

Some students can identify these forms in an argument if the argument is made about concrete objects, such as cars or specific people. Some of these students have difficulty transferring this ability with concrete objects to application to abstract entities such as sets, algorithms or symbols. Some students who achieve with difficulty the ability to recognize the application of rules of inference in one argument about abstract entities, become quicker at recognizing arguments of similar form about other abstract entities, and some continue to achieve with difficulty, as if learning the first argument did not facilitate learning the second argument of the same form.

Students would attend to diagrammatic representation of proofs, such as a block diagram depicting machine descriptions packaged as input for yet other machines to process, but were not observed to employ such diagrams.

Students have been seen to employ decision tree diagrams.

Students would attend to algorithm representation of proofs, such as a recursive process that determines a prime factorization, but were not observed to employ such algorithmic descriptions.

Students do not always attempt to understand proofs they are shown. For those who do try, some students do not succeed.

When students attempt to understand proofs, they sometimes get stuck. They reported preferring a representation in code, which they could exercise in a development system. They did not know of a similar system, such as ProofGeneral[17] that

would help them tinker with or otherwise examine a proof.

It appears that, as students attempt to understand specific proofs, they try to find an example in which the symbols refer to concrete objects. In a proof about primes, some students will substitute specific primes to “check the idea”.

Some students have had trouble transferring their understanding of proof patterns applied in one domain to proof patterns applied in a similar domain. In particular, some students who had studied proof by mathematic induction with natural numbers in one semester, when faced with understanding a proof by mathematic induction with natural numbers designating the level of a pushdown stack in a subsequent semester, claimed not to have understood proof by mathematic induction. The yet more general idea of structural induction, implicit in tree data structures, might elude them in the data structures course. These students did not detect the problem with the traditional “all women are blue-eyed blonds” argument.

Students answering a list of questions, representing computer science ideas mathematically, in algorithms and in figures found the questions “interesting”, “fun”, “different” and “non-trivial”.

Except when assigned to do so, students were not observed to attempt to solve simpler problems, such as by imposing partitioning into cases. Except when assigned to do so, students were not observed to attempt to solve more general problems, as is sometimes helpful[81].

When examples are given, the students attempt to infer definitions themselves. Some students will compare the definitions they infer with the mathematical community’s definitions. Some students do not, continuing to use what they have constructed.



TABLE 4.2.2: Illustrative Quotations for How Students Approach Comprehending Proof

| Category   | Quote  |
|--|--|
| Appreciate<br>a gener-<br>alization<br>/<br>special-<br>ization<br>hierarchy of<br>definitions | i think the thing a lot of people hadn't really had to deal with before was just the level of abstraction that comes with proof writing, which is inherent with computer science, but a lot of time when we talked about problems it's always through analogies, i mean the traveling salesman problem is about cities and moving but that's not really what it's about, it's about graphs and paths |
| Look up the<br>definitions<br>and use<br>them (Math<br>major)                                  | you have to understand all the things that are being accepted as true if the proof relies on them, then logically follow the proof; you have to accept all the rules that the writer of the proof has accepted   |
| Use<br>different<br>representa-<br>tions   | <p>a lot of it has to do with fluency, because we don't speak math ...we don't actually use the math language to accomplish anything, whereas the code actually accomplishes something</p> <p>Q:I wonder whether if proofs like mathematical formulations, could be rewritten as algorithms would the computer science students find them more readily understood. A: absolutely</p>                 |
| Use<br>dynamics  | that makes a little more sense than some of the assertions, the equalities, an algorithm you can <i>trace through</i> , you can write it out, things like that, it's very beneficial   |
| Continued on next page   |  |

Table 4.2.2 – continued from previous page

| Category   | Quote   |
|--|---|
|  | in pseudocode or in a formula, i would like it's more of an algorithm, you can actually go through and <i>run it in your head</i> at least, uh, so you have a method for obtaining results,   |
| Use a diagram, visualization   | visual proofs were just always easier even to this day, i find that things that i can visualize i tend to do a lot better with. the visual aspect made it a lot easier. I understand how visual proofs are not complete, ... a visual explanation only shows one particular situation you know whereas in proofs you have to go for all of them but you know sort of visualizing it has always made it easier at least for me |
|  | being able to visualize it is definitely a key for me and for other people as well, so I don't know how you could truly understand that sort of thing without, <i>it's an abstraction, man this is really hard to say</i>   |
| Go over all the logical elements from Class, related axioms and theorems | the biggest thing that changed in my proof writing in the math side I didn't really have a good understanding of logical statements, like an if and only if   |
| Continued on next page   |   |

Table 4.2.2 – continued from previous page

| Category                                    | Quote  |
|---|--|
| Apply the Proof                             | considering it's algorithms, it's usually a certain type of proof that you can recognize so it's fine  |
| Pattern from Class                          | aha moment have always been proofs written for induction, despite the fact that I've done them multiple times, they go over my head and I have to relearn proofs by induction. |
| Just Like the Examples from Class           | leaving it at the formal definition is kind of aaahh, I kind of work backwards with those, like I get an example, then ok this relates to this step that's what this means.    |
|   | after understanding a proof, I try a few examples to check the idea  |
| Emotional rather than intellectual response | the second I see a summation, I'm like oh (profanity) this is some really long thing, . . . when you look at that little squiggly (profanity)*6.                               |

### 4.2.3 Relations

The relations are shown in Table 4.2.3. Relations are inferred by comparing two conceptualizations that are adjacent. Then, they are further processed to obtain critical factors. The first relation arises from the comparison of a single proof with a pattern for a proof, which implies similarities among claims to be demonstrated.

The second relation is an extension of source material for understanding proof, going beyond patterns to application of theorems and axioms.

The third relation broadens the representation type for the material consulted, augmenting theorems and axioms with diagrams.

The next relation extends diagrams in the time dimension.

The next relation further extends representations to incorporate code.

The next component brought to bear in understanding is noticing warrants from definitions.

The most sophisticated relation is that many interrelated (defined) terms exist.

While the relations are in the form of an inclusion hierarchy because descriptions from students give them in order, there does not seem to be an inherent need for diagrams or algorithms.

#### **4.2.4 Critical Factors**

The critical factors are shown in Table 4.2.4.

Critical factors are derived from relations. Critical factors are items to be discerned, and are recommended to teachers for exposition with contrast, generalization, separation, and fusion. Therefore the derivation process is to turn a relation into a learning objective.

To help students discern that a proof is general and can be used as a pattern, we might present contrasting concrete situations that can be generalized by using mathematical formulation. This suggests as a critical factor the idea that problem statements can have similarities, to the extent of sharing a solution with a solved problem.

To help students discern the larger proofs can be built from smaller proofs, and from axioms separation and fusion seem appropriate. This suggests proofs having

TABLE 4.2.3: Relationships for Student Conceptualizations of How to Understand Proofs

| Category  | Relation  |
|---|---|
| Appreciate generalization / specialization hierarchy of definitions | sets of terms related by generalization / specialization make clear the properties that distinguish the terms from each other |
| Look up the definitions and use them (Math major)                   |   |
| Look up the definitions and use them (Math major)                   | Recognize that definitions warrant transformations.   |
| Use different representations                                       |   |
| Use different representations                                       | consider whether algorithmic or other representation might supplement the mathematical formulation                            |
| Use dynamics  |   |
| Use dynamics  | consider whether animation might help   |
| Use visualization   |   |
| Use visualization   | Consider whether a diagram might help.  |
| Go over all axioms, theorems from class                             |   |
| Go over all axioms, theorems from class                             | Consider recently taught component parts of proofs.   |
| Study proof patterns from class                                     |   |
| Study proof patterns from class                                     | Notice similarities among problems solved by the same pattern.  |
| Study the proof from class  |   |

components as a critical factor.

Students report enlisting the aid of diagrams to help them see what the claim, and the proof, is about. Thus, a critical factor is that relationships among parts of a proof, and ideas dealt with in the proof, can be illustrated in diagrams.

Students report animating diagrams; a critical factor is showing that entities relevant to a proof can change with time.

Some students describe animation with code, bringing to bear another strength. Thus a critical factor is use of multiple representations.

Some of our students delay attention to definitions until after these earlier tools for comprehension have been tried; nevertheless definitions are necessary as well as important, so they are a critical factor.

Understanding the relationships among related mathematical definitions is a more advanced critical factor than understanding that individual definitions must be used.

TABLE 4.2.4: Critical Factors for How Students Approach Comprehending Proof

| Critical Factor   |
|---|
| Use abstraction; generalize; see analogies  |
| Mathematical (not informal) definitions are important for the justification of an inference step.   |
| Personal preferences for representations, code, diagrams, animations can all be helpful; expressing one idea in multiple ways can be helpful. |
| Explore the context of the visualization by projecting into the time domain.  |
| Thinking with diagrams can be helpful.  |
| Proofs can have parts, consider recently taught component parts.  |
| Compare problem statements, look for similarities between problems and solved problems.   |

### 4.2.5 Dimensions of Variation

The dimension of variation found begins with the idea that inference from examples is sometimes used to anchor the effort at understanding. These examples can be of concepts discussed in class. First, the instance is examined to check whether it is the same as what was done in class, and if not, then, an example of a known pattern. Second modifications of known patterns are tried to see whether the instance can be so constructed. Failing to match such a pattern is followed by switching from categorization to investigation of behavior. Visualization of a transformation is attempted. Dynamics, as in tracing through program execution is helpful for understanding, to those who can see the sequence of transformations. Some relatively sophisticated students will avail themselves of definitions, to examine whether transformation steps appear warranted. More than one student appreciated that a collection of definitions in a specialization/generalization hierarchy, and reasoning by analogy, which exploits the network of relationships found in such a hierarchy, was a powerful tool for understanding.

### 4.2.6 Validation

Consistency with existing literature is helpful here, as there are adjacent categories in Harel and Sowder's[118] categorization, namely internalization and interiorization, that are useful in the pattern matching part, and another pair of adjacent categories in the same source, perceptual and transformational. Like students of mathematics, our students use visualization, and apply dynamics; in contrast with students of mathematics, for our students the dynamics emphasizes switching to a programming representation, rather than involving diagrams that evolve.

### 4.2.7 Outcome Space

The outcome space from the research question “How do students attempt to understand proofs?” has as its foundation the conceptualization that one can begin to learn about comprehending proofs by studying how one proof works, as demonstrated in class.

The conceptualization of “Study the proof from class” is important to note, because it is a significant improvement over the pre-conceptual anxious emotional reaction that has been observed to precede understanding a first proof. The value of comprehension of an individual proof can be somewhat reduced for students exercising the idea that once a statement has been proved, the proof itself is no longer interesting, it is “over”. The value of comprehension of an individual proof can be somewhat reduced for students exercising the idea that the purpose of the study of the proof is to learn the truth of the example consequence, rather than to learn the proof technique itself.

A critical factor that enables students to make use of a more advanced conceptualization is that a specific proof may be a representative of a reusable proof technique.

The conceptualization “Study proof patterns from class” extends the idea that a proof technique is exactly the proof shown in class to the idea that patterns for proofs are available, that is, that one may customize a general proof technique to fit the proposition to be shown. In this conceptualization, a student faced with understanding an individual proof might fruitfully compare known patterns to see whether any pattern seems to match the proof the student is trying to understand.

A critical factor that enables students to make use of a more advanced conceptualization is that proofs, once they are large enough, can have parts.



The conceptualization “Go over all axioms and theorems from class” incorporates the aspect that proofs need not be monolithic; they can have parts. This idea deepens the understanding of the ability to customize a proof pattern, to include making use of axioms and theorems that have been recognized by the classroom community as component elements of a proof.

A critical factor that enables students to make use of a more advanced conceptualization is that insight into transformations expressed mathematically can be gained by the judicious use of diagrams.

The conceptualization “Use visualization” extends the technique of comprehension to supplementing the formal expression of elements of the proof by recruiting other means of understanding, namely incorporating insights from processing visual representations.

A critical factor that enables students to make use of a more advanced conceptualization is that while diagrams might appear as a single event, they also can represent a snapshot in time. One diagram can be thought of as embedded into a sequence of diagrams in which relationships might evolve.

The conceptualization “Use dynamics” adds a dimension of change with time.

A critical factor that enables students to make use of a more advanced conceptualization is that multiple representations, beyond dynamic diagrams and mathematical formulation, exist. One student might gain more insight from imagining an algorithmic expression of a transformation under consideration.

The conceptualization “Use different representations” extends mathematical formulation and dynamic diagrams to additional representations. It invites the students to recognize similarities with understanding algorithms and programs, with which they may have some familiarity and comfort.

A critical factor that enables students to make use of a more advanced conceptualization is that definitions as agreed within the mathematical community are critical for warranting transformation steps in a proof. Students who prefer to develop their own informal definitions, and hope to work with those, can be at a significant disadvantage.

The conceptualization “Look up definitions and use them” extends techniques offering insight into the transformations occurring in a proof to students attempting to comprehend it. The precision of understanding with agreed definitions can be a significant advance over that achievable by students using their own informally acquired approximate definitions.

A critical factor that enables students to make use of a more advanced conceptualization is that definitions are not isolated, rather, they reside in relationships with other terms, in possibly multiple specialization / generalization hierarchies. When some concepts are lodged in a structure of relationships with each other, it is sometimes possible to find another set of concepts that share the relationship. Reasoning by analogy involves exploiting an understanding of relationships among one set of concepts, to aid an understanding of relationships among another set of concepts. Students have requested the use of analogies in explanations.

The conceptualization “Appreciate generalization / specialization hierarchy of definitions” makes use of this care in the use of definitions, and allows reasoning by analogy.

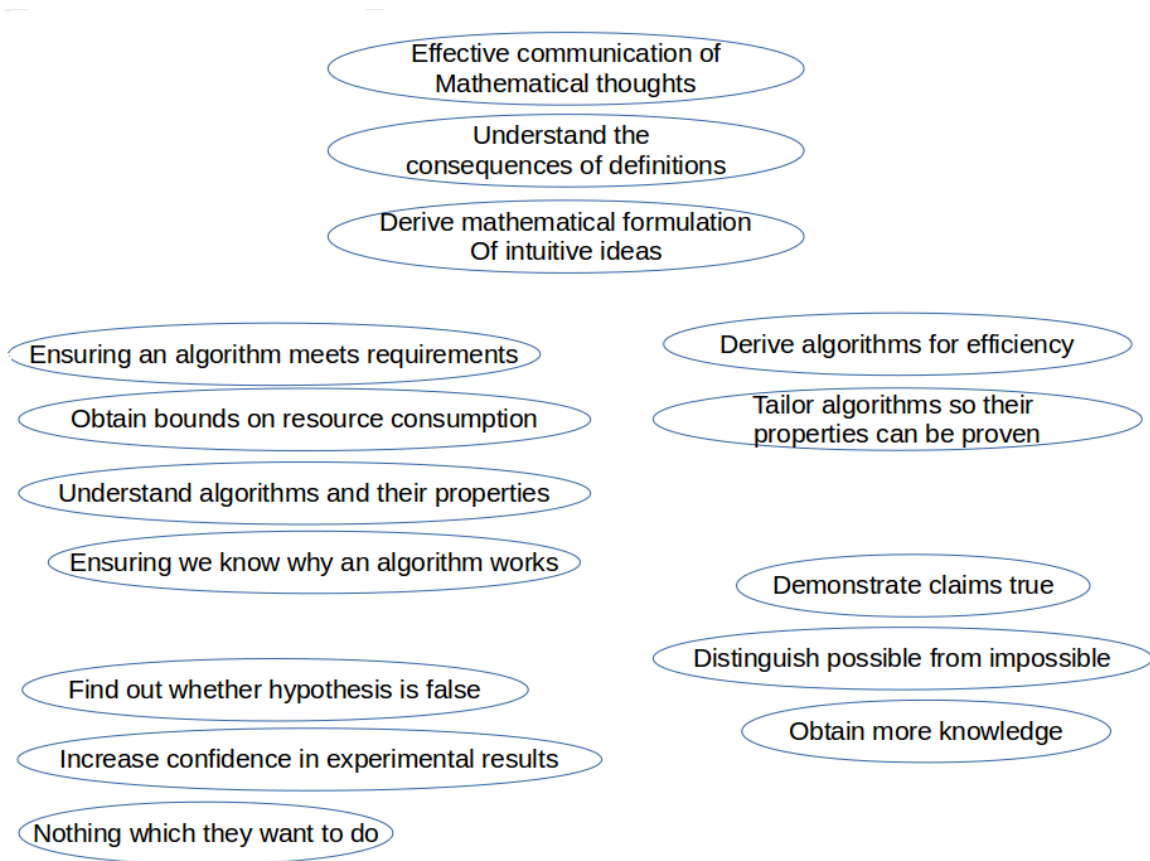


FIGURE 4.3.1: Initial Categories from What students think a proof is for

## 4.3 Phenomenographic Analysis of What Students Think Proof is For

### 4.3.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.3.1 and depicted in Figure 4.3.1. Groups of categories are listed in Table 4.3.2 and depicted in Figure 4.3.2.

Because there is an unusually large number of categories in response to this research question, the categories have been grouped. The least sophisticated group of

TABLE 4.3.1: Categories for Student Conceptualizations of  
Reasons for Teaching Proof

| Category   | Description  |
|--|--|
| Effective Communication of Mathematical Thoughts         | Mathematical maturity  |
| Understand the consequences of definition                |  |
| Derive mathematical formulation of intuitive ideas       |  |
| Derive algorithms for efficiency                         | proofs help us design; we can see the efficiency and correctness                   |
| Tailor an algorithm so that its properties can be proven |  |
| Show that an algorithm meets requirements                | proofs support our claims for our algorithms                                       |
| Establish bounds on resource utilization                 |  |
| Understanding Algorithms and Their Properties            |  |
| Ensuring we know why an algorithm works                  |  |
| Demonstrate claims (conclusively)                        | appreciates proof of existential and possibly universal statements                 |
| Distinguish the possible from the impossible             | partial truth  |
| Obtain more knowledge                                    | psychological vs. inherent   |
| Find out whether hypothesis is false                     | confusion about terms, claim vs. hypothesis or confusion about nature of deduction |
| Increase confidence in experimental results              | confusion about universal unequivocal nature of argument                           |
| Do not know why  | “we do not accomplish anything”  |
| Nothing desirable  | insufficient structural relevance  |
| Nothing of relevance                                     | don’t see the general nature of results for numbers, graphs                        |

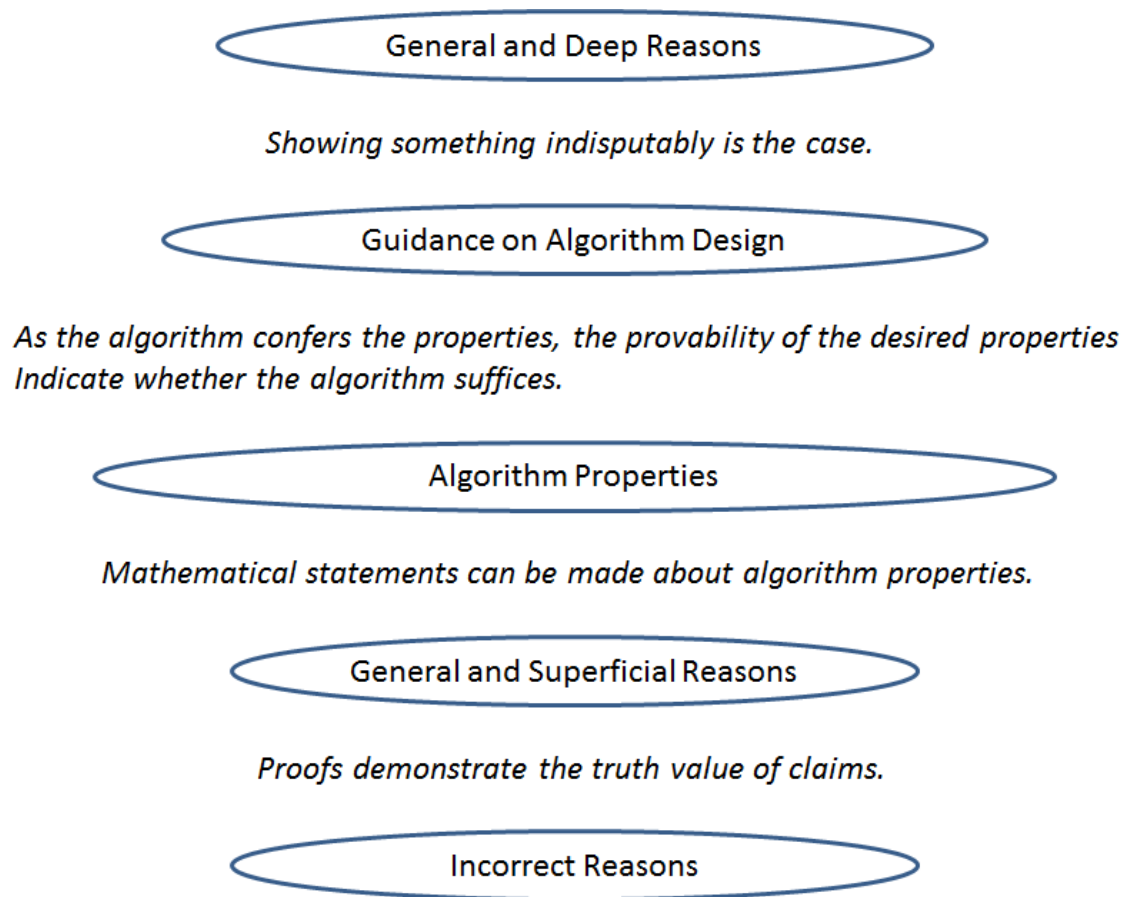


FIGURE 4.3.2: Outcome space from What students think a proof is for

TABLE 4.3.2: Grouped Categories for Student Conceptualizations of Reasons for Teaching Proof

| Grouped Category                | Description  |
|---------------------------------|--|
| General and Deep Reasons        | non-specific to computer science                         |
| Guidance on Algorithm Design    | Proof is for tailoring our algorithm creation.           |
| Algorithm Properties            | Proof is for knowing our algorithm properties assuredly. |
| General and Superficial Reasons | non-specific to computer science                         |
| Incorrect Reasons               | unguided speculation                                     |

categories includes misunderstandings, not just incomplete, but containing misinformation. The critical aspect differentiating this group of categories from the next group is the idea that proof establishes, unequivocally, the truth value of the claim. The context of assumptions in which the claim is made is included in the understanding of the claim, in so far as that context has supplied warrants for the argument.

The next group of categories are true conceptualizations that are not very detailed, such as, a proof demonstrates that a claim is true.

### 4.3.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.3.3.

Students offered many reasons why proof is taught, and it seemed interesting to capture the variety. However, the large amount of data made it seem helpful to take two steps in grouping. We called the first step finding categories, but then we dealt with groupings of these categories.

The lowest level of conceptualization contains no reason why proofs are taught, because students propose such reasons as “The professors were math majors, and feel we should know some math.”

Some students know that proofs are used for justifying claims about the correctness and resource consumption of algorithms. No student was found who considered provability to be a guide for algorithm construction, although one was found who was required to use proof in publication, and consequently used proof to know when she was finished with her algorithms.

Other students connect learning proof by induction with explaining why algorithms work: “yes, of course, the first thing i thought of when i saw induction, was

TABLE 4.3.3: Illustrative Quotations for What Students Think Proof is For

| Category                        | Quote   |
|---------------------------------|---|
| General and Deep Reasons        | making sure you understand the principles   |
|                                 | seeing why things are the way they are, and showing indisputably that something is the case                     |
| Guidance on Algorithm Design    | If the proof of part of the algorithm is too long or hard, I can change the algorithm to make the proof easier. |
| Algorithm Properties            | using proof to as a way to explain the algorithm and prove its validity   |
|                                 | when we're deciding how to make programs efficient, make them do what we want, proofs could help us             |
| General and Superficial Reasons | It is a required part of a paper for the conferences at which I try to publish.                                 |
| Incorrect Reasons               | The professors were math majors, and feel we should know some math.   |

recursion”.

Some students were asked whether they ever employed proof for any reason other than having been assigned. No student said they employed proof for any purpose other than responding to an assignment to provide a proof. When students said they never used recursion because they never knew whether a situation warranted the use of recursion, the interviewer suggested that proof might be a useful tool for understanding the situation. Students acquiesced. Given the difficulty with transfer, it appears that students do not choose to exercise this transfer.

### 4.3.3 Relations

The relations are shown in Table 4.3.4.

Some students give incorrect reasons for studying proofs. The next step is appreciating that demonstrating the truth value of claims is valuable.

This idea becomes more specific when it is realized that claims about algorithm

TABLE 4.3.4: Relationships for Student Conceptualizations of Reasons for Teaching Proof

| Grouped Category                | Relation  |
|---------------------------------|---|
| General and Deep Reasons        | Perhaps generalization  |
| Guidance on Algorithm Design    |   |
| Guidance on Algorithm Design    | As the algorithm confers the properties, the provability of the desired properties indicate whether the algorithm suffices. |
| Algorithm Properties            |   |
| Algorithm Properties            |   |
| General and Superficial Reasons | Mathematical statements can be made about algorithm properties.   |
| General and Superficial Reasons |   |
| General and Superficial Reasons |   |
| Incorrect Reasons               | Demonstrating the truth value of claims.  |
|                                 |   |

properties can be addressed.

With further progress, students consider the possibility that some programs, though they may satisfy the compiler, may not satisfy functionality requirements, or even computational complexity requirements, and that proofs can address claims of this nature.

Subsequently students gain in their appreciation for the utility of proof, for demonstrating correctness and other qualities, more thoroughly than tests can do.

#### 4.3.4 Critical Factors

The critical factors are shown in Table 4.3.5.

Some students do not discern on their own that it is the proof technique that is being taught, rather than the truth of the theorem. The critical factor is, that the subject is the proof technique. To help these students, varying the attempt to show the truth value of a claim, by contrasting inadequate proof-attempts with successful



proofs calls more attention to the technique than to the theorem being proved.

The next critical factor is that at some point in their development the students will be able to apply proof techniques for the purpose of demonstrating that algorithms have certain properties. To emphasize that properties of algorithms are relevant, different specific properties can be presented, and the idea of properties generalized. It is possible that multiple properties can be fused, and the proofs of each likewise fused.

The idea of algorithm properties satisfying requirements, and as a result, algorithms and their implementations satisfying requirements, is a generalization of the idea of individual properties being sufficient.

There is a further generalization beyond algorithms, for example, products.

TABLE 4.3.5: Critical Factors for What Students Think Proof is For

| Critical Factor  |
|--|
| Showing something (a generalization from algorithm properties) is indisputably the case.   |
| As the algorithm confers the properties, the provability of the desired properties indicates whether the algorithm suffices.   |
| Mathematical statements can be made about algorithm properties.  |
| In class have seen proofs being used to demonstrate the truth value of claims. As some conceptualizations hold that once a claim's truth has been demonstrated (a once-only activity), the proof technique is no longer of interest, it can help to foster the realization that similar techniques can solve similar problems. |

### 4.3.5 Dimensions of Variation

Students' conceptualizations build on a superficial and general understanding of what proof is for, deepening in understanding as they study examples of proofs applied to algorithms. In particular, they see proofs applied to algorithms in the Algorithms

class. Here properties of algorithms such as resource utilization are shown with proofs. Students report not having to synthesize these proofs, rather, to understand them. Later, in the Introduction to the Theory of Computing, proofs are seen by students to be used on abstract classes of algorithms. Here students report more trouble understanding and applying proofs.

An alternative dimension of variation could be developing deeper understanding of the nature of computation. Engaging in proof gives students the opportunity to contemplate the operations they carry out in algorithm construction with respect to how the complexity of an algorithm may change as aspects of the computation change, as from 2-SAT to 3-SAT.

#### **4.3.6 Validation**

Here validation is by internal consistency. Excerpts of student transcripts were selected on the basis of being related to this question. A dimension of variation emerged from the data, such that the excerpts seemed readily organized along this dimension.

#### **4.3.7 Outcome Space**

The outcome space from the research question “What do students think proof is for?” begins with unguided speculation on the part of the students, that arrives an incorrect idea. It is important to notice that there are some students who are unaware, at least at the beginning of the time they are taking discrete structures, that the material is highly relevant to subsequent courses.

A critical factor for students to realize that there are at least general reasons for proof is to recognize a relationship between a problem being solved and a proof that

is part of the solution.

The conceptualization of “General but Superficial Reasons” is a qualitative advance in understanding, compared to the starting conceptualization of some students. There is a reason for studying proof, yet to be determined.

A critical factor for students to advance to the next level of understanding is that proofs are relevant to computer science.

The conceptualization of “Algorithm Properties” provides specific reasons for studying proof. Students know, to some degree, that they are concerned with the properties of algorithms. That some of these properties can be proved, and that statements about algorithms can be made with assurance are ideas conferred by this conceptualization.

A critical factor for students to advance to the next level of understanding of what proofs are for, in the computer science curriculum, is that provability of algorithm properties can provide guidance to algorithm design.

The conceptualization of “Guidance on Algorithm Design” includes an understanding that design of algorithms might involve meeting requirements, and that algorithms can be judged according to how well they meet requirements, and that in some cases, satisfaction of requirements by an algorithm can be guaranteed by proof.

TABLE 4.4.1: Categories for How Students Attempt to Apply Proofs (When Assigned)

| Category                | Description   |
|-------------------------|---|
| Improve Efficiency      | develop a sense of which transformations are helpful in the circumstances |
| Attempt Transformations | consider different representations  |
| Adapt Known Proofs      | start with class notes and modify   |
| Use Known Proofs        | copy from class notes   |

## 4.4 Phenomenographic Analysis of How Students Attempt to Apply Proofs (When Assigned)

### 4.4.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.4.1 and depicted in Figure 4.4.1.

### 4.4.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.4.2.

This data comes from analysis of tests and interviews of students and teaching assistants.

Students claim to apply proof by mathematic induction without reference to the problem statement, citing reasons including “They taught us that the most.” and “I know I can carry out that process, it sort of checks itself.”

Teaching assistants state that proofs from class notes are provided as answers even when they do not match the problem that is posed.

Students sometimes attempt to adapt a pattern, but in an incorrect fashion. For example, a proof by contrapositive is blended with the notion of proof by contradic-

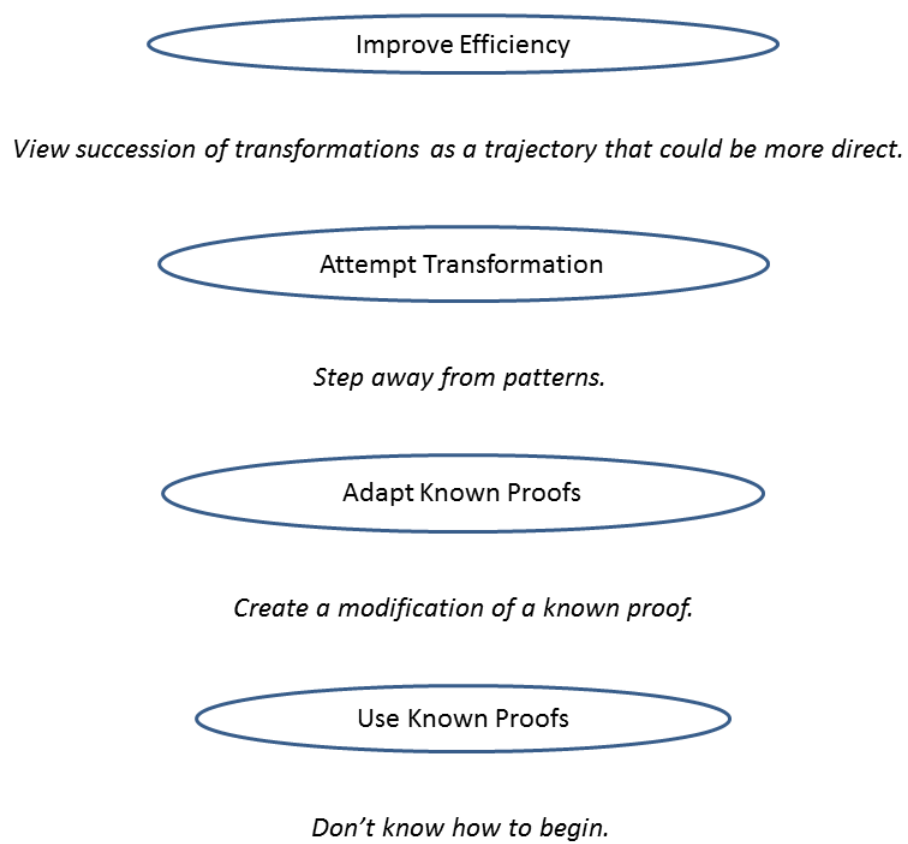


FIGURE 4.4.1: Outcome space from How Students Attempt to Apply Proofs When Assigned

TABLE 4.4.2: Illustrative Quotations for How Students Attempt to Apply Proofs (When Assigned)

| Category                | Quote  |
|-------------------------|--|
| Improve Efficiency      | My former proofs were wandering, I laugh at them now.  |
| Attempt Transformations | I like proofs that are procedures, I don't like "logic proofs"   |
| Adapt Known Proofs      | a proof by contrapositive is blended with the notion of proof by contradiction, and some property is assumed for the inverse of an implication, resulting in an attempt to derive a conclusion from a proof of an inverse. |
| Use Known Proofs        | They taught us that (induction) the most.  |
|                         | I know I can carry out that process, it sort of checks itself.   |

tion, and some property is assumed for the inverse of an implication, resulting in an attempt to derive a conclusion from a proof of an inverse.

Students who succeed in proving report that over time they improve their proof attempts, and sometimes find their older proofs amusing for their wandering nature.

### 4.4.3 Relations

The relations are shown in Table 4.4.3.

The relation between use of known proofs, and the modification of a proof, is that modifications are conceivable.

The relation between working with modifications of patterns, and applying rules of inference outside the template of a pattern, is to step away from patterns.

The relation between creating sequences of application of rules of inference, and being concerned with short proofs is to develop a sense of efficiency.

TABLE 4.4.3: Relations for How Students Attempt to Apply Proofs (When Assigned)

| Category                | Relation  |
|-------------------------|---|
| Improve Efficiency      | View the transformations as a trajectory that can be more direct. |
| Attempt Transformations |   |
| Attempt Transformations | Step away from patterns.  |
| Adapt Known Proofs      |   |
| Adapt Known Proofs      | Create a modification.  |
| Use Known Proofs        |   |

#### 4.4.4 Critical Factors

The critical factors are shown in Table 4.4.4.

The earliest critical factor is for the benefit of those students who have trouble even beginning a proof. They can start by examining the problem statement. It might resemble a problem they have seen solved. This can be illustrated by teaching solution of several problems that differ in concrete but not structural ways, urging the student to generalize.

The next critical factor is that proofs can be created, possibly by modification of a known pattern. Showing a set of proofs that differ only slightly, and involving class participation in creating modifications uses contrast to help students discern this point.

The next critical factor is that it is not necessary to use a pattern, that rules of inference can be used when they are justified. Contrasting those rules of inference that are justified from those that are not, and contrasting those rules of inference that are applicable in any given situation, from those that are not, might help students discern this constrained freedom.

The next critical factor is the sequence of transformations can be longer than

TABLE 4.4.4: Critical Factors for How Students Attempt to Apply Proofs (When Assigned)

| Critical Factor  |
|--|
| View the transformations as a trajectory that can be more direct.  |
| Step away from patterns.   |
| Create a modification.   |
| There are students do not know how to begin; for them, a critical factor could be that one can examine the structure of the problem statement. |

necessary. Contrasting long proofs with short ones, demonstrating the same claim, should help students discern this.

#### 4.4.5 Dimensions of Variation

The dimension of variation begins with the student reiterating what has been shown in class. Deepening understanding is shown by evidence of modification of what has been shown, as far as a variation on a pattern that has been shown. Further understanding of the logic of transformation is demonstrated by correct (warranted) application of rules of inference. The most advanced conceptualization found includes that students can judge their proof attempts as to length, and seek improvements with more efficient transformations.

#### 4.4.6 Validation

This analysis is supported by commentary from multiple teaching assistants.



#### 4.4.7 Outcome Space

The outcome space from the research question “How do students attempt to apply proof when assigned?” has as a foundation to use known proofs. This conceptualization was discovered from interviews of teaching assistants who reported that students submitted, as solutions to requests for proof, the same proof used in class, even though it was a proof of something else.

A critical factor for students to advance to the next level of understanding is that they can learn how to make valid modifications to existing proofs.

The conceptualization “Adapt Known Proofs” expands the understanding that proofs can be re-used. It allows students to see previous proofs as candidates for patterns, that can be modified. It allows that by making one or a succession of small valid changes, a student may begin to be creative in proof design.

A critical factor for students to advance to the next level of understanding is that by understanding what is a valid transformation, one does not need to depend upon patterns, however useful they may be.

The conceptualization “Attempt Transformation” incorporates the additional notion that one can create a proof without relying on a pattern. Thus, the critical factor that knowing valid rules of inference allows one to create valid proofs “from scratch” is a critical aspect, as well.

A critical factor for students to advance to the next level of understanding is that proofs carry out a succession of transformations that can make use of more or fewer steps, and that shorter proofs are often preferred.

The conceptualization “Improve Efficiency” implies an evaluative capacity on the part of the creator of a proof, that can compare two proof attempts on the basis of

how many steps they take to achieve the given derivation.

TABLE 4.5.1: Illustrative Quotations for How Students Attempt to Apply Proofs  
(When Not Assigned)

| Category   | Quote  |
|--|--|
| Don't seriously<br>apply proofs<br>outside of assign-<br>ments | Q: do you ever decide on your own that you want to do a proof? A: no, I just tend I tend to just write code, it's always been proof enough for me  |
|  | Q: Do you ever find yourself doing proofs? associated with computer science? that haven't been assigned? A: That have not been assigned? Q: Right, for fun, or because you want to know something? A: um, he-he, well, i did find myself doing proofs, they were silly proofs, just like things about like things stuff up, yeah since i didn't have very solve it base, it was just like statements, not really just proof, just where you want to get to, so like the end result that you want to get to |

## 4.5 Phenomenographic Analysis of How Students Attempt to Apply Proofs (When Not Assigned)

### 4.5.1 Categories

There is only one category for student responses to this question. They do not attempt proofs when not assigned.

### 4.5.2 Illustrative Quotations for Categories

Some students claimed they never constructed proofs when not assigned. Illustrative quotations for the categories are shown in Table 4.5.1.

TABLE 4.6.1: Illustrative Quotations for Whether Students Exhibit Consequences of Inability With Proof (such as avoiding recursion)

| Category             | Quote  |
|----------------------|--|
| Non-use of recursion | and then you split, what do i have to do to get to that point, so you have to actually find what are the required pieces for you to solve the problem so and every single piece, then you have to prove by itself, so that can i get to the second step, can i get to the third step, because if i lose the first proof, i will never get to the second, because i already established that my second piece depends upon my first piece, so i cannot move forward so i have to divide into small pieces and try to prove them. |
| Lack of confidence   | I still feel very shaky with proofs, sometimes, still getting the hang of it, it hasn't become second nature to me   |
|                      | it's like I kind of understand like I can see why this would take how long it is but I don't feel it very solid  |

## 4.6 Phenomenographic Analysis of Whether Students Exhibit Consequences of Inability With Proof (such as avoiding recursion)

### 4.6.1 Categories

There is only one category from the traditional phenomenographic method: Students do not have confidence enough to know when they can apply recursion. They felt they were asked to produce recursive algorithms in situations in which it applied, and that they could. They felt that such situations did not occur subsequently. Some asked their employed friends who echoed this opinion.

### 4.6.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.6.1.

### **4.6.3 Relations**

There are no relations, as there is only one category.

### **4.6.4 Critical Factors**

There are no critical factors, as there is only one category.

### **4.6.5 Dimensions of Variation**

There is no dimension of variation, as there is only one category.

### **4.6.6 Validation**

Every participant agreed. Some students contributed commentary from relatives and friends at work, including an uncle working at Microsoft.

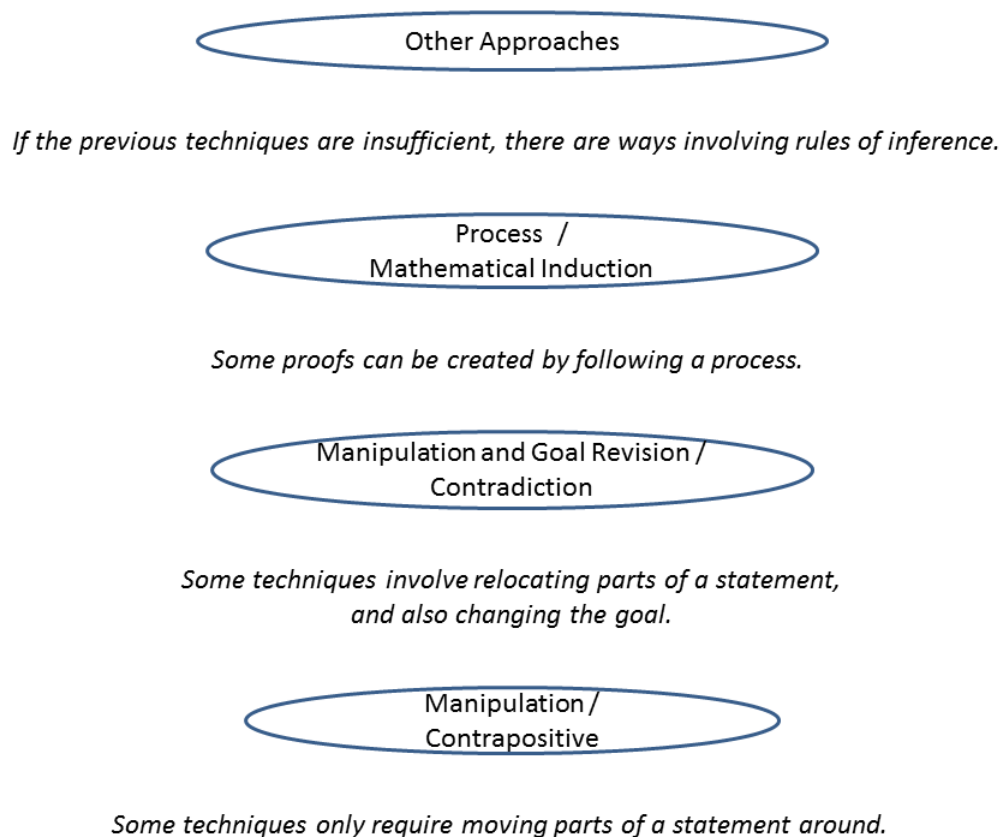


FIGURE 4.7.1: Outcome space from How Students Use Structure in Proof

## 4.7 Phenomenographic Analysis of Student Familiarity with Specific Proof Techniques

### 4.7.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.7.1 and depicted in Figure 4.7.1.

Disconcertingly, students refer to what is not mathematical induction as “logic proofs”.

TABLE 4.7.1: Categories for Student Conceptualizations of Specific Proof Techniques

| Category                         | Description  |
|----------------------------------|--|
| Other                            | Try what seems possible  |
| Mathematical Induction / Process | some proofs can be constructed by carrying out a process                               |
| Contradiction / Goal Redirection | some proofs can be constructed by a mechanical modification and adoption of a new goal |
| Contrapositive / Syntactic       | Some mechanical rearrangement be enough  |

### 4.7.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.7.2.

This data collected for this question was not diverse enough to be pursued with the traditional phenomenographic method. The data include:

TABLE 4.7.2: Illustrative Quotations for Student Familiarity with Specific Proof Techniques

| Category               | Quote   |
|------------------------|---|
| Contradiction          | we're trying to prove that the opposite is incorrect  |
| Contrapositive         | you just flip it around and then it's all better  |
| Mathematical induction | i guess I can just induct here somehow  |
|                        | I'm a fan of like having like a set of steps to do something with, rather than so i know like what to do next |
| Continued on next page |   |

Table 4.7.2 – continued from previous page

| Category | Quote  |
|----------|--|
|          | the thing that that induction is there are steps to it, you prove for this case, you prove for that case, plus one, I can go through those steps and by going through the steps I'm sure it's correct, because it's the right steps but in my mind it's a little shaky |
| Other    | laws of logic proofs, they're a little more difficult it's almost like a puzzle, i'll sometimes work forwards and backwards at the same time, i'll start at the bottom and kind of modify that ...start thinking from both ends  |
|          | analyzing every possible manipulation  |

When asked about specific proof techniques, some students mentioned proof by mathematical induction

Students claimed to prefer proofs by mathematical induction on the basis that they were formulaic; supposedly, a procedure could be used to synthesize them.

When asked for specific proof techniques other than proof by mathematical induction, students knew the words contradiction and contrapositive, but sometimes could not distinguish between them.

When asked about proof by construction, some students thought this referred to construction of any proof.

Some students thought proof by contradiction referred to proving the opposite of something, rather than disproving the opposite of something.



TABLE 4.7.3: Relationships for Student Conceptualizations of Familiarity with Specific Proof Techniques

| Category                                       | Relation  |
|--|---|
| Other Approaches                               | Where a process is not thought available, one can attempt to apply logic. |
| Process / Mathematical Induction               |   |
| Process / Mathematical Induction               | There are process steps for creating some proofs.                         |
| Manipulation and Goal Revision / Contradiction |   |
| Manipulation and Goal Revision / Contradiction | One can work from the goal towards the problem statement.                 |
| Manipulation / Contrapositive                  |   |

### 4.7.3 Relations

The relations are shown in Table 4.7.3.

One early relation is that one can work not only from the premises to the consequence, but also transform the consequence in the direction of the premise.

Another early relation is that there are proof types for which a procedural approach for creating such a proof, exists.

A later relation is that, even when process steps are not available, or obvious, one can examine the rules of inference.

### 4.7.4 Critical Factors

Critical aspects are listed in Table 4.7.4.

One early critical factor is that both premises and claims are mathematical statements, that can be transformed into statements of the same truth value. Thus, one may advance a proof from either end. These two starting points for a proof can be contrasted with each other.

Another early critical factor is that there are proof techniques that are procedural. By showing proof techniques, such as proof by contradiction and proof by mathematic induction, and contrasting these with proofs that are not procedural, students can be helped to discern these procedures. Students also need to distinguish the types of problem for which the procedural techniques apply. Several examples and generalization, combined with contrasting examples, should help the students perceive this.

That rules of inference can be applied singly (without being part of a pattern) is a critical factor. Separation of an existing proof into its steps, examination of the warranting of the individual steps and subsequent fusion may help students discern this.

TABLE 4.7.4: Critical Factors for Student Familiarity with Specific Proof Techniques

| Critical Factor   |
|---|
| If the previous techniques are insufficient, there are ways involving rules of inference. |
| Some proofs can be created by following a process.  |
| Some techniques involve relocating parts of a statement, also changing the goal.          |

### 4.7.5 Dimension of Variation

The dimension of variation is depth of understanding. The rearrangement and modification of statements is perceived, though rules governing it are not. Subsequently an appreciation develops that sometimes, it is possible to generate statements by following a procedure. A deeper understanding, that applying rules of inference corresponds to performing logical deduction evolves from recognition that the procedurally generated statements form a valid argument – there is underlying reasoning that is

accessible.

#### 4.7.6 Outcome Space

The outcome space from the research question “What familiarity do students have with specific proof techniques?” has a foundation in the ability to perform manipulation of mathematical statements. It is important to note that some students begin being uncomfortable with understanding mathematical formulation, and being unsure what a mathematical statement is expressing. For these students, being able to operate with the conceptualization that mathematical statements can be knowledgeably manipulated is itself an achievement.

Critical factors for reaching the conceptualization that mathematical statements can be manipulated in ways that maintain their truth value is an understanding of what variables can represent (such as possibly empty ranges of values, not necessarily a single one), and what a relation is.

The conceptualization “Manipulation / Contrapositive” includes that the truth value of a statement can be maintained when certain manipulations are performed upon it.

A critical factor for understanding proof by contradiction is that a manipulation can make sense together with a revision of the goal of the proof.

The conceptualization “Manipulation and Goal Revision / Contradiction” adds the idea of contexts and the idea that a manipulation together with a goal revision can be a valid transformation step. Students may see examples of a new context being established without also learning about the closing of the context.

A critical factor for understanding proof by mathematical induction is the un-

derlying logic of the base case forming an instance of a premise of an inductive step being known to be true. Students who learn to produce a proof by mathematical induction, by following a recipe, can have trouble with problems, such as that created by Polya and used by Sipser. This problem, in the case of Sipser being about horses, mimics the recipe construction of a proof by mathematical induction, and requires for solution that the student recognize that the inductive step must have a base case that transforms the derivation of the inductive step into a proof that the result of the induction is true.

The conceptualization “Process / Mathematical Induction” includes that the process by which proofs by mathematical induction is known. The construction process offers a convenient context to dwell on the reasoning underlying this proof method.

A critical factor for understanding construction of proof without patterns, such as mathematical induction, is the understanding of the rules of inference.

TABLE 4.8.1: Categories for Student Conceptualizations of What Structural Elements are Found in Proofs

| Category                         | Description   |
|----------------------------------|---|
| Components                       | Proofs can be built from components                   |
| Pattern(s)                       | proofs can follow patterns, such as in contradiction  |
| Process Steps, State Transitions | proofs can be generated by following a process        |
| Like Programs                    | similarity to programs in that sequence of statements |

## 4.8 Phenomenographic Analysis of How Students Use Structure in Proof

### 4.8.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.8.1 and depicted in Figure 4.8.1.

Ideas that would have been welcome but did not appear include use of lemmas, use of cases, nested assumptions.

### 4.8.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.8.2. Some students, in the context of hearing a presentation in an algorithms course, of a proof with a lemma, do not know, by name, what a lemma is.

Some students describe proofs as a sequence of statements, not commenting on any structure. Some students, in the context of planning to construct a proof, do not choose to divide and conquer the problem, breaking it into component parts, such as

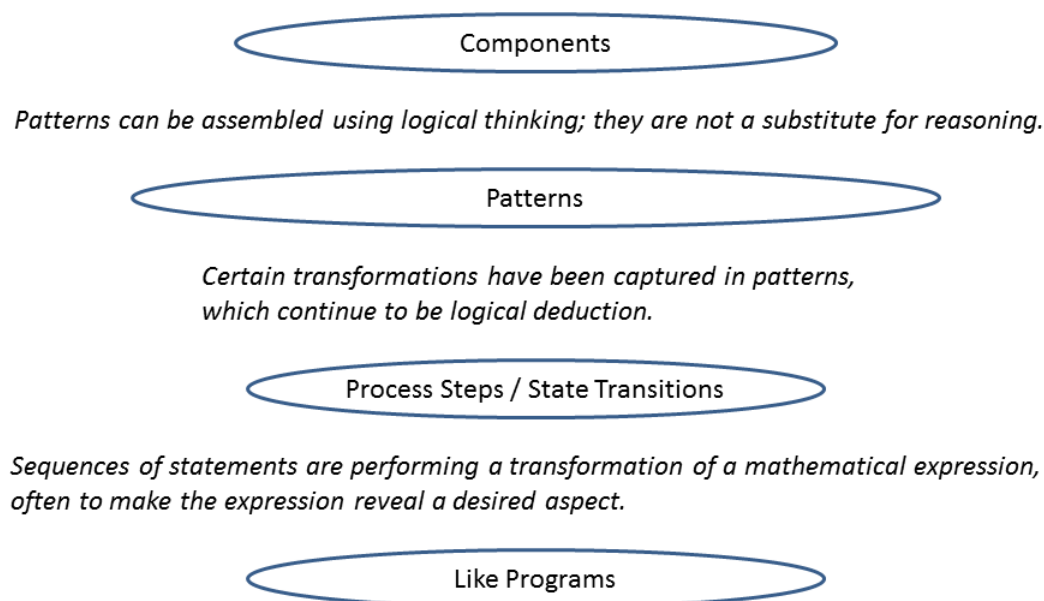


FIGURE 4.8.1: Outcome space from What Structural Elements Students Find in Proof

TABLE 4.8.2: Illustrative Quotations for How Students Use Structure in Proof

| Category                         | Quote  |
|----------------------------------|--|
| Components                       | What's a lemma?  |
|                                  | like with programming there's no ambiguity, everything is very structured, like proofs are structured in much the same way |
| Patterns                         | there are patterns you can use, a list of patterns, to identify if proofs are logical                                      |
| Process Steps, State Transitions | There are the ones that use process steps, and logic proofs, that use rules of inference.                                  |
| Like Programs                    | sequence of statements   |

TABLE 4.8.3: Relationships for Student Conceptualizations of What Structure Students Notice in Proofs

| Category                         | Relation  |
|----------------------------------|---|
| Components                       | Patterns can be combined, be used as components.  |
| Pattern(s)                       |   |
| Pattern(s)                       | It is helpful to know patterns of process steps, because the steps can be applied as a group. |
| Process Steps, State Transitions |   |
| Process Steps, State Transitions | There are organizing principles for these sequences of statements.                            |
| Like Programs                    |   |

cases. Some students appreciate structure.

### 4.8.3 Relations

The relations are shown in Table 4.8.3.

The first relation is that sequences of transformations are organized.

Next, patterns of organization have been learned, and are usefully applied to problems of certain types.

Last, problems can sometimes usefully be broken into components, such that the components may be susceptible to being solved by the patterns.

### 4.8.4 Critical Factors

The first critical factor is to see that the transformations in a sequence are connected, they form a trajectory of transformation. This can be contrasted with an unordered set of transformations.

The second critical factor is that some useful sequences have been discovered and formed into patterns. Though such a pattern, for example proof by mathematic induction, is usefully shown in multiple applications, Marton and Pang[177] have shown that varying the foreground concept, in this case the proof technique, against a fixed background, in this case the problem to be solved, is more effective in helping the student discern the learning objective. Thus, the existence of an applicable pattern would be more effectively shown by attempting a non-patterned approach, contrasted with application of the patterned approach.

The last critical factor is that components (possibly involving solution by pattern) can be combined. One contrast might be a monolithic proof with a proof in component parts. For example, separation could be used, on a carefully constructed monolithic proof, to show the parts.

Critical aspects are listed in Table 4.8.4.

TABLE 4.8.4: Critical Factors for What Structure Students Notice in Proofs

| Critical Factor   |
|---|
| Patterns can be assembled using logical thinking; they are not a substitute for reasoning. Moreover, building blocks used in a proof can be custom created for that particular proof. |
| Certain transformations have been captured in patterns, which continue to be logical deduction.   |
| Sequences of statements are performing a transformation of a mathematical expression, often to make the expression reveal a desired aspect.   |

## 4.8.5 Dimensions of Variation

The dimension of variation seems to take support from, be scaffolded by, development of software. The most basic conceptualization students exhibit for structure is the



individual line, similar to the line of code. The next more inclusive conceptualization recognizes groups of lines that show transitions or steps of transformation. The next more inclusive conceptualization captures that such groups of lines can form a pattern, as for reuse on multiple occasions. The next more inclusive conceptualization encompasses more freedom in the choice of sequences of lines. More significantly, the focus moves from pre-considered solutions to the problem. The most inclusive conceptualization admits the idea of solving the problem in component parts.

#### **4.8.6 Validation**

These data are internally consistent with how students construct a proof when assigned. They are also compatible with the material on recursion and proof by mathematical induction.

#### **4.8.7 Outcome Space**

The outcome space from the research question “How do students use structure in proof?” is founded on the students’ experience with programs. Though students do have experience with programs, and from the point of view of the instructor, might predictably have noticed analogies between the structure in programs and the opportunity to employ structure in proofs, it is not the case that all students with programming experience can transfer that experience to proof construction.

A critical factor is awareness of specific correspondences, as between invocations of an auxiliary function, and use of a lemma, or as in a programmatic transformation step that eases subsequent processing, without changing the data carried within a data structure.

The conceptualization of “Like Programs” is readily available to students in a vague way, and can be made much more specific.

A critical factor that enables students to make use of a more advanced conceptualization is that the work of a proof is to transform a mathematical expression, such that a desired aspect of what we want to show is more evident.

The conceptualization “Process Steps / State Transitions” involves deeper understanding of the manner in which proofs are like programs. The structure of a proof may include transitions from one representation of information into another representation. A program often moves a collection of memory locations through multiple states.

A critical factor that enables students to make use of a more advanced conceptualization is that some transformations are routinely useful, and appear in patterns. The patterns retain the logical deduction that was used when they were constructed.

The conceptualization “Patterns” helps students see how to apply routine sequences of logical deduction steps in situations that support that logic. It also helps students see what customization of the pattern might be useful when trying to prove a given proposition.

A critical factor that enables students to make use of a more advanced conceptualization is that patterns may themselves be applied in combinations. It is important that students realize that combining patterns must be accompanied by reasoning. Even though patterns carry with them the logical deductions that were used in their creation, one must know when it is warranted to apply the pattern.

The conceptualization “Components” helps students see that proofs can be assembled making use of component parts, when warranted.

There are rules the sequence of statements in a proof must adhere to, more than

there are rules for sequences of statements in code. In particular, statements, that are complete and correct when considered by themselves, must be warranted in proofs, else it does not form an argument. Sequences of statements that satisfy a compiler may be said to be a program. Whether these statements taken together calculate anything of any value is a different matter. Whether mathematical statements in a list form a proof, depends upon whether the subsequent statements are warranted by those going before.

TABLE 4.9.1: Categories from What Students Think Makes a Proof Valid

| Category                          | Description   |
|-----------------------------------|---|
| Understand the Reasoning          | Transformations that advance toward the goal help reduce domain of allowed transformations. |
| Stick to valid rules of inference | rules of inference tell us some valid transformation we can do                              |
| Re-use proof patterns             | can be apply patterns without understanding justification                                   |
| Know what's true and why          | this is not about new arguments, is about memory  |

## 4.9 Phenomenographic Analysis of What Students Think Makes a Proof Valid

### 4.9.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.9.1 and depicted in Figure 4.9.1.

Ideas that would have been welcome but did not appear warrants based upon definitions, preconditions, postconditions.

### 4.9.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.9.2. Some students are not sure how to construct an argument.

Some students do not understand that statements should be warranted. Without an appreciation of definitions, understanding warrants can be expected to be fraught with difficulty.

Some students do not recognize a good argument when they are looking at one.

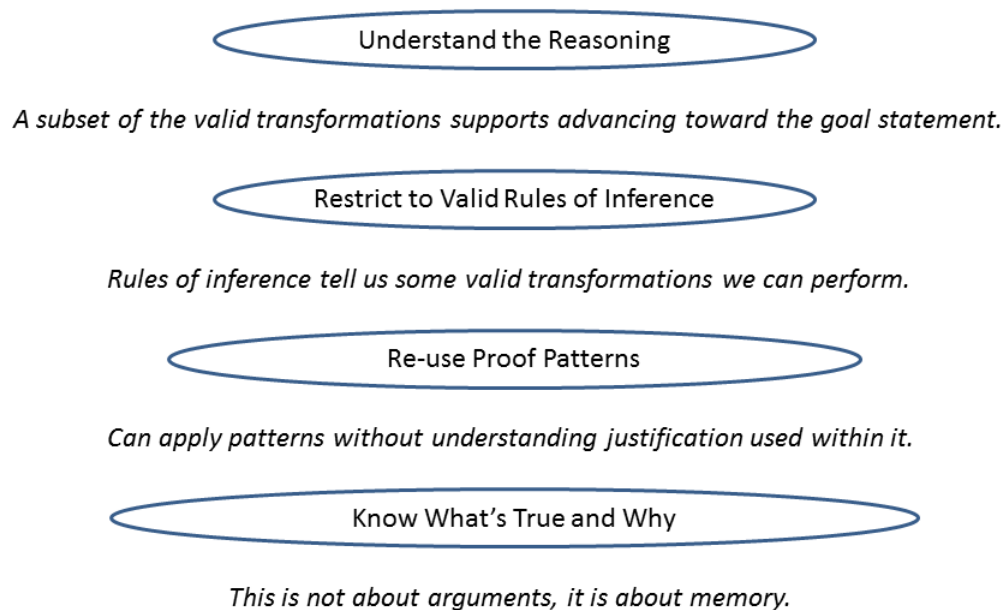


FIGURE 4.9.1: Outcome space from What Students Think Makes a Proof Valid

TABLE 4.9.2: Illustrative Quotations for What Students Think Makes a Proof Valid

| Category                          | Quote   |
|-----------------------------------|---|
| Understand the reasoning          | it's the steps in between that no one has an idea about, it's like a bridge, you start at a, you get to c, but b is the journey, and everyone skips that, they understand a, c but they don't |
| Stick to valid rules of inference | You have to stick to valid rules of interference.   |
| Reuse Proof Patterns              | you should really see why this works, that's where the gap is where everyone kind of loses it.  |
| Know what's true and why          | You have to just know what's true, remember it, and why it's true.  |

Some students used confused/incorrect forms of rules of inference.

Some students do not notice that the imposition of a subdivision into cases creates more premises.

Some students do not notice that proof by contradiction introduces (for purposes of contradiction) a premise.

Instructors do tell students about valid forms of arguments. There are some students who can recite the names of valid forms, but cannot produce arguments using them. Students do reiterate valid proofs from class. If the assigned example matches the proof from lecture sufficiently, false positive results for understanding can occur.

### 4.9.3 Relations

The relations are shown in Table 4.9.3.

The earliest relation is that though one may believe it is necessary to remember all axioms and previously proved theorems, it may be more useful to remember some proof techniques (including the reasoning captured within them), and be able to tell when they apply.

The next relation is that rules of inference may be applied without being embedded in a pattern, so long as they are warranted.

The most advanced relation is that there is the notion of a proof plan, and that some proof plans are preferable to others, direct proof being preferable to indirect.

### 4.9.4 Critical Factors

Critical factors are listed in Table 4.9.4.

TABLE 4.9.3: Relationships from What Students Think Makes a Proof Valid

| Category                          | Relationship  |
|-----------------------------------|---|
| Understand the reasoning          | Have a proof plan, develop a preference for direct proofs over indirect   |
| Stick to valid rules of inference |   |
| Stick to valid rules of inference | Even in the absence of a known pattern, one may apply rules of inference that match the present expression. Warrants are important.                   |
| Re-use proof patterns             |   |
| Re-use proof patterns             | Besides using memory, one can match certain situations with applicable patterns, and explore whether transformation by that pattern would be helpful. |
| Know what's true and why          |   |

The earliest critical factor is that there is valid reasoning embedded in proof techniques, which can be employed, though it should be understood in general and in its application to a given problem. Contrast with fallacious application, such as Sipser's[241] proof-attempt that all horses are of the same color, might help students discern this.

The next critical factor is that even in the absence of patterns, validity can be achieved by using individual rules of inference, when warranted. Contrast between warranted and unwarranted rules of inference could be helpful.

The last critical factor is that there may be multiple valid transformations, and some might be more helpful in the domain of efficiency than others.

#### 4.9.5 Dimensions of Variation

The dimension of variation that most strongly suggests itself is degree of appreciation of logical deduction and of definitions as warrants for the logical transformation steps

TABLE 4.9.4: Critical Factors from What Students Think Makes a Proof Valid

| Critical Factor  |
|--|
| A subset of the valid transformations supports advancing toward the goal statement.  |
| Rules of inference tell us some valid transformations we can perform.  |
| One can apply patterns without understanding the justification used within it, but it good to know that patterns are sequences of logical deduction. |

that are to be carried out. Students' conceptualizations show a low appreciation for definitions, including a reluctance to learn them.

Once students are aware of the power of a carefully crafted definition, which we do see in our subpopulation of students who are dual majors in mathematics, the enthusiasm for building collections of related concepts and reasoning about them appears. Appreciation of definitions as precision tools, and appreciation for reasoning seem to be mutually reinforcing concepts.

More practice distinguishing valid arguments from invalid ones, and more practice writing arguments in valid forms might help.

Students need to pay attention to how the context of definitions and the item to be proved relate to the progression of statements that demonstrates what is to be shown. Practice explicitly providing warrants might help.

#### 4.9.6 Validation

Alcock and Weber[6] in mathematics education observe the role of appreciation of definitions in warranting.



### 4.9.7 Outcome Space

The outcome space from the research question “What do students think makes a proof valid?” has as its foundation, the conceptualization that students should know what’s true, and why, from memory. This view seems consistent with views found in the student population about proofs practiced in middle school or high school geometry.

A critical factor is that we study proof techniques, and the role of axioms and theorems is that of data to the proof technique’s algorithm.

The conceptualization “Re-use Proof Patterns” incorporates the presence of and interest in proof techniques that describe the transformation undergone by the problem statement’s proposition, as it is re-expressed to make evident what we wanted to show.

A critical factor is that it is not necessary to restrict logical reasoning to that of the studied patterns, such as proof by mathematical induction. What some students call “logic proofs”, namely those which employ logical deduction, even if in a novel fashion, are also possible, so long as only warranted transformations are made.

The conceptualization “Restrict to Valid Rules of Inference” extends the previous conceptualization with the freedom to apply warranted rules of inference in any combination.

A critical factor is that there is a notion of a goal, (namely that which we want to show) and of progress toward the goal, which could be measured in number of transformation steps. The structure of the statement that we want to show can provide clues to transformation steps that could be helpful.

The conceptualization “Understand the Reasoning” has increased depth of under-

standing compared to the previous conceptualization. Beyond focusing on warranting each transformation step, the idea of guiding the transformations along a path, and that some paths (such as direct ones) are more desirable than others is included in this conceptualization.

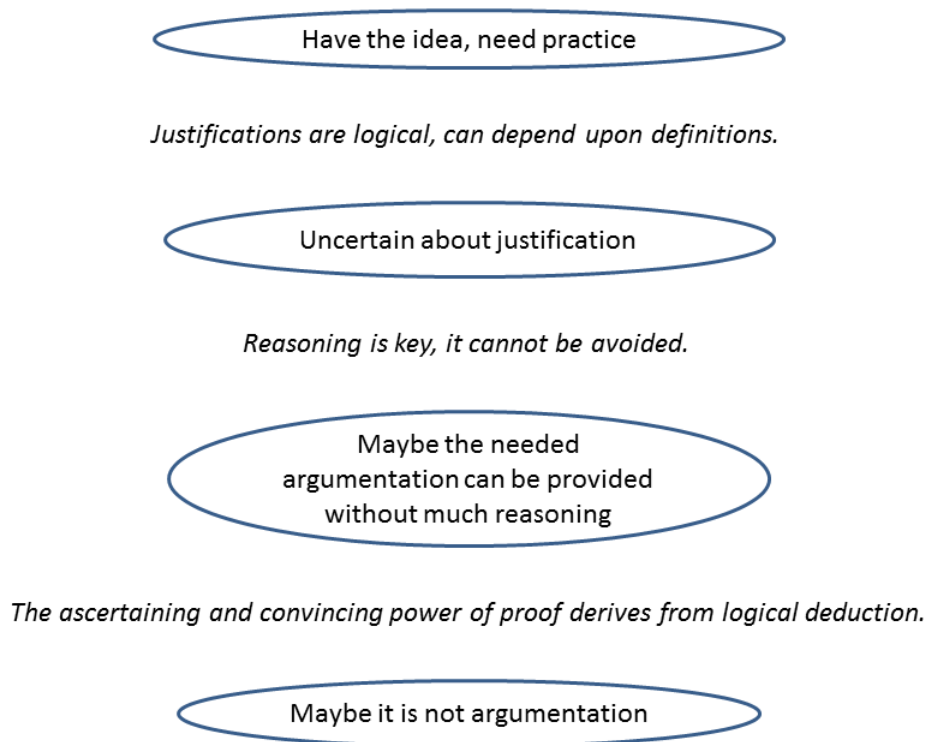


FIGURE 4.10.1: Categories from Combined Data

## 4.10 Phenomenographic Analysis of Combined Data

### 4.10.1 Categories

The categories, developed in the traditional phenomenographic analysis, are listed in Table 4.10.1 and depicted in Figure 4.10.1.

The most basic category, “Maybe it isn’t argumentation”, includes that examples are preferred over definitions. This category also includes that code implementations serve as well as proofs (which might suggest that not all students are remembering some universal statements can be proved). This category misses the point that argu-

TABLE 4.10.1: Categories for Combined Data

| Category   | Description   |
|--|---|
| “Have the idea, need practice”   | This conceptualization recognizes that improvement should and probably can take place. This idea that practice might help is included.  |
| “Uncertain about justification”  | The conceptualization knows that reasoning is to be carried out. It does not contain a complete understanding of how any needed reasoning might be supported.                   |
| “Maybe the needed argumentation can be provided without performing much reasoning” | This conceptualization recognizes that reasoning is being used. It does not understand how to reason.   |
| “Maybe it isn’t argumentation”   | This conceptualization does not notice that argumentation is the activity. It recognizes concrete conclusions by other means, and is lost when abstract entities are discussed. |

mentation, (which can be greatly facilitated by mathematical notation) can be based upon definitions. It misses that, with definitions, some universal statements can be shown to be true beyond any doubt. Some students’ belief that proofs of theorems in Discrete Structures is about the theorems rather than being about the proofs could be evidence of this category. Some students’ response of “Duh” to arguments about concrete entities could be explained by this conceptualization. The difficulties some students have generalizing from arguments about concrete entities to abstract entities could be explained by this conceptualization.

The next most basic category, “Maybe the needed argumentation can be provided without performing much reasoning”, includes the application of arguments realized as a sequence of process steps. While the first category misses argumentation save by use of example, the second category seeks to provide argumentation without en-

gaging in much reasoning. Note that the behavior of choosing proof by mathematic induction as a solution because it was most frequently taught or is remembered best as opposed to because the nature of the problem seemed matched to it is evidence of this. Confusion between proof by contrapositive and an admixture of proof by contradiction and use of inverse of an implication are evidence of this category of conceptualization. The variety of incorrect reasons students propose, for why proof is taught in the computer science curriculum, is consistent with this conceptualization.

The next category “Uncertain about justification” includes the use of steps presumably seen in instructional materials, but in situations in which they are not justified, such as the assertion that all integers can be represented as  $2k + 1$ , for  $k$  an integer, because “all integers are odd”. The absence of the idea, that the collection of mathematical definitions is a resource, has consequences. One consequence is that students with this conceptualization aren’t necessarily able to draw on their knowledge of definitions, even to suggesting what they must know is false.

The next category “Have the idea, need practice” includes reasoning by application of warranted rules of inference. Students who reported spending hours trying out possible inferences, and developing a sense of which transformations represented progress give evidence for this category.

### 4.10.2 Illustrative Quotations for Categories

Illustrative quotations for the categories are shown in Table 4.10.2. The data are the whole dataset collected for any of the research questions.

In the data of three of the research questions the idea of patterns appears. The ideas of process steps and of visualization also feature prominently. Some students



seem to be attempting to understand the structure and validity of proofs by using their knowledge of patterns, process steps and visualization.

The approach of building up process steps for proof construction differs from the axiom and definition based approach of Hilbert[126, 129], which we might hope our students would learn.

Particularly of value are the data that students discount definitions in favor of examples, that students find examples are easier, that students do not necessarily appreciate the value of definitions, and that students think they can adequately reconstruct the definitions from examples. Some students don't realize how time consuming that reconstruction would be.

### **Maybe it isn't argumentation**

Student interview, student help session, test data and faculty interview combined on the subject of some students have trouble moving from reasoning about concrete entities to reasoning about abstract entities. This difficulty was initially thought to stem from an inability to think with abstractions. Written logic was carefully populated with concrete entities and then abstract entities in a completely parallel construction (in the sense of English grammar). The reasoning was found to be clear in the case with concrete entities and obscure in the case with abstract entities. Even when abstractions were "approached" by contracting the names of the concrete entities, the sense of being convinced of the conclusion was lost immediately in the contraction. In this manner it was discovered no use was being made of the argument, rather an appeal to already established knowledge ("Maserati is expensive. Duh") was being made, and the reasoning steps were being ignored. As soon as contraction invalidated

the established knowledge, use of the reasoning steps was all that remained.

### **Generalization**

The above-mentioned test, in which parallel construction was used, to populate one argument in turn with first concrete entities and second abstract entities, received student thanks, as a helpful teaching intervention. Certainly the suspicion that it would be helpful drove the creation of the exercise. Student confirmation that the argument is somehow more readily appreciated with concrete elements, and can then be seen on its own, and applied to abstract entities.

### **Generalization vs. Argumentation**

By these two data we see an example where multiple causes can manifest in the same outcomes of student confusion.

### **Interiorization vs. Internalization**

Prior work by Harel and Sowder[118] among students of mathematics revealed the distinction between interiorization and internalization. It seemed to make sense that students of computer science in comparison with students of mathematics, given the emphasis on programming and the meaning of processes, that students of computer science would advance to the more comprehensive conceptualization rapidly, leaving the less comprehensive unpopulated. This raised consciousness of the topic was brought to interviews, but resulted in the unexpected outcome that some students of computer science can learn to carry out a process without being able to speak articulately about what the process is and how it works.



## **Representation**

This topic was brought to the forefront in an interview with a student. The original subject of the interview was proof techniques, and proof by mathematic induction was being discussed. Recursion was also being discussed. This student, who had performed well in courses in which these were taught, had never noticed any similarity between the two until during the interview. Because he commented upon this during the interview, the follow up question, “Why not?” occurred. The answer was that mathematic induction was taught in mathematical formulation and recursion was taught using diagrams. No thought connecting the two occurred because the representations were different.

This impermeability of representation domains occasioned further thought about generalization. Why should thoughts learned in one representation remain trapped there, rather than being generalized, and possibly re-expressed in a different representation?

This question was pursued with the list of questions found in appendix B. The questions were administered to several volunteers, both faculty and students. The results were that changing representations was challenging, and in some cases fun.

## **Structure**

These data are mainly those mentioned in section 4.8.

## **Steps, Leaps and Direction**

This topic was discussed with students who are dual majors in computer science and mathematics. Such students found that a sense of direction for transformations in

TABLE 4.10.3: Relationships for Combined Data

| Category   | Relationship                                       |
|--|--|
| “Have the idea, need practice”   | The idea of what justifications are                |
| “Uncertain about justification”  |  |
| “Uncertain about justification”  | The recognition that reasoning should be exercised |
| “Maybe the needed argumentation can be provided without performing much reasoning” |  |
| “Maybe the needed argumentation can be provided without performing much reasoning” | The idea that reasoning is the main activity       |
| “Maybe it isn’t argumentation”   |  |

proof is acquired through practice. They reflected upon older proof attempts they had made, and newer, and found the earlier attempts rather laughable due to their circumambulating nature.

### 4.10.3 Relations

The relations are shown in Table 4.10.3.

The earliest relation is that constructing a logical, deductive argument is the activity that is being carried out; that is the means by which ascertainment and persuasion is accomplished.

The next relation is that reasoning is not to be avoided. Procedurally generating a pattern, without reasoning about why and how it is applicable, is not a sufficient practice.

The most advanced relation is an understanding of warranting. This implies use of definitions.

#### 4.10.4 Critical Factors

Critical factors are listed in Table 4.10.4.

The first critical factor is that logical deduction is the activity that creates proofs, causes ascertainment and give the convincing power to proofs. Contrast between an unsupported assertion, and use of a theorem, which has its own proof might help students discern the central role of logical deduction. The prevention of infinite regress by appeal to axioms may further clarify that logical argumentation is the only justification used for belief.

The next critical factor is that it is not sufficient to be able to correctly apply a pattern. Its reasoning must be carried out. Otherwise students will not be convinced by proof, and will lose the benefit of its use. Contrasting applications of procedurally generated proofs to situations in which reasoning connects the proof to the problem vs. when such reasoning is omitted should help students discern the role of reasoning.

The last critical factor is that warrants can depend upon definitions. To make this more discernible, a fake and inadequate definition can be drawn from some examples, and a defective proof-attempt created from it. This can be contrasted with use of a real definition, and correct warranting.

TABLE 4.10.4: Critical Factors from Combined Data

| Critical Factor  |
|--|
| Justifications, warrants, are logical and can depend upon definitions.   |
| Reasoning must be carried out in the construction of the proof; process steps applied without exercising the reasoning associated with them are unsatisfying.  |
| The ascertaining and convincing power of proof derives from logical deduction. The principle activity is reasoning, logical deduction. Appealing to existing theorems might seem like invoking prior authority, but it is actually invoking the reasoning supporting them. |

#### **4.10.5 Dimensions of Variation**

There is a dimension of variation that extends from the idea that the student is being called upon to synthesize arguments, through the means of expression of mathematical ideas, through the often creative construction of a sequence of warranted transformations. The idea that a warrant can be the consequence of a definition, and that therefore definitions are critical for reasoning is developed in support of transformations. The idea that reasoning based on definitions can support the proof of universal and not just existential statements then informs those students who had hoped that finding examples would be a sufficient substitute.

#### **4.10.6 Validation**

That examples loom vastly more important in the minds of students than definitions has been noted by several, and summarized by Chazan[50].

This suggests that explicit use of patterns, process steps and visualization, comparing and contrasting these existing ideas with ideas from proof will help students learn the new ideas.

## Chapter 5

# Interpretation/Discussion

In this chapter we discuss the results obtained in the analysis. First we discuss and interpret the results for the individual questions. Then we combine the data from the individual questions, to discuss what the data taken altogether seem to suggest.

Concerning the amount of discussion, no implication about fractions of the student population holding one or another of the conceptualizations is intended. As when we write an algorithm or program in which we handle special cases, the amount of code for the more frequent cases may be much less than the amount of code for the unusual cases. So too, here, the amount of discussion should not be thought to match the relative frequencies with which ideas are held by students. Conceptualizations are deemed to be held by populations of students; individual students may simultaneously hold multiple conceptualizations, such as the student Lasry and Mazur quoted, “How should I answer these questions? According to what you taught me, or according to the way I usually think about these things?”[156]. Qualitative research often seeks information without putting a premium on relative frequency. For example, there are

researchers[64] to whom it matters greatly that the viewpoints of a minority are not lost by being averaged into those of a majority.

In a phenomenographic study, the results include categories of conceptualizations found in the collection of students. Much as a detective assembles clues to arrive at a coherent explanation of events, we assemble phrases (fragments of interviews, parts of proof attempts) to infer categories of conceptualization from the data produced by the participants. A description of each category is part of the result, and also pairwise relationships between some pairs of categories are part of the result. Together the categories and relationships form the outcome space.

One expectation in a phenomenographic study is that one or more dimensions of variation will be discovered[176]; one or more dimensions of variation is made more apparent, by considering the several relations in the outcome space. Arranging the categories as suggested by these pairwise relationships helps exhibit a dimension of variation. The dimension of variation, by means of the categories arranged along it, may in turn suggest distinguished points, called critical factors (also critical aspects<sup>1</sup>). Points along the dimension of variation are concepts, awareness of which is thought to be especially beneficial to students[177], for the purpose of enlarging and or deepening their understanding.

As we discuss the results, it can make sense to consider alternate dimensions of variation, comparing them with the chosen. This is a process step thought to contribute to validation[65]. For some research questions we are able to consider alternate critical factors, both along alternate dimensions of variation, but also along

---

<sup>1</sup>The phrases critical factors and critical aspects are both used. We prefer critical factors, as the idea aspects has been used more specifically to discuss a component hierarchy of a learning objective. We choose the idea critical factor because it allows us to discuss a generalization/specialization hierarchy.

the proposed dimensions of variation.

Critical factors are ideas that are necessary for a better conceptualization[177]. We conjecture that not all points along a dimension of variation are equally valuable. Some ideas encounter more resistance from students than others [213]. While students resist ideas, they are less likely to adopt them on their own[213]. Those ideas that are less readily adopted by students, but are also necessary for advancement along the dimension of variation, might be preferable as critical factors, in that these match the helping role of teachers to the area where help is most needed[177].

We attempt to trace development of student's conceptualizations, looking especially for stages at which some students seem to miss some components of the material we wish them to acquire.

The organization of this chapter matches that of the previous chapter, so, the research questions are considered in the same order.

## **5.1 Interpretation of What Students Think Proof Is**

Here we interpret the results from section 4.1, see Figure 4.1.1. Please recall that the categories are of conceptualizations of what proof is; because the conceptualizations are incomplete, the category names do not correspond to understanding of proof, rather to partial understandings from which the ability to understand what proof is are built. For the convenience of the reader, these were, earliest to most advanced, awareness of abstraction, contains certain syntactic elements, composed of mathematical statements, combinations of standard argument forms, argument in support of idea or claim, and makes a claim obviously correct.

Some students have not adopted the view that truth is made clear by logical deduction. To imagine how this could be, consider that students have been exposed to much material whose intention is to convince, such as sales literature, political oratory and guidance from authority figures, and relatively little material that includes logical deduction, such as geometry class and debate club.

When the conceptualization in which persuasion is not logical, rather opinion comes from asking someone the person knows or trusts, or by personal experience, is operating, then the students' readiness to engage with logical reasoning might need to be improved.

When thinking about individual concrete entities, those students having not yet adopted logical argument as a basis for belief, might not feel a need to make this advance. Were students thinking in terms of abstract entities, they might more easily discern the need for logical reasoning. It is the case that there are some students who do not appreciate abstraction. In software engineering, where students use metrics tools that report depth of inheritance hierarchies, students have asked what purpose such hierarchies serve. Students selected for merit in the student advisory committee in our department have asked why we spend as much time as we do on inheritance hierarchies in Unified Modeling Language(UML®). The recommendation to use sub-classes instead of copying and pasting code is sometimes received without enthusiasm.

It appears that awareness of the benefits of thinking with abstraction could advantageously be further cultivated among the students. Once trust in these benefits is present, the use of reasoning about abstractions can be motivated.

Another basic conceptualization, "contains certain syntactic elements", recognized by researchers[118], (who name it "ritual proof"), in students of mathematics, is that the markings that accompany, call attention to, and set off a proof, such as the tokens



“Proof:” and “QED” are part of the proof in the sense that they contribute to what makes a proof a proof.

A conceptualization showing deeper understanding of what proof is recognizes that proof is composed of mathematical statements. Not all students effectively compose individual mathematical statements. Some students are confused by individual mathematical statements with multiple quantifiers, especially if negation is included. With this context, one can see that some students can benefit from help in understanding sequences of mathematical statements. Likewise, some students can benefit from help in learning to construct sequences of mathematical statements. When difficulty in understanding the individual statements is present, noting the need for a warrant to support a subsequent statement could be more fraught.

One conceptualization, “combinations of standard argument forms”, builds upon the understanding that a proof is a sequence of mathematical statements, observing that standard forms, routine proof techniques, exist. Some students have the understanding that some proofs can be constructed by following a procedure, for example, students described a process for generating a proof by mathematic induction. Some students describe a (partial) method for carrying out a proof by contraposition. This conceptualization does not necessarily include understanding the logical deduction embedded in the proofs constructed by such a process. Students have described proofs that must be carried out without the benefit of process-step generation as “logic proofs”.

The conceptualization that a proof is an argument consisting of logical deduction is more comprehensive and deeper than these previous conceptualizations.

The conceptualization that a proof makes a claim obviously correct rests upon the notion of a logical argument and turns from looking at what a proof is internally, to

what a proof produces, namely confidence in some clearly defined assertion.

## 5.2 Interpretation of How Students Attempt to Understand Proofs

Here we interpret the results from section 4.2, see Figure 4.2.1.

For the convenience of the reader, the categories were, from earliest to most advanced, study the proof from class, study proof patterns from class, go over all axioms and theorems from class, use visualization, use dynamics, use different representations, look up definitions and use them, and appreciate generalization / specialization hierarchy of definitions.

An early conceptualization of how to approach understanding a proof is to review the description of a proof (not necessarily the same one) shown in class. This technique is consistent with a theme, of which we shall see more later, that students like to learn by inference from examples. Using this conceptualization, “study a proof from class”, a student may compare a proof to be understood with a proof that has been discussed in class, to determine whether they are the same.

A more general conceptualization, “study proof patterns from class”, enables students to compare a proof to be understood, or parts of it, with patterns for proofs, discussed in class. Reminiscent of learning to understand programs by making changes in them, to see what effect may be observed in the output, students can detect a proof technique by learning it as a theme with variations, and compare a proof to be understood with a collection of known proof techniques.

Some students describe going over all axioms and theorems discussed in class, comparing them with part or all of the proof to be understood.

A more advanced conceptualization switches from using examples of proofs or parts of proofs from class to using reasoning. With this conceptualization, “use visualization”, students try to incorporate a visual, diagrammatic dimension into their method of understanding. There may be items discussed in the proof that have elements whose relationships can be diagrammed. Students using this conceptualization speak of “seeing” what the proof is about.

Adding, to the previous conceptualization, the dimension of time, yields a conceptualization called “use dynamics”. With this conceptualization, students talk about “seeing what is going on”. It is not always the case that students who wish to see what is going on know that it is transformation of mathematical expressions to reveal the desired proposition that is “what is going on”.

Diagrams, static or animated, are a different representation. There is a conceptualization which includes deliberate use of alternative representations. Some students appreciate the rephrasing of mathematical statements into programming language. Use of a programming language brings a sense of familiarity which somehow seems to aid in comprehension, perhaps by facilitating associations of ideas that are less accessible for our students, when mathematical notation is employed.

A deeper conceptualization is that definitions, built up and maintained by the mathematical community, are not only implicit in the use of mathematical terms, but key to the warranting of transformation steps.

Beyond this, the appreciation of the collection of definitions, in their generalization / specialization hierarchy, is a deeper, more comprehensive conceptualization, that is a tool for understanding proofs.

### 5.3 Interpretation of What Students Think Proof is For

Here we interpret the results from section 4.3, see Figure 4.3.2.

For the convenience of the reader, the categories were, from earliest to most advanced, that there are some general reasons for proof, such as demonstrating claims, and some computer science related reasons for proof, including providing assurance of algorithm properties such as resource utilization, providing guidance as to whether an algorithm provably meets its requirements, and some deep reasons, also not related to computer science.

Some students do not know why proofs appear in the computer science curriculum, at the time they are studying them. Some students do not know why discrete structures is taught in our curriculum. Of these, not all ask why; instead, some conjecture for themselves, and are sometimes incorrect. When operating with a conceptualization that does not include an understanding of the place of proof in our curriculum, students can lack that motivation which arises from knowing the application of the material. Marton and Booth[176] use the phrase “structural relevance” to refer to the relationship of the material being studied to a domain of applicability. We expect structural relevance to be learned when students see proofs used in subsequent classes, and sometimes it is, but sometimes students don’t recognize proofs, at least not right away, when they subsequently see them. Some students do not notice that the material being taught in discrete structures is not as much about the theorems whose proofs are shown, as it is about the proof techniques. Thinking with this conceptualization would explain questions such as “Do programmers really have to know these facts about prime numbers?”

The simplest conceptualization that includes correct reasons for studying proof encompasses the idea that proofs are both generally useful and a required part of the activities of computer scientists. Students exercising this conceptualization understand that proof by mathematic induction has many applications in computer science, but they might not be able to name one. They might not recognize, for example, that showing a recursive program correct may use proof by mathematic induction. Some students do say, “When I am looking for a proof technique, I try proof by mathematic induction first, because they taught that one the most.”

Enlarging and deepening the simplest conceptualization is one, “Algorithm Properties”, that recognizes that proofs are used in demonstrating the resource consumption of algorithms. With this conceptualization, students can understand that a non-deterministic polynomial time algorithm can process a certificate for a solution in polynomial time, yet might not be able to find a solution in polynomial time.

It is a yet deeper level of understanding, and another conceptualization, “Guidance on Algorithm Design”, that turns the ability of proof to assure us that we understand resource consumption into a tool that guides us as we design algorithms. That with proof we may ascertain that we understand and can explain an algorithm to others, and that without proof we might be mistaken about our design, so we use proof in creating algorithms, is a more advanced conceptualization than “Algorithm Properties”.

Some students generalize from the ability to understand and create algorithms assuredly to the ability to know the truth of other propositions beyond doubt. Then, they are exercising the conceptualization we call “General and Deep Reasons”.

Students have trouble generalizing, so even when they have seen a proof by mathematical induction with natural numbers in Discrete Structures, they do not neces-

sarily transfer that knowledge when they see a proof by mathematic induction on the depth of a stack. Those who see that what has been thought of as a pattern can be used in such a way that its meaning is important may choose at this time to request explanation of this proof technique.

The interpretation is that there is a very broad distribution of conceptualizations, and that continued study improves the least informed conceptualizations. Continued study, such as Introduction to the Theory of Computation can be hampered by operating with the less informed conceptualizations.

There is a gap between obtaining bounds on resource consumption on the one hand, and deriving mathematical formulation of intuitive ideas, as might drive on the other.

Students might benefit from an optional supplementary course, possibly over winter or summer break. The effects of aspects of algorithms on the resulting complexity class might be a valuable stepping stone in between.

## **5.4 Interpretation of How Students Attempt to Apply Proofs (When Assigned)**

Here we interpret the results from section 4.3, see Figure 4.4.1.

For the convenience of the reader, these categories were, from earliest to most advanced, that students copy proofs seen in class; adapt, or modify proofs seen in class; attempt to make use of transformation steps; and, once they can make correct proofs, sometimes strive for efficient transformations.

That some students provide as a proof attempt on homework, the proof given in lecture, whether or not that proof is appropriate to the assigned proposition, indicates

that some students are approaching their assignments with little insight into how to choose a proof technique based upon the problem statement.

Some students have trouble understanding the mathematical formulation of the problem statement. Some students have trouble with negation, even of statements with a single quantifier. We might ask: “What if students are not thinking of universals?” Some students express that they do not have any comfort level with mathematical formulation as a language. Some students produce mathematical notation by copying examples, or parts of examples, much as a traveler might produce from a phrase guide to an unfamiliar language. One student remembered a process in which some expression  $A$  is negated, logical deductions are made and a contradiction reached. Based on this contradiction, in classical deduction, something can be deduced to be true. However, in this proof-attempt, the student claimed something other than  $A$ , not sufficiently related to  $A$ , to have been shown true. This could be because the steps of proof by contradiction have been learned in the absence of understanding the underlying logic.

The Nobel-laureate Kahneman[267, 139] highlights the distinction between intuitive and deliberate thought processes. Intuitive thought processes develop from many exposures to relevant data, and include unconscious operations of the mind. Deliberate thought processes consist of steps; the steps were consciously learned, and are consciously executed. Lasry and Mazur[156] point out that when we teach students to apply deliberate thought processes to ideas about which they have already developed some intuition, losses can occur. For example, Lasry cites Hutchison and Elby, who state “observed that students’ likelihood of correctly answering a kinematics question easily solved through common sense depended on whether preceding questions on the survey were designed to prime “sense-making” or schoolish “answer-making” [132].

Common sense answers, extracted from memory built up from many exposures to relevant data, are provided by the “system one” thinking described by Kahneman. Deliberation about some thing ‘a’, during early stages of students’ learning about ‘a’, carried out by “system two”, can be confused. As Lasry states “Only students who had taken physics got this wrong”[156, page 403].

For material about which students may have intuition already, it may be useful to coordinate the existing intuition with the deliberation being learned. Exploring common sense notions at the same time as learning theory-based ideas might reduce confusion that may otherwise ensue.

According to Marton[182], there are cultural differences about the order in which a process to be understood is learned. In some situations the process is expected to be learned by rote, with understanding following and in others understanding is expected first making the procedure easy to remember. Uncertainty about the appropriate time to ask questions can result. Some students have reported that their cultural background discourages asking questions at all; combined with uncertainty about timeliness, this discouragement could overwhelm the desire to get help for understanding.

## **5.5 Interpretation of How Students Attempt to Apply Proofs (When Not Assigned)**

There is insufficient evidence of students applying proofs when not assigned. Even though students had a lot to say about what they think proofs are for, they also did not find that they used proofs when not assigned. Some students said they found coding a preferred technique when proof might have served.



## 5.6 Interpretation of Whether Students Exhibit Consequences of Inability With Proof (such as avoiding recursion)

Students claim to be unsure of when certain algorithmic approaches, such as recursion, are appropriate, and do not think of proof as helpful in a determination. We do know that students say they eschew making use of algorithmic techniques they know, for want of recognizing when these are applicable. Some students say their approach to choosing among algorithms they know includes consulting their relatives and friends. Some students say they never use recursion, and have asked relatives at Microsoft, who have told the students that they do not use recursion. We could in future work seek a dimension of variation about the relative value of the methods students use when they reason about the applicability of algorithms to situations.

We might try to provide Marton and Booth's structural relevance [176] for use of algorithms, arranging algorithms according to some guidance about how generally useful they are, and discuss variations in situations that make algorithms more or less useful.

## 5.7 Interpretation of Student Familiarity with Specific Proof Techniques

Here we interpret the results from section 4.7, see Figure 4.7.1. For the convenience of the reader, the categories were, from earliest to most advanced, manipulation / contrapositive, manipulation and goal revision / contradiction, process / mathematical induction, and other approaches.

The understanding having the least depth contains the appreciation that mathematical symbols are being rearranged, and sometimes transformed. Some students have this understanding, but do not develop a sense of distance and direction that guide them to transformations that connect the hypothesis to the consequence.

A deeper understanding, “manipulation and goal revision / contradiction”, includes insight into changes that may help exhibit the relationship between the hypothesis and the consequence. With this conceptualization students notice that one may bring the consequence closer to the hypothesis.

Another conception, “process / mathematical induction”, is more extensive, because it also includes the ability to generate transformation steps by a procedure. This allows some students to provide valid proof attempts, even in the absence of understanding why these assembled transformation steps prove.

A broader and deeper conceptualization marshals the rules of inference. Some students perceive a frontier, such that, if they were to cross it, they would be able to employ rules of inference as tools to perform valid transformations, in a manner similar to the employment of programming statement sequences to effect functionality.

### 5.7.1 Processes and Conviction

Some students are familiar, in the sense of name recognition, with proof by mathematical induction, contrapositive, contradiction and cases. Some students see these as patterns for proof construction, i.e., steps that can be carried out. Some students are aware that, but are not always possessed of intrinsic conviction that the patterns achieve the desired result. Thus some students do not always achieve confidence in what Carnap[48, p. 21-22] calls the “psychological content (the totality of associa-

tions)” of the consequence of the logical deductions.

Students who rely on the process of generating a proof, without understanding the logic supporting the proof pattern, derive what confidence they have from the source of the pattern, it is extrinsic conviction, in the sense described by Harel and Sowder[118].

When students using this conceptualization subsequently view a proof, its explaining and convincing power is correspondingly diminished.

### **5.7.2 Alternate Dimensions of Variation**

The proposed dimension of variation is that comfort derives from degree of similarity to a process, as in, students are more comfortable with proof by induction because there is a process that they have learned for producing its parts. Students have somewhat less comfort with proof by contradiction, which also has a process, but also has a scope within which a premise obtains, and an outer scope in which it does not. Students have even less comfort with proofs for which “application of logic” is the only guidance they perceive.

An alternate dimension of variation is that comfort derives from the frequency with which they have practiced a certain kind of proof. Students have remarked that they begin a proof by selecting the technique they have practiced most often, without regard for the problem statement.

### **5.7.3 Alternate Critical Factors**

We could emphasize the kinds of transformations that are made. An applicable rule of inference allows us to make a transformation. We might raise the students’ comfort

level by increasing their “sense of direction”, their sense of how distant the presently known representation is from the desired representation, and developing a sense of whether application of any given rule can move the representation in the desired direction.

## 5.8 Interpretation of How Students Use Structure in Proof

Here we interpret the results from section 4.8, see Figure 4.8.1.

For the convenience of the reader, the categories were, from earliest to most advanced, proofs are like programs, being composed of statements; statements carry out process steps; certain patterns of process steps exist (such as proof by mathematic induction); and divide and conquer – solving a problem in parts.

It seems that some students are using their backgrounds from programming in discerning structure in proof. Nevertheless, mention of cases, which appear in the programming construct switch-case, and in proofs, were lacking. Mention of dichotomies, as in the programming construct if-else and in proof by contradiction were missed as well. The situation with proof by mathematic induction and recursion may be useful here. Could it be that the representation of the switch-case construction is sufficiently different that students do not on their own translate from the code representation to the mathematical formulation? The OR-elimination rule from Huth and Ryan[133], for example, might not be emphasized in Discrete Structures.

From Huth and Ryan[133] we see “How did we come up with the proof above? Parts of it are *determined* by the structure of the formulas we have, while other parts require us to be *creative*.”

It would help students to be able to recognize and exploit structure of the statement to be proved. They could be guided by the structure of what they are to prove, in choosing some parts of the proof they are to construct. Therefore it is of interest to know whether they notice structure within example proofs they are shown.

Because some students do not recognize structures, such as implication introduction, in proofs, and do not correlate these with the structure of statements to be proved (“When I need to do a proof I try proof by mathematical induction because we practiced that most often.”) it lends more confidence to the conjecture that certain approaches to proofs are memorized as monolithic processes. It could happen, when learning a proof is scaffolded by the idea of a single (software) function. Scaffolding a proof with a single function can be compared with scaffolding a proof with a program. Similar to a program, with parts that can be assembled, we can consider proofs, where the parts, being smaller arguments, might be thought of as methods within programs. There is some significance to the assembly of the smaller parts. Some part of a plan of assembly may be pattern matching[133], but there may be a requirement for reasoning, for creating an argument. In a similar way software architecture can be designed with modules. We might be expecting students to see this by themselves. There is a generalization step from designing a program to designing a system that performs transformations, and a specialization step that focuses a system that performs transformations onto construction of arguments. Insight into the students’ ease of generalization and specialization, can inform our opinion of how easily students see a correspondence between the architecture of a proof plan and the architecture of a solution in program design.

Students were familiar with the idea of proof by contradiction, though when it was described as preparing a context in which an additional premise was available,

and therefore similar to introduction of special cases, which also introduce contexts with additional premises, students did not find this to be a familiar idea.

One interpretation of this is that some students are comfortable with the idea of proofs having structure, beyond the practiced structure of proof by mathematic induction and proof by contradiction.

When students consider whether a rule of inference is applicable to, and helpful for the transformation of logical representation they are attempting to accomplish, we might ask whether there are two levels of abstraction. One for the rule of inference in isolation, and another for that same rule of inference when in use in the specific proof.

Without abstraction, definitions are more cumbersome to remember and use.

## 5.9 Interpretation of What Students Think Makes a Proof Valid

Here we interpret the results from section 4.9, see Figure 4.9.1. For the convenience of the reader, the categories were, from from earliest to most advanced, that students should know what's true and why (from memory), should reuse proof patterns (such as proof by mathematic induction), should stick to valid rules of inference and should understand the reasoning.

These conceptualizations seem to suggest that students are not comfortable with their ability to take on logical deductions of unspecified nature. Rather, the conceptualizations develop an understanding of restrictions, compliance with which keeps a proof safely valid. The first conceptualization seems daunting in its magnitude, when scaled beyond the specific examples taught in one semester of Discrete Structures, to

the mathematical domain of interest to computer science. This restriction limits the creation of proofs to those that are already accepted.

The next conceptualization is less restrictive, in that it allows modification or adaptation of patterns. The pattern includes its underlying reasoning, and some students do not understand that underlying reasoning. Rather, the conceptualization is that limited modification of the pattern, the safe extent of which students may hope to infer from examples, can be done while maintaining the validity.

The third conceptualization frees the students from the boundaries imposed by using only proof patterns they have learned. It allows them to create proofs using combinations of inference rules. What distinguishes this conceptualization from complete freedom (“understand the reasoning”) is that without understanding the reasoning associated with the sequence of rules of inference it is harder to express intuition in mathematical formulation. The difference may be similar to that between tinkering and designing.

## 5.10 Interpretation of Combined Data

The interpretation of combined data is divided into parts: The interpretation of the analysis from section 4.10 is in section 5.10.1. The combination of the interpretations in earlier sections of chapter five is in section 5.10.2. Comparison of sections 5.10.1 and 5.10.2 is found in section 5.10.3. By performing both interpretations separately, more insight into conceptualizations held by the students might be obtained. The operations of analysis and interpretation might not be expected to commute (in the sense of a commutative diagram from abstract algebra), because selecting data rele-

vant to a specific research question results in less attention to utterances (likewise any student production), that are off the point of the question, and a subsequent process of interpretation would not be expected to restore these discarded data.

### **5.10.1 Interpretation of Analysis from Section 4.10**

Here we interpret the results from section 4.10, see Figure 4.10.1. For the convenience of the reader, the categories were, from earliest to most advanced, did not yet recognize that argumentation by logical deduction is the subject when learning how to prove, followed by the more comprehensive conceptualization that includes the idea of reasoning, but (significantly) encapsulated within processes that can be reused without necessarily understanding; followed by a more comprehensive conceptualization in which proofs can be constructed outside of patterns, but the process of warranting is not soundly understood; followed at last by understanding justification of steps and valuing efficiency.

It seems worthy of note that some students do not see, at least not right away, that argumentation is being taught.

Some students do not pay attention evenly to the material presented in lecture. Some students choose to pay more attention to examples, and even ignore definitions. It can easily be problematic for students if they choose naively the material to which they attend most.

### **Argumentation**

One interpretation, consistent with the data, is that students don't pay as much attention to the nature of a proof as an argument as they pay attention to the apparent



similarity to programs.

When a proof has a form that students can memorize, such as proof by mathematical induction, they can memorize the form and write proofs as sequences of steps, as they describe. When a proof does not have this form, they know they need to use logic, but have been inarticulate about how they use logic. They do not speak of warrants, and they do not mention finding structure, either in proofs they are trying to understand or in statements they are trying to prove, and no student related the structure of the statement they were trying to prove to the structure of the proof they were constructing.

By feeling uncomfortable with logic, not paying attention to definitions, and not noticing structure, and not having the purpose of constructing an argument, they are at a disadvantage in creating proofs.

It could be that students are scaffolding their learning of proof with their knowledge of programming. Though the Curry-Howard isomorphism provides strong support for scaffolding learning of proof by studying how compilers and/or interpreters transform programs from source code into more platform-specific expressions, this is not generally the approach to this scaffolding that students are taking. Instead, we find some students observing that some proofs, for example, proofs by mathematic induction, seem to proceed according to a process. We have seen the student conception “logic proofs”, placed in contrast with proofs, production of which can seem like a process. One significant difference is the creativity called for by the so-called logic proofs, compared with the process-like proofs. Another difference is that a proof constructed by following a process might appear successful in an assessment even though unconvincing to its writer.

Some students report beginning with a premise and trying every transformation

provided by every available rule of inference. This impractical approach, when tried, suggests that some students might not, early in their practice, understand the syntax of the logical formulations with which they are working. This would be consistent with the work of Almstrum[8], and observations by Sheehy (reported herein).

When we ponder a preference for examples over definitions, in the context of scaffolding learning of proof with learning of programming, the utility of examples in learning programming is of interest. Examples, such as “Hello, world” are commonly used in learning programming.

It may be useful to consider an approach to definitions from examples described by Carnap[48, page 137]. He describes starting with the idea of parallel lines in a fixed plane. From the idea of parallel lines, equivalence classes of parallel lines can be obtained. From the idea of differing equivalence classes, the idea of direction can be abstracted.

This furnishes an example of a dimension of variation. By noticing that direction can vary, the concept of the direction of a parallel line is made distinct from the concept of parallel lines. We may conjecture, in the manner of Marton and Pang[177] that, if every parallel line had the same direction, no distinction between lines being parallel and lines having the direction would occur.

If we were to build upon the preference of some students to work with examples, or even to appreciate the task they take upon themselves when they attempt to proceed in this way, we might wish to explore the nature of learning about proofs through examples.

To introduce proof techniques, we use domains such as natural numbers, integers, and graphs. This is deliberate, so that time need not be spent acquainting students with the domain from which examples are chosen. We use these domains in instruction

and assessment.

In later courses we employ these proof techniques with other domains, such as grammars and state machines.

### **Argumentation Without Much Reasoning**

Whether or not students have accepted that reasoning is what they are doing, they can learn to assemble arguments from parts of arguments. Even when they do accept that reasoning is what they want to do, they do not necessarily understand the reasoning that is inherent in the proof pattern they are constructing. Especially when they can get a full score for creating a proof, they might not prioritize finding out what the reasoning is, that makes their successful construction a proof. Even when they want to know how the construction proves the proposition, they may not find time to address their uncertainty.

### **Uncertain About Justification**

For proofs that are not created using a pattern, what some of the students call “logic proofs”, reasoning must be used. In this conceptualization, students understand that individual rules of inference are allowed steps; students may not understand when a rule may be used, or when application of a rule is apt to help them in their goal. Semantics are sometimes helpful in warranting proof steps. When students are not appreciative of the role of definitions, they may be hindered in creating steps in proofs.

## **Have the Idea, Need Practice**

With this conceptualization, students are conscious of deliberate reasoning expressed in mathematical formulation, using inference rules warranted by the circumstances. Some students expressed that they had seen their ability to write proof improve over their academic careers, and expected continued improvement.

### **5.10.2 Combined Interpretations**

Here we combined the interpretations given in sections 5.1 through 5.9 of this chapter.

Some students understand a proof to be a set of statements in mathematical formulation that is convincing to a grader. Students recognize with dismay this extrinsic conviction. Some students do try to comprehend proofs, scaffolding their understanding using their familiarity with code. Some students appreciate that demonstrating properties of algorithms is an important role for proof; some students feel that the code itself is sufficient for this purpose. The difference between an existential and a universal quantifier, though present in some students' understandings in notation, can be problematic in application to specific situations. In this light we can see why some students offer examples in lieu of proofs for propositions making universal claims. Given the concentration on algorithm properties and proof by mathematic induction, combined with the difficulty specializing from notation to specific situations, it is not surprising that the use of modus tollens in some decidability arguments is hard for some students to understand. The reluctance of some students to engage in proof when not assigned deprives them of the use of proof to establish that situations warrant the use of certain algorithmic techniques, such as recursion. Some students have said they do not use recursion, even though they know how, because they do not know

how to be sure it is applicable. Some students do not make much use of the analogy in structure between algorithms and proofs. Though some students readily admit to the utility of switch-case statements and to division into cases, some students do not see that another premise is gained when cases are used, or they do not notice the value in having another premise. Some students understand that rules of inference allow transformation steps, but when there is difficulty specializing from the general notation to a specific use, the understanding of validity can be overburdened.

### **5.10.3 Comparison of Combination Before vs. After Interpretation**

A check was made on whether any difference was found, between the combination of interpretations, which could be thought of as a meta-interpretation, on the one hand, and the interpretation of the combined analysis, on the other hand.

In the interpretation of the analysis of the combined data, we see that argumentation is not as clearly perceived by some students as we might hope. For the meta-interpretation, we see that some students notice that proofs look like sequences of statements in mathematical formulation. One possible explanation of these two outcomes is that they are the same idea seen from two different perspectives – is not always central to students' understanding of what proof is, that reasoning by logical deduction provides the essence of proof.

## 5.11 Report: Navigating Through Space of Conceptualization of Proof

In the Methodology section 3.6, the process used to generate this report was described.

In summary:

- A ‘What if?’ question, the main part of which is a conjecture, is evoked when the researcher is examining the data.
- A conjecture is used to imagine a perspective.
- A perspective is used to predict student productions (e.g., utterances, responses to assignments).
- The predictions are used to estimate the degree of confidence with which we might hold the conjecture.
- With those conjectures about which we have sufficient confidence, the report is constructed.

The report is used for member checking, one activity related to validity. A good report furnishes a thick and rich description of the sending context, so that the reader may judge whether the material can usefully be transferred to the reader’s context.

### 5.11.1 Context

We describe the context, and then describe our students’ conceptualizations.

At a large university in the northeastern United States, in the computer science and engineering department, there are approximately 2700 undergraduate students.

There are over one hundred graduate students. Among the undergraduates, women are quite noticeably in the minority. The computer laboratories are open, staffed with monitors, until midnight during the semester, and is generally lightly occupied at night, but rarely crowded. Around the times exams or projects are due, the population in the lab increases. The students have many clubs and activities, and can be seen using the building's public spaces for extracurricular as well as curricular activities in the evenings. Most of the students live on campus, rather than commuting, and the campus, even after midnight, will have many students walking about, though not nearly as many as in the daytime. Some computer science students form friendships that assist in the formation of teams for group projects. The number of students who are overseen by the Center for Students with Disabilities, for having being located on autism spectrum, seems to be increasing. The economic welfare of the students seems generally to be adequate or better, though the rate of loan accumulation is a source of concern among most students. Many students attempt to study without purchasing or even renting textbooks. While apparently, generally not deprived of food, shelter or clothing, it can often be the case that students deprive themselves of sleep. The number of all-nighters per week is a statistic that is commonly esteemed. Most students seem enthusiastic about their studies. Shared passions for the material accompanies gentle competition. Students help each other prepare for interviews. Students ask others, who are seen to excel in class, for help, and help is generally given. The level of friendly mutual support is similar to that in the mathematics department, and noticeably higher in computer science and mathematics than in physiology and neurobiology, for example. The faculty are friendly and accessible, maintaining open office doors most of the day. Some students have been seen to help each other more than university policy permits. Students often share information,

such as their grades, with each other.

### 5.11.2 Conjectures and Conceptualizations

There is a conceptualization, related to learning proof, that argumentation is not key to proof (see section 4.10.1).

**What if students are trying to learn “what”, but not asking “why”?**

- They would not be thinking in terms of arguments addressing “why”.
- They would be non-plussed about arguments for things they “know”.
- They would not be on the lookout for arguments about “why”, and might have trouble seeing them.

There is a conceptualization, related to learning proof, that a sequence of statements does not require justification (see section 4.10.1).

**What if students had a foundation in process/algorithms and did not have much experience thinking about justifications for specific reasoning steps?**

- They might favor proofs that look like process steps and feel wary about “logic proofs”.
- They might attempt proofs that have some but not all of the process steps from correct proofs.
- They might fail to understand when these process steps are applicable vs. when they are not.



There is a conceptualization that “a proof” can be constructed without benefit of careful definitions (see section 4.10.1).

**What if students were not sufficiently patient with themselves to notice careful distinctions?**

(e.g.,  $\exists n \in \mathcal{N} : \forall m \in \mathcal{N} n \leq m$ .)

- They might not notice the role of “no bigger than” in the phrase there exists a natural number that is no bigger than any other natural number. Instead they might seize on bigger than, and respond that there is no largest natural number.

There is a conceptualization that precise definitions are not important, or that they can be inferred by the student from examples, and need not be checked.

**What if students do not listen with equal attentiveness to all that is said?**

(By their own testimony about not paying attention to definitions and engaging only when examples are given.)

- It might be that they also do not pay attention equally to all parts of a written sentence.
- Attending insufficiently to definitions combined with not placing enough importance upon warrants for next statements in proofs would generate difficulties in the composition of proofs.

There is a conceptualization that proof assignments should not be difficult.

**What if students didn't realize that the work they see is a polished product?**

(What if they didn't know others make many changes on the way to the end?)

- Students are able to comprehend programs written by others, and students notice that in the composition of their own programs, editing may occur.
- Early in their programming efforts, they might worry whether their difficulties signal a poor career choice.

There is a conceptualization that the sequence of statements in a proof can be constructed successfully without reasoning, by following a pattern.

**What if students hoped that process-step proofs would be sufficient?**

- Students might not realize that proving also can involve choices among possible transformation steps, and that false starts and subsequent attempts are normal in the community.

There is a conceptualization that clarity, such as can be achieved with mathematical formulation, is not always necessary, even in proof.

**What if students were not finding a clarity benefit to mathematical formulation?**

- Chicken and egg / boot strapping synergy between clear thought and mathematical formulation. Attention to positioning students onto this self-reinforcing thought pattern. brackets, right associative practice structure

There is a conceptualization that extracting the structure during comprehension, or developing a structure, during synthesis, is not necessarily beneficial.

### **What if students did not notice structure in proofs?**

(By students' own testimony, some are not noticing structure in proofs.)

- Though students notice structure in code, and it seems to make sense they would notice structure in proofs, some do not.

The conceptualization that generalization from a representation, and re-expression of an idea so generalized in another representation is a useful exercise seems to be absent.

### **What if students did not easily generalize from one representation and specialize an idea into another representation?**

- We have investigated, a little, students' and others' facility with switching representations, and given that some very capable students find representation switching to be challenging, it is not that surprising that some students do not immediately see structure in proofs even when they can see and employ structure in code.

There is a conceptualization that negation cannot really be sufficiently difficult to be worthy of much attention.

### **What if students really did have trouble with negation?**

- Some students have trouble with negation, even of statements with only a single quantifier. We might ask, what if students are not thinking in terms of

universals. Students who find that working code is sufficient for the purposes addressed by proofs could be thinking about proofs of existential statements. Proving (not disproving) universal statements might not be of much interest to these students. They are not involved in making proofs of safety.

### 5.11.3 Report

When students first take up proof at university, there are some who are not so sure what the main message is. They might react to arguments about concrete entities that they would have known, with confidence, that conclusion without argument. To some, it is not clear that it is the argument, to which they are trying to direct their attention. Some students use argument forms “It must be  $x$ , because if it were not  $x$  some contradiction would occur.” We can infer from their reactions to the teaching of proof, that it is not the proof technique that claims their attention. (“Do programmers have to know these prime number facts?”) Students have a difficulty moving from the stage of an argument reasoning about concrete entities, to the stage of an argument reasoning about abstract entities. It could be generalization from the concrete to the more abstract, but it could also be, not paying attention or resisting paying attention, or not making the effort to pay attention, to the argument.

Even for students who are conscious that the subject is an argument, they are not so clear how to make one, especially if it is not one of those for which there are process steps (“Logic proofs”). Some students are also not clear on why the product of the process steps forms a convincing argument.

Moreover, not all students are clear that a proof shows unequivocally, suggesting that the proof could be “backed up by experimental data”.

Also, some students say they do not know what they would use a proof for.

If it were not unequivocal, then what benefit would it have over code? Another insight contributing to this view is that students offer alternative activities to proof, that produce examples. When a statement to be shown is an existential statement, an example is fine. For disproving a universal statement, a counterexample is fine. For proving a universal statement, unless it is a generic particular, the example is not adequate. This casts doubt over whether students are really understanding universal quantifiers. There is other evidence that students have trouble with quantifiers, as some students do not succeed in negating statements with quantifiers.

Students have been disappointed that all the applications of proofs that they saw were about things (theorems) that were already known (by someone) to be true. Some students, at least early in their careers as students, think proofs are supposed to show some new fact, and are disappointed to see that the product is (“only”) a different representation of an implied fact. Students seem not to appreciate what benefit a new representation can have.

Also, students do not necessarily see the connections between multiple representations. When data structures are taught with tree diagrams for binary search trees, some students do not know how to represent the same idea in code.

Some students also do not see any connection between the tree diagram and recursive definitions and proof by mathematical induction.

Some students do not see any benefit in spending time on UML<sup>®</sup> diagrams, either class hierarchy, components or sequence diagrams.

Some students learn how to manipulate mathematical formulation without knowing how to apply mathematical formulation to problems they may encounter (such as discovering whether a context supports the use of an algorithm).

Maybe students are not noticing that proofs for universal statements are possible, as well as proofs for existential statements (“Think code is perfectly satisfactory for the purposes for which we have seen proofs applied.”)

Some students are not aware that proof can be used to determine whether a situation is suitable for the application of an algorithm they have learned. One consequence of this is that they make less use of these algorithms they have learned than they otherwise might.

The students who were dual majors in math knew that proofs were arguments by logical deduction, that careful definitions enabled the construction of arguments, and that the results of proof were unequivocal.

One graduate student, whose undergraduate education was elsewhere, who was not a mathematics major, used proofs in papers, because it was required in that publishing venue. She said she did not use proofs at any other time. She developed algorithms without using proofs, and in the process of publishing the algorithms, furnished proofs, adjusting the algorithm if the proof seemed as if it would be too long.

## Chapter 6

# Validity and Reliability

In the first section of this chapter we address the components, in the goals of validity and reliability, that are appropriate, according the Denzin and Lincoln[65], for research in the social constructivist paradigm. In the second section we highlight those portions of our methodology which were directed towards those goals. In the third section we describe these efforts as they took place in support of some published papers and a manuscript in preparation.

### 6.1 Introduction

Several researchers have written advice about how to pursue validity and reliability.

Correspondingly, several requirements exist, at differing degrees of formality, for a qualitative research study to be valid.

Denzin and Lincoln[64, 163, 164, 65] are widely cited; we have adopted their advice. We also follow the recommendations of Merriam[190].

This is intended to differ from quantitative research, in which the perspectives of the majority are thought to predominate, discounting viewpoints contributed especially by minorities.

Following the advice of Merriam[190], in this section we report on how (using the methods described in Chapter 3) we developed a degree of confidence in our findings, through the multiplicity of practices and perspectives we used, including the software programs we used to manage and organize data, the parallel pipelined nature of collecting and analyzing data as the research proceeded, and our style of inductive and comparative analysis.

## **6.2 Validity and Reliability Goals in the Social Constructivist Paradigm**

Denzin and Lincoln [65] state that, for a constructivist paradigm, as our social constructivism is, the criteria are trustworthiness, credibility, transferability and confirmability.

Merriam[188] advises beyond these that a statement of researcher bias assists the reader's assessment of whether an interpretation is transferable to the reader's context of interest.

In this chapter we describe the components of our strategy to make our work valid, reliable, trustworthy, credible, transferable and confirmable.



### 6.2.1 Trustworthiness

In their article “But is it Rigorous: Trustworthiness and Authenticity in Naturalistic Evaluation”, Lincoln and Guba[164, p. 18] state

The axiom concerned with the nature of “truth” statements demands that inquirers abandon the assumption that enduring, context-free truth statements – generalizations – can and should be sought. Rather, it asserts that all human behavior is time- and context-bound; this boundedness suggests that inquire is incapable of producing nomothetic knowledge but instead only idiographic “working hypotheses” that relate to a given and specific context.

They observe that it is the naturalistic inquirer’s obligation to provide a “thick description” of the sending context. What is “thick” enough is, that it enables a comparison between the context in which the data were found with a receiving context, in which the interpretation might be applied.

The criteria making up trustworthiness are credibility, transferability, dependability and confirmability[164, p. 18].

### 6.2.2 Credibility

Some criteria for judging the credibility of interpretations include[231] persistent observation and prolonged engagement, triangulation (as can be obtained by asking others for their views) and searching for evidence that contradicts the interpretation.

We observed student work and conducted interviews over a three year period. For triangulation, we used homework, help session, test and interview data, we talked

with professors and teaching assistants as well as with students.

To check upon inclusiveness, we counted our volunteer population according to the demographics reported by our university, see section 3.3. The demographics of our department differs from that of the university as a whole (for example, the percentage of women is smaller). We used theoretical sampling to apparent saturation (see section 3.4). We used memos (see section 3.5) and constant comparison (see section 3.5), we used diverse sources (interview, homework) (see section 3.4).

We performed member checking (see section 3.5). This resulted in affirmation about students might be trying to learn “what” instead of “why”. They “don’t care why, as long as it works”.

We discussed preliminary findings with colleagues (peer examination); they found our results believable. Our findings were compatible with those of peer reviewed literature in math and computer science (see chapter 7 on Related Work). We describe our population and context so that readers may judge transferability to their contexts.

### **6.2.3 Transferability**

Lincoln and Guba [163, p. 125] state that a “thick description of the sending context” supports the ability of a reader to decide to what extent any given study may be applicable to a receiving context.

This description is found in Chapter 3.

Our population, sources and analysis methods have been described in Chapter 3, to aid readers in judging the extent to which the work may be generalizable.

### 6.2.4 Confirmability

Krefting[151, p. 217] lists means of approaching confirmability. These include keeping an audit trail, using triangulation and reflexivity.

For audit trail, our methods are described to an extent that permits a replication study.

Our triangulation is described in section 6.3.

Krefting's[151, p. 218] reflexivity seems to match Merriam's[190] statement of researcher bias. We provide this in section 6.5. Our interview protocol is available. Homework assignments included proofs of concepts treated in popular textbooks in Discrete Mathematics, such as that by Epp[85] and Hammack[112]. Some confirmation has already been obtained, by consistency with related work.

## 6.3 Approaches

In this study we applied triangulation in several ways. We interviewed faculty teaching the courses involving proofs. We interviewed TAs assisting in the courses involving proofs. The students in these courses are from our same population. To get an idea of the background preparation of our students, we substitute-taught geometry and algebra II classes in a high school. The high school population was quite similar to our university population, but differed by consisting almost entirely of domestic students, studying in their first language, and by having a larger percentage of women students, and of declared transgender students. Though the community served by this high school is diverse over socio-economic status, this component of diversity is probably greater in our university population.

Consistency with the work of other researchers is a check on the validity of an analysis.

In this study we compared our results with those achieved by some other researchers in computer science education and also by some researchers in the mathematics education community.

Checking possible interpretations is a technique that may aid in increasing confidence in validity.

We prepared a list of questions that was addressed by several faculty and several students, that began an examination of the role of specific representation styles (mathematical notation, figures and pseudocode) for proof related problem statements (see appendix B). Results were reported in the combined data (see section 4.10.1).

We used member checking of the summary report to contribute to validity.

### **6.3.1 Purposefully Seeking Diversity**

We sought diversity in participants: students, TAs, instructors; in sources: interviews, homework, tests; over a period of time in two ways, by talking with new students, longer term students, employed former students, and students who left the major and by spending several years in the investigation. Participants included persons identifying male and identifying female. Participants included persons who were domestic students, and also international students. Participants included students who remained in the major and some who left. Participants included students who had been enrolled in school since childhood, and those who had taken time off. Participants included current students, and employed former students. Participants included students who were enrolled as students of computing and not mathematics, and stu-

dents who were also enrolled in mathematics. Participants included undergraduate and graduate students, and also professors. The teaching assistants and professors were used as diverse observers.

### **6.3.2 Diversity in Participants**

Having seen students in minority groups struggle, possibly related to cultural sensitivities, with proof, we care about the welfare of women, Latino/as and persons of color in computer science. We included participants from each of these groups. Noting that some of our Latino fellow students were declining offers for help, we inquired among friends and learned that consultation is a more culturally sensitive term. Whether or not a student was a mathematics enthusiast was the only noticeable factor that differentiated the answers, in retrospect, of our participants.

The literature of mathematics education includes work on students' learning about proof. Our work with computer science students has benefited from having some participants who are dual majors of math and computer science. This offered us hints about similarities and differences between students having more or less math background. The significance of definitions, the necessity and utility of proof, appeared different in these groups.

### **Theoretical Sampling**

Theoretical sampling[102] is a method of proceeding in qualitative investigation. By performing some data collection and beginning analysis as soon as data are available, one obtains the opportunity to allow earlier data to guide later collection of data. Institutional Review Board (IRB) procedures must be followed. Therefore, either

the protocol is sufficiently broad to enable variation in the collection, or multiple protocols must be approved, to cover the range of questions that the investigation develops.

## **Saturation**

Saturation is a state of affairs, that arises in co-occurring data collection and analysis, in which the incorporation of new data no longer results in collection of new ideas. By collecting and analyzing data into saturation, we hope to avoid missing any conceptualizations.

With both the interviews and the documents, towards the end of collecting data the amount of material that sounded different dwindled. As analysis continued, the number of newly needed codes became reduced, and the rate of category creation vanished.

### **6.3.3 Diversity in Sources**

We combined interview data from students, TAs and faculty, and homework. We also included anonymous, aggregate data from classes and consultation sessions.

Interviews of students formed the core data. We supplemented these with interviews of faculty and teaching assistants. We compared these reports with evidence from homework, and written discussion by students. With gratitude to the participants, we were able, as reported in section 3.4 to interview students from any differing cultural, genetic, and international heritages, and economic backgrounds.

## Interviews

Part of attaining validity is bracketing away researcher bias. If the research were more reflective of researcher bias than of data, surprise would rarely occur in the researcher.

Vignettes of validity include surprise associated with an outcome. The experience of surprise contradicts the concern that researcher bias has merely obscured what information the data might contain. For example, the earlier interpretation, that students were experiencing difficulty as they tried to generalize from an argument bound to concrete elements to an argument with free variables, to an argument bound to ideas that were more abstract, was replaced with a more surprising insight that some students are not engaging with the nature of argumentation as a means of convincing.

It was a surprising idea that these students are attempting to verify the correctness of a conclusion, and with concrete entities, they have their own wellsprings of confidence about the correctness of the conclusion and disengage from the argumentation. However, this does not give them a bridge to understanding an argument with free variables or variables bound to more abstract concepts.

## Documents

The use of documents is thought to increase the multiplicity of our sources, thereby being a source of triangulation according to Denzin[64].

One reason why homework may differ from interviews is the students' objectives.

What students write on assessments, such as homework, is taken by us to be the students' best efforts to earn credit, and therefore a reflection of what they hope

is true. Students in interviews have been assured that revealing what they do not know will not hurt them, and students have benefited from revealing what they do not know, because they have been given explanations in a tutoring situation. For tests and homework it is probably the case that revealing what they do not know is thought to be counterproductive.

### **Other Sources**

Crosschecking among sources, as described in section 3.5.1, is thought to be, like constant comparison, a source of validity. We crosschecked our own observations with those from faculty members having these students in courses in which proof was relevant. We crosschecked with faculty of different institutions. In both cases we found agreement and inspiration, but no disagreement. We checked the published literature, to see whether findings were conflicting or compatible. Here also we found only agreement.

### **6.3.4 Alternative Explanations**

The exercise of seeking alternative explanations was very useful. It forced us to find a differing perspective. This other perspective inspired new ideas.

### **Looking for Supplementary Explanations and Supporting Data**

According to Patton[199], credibility is gained by considering alternative explanations. As seen in Chapter 5, our interpretation was carried out with concentration on finding alternative explanations. For the question what is proof, in which we sought to gain insight into some students not being able to generalize from an argument about



concrete entities to the form of the argument, the search for alternative explanations allowed to appear the supplementary interpretation that with concrete entities, some students might abandon paying attention to the argument as a source of confidence, substituting their own reasons for believing the conclusion.

### 6.3.5 Consistency Checking

The consolidation of collected diverse material into an internally consistent and externally believable whole is a product of qualitative research.

Merriam[190] has recommended steps for achieving this.

- Merriam[190, p.209] states: “Ensuring validity and reliability in qualitative research involves conducting the investigation in an ethical manner.” We conducted our research under the supervision of the Institutional Review Board at the University of Connecticut, under several, similar protocols, H13-065, H14-112, and H15-022.
- Merriam[190, p.210] states: “Regardless of the type of research, validity and reliability are concerns that can be approached through careful attention to a study’s conceptualization and the way in which the data are collected, analyzed, and interpreted, and the way in which the findings are presented.”
- Adopting Denzin and Lincoln’s[65] metaphor of quilting, we mentally check whether the quilt obtained by piecing together our insights from the analysis of interviews and documents, and from the viewpoints of students, assistants and instructors is connected and coherent and covers the space we are examining.

- Good practice[65] requires that we include the perspectives of all members of the domain for which we claim to apply. Moreover, no members of this group should be marginalized. This deliberate seeking of differing perspectives is called theoretical sampling. The process of seeking these perspectives is judged concluded when additional data seem to produce no additional codes. This condition is called saturation. (Addressed for this work, in Chapter 3.)
- A diversity of assessment methods should be used. (Addressed for this work, in Chapter 3.)
- Adherence to a methodology, that should be described in sufficient detail that readers could imagine carrying out the method, contributes to validity. (Addressed for this work, in Chapter 3.)
- Production of a report, that contains a summary description[65] of the results, contributes to validity. (Addressed for this work, as “Navigating Through Space of Conceptualization of Proof” in section 5.11.)
- Participant assessment of that summary, such that the authenticity of representation of the domain being described is confirmed by the people in that domain, contributes to validity. This process is called member-checking.
- Internal consistency, such as checked by constant comparison, and also external consistency, such as checked by others with a view to related data (in this case, several instructors experience the students’ use of knowledge about proof).
- There has been much discussion[190, 163, 302] about which terms should be used to address the concept of the bases for confidence in the results of qualitative

study. We choose the terms validity and reliability, and attempt to illustrate what we mean by those terms.

- In this research, we hope to do what Denzin and Lincoln [65] describe as: “secure an in-depth understanding of the phenomenon in question”.

## Peer Review

Merriam[190, p. 220] states that peer review is a useful source of validation. Some papers resulting from this study have been published in conference proceedings. Validation method applied and reported in those papers is summarized below.

## 6.4 Specific Papers and Manuscripts

For some published works and a manuscript in progress we can describe approaches to validity.

### 6.4.1 Validity and Reliability in Proofs Using the Pumping Lemma for Regular Languages

In this paper, we proposed possible student conceptualizations that predicted the errors we saw. Pros and cons for validity for these proposed explanations were considered. One conceptualization seemed to be that universality of a statement was not noticed, or not regarded as significant. When a universal claim (for all  $a \in A$ ) is treated as if it were an existential claim (there exists  $a \in A$ ), the application of unwarranted restrictions (“Let  $x$  be empty”, “let  $|xy| = p$ ”) is no longer problematic. We predicted from this conjecture about insignificance of universality that students

would offer examples as proofs, would consider experimental evidence as helpful in the presence of a proof, and would consider an algorithm or code demonstrating a result as adequate. All of these predictions were consistent with our data and the data of others. We check whether there are multiple variations of error that might be explained by lack of appreciation of this components of the material of discrete systems, and found some possibilities.

Some support for the validity of the results comes from seeing several variations of each proposed error type. We found in our data multiple versions of unwarranted restrictions: choosing  $x$  to be empty or choosing the length of  $xy$  to be  $p$ , and others. We found in literature warnings against attempts to prove statements with universal qualifiers true by means of showing the existence of examples [68, 91]. These warnings suggest these errors have occurred before.

We also found in our data, several versions of misunderstanding inequalities. We found support in literature for errors of misunderstanding how to work with inequalities, by students of this level[184]. Difficulties with inequalities is proposed as a useful grouping of several errors we found while examining documents. We would like to consider inequalities with student having difficult expressing ideas in a different representation from that in which the ideas were taught. Inequalities are often illustrated. It may provide some assurance of validity, when proposed conceptualizations, obtained from different research questions, coordinate well together to explain student progress.

Student difficulties with the pigeonhole principle offers an example of this synergy-based validity. Students learning the pigeonhole principle sometimes apply a restriction that pigeons must be divided as evenly as possible among the nests. They are understanding that no later than when the number of pigeons exceeds the number

of nests, sharing must occur. They use the evenly divided idea to find the first time at which we are certain that sharing or reuse occurs. They do not always recognize that reuse can occur sooner than that earliest time at which we are certain it must have occurred. So, we find evidence of this key idea of inequality being problematic in multiple contexts. Representation and inequality are acting with synergy in this explanation.

Variation theory supports our observation that comparing and contrasting fine distinctions in material being taught aids the process of learning. We used the difference between assignment and equality testing, manifest in the java expression of “==” vs. “=”. We compared a software procedure representation with a mathematical formulation (the latter using only “=”), for comprehensibility by students. This helped us to see that barriers to student understanding exist, for some students of computer science, at the level of formulation. It also helped us see that the barrier between the internalization and interiorization of Harel and Sowder[118] might be less of a barrier in students of computer science who are routinely conscious of the need to analyze procedures.

### 6.4.2 Proof by Induction

Here we see agreement among participants, and a spectrum of depth of understanding, which lend confidence to the interpretation.

We were encouraged by the overlap in description among interview participants. The interviews were certainly not the same, but common elements, specifically that there is a form to proofs by induction, appeared. Some students referred to this form as steps, others as a procedure, or framework. Moreover, degrees of understanding

filled in a spectrum, from joyful deep understanding to admissions of not understanding why the steps of proof by induction prove anything, and conceptions in between. These included a supposition why a proof of an induction step would, in combination with an established base case, constitute a proof by induction, that its originator characterized as “weird”.

## 6.5 Statement of Researcher Bias

The researcher believes that students who are able to be successful in other courses in computer science will be able to learn the basics of proof, if they are willing to pay attention to all of the material presented in class, and carry out a reasonable number of well-chosen exercises. The researcher finds differing mental experiences of the same information, presented in multiple representations, such as linear temporal logic and Büchi automata, or pseudocode and mathematical formulation, fascinating. The researcher believes that emotional states and quality of sleep, in the students can have a significant effect upon the student’s ability to understand and to remember.

## Chapter 7

### Related Work

The search for related work was conducted with the purpose of being informed about what had been learned before, and what recent investigations were being carried out, especially if it were helpful to our goal of understanding the students' conceptualizations of proof. Boote and Beile's work[33] was helpful in evaluating this chapter as it developed.

The organization of the literature review is:

1. Research questions
2. Phenomenography / Variation Theory for theoretical framework
3. Social constructivism for epistemological perspective
4. Qualitative inductive analysis in phenomenography and variation theory for methodology
5. Mathematics education

## 6. Computer science and engineering education

**Research Questions** The focus used for literature search for research questions was to ascertain whether these questions were already addressed. The goal for literature search for research questions was to find an area related to these questions, that remained to be discovered, and promised to suggest improvements in pedagogy. What was achieved toward this goal was identification of an area for contribution, and a publication[245]; work that connected literature found in mathematics education with that found in computer science and engineering education, offering an explanation of the previous findings[34] in computer science and engineering by means of identifying in our computing student population, a phenomenon previously noticed[243, 7] in students of mathematics. More progress on this goal appears possible. The perspective on the research questions includes that students may not necessarily avail themselves of the socially-mediated opportunities for learning, and that this forbearance may depend upon cultural and emotional factors. The coverage on the research question literature is purposeful sampling.

**Theoretical Framework (Phenomenography and Variation Theory)** The focus used for literature search for theoretical framework was to learn the theory, including its historical development. The goal for literature search in the theoretical framework was to understand the strengths of the theory, and the domain in which these strengths could be expected to manifest. What was achieved toward this goal was to understand the theory well enough to see its utility in gaining insight into the learning of our students. More progress on this goal would extend the domains



to which it applies in our students. The perspective on the theoretical framework includes that it can be enriched by pursuing its relations with cognitive neuroscience. The coverage on the theoretical framework literature is purposeful sampling, beginning with the early work of the founders, namely Marton and his associates, and continuing by both `scholar.google.com` search and following the trail of references.

**Epistemological Perspective (Social Constructivism)** The focus used for literature search for epistemological perspective was to learn the theory, including its historical development. The goal for literature search in the epistemological perspective was to gain an understanding based on the work of the originators, and the development of this theory, of the meaning obtained by more modern applications. What was achieved toward this goal was to appreciate the work of Piaget and the neo-Piagetians, which gives educators insight into the influence of knowledge in place in the student upon any attempt at building new knowledge. More progress on this goal would focus on the use of analogy, such as building up from (student-preferred) examples into abstractions, using non-examples in the context of variation theory, to guide the abstraction building of students. The perspective on the epistemological perspective includes that there may be a preponderance of contributions from researchers who prefer socio-cultural cognition theory to cognitivism. The coverage on the epistemological perspective literature is purposeful sampling, beginning with the early work of the founders, namely Piaget and Vygotsky, and continuing by both `scholar.google.com` search and following the trail of references.

**Methodology (Phenomenographic / Variation Theoretic Analysis)** The focus used for literature search for methodology was to understand its practice, especially the ways in which consideration of validity guide and constrain methods. The goal for literature search in the methodology was to master the methods for application to our research questions such that we would obtain results with a validity acceptable for respected practice. What was achieved toward this goal was to be able to execute, modify and explain this methodology to a degree that was enough to result in published papers. Finding results similar to those in published papers by using these techniques has been reassuring. More progress on this goal might include learning how to incorporate recent results from cognitive neuroscience, for example, see Ansari and Lyons[11]. The perspective on phenomenographic methodology is that it bears some similarity to unsupervised machine learning, in that categories are developed and features that distinguish categories are developed, from the data. The coverage on the methodology literature is purposeful sampling, beginning with the textbooks recommended in a year-long course offered in the Neag school of education at the University of Connecticut, namely Merriam[189, 190], Wolcott[302], Creswell[58, 59].

**Mathematics Education** The focus used for literature search for mathematics education, which is extensive, was to find widely cited studies investigating work similar to ours. The goal for literature search for mathematics education was to find descriptions of conceptualizations of proof among mathematics students at a level of advancement similar to that of our computing students. What was achieved toward this goal was to find much useful, insightful work, while always becoming aware of

more. More progress on this goal can readily be achieved by continued reading, especially seeking techniques that have shown promise, such as those of R. L. Moore[137], the van Hiele[272], Fawcett[88] and Schoenfeld[229, 230]. The perspective on the mathematics education literature is that it provides a wealth of insight. The coverage on the mathematics education literature is purposeful sampling, beginning with textbooks (Krantz[150, 148, 149], Dean and Hinchey[63], Hanna, Jahnke and Pulte[115], Reid and Knipping[216], Stylianou[252], Polya[208, 210, 209], Chartrand Polimeni and Zhang[49], Devlin[68]), work of some widely-cited authors (Tall[254], Harel[118], Weber[296], Ball[251], Moore[137]) and continuing by both scholar.google.com search and following the trail of references.

**Computer Science and Engineering Education** The focus used for literature search for computer science and engineering education was on phenomenographic studies of any topic, taken in union with any work on students understanding of proof. The goal for literature search for computer science and engineering education was to find a way to situate this research in the work of this community. What was achieved toward this goal was some publications[245, 244]. These publications show that the investigation into our students' understanding of proof has been practical; students unconvinced by proof-attempt activities do not find proof a helpful tool in deciding whether an algorithm they have learned is application to a problem situation with which they are faced. More progress on this goal would be following the literature, more work and more publications. The perspective on the computer science and engineering education literature is that it provides a wealth of insight. The coverage on the computer science and engineering literature is purposeful sampling, beginning with

current ICER and Koli conference papers, and continuing by both scholar.google.com search and following the trail of references.

## 7.1 Summary

Mathematics education literature is useful in understanding the conceptualizations of our students. There is more. Our computing students differ from the students described in the mathematics education literature. There are strengths, such as exposure to algorithms and coding, that we can exploit. There are weaknesses, such as a lack of enculturation about proof and definitions, that we can remediate.

As we help our students make sense of instructional material on proof, we can warn them about being satisfied by learning a procedure, and urge them to describe the warrants between proof transformation steps. Ulricksen et al.[269] gave us the felicitous phrase “Weaving a Bridge of Sense”. We want to weave a web of sense that incorporates the strands of argumentation (warrants), mathematical formulation, insight from diagrams and transformations between representations, so that students may use this web to capture, evaluate and use ideas of proof to understand, apply and synthesize algorithms.

We note some remarks of Balacheff[20] bearing on the choice of theoretical framework and epistemology for qualitatively investigating students’ conceptualizations of proof: “Whether we consider mathematical proof as a universal and exemplary type of proof (1), or being first of an idiosyncratic nature (2), at the core of mathematics (3), or a tool needed by mathematics (4), getting its meaning from applications (5) or being specific to mathematics as an autonomous field (6), makes a big difference.

These views witness radically different epistemologies of mathematical proof, they correspond to very different understandings of mathematical proof in a teaching-learning perspective and hence they will determine the choice for very different research programmes, research design and, above all, radically different understanding of what students could produce. Indeed, we cannot avoid involving in our work our own epistemology of mathematical proof, and beyond our own epistemology of mathematics. But if we are not aware of the differences among these epistemologies and the implications of these differences on sharing theories and methods, problems and results, these epistemologies will become the essential obstacle to making progress in our field of research.”

Because our goal has been to inform supposed later efforts curriculum and instructional design in computer science and engineering, with information about the conceptualizations that students bring to courses, we chose a phenomenographic study. Phenomenography supports us in discovering the conceptualizations found in a cohort of students, and making sense of that information in a way that leads directly to suggestions for helping students discern what is needed for advancement.

## 7.2 Research Questions

Zendler et al.[305] wrote in 2008 about central concepts in computer science education. The only mention of proof there is within an example taken from mathematics education. Schwill[233, 232] wrote about fundamental idea in computer science education, mentioning proof of worst case execution time in analysis of algorithms. Petrovic[200], and Simic[236] wrote about using incremental refinement proof assistant software to

help students learn to prove correctness of algorithms.

Schäfer et al.[228] interviewed students of computer science, to learn their difficulties. Mathematical logic was the topic they found to be most significant difficulty. Within this domain they found[228, p. 89] that “the main difficulty lies in coming up with the right idea how to solve the problem and which proof method to apply.” They also found that students experienced time pressure using proof techniques they knew. Subsequently they built a game as an instructional tool, and evaluated it. Enström[82] sought to “understand what subject matter is particularly difficult to students.” This is in the context of students of theoretical computer science. Enström discovered the top three difficulties to include proving correctness and performing reductions.

Knobelsdorf et al.[146] wrote, of students learning and using proof in an introduction to theory of computation course “most students remained very passive during the student sessions and did not participate in discussions even when their own solutions contained mistakes”.

Based on recent work it appears that more remains to be done in finding out what conceptualizations of proof are found in groups of students of computer science and engineering.

## 7.3 Constructivism

Von Glasersfeld[277] credits Vico[275] as the first true constructivist, quoting him as saying “human truth is what man comes to know as he builds it”.

Locke[169, Book I Chapter 4 §2] foreshadowed the work of Piaget, writing “If we will attently(sic) consider new born children, we shall have little reason, to think,

that they bring many ideas into the world with them. For, bating, perhaps, some faint ideas, of hunger, and thirst and warmth, and some pains, which they may have felt in the womb, there is not the least appearance of any settled(sic) ideas at all in them.”

A major figure in constructivism, Piaget was an evolutionary biologist[203, p. 26]. He created theory about how human intelligence develops, from the earliest post-natal life, always building upon what was already present, supporting his theory with years of meticulous observations.

Piaget and Beth[28, p. 195] wrote that ontogenetic construction of evidence of a new domain integrates the former domain as subdomain.

This natural learning process, in which the new is accommodated by means of what has gone before, may be quite relevant to our students’ preference to receive new proof-related ideas not as abstract members of a class being defined, but as concrete entities serving as examples.

Locke discusses abstraction[169, Book II, chapter XI, §9] “the mind makes the particular ideas, received from particular objects, to become general; which is done by considering them as they are in the mind such appearances, separate from all other existencies, and the circumstances of real existence, as time, place or any other concomitant ideas. . . whereby ideas taken from particular beings, become general representatives of all of the same kind”

Piaget described a process called *reflective abstraction*.

Harel and Tall[120, p. 39] wrote “An abstraction process occurs when the subject focuses attention on specific properties of a given object and then considered these properties in isolation from the original.”

Moström et al.[197], Eckerdal et al.[76], Kramer[147] and Devlin[67] wrote about

the significance of abstraction for students of computing. Kramer wrote “Both the ability to abstract and the ability to move flexibly from one level of abstraction to another are key skills in computer science” [147].

Hazzan and Dubinsky [123] wrote “since abstraction can be addressed on different levels, the shift between different levels of abstraction can also support problem-solving process. However, the knowledge of how and when to move between different levels of abstraction does not always come naturally, and requires some awareness.”

K. C. Moore has written about a student augmenting conceptual structures [195], using multiple representations. He states “One way to explain how the diagram aided her understanding of the situation is that Marie constituted her diagram in terms of quantities and relationships between quantities that she then symbolized using equations and variables, whereas her initial approach to the problem was entirely symbolic and devoid of imagined quantities and relationships.” [196, p. 373] He goes on to say “That is, reconstructing, recognizing and re-presenting quantitative structures enabled her to construct a sense of invariance or similarity across the tasks, eventually leading to her anticipating the implications of this structure.” Marton and Pang [177, p. 219] referred to representation stating “It seems that to experience the effect of two simultaneously changing variables (supply and demand) on a third variable (price), it is an advantage for learners to be exposed to the different cases as represented in one diagram rather than separate diagrams. It is a further advantage for learners to be exposed to the changes represented as movements in a diagram (i.e., being represented dynamically) rather than simply being exposed to the changes as differences between different cases in the same diagram (i.e., being represented in static form).”

Harel and Sowder [119] pointed out that there is a difference between what they call *non-referential symbolic* proof scheme and *referential symbolic* proof scheme. In the



former case, students carry out transformations, but there is no potential coherent system of referents for their symbols and operations. When our students cannot translate from one representation, such as a diagram, or pseudo-code, to another, such as a mathematical formulation, they might be restricted, when using mathematical formulation, from more advanced understanding than the non-referential symbolic stage.

Lesh and Harel have studied the ability of some students to develop models, an activity exercising abstraction [161]. They state

Results show that, when problem solvers go through an iterative sequence of testing and revising cycles to develop productive models (or ways of thinking) about a given problem solving situation, and when the conceptual systems that are needed are similar to those that underlie important constructs in the school mathematics curriculum, then these modeling cycles often appear to be local or situated versions of the general stages of development that developmental psychologists and mathematics educators have observed over time periods of several years for the relevant mathematics constructs. Furthermore, the processes that contribute to local conceptual development in model-eliciting activities are similar in many respects to the processes that contribute to general cognitive development.

Kuhn et al.[152] reported from a sample of 265 subjects, that “Only about 30% of adults had completely achieved the transition to consolidated formal operations. Most remained transitional between concrete and formal operations; about 15% showed no formal thought at all. The longitudinal study confirmed that early adolescence is a

period of emergence and development of formal operations.”

This helps set the context for the results of Almstrum[8], that high school students of computing have more difficulty with problems involving logic than problems not involving logic. Herman et al.[127] wrote that “The results of this study demonstrate that students who passed digital logic classes with grades of B and C are unable to solve basic conceptual problems even shortly after completing a digital logic class.”

Devlin[67] observed that study of mathematics addresses students’ learning of abstraction. Hazzan and Dubinsky[122, 123] discussed exercising abstraction with software engineering.

Moström et al.[197] gave examples of student improvement during a course on object oriented programming.

Reflection is a component of constructivism, in that it contributes to the construction of conceptualization. Locke[169, Book II, chapter I, §2], in 1690, described reflection as ideas that the mind gets by reflecting on its own operations within itself. “experience: In that, all our knowledge is founded; and from that it ultimately derives itself. Our observation employ’d either about external, sensible objects; or about the internal operations of our minds, perceived and reflected upon by ourselves, is that, which supplies our understandings with all the materials of thinking.” and “reflection, the ideas it affords being such only, as the mind gets by reflecting on its own operations within it self. By reflection then, in the following part of this discourse, I would be understood to mean, that notice which the mind takes of its own operations, and the manner of them, be reason whereof, there come to be ideas of these operations in the understanding. These two, I say, viz., external, material things, as the objects of sensation; and the operations of our own minds within, as the object of reflection, are, to me, the only originals, from whence all our ideas take

their beginnings. The term operations here, I use in a large sense, as comprehending not barely the actions of the mind about its ideas, but some sort of passions arising sometimes from them, such as is the satisfaction or uneasiness arising from any thought.”[169, Book II, chapter I, §4]

In the light of modern day insights into reflection, including Rota’s observation about the importance of beauty[219], and Amalric and Dehaene’s localization of mathematical comprehension away from verbal processing areas of the brain[9], Locke’s comments seem outstandingly prescient.

Locke pointed out the importance of attention[169, Book II, chapter I, §7] “they may come in his way every day; but yet he will have but a confused idea of all the parts they are made up of, till he applies himself with attention, to consider them each in particular”.

Locke addressed getting, and keeping, content in memory:[169, Book II, chapter X, §3] “Attention and repetition help much to the fixing any ideas in our memory: but those, which naturally at first make the deepest, and most lasting impression, are those, which are accompanied by pleasure or pain.” and [169, Book II, chapter X, §4 & §6] “But concerning the several degrees of lasting, wherewith ideas are imprinted on the memory, we may observe, first, that some of them ... quickly fade ... those that are oftenest refreshed (amongst which are those that are conveyed into the mind by more ways than one) by a frequent return of the objects or actions that produce them, fix themselves best in the memory, and remain clearest and longest there.”

Locke touched on ideas emerging into consciousness:[169, Book II, chapter X, §7] “sometimes too they (ideas) start up in our minds of their own accord, and offer themselves to the understanding; and very often are rouzed and tumbled out of their dark cells, into open daylight, by some turbulent and tempestuous passion,

our affections bringing ideas to our memory, which had otherwise lain quiet and unregarded.”

Locke wrote on distinct and confused ideas[169, Book II, chapter XXVIII, §7] pointing out that incomplete definitions (“too small a number of simple ideas”), because they fail to distinguish classes that are intended to be distinguishable; disorderly definitions (“the particulars that make up any idea, are in number enough; yet the are so jumbled together”); definitions composed of ill-defined terms (“when any of them is uncertain, and undetermined”) are sources of confusion.

Harel and Tall[120, p. 39] wrote “The process of formal definition in advanced mathematics actually consists of two distinct complementary processes. One is the *abstraction* of specific properties of one or more mathematical objects to form the basis of the definition of the new abstract mathematical object. The other is the process of *construction* of the abstract concept through logical deduction from the definition.”

Truth and falsity, the atoms of logic, are introduced by Locke[169, Book II, chapter XXXI, §7] as applying to ideas. Moreover, in this connection, it is only when the relation of a specific idea to a specific external entity is formed, that ideas obtain the property of truth or falsity. Thus one’s ideas are unassailably right, if no relation with outside entities is supposed.

Locke wrote on abstraction [169, Book II, chapter XXXI, §6], saying that it occurs naturally, readily “the first thing it does, as the foundation of the easier enlarging its knowledge, either by contemplation of the things themselves, that it would know; or conference with others about them, is to bind them into bundles, and rank them so into sorts, that what knowledge it gets of any of them, it may thereby with assurance extend to all of that sort; and so advance by larger steps in that which is its great

business, knowledge. This, as I have elsewhere shewed, is the reason why we collect things under comprehensive ideas, with names annexed to them into genera and species; i.e. into kinds and sorts.”

Mason claimed that abstraction is key “Students of mathematics often say that they find mathematics abstract, and give this as the reason for begin stuck, for disliking mathematics lessons, or even for withdrawing from mathematics altogether. Yet the power of mathematics, and the pleasure that mathematicians get from it, arise from precisely the abstract nature of mathematics. ...abstracting lies between the expression of generality and the manipulation of that expression while, for example, constructing a convincing argument. In that ever so delicate shift of attention occurs the drawing away of form from the sensible”[183].

According to Davidson[62], who stated “These semantic considerations are relevant to constructivist epistemology, in which the role of action is fundamental”, there is an interpretation of category theory associated with Goldblatt[106] and Lawvere[157], in which the arrows of category theory connote activity and emergence. Davidson explained how Piaget used category theory in this sense to describe how students reflect, and perform abstraction; they use morphisms of a category as comparisons that draw attention to correspondences among patterns. Construction of correspondences can occur through comparisons and analogies[204].

Harel and Tall[120, p. 40] wrote “we may help students ...attain the reconstructive generalization required for the formal abstraction. ... by focusing on a mid-way development in which a specific example is seen by the teacher as a representative of the abstract idea, which we term a generic example.”

Davidson [62] related “Piaget and his colleagues claim precocious attainment of several types of conservation when young children are trained by techniques that draw

attention to these morphisms, such as slowly transforming the material by removing parts from one side and reattaching them to the other side. Training that draws attention to children's own actions of inducing such correspondences is particularly effective[201]. Davidson stated that "Nor should it be inferred that the categorical formalism is applicable only to competencies of early and middle childhood. Much of its attractiveness is precisely that it puts cognition from infancy through adulthood in reach of a single formal theory." [62, p. 231] Goodson-Espy wrote "In order to encourage these reflective abstractions, one has to place careful emphasis on offering students appropriate mathematical learning tasks....radical constructivism adherents would argue that the solver creates a unique mental structure through her own activity." [108] According to Goodson-Espy, Cifarelli[54] "defined levels of reflective abstraction attained by college students ... Recognition, Representation, Structural Abstraction, Structural Awareness." Cifarelli gave eight levels of solver ability. Dubinsky et al.[73] described an idea of reflective abstraction, namely "A *schema* is a more or less coherent collection of mental objects and mental processes for transforming objects. When faced with a new situation or, what we may refer to in mathematics as a *problem*, and individual is said to be *disequilibrated* and may attempt to *reequilibrate* by *solving the problem*. The process of *equilibration* results in the construction or reconstruction of schemas."

Fuys, Geddes and Tischler[96] wrote about the Van Hiele model of thinking in geometry among adolescents. Informal reasoning in two earlier levels transforms into rigorous study of axiomatic geometry in two more advanced levels. One very significant aspect of the van Hiele model is that people operating at different levels have difficulty communicating, resulting in failure when the teacher uses "the language of a higher level than is understood by the student." [96, p. 7]. Types of instructional

experiences can affect progress. Progress from one level to the next is thought to occur in five phases. The phrases are information, guided orientation, explication (in which the student becomes conscious of relations and tries to express them in words, motivating technical language), free orientation, and integration. At level 2, informal arguments and then conclusions occur, motivating justification by use of logical arguments.

Tall[259, p. 9] extended the types of abstraction described by Piaget to include “a fourth type of abstraction that generalizes empirical abstraction of the properties of physical objects, to imagine mental objects that can exist only in the mind, such as points that have no size and straight lines that have a length but no width. This may be termed *Platonic abstraction* as it forms Platonic mental objects by focusing on the essential properties of figures.” The separation of abstraction related to objects and to processes is related to van Hiele’s Structure and Insight[272]. Lakoff[153, p. 12-13] wrote about *conceptual embodiment* and *functional embodiment*, referring to use of mental images and ‘the automatic, unconscious use of concepts without noticeable effort as part of normal functioning’. Tall considered diagrams, both static and dynamic, as relevant to learning mathematics.

Harel and Tall[120, p. 38] described three kinds of generalization, expansive, reconstructive and disjunctive. Students may have a deep understanding, ready to be expanded, or they may take the occasion to deepen their understanding, such that they reconstruct previous ideas to accommodate new ones, or they might just accumulate new ideas. (One role a teacher may serve is to expose the student to good analogies, so that expandable ideas are built, and occasions on which reconstruction is required are reduced.)

### 7.3.1 Social Constructivism

The means by which we may advance our understanding in conference with others is a significant contribution of Vygotsky[282], likewise the means by which we as educators may further the growth of others.

Locke[169, Book III, chapter I, §2] talked about the use of language for sharing our ideas with others.

Harel and Sowder[119, p. 5] observed that “the concept of proof is social – in that what is offered as a convincing argument by one person must be accepted by others – one must take into account the social nature of the proving process.”

#### Origin of Social Constructivism

Vygotsky, in the time immediately after the Russian Revolution, led a school for children whose physical and mental ages were not much correlated, and whose mental ages had to be assessed. Vygotsky observed that, in assessing the mental age of such a student, not only the capability of the student acting alone, but also the capability when the student received a small amount of assistance from peers or an instructor, was very informative. He named the difference between these measures the “zone of proximal development”.

We take the zone of proximal development(ZPD) to identify those new ideas which the student is well prepared to learn with a small amount of assistance.

Given that we wish to teach the abstract ideas of mathematical definitions as important for warranting transformation steps in proof, we may reduce that problem to the cultivation of the development of the ZPD, so that it follows a trajectory supporting the material in our course. Of course the ZPD is within the student.



Vygotsky in *Language and Thought*[282] said we do as individuals build up thoughts and then as we become socialized with shared language, some accommodation would need to be enforced onto the child. [p.17] ...the psychological problem is to become convinced that always, necessarily a given picture has to appear as one of a multiple of possible graphs of the same category (i.e., only as a representative of a class ...must be grasped not in a final fixed state but rather *in construction* the point moving).

Vygotsky[283, p. 49] noted that “one child selected a picture of an onion to recall the word ‘dinner’. When asked why she chose the picture, she gave the perfectly satisfactory answer, ‘Because I eat an onion’. However, she was unable to recall the word ‘dinner’ during the experiment. This example shows that the ability to form elementary associations is not sufficient to ensure that the associative relation will fulfill the *instrumental* function necessary to produce recall.”

As instructors attempt to nurture the ZPD, we should remain cognizant that, when students are not passive, we may be out of coordination with the direction of advanced spontaneously used by the student, and we should take care to provide associations that students can use to recall the new material.

Frawley[92, p. 21] summarizes Vygotsky’s work as “a theory of the *internalizing of the external*, a theory of *internal-external relations*.”

Frawley discusses inference “Speaking drives thinking. This *language for thought*, the framing of the inferential language of thought in which the solutions are finally computed, has an inhibitory function, ruling out options and giving a direction to the representational thinking.”[92, p. 31]

## Vygotsky and Cognitive Science

Frawley[92] writes about some compatibilities between Vygotsky's social constructivism and cognitive science.

According to Frawley[92], there are disagreements among cognitive scientists. Though one might agree that ideas are shared between people using words in part, and of course sometimes (possibly animated) diagrams, and the mind works with ideas, there is disagreement about how the brain represents the ideas. Though functional magnetic resonance imaging tells us which parts of the brain are metabolically active during thinking, we shall attempt to operate at a level of abstraction above that of how ideas are represented physiologically. Moreover, when we talk of representations, and re-presentations, we are not intending to use these terms in the sense they are used in cognitive science. Rather, we use re-presentation as it is used by von Glasersfeld, and representation as it is used generally.

Miller's famous  $7 \pm 2$  units limitation on human working memory[92, 193, p. 77] refers to distinguishable alternatives. Perhaps these alternatives can be pointers to items in Tversky and Kahneman's system 1 intuition[139] (Reif's so-called precompiled[217]).

Vygotsky[281] stated "The child's work on a word is not finished when its meaning is learned". There remains work to be done, by the individual. I think this might refer to integrating the word and its definition into a hierarchical network of related words and definitions.

## Cognitive Science and Science Education

Cognitive science addresses questions including “Can people increase their efficiency by replacing cumbersome thinking with more intuitive processes? ... *how* can they?” [217, p. 257]

Reif[217] addressed, from the perspective of cognitive science, several topics pertinent to education:

“Declarative knowledge is factual knowledge that specifies relevant entities and the relations among them. Procedural knowledge specifies methods that indicate how to perform various tasks. Declarative knowledge is compact and flexibly usable. But, it requires reasoning or problem solving to figure out how it can be used for any particular task. Procedural knowledge can be easily used for an intended task. But extensive procedural knowledge is required for performing many diverse tasks. Furthermore, without adequate declarative knowledge, such procedural knowledge cannot be readily modified in case of need. Declarative and procedural knowledge are both needed for good performance since either one is inadequate without the other.” [217, p. 41]

“Students’ difficulties are predominantly due to inadequate discriminations, to defective interpretations of basic definitions, to faulty bits of remembered knowledge, and to fragmented conceptual knowledge leading to irresolvable paradoxes.” [217, p. 83]

“Effective encoding is helped by repeated practice and by elaborative processing of the acquired knowledge. Practice should be properly spaced since massed practice leads to rapid forgetting. The resulting stored knowledge gradually decays with time, but can be refreshed by reviews and further practice. However, care is needed

to reduce interference effects between successively learned knowledge. An important practical implication of these findings is that effective learning requires active processing by the learner. Memory can be managed more effectively by adequate practicing, by elaborating and using newly acquired knowledge, by deliberately examining important discriminations, by redescribing and organizing this knowledge in useful forms, and by repeatedly reviewing previously acquired knowledge.”[217, p. 100] “Facilitate learning by active processing and careful managing of cognitive load. Exploit a useful knowledge organization”[217, p. 375]

“Plausible inferences are useful for discovering new knowledge and generating hypotheses, but deductive inference from a few scientific principles are needed to yield reliable predictions and explanations. Although procedures are often used in teaching and in everyday life, mindlessly used procedures can result in little learning and may lead to inflexible or inappropriate performance.”[217, p. 117]

“Good performance can be greatly enhanced by the availability of multiple descriptions so that appropriate descriptions can be chosen for the tasks for which they are best suited. For example, good scientists (even in highly quantitative fields) often use *both* qualitative and precise mathematical descriptions. The use of multiple descriptions also requires the ability to translate between them so that the same situation can be described in alternative ways.”[217, p. 136]

“organization of knowledge is very important . . . A hierarchical knowledge organization displays knowledge in the form of successive elaborations. Hierarchical forms of knowledge organization are particularly useful (especially in science) for organizing large amounts of knowledge in a compact, coherent, and readily elaborated form that facilitates the retrieval of specific information. (A familiar example is the organization of geographical knowledge in the form of maps elaborated at increasing levels of

detail.) . . . Instructional efforts need to pay as much attention to the organization of acquired knowledge as to its content.”[217, p. 161] Decision making is also assisted by well-organized information.[217, p. 200]

“Successful problem solving can be facilitated by . . . an initial clear description of the problem”[217, p. 227] and “by well-organized and easily retrievable supportive knowledge”[217, p. 252]

“Only after several months do some students realize that drawing diagrams, that help them to visualize the relevant situations, can ultimately save them time by avoiding many of their problem-solving difficulties.”[217, p. 258]

“Efficiency for work in a particular domain can be improved by *compiling* knowledge – that is, by acquiring knowledge about the domain and remembering it in a readily usable form. The knowledge learned in this way can afterward be used again with less need to engage in deliberate thinking. Laborious thought processes can then be replaced by easier and faster recognition and retrieval processes. . . . Compiled case-specific knowledge must be accompanied by applicability conditions indicating when this knowledge is valid and useful. Such conditions must be stored together with the case-specific knowledge – and must then be properly retrieved at the time when this knowledge is invoked”[217, p. 260]

Consistent with Kahneman and Tversky’s idea of systems 1 and 2, is Reif’s split between deliberate processing (system 2) and recall from memory (system 1). The activity that converts the results of deliberation produced by system 1 into the memory accessed by system 2 is practice: “When initially performing some task, one proceeds deliberately by repeatedly deciding what to do, implementing the decision, and assessing whether the performance was satisfactory. But after the same task has been performed a few times, the task can be performed more *routinely* (that is, with less

deliberately thought). The performance thus becomes more efficient and is carried out more rapidly and effortlessly.”[217, p. 261] “Sufficiently repeated and consistent practice can lead to the extreme situation where routine performance becomes *automatic* (that is, to a situation where the performance can be reliably carried out without conscious awareness, even while one may be deliberately performing some other demanding task.)”[217, p. 263] One benefit of automaticity is genuinely good performance. “One part of the answer lies in humans’ ability to think symbolically so that they can work with useful special descriptions of various situations. But the other part of the answer lies in their ability to combine deliberate thinking with subconscious processing that exploits learning and the large capacity of their long-term memory. Thus people can use their deliberate thinking to deal with the central aspects of some task. But they can perform subsidiary parts of a task routinely, without much deliberate thought, by relying on recognition processes and previously learned activities that can now be carried out without much conscious attention. Furthermore, they may have learned some subsidiary tasks so well that these have become automatic and require *no* conscious thought. A considerable part of a task can then be downloaded onto subconscious processes, thus freeing deliberate thinking for the more centrally important parts of a task.”[217, p. 266]

Consistent with the description of peer instruction by Crouch, Mazur et al.[60], “they are often not consciously aware of all their acquired knowledge (some of this knowledge is for them *tacit*) so that they may have difficulty communicating some of this knowledge to other people.”[217, p. 267] Tall[259, p. 6] wrote “The strengthening of useful links between neurons provides new and more immediate paths of thought, so that processes that occur in time ... are shortened to ... immediately output the result ... This involves a *compression* of knowledge in which lengthy operations are

replaced by immediate conceptual links.”

“Students, who are attempting to learn science, find it difficult to modify or transcend their preexisting naive scientific conceptions. Hence such conceptions often persist for a long time and cause many confusions. To be effective, science instruction must be explicitly aware of students’ naive prior notions. But such naive conceptions cannot merely be replaced by miscellaneous conceptions of greater scientific validity. They must be transcended by *coherent* scientific knowledge that is recognized to be more valid and useful”[217, p. 332]

“To describe a learning problem, one needs to specify a student’s initial performance capabilities”[217, p. 355]

Sherin [234] reviewed two articles, stating they “make a compelling case that the development of expertise in physics subject matter requires the weaving together – coordination – of old and new knowledge resources.”.

Balacheff[20] observed that the study by Healy and Hoyles[124] at least partially supported the suggestion of Tall[257], namely “The cognitive development of students needs to be taken into account so that proof[s] are presented in forms that are potentially meaningful to them. This requires educators and mathematicians to rethink the nature of mathematical proof and to consider the use of the different types of proof related to the cognitive development of the individual.”

Gentner and Markman[97] wrote about structure mapping in analogy and similarity. They suggested that “both similarity and analogy involve a process of structural alignment and mapping, that is, that similarity is like analogy.”

## Social Constructivism Applied to Mathematics Education

Archavi et al.[13, p. 6] wrote: “Students’ mathematical activity takes place in an inherently social milieu.”

Goodson-Espy[108], citing Cobb[55], wrote “Collective abstraction was defined during a series of classroom design experiments targeted at understanding the interactions in the immediate social situation of students’ mathematical learning ... He described collective abstraction as occurring when members of a community collectively use prior group experiences as an explicit object for class discourse, meaning the groups’ previous discussion and activity becomes an object of reflection.” Goodson-Espy[108] went on to say “The actor-oriented abstraction approach is aimed at coordinating individual and social levels of abstraction. ... Actor-oriented abstraction includes modifications to the Piagetian construct of reflective abstraction ... describe[s] both individual and social levels of abstraction, and uses the device of *attention focusing* to do so.” Lobato[168] wrote about the actor-oriented approach, which, according to Goodson-Espy[108, p. 383], “seeks to describe how features of instructional environments, including curricular materials, social-cultural norms, tool use and classroom discourse, interact to affect the conceptual attributes to which students pay attention.”

Fischbein wrote [90, p. 214] about a “concept of intuitive loading — have to know students first before knowing how to teach them.”

Bransford et al.[41, p. 296] attempt to address the problem identified as inert knowledge (in the sense of Whitehead[300]). They situated class activity in a problem solving environment, and they showed[271] that this instruction had better results for students’ ability to transfer skills to new word problems than traditional instruction.



Lehrer et al. [158, p. 334] found that “at least in some circumstances, giving children models may be less helpful than fostering their propensity to construct, evaluate, and revise models of their own to solve problems that they consider personally meaningful.”

Also specific to students concerned with algorithms, we may wish to extend the notion of social constructivism from that of Piaget [203] and of Vygotsky[283, 284], where it was necessarily a person with whom the learner was communicating, and therefore with whom it was necessary to share a basis for communication, to include a compiler and runtime execution environment, as students of computing disciplines must also comply with rules (e.g., syntax) used in these systems. Recalling the work of Papert and Harel[121], we might call this constructivism with constructionism. Constructionism is an approach to learning in which the person learns through design and programming.

Fukawa-Connelly[94] wrote about the utility of norms for presentation of proof in the classroom. He wrote “the behaviors that norms prompted the students to engage were those that literature suggests leads to increased comprehension of proofs.” He wrote about the knowledge and actions that support proof-comprehension and proof writing, including “the value of examples and diagrams in generating the ideas for the proof methods and critically examining the proof that was produced.” The students were required to present and defend their work, which led them to view proof as a means of communication.

### 7.3.2 Radical Constructivism

According to Cifarelli and Sevim[53] radical constructivism, in particular re-presentation, has a role in understanding mathematical problem solving. It is desired to understand “constructive learning process, characterized by the construction of knowledge through ordering and organizing experience” [53, p. 360].

#### Correspondence with Physical Entities, or Not

Von Glasersfeld[277] states that his version of constructivism is radical “because it breaks with convention and develops a theory of knowledge in which knowledge does not reflect an objective, ontological reality but exclusively an ordering and organization of a world constituted by experience”. Von Glasersfeld, in his book with Smock[246] also wrote “We call this school of constructivism ‘radical’ because it holds that the knower’s perceptual (and conceptual) activity is not merely one of selecting or transforming cognitive structure by means of some form of interaction with ‘existing’ structure but rather a constitutive activity which, alone, is responsible for every type or kind of structure an organism comes to ‘know’”. He observes that the disconnect between the real world and whatever our experience of it might convey to us has been recognized since at least the time of Plato[277]. Locke[169, Book IV Chap. IV §3] asks “Tis evident, the mind knows not things immediately, but only by the intervention of the ideas it has of them. Our knowledge therefore is real, only so far as there is a conformity between our ideas and the reality of things. But what shall here be the criterion? How shall the mind, when it perceives nothing but its own ideas, know that they agree with things themselves?”. Von Glasersfeld concludes his first section stating that radical constructivism “is not quite as radical as it appears

at first sight.”

Locke[169, Book IV Chap. IV §8] continues “All the discourses of the mathematicians about the squaring of a circle, conick sections, or any other part of mathematics, concern not the existence of any of those figures; but their demonstrations which depend on their ideas are the same, whether there be any square or circle existing in the world, or no.”

Herein we take the use of “radical” to signify recognition of possible lack of correspondence of knowledge – what one knows – from whatever an objective, ontological reality might be. Von Glasersfeld stated “Once knowing is no longer understood as the search for an iconic representation of ontological reality but, instead, as a search for fitting ways of behaving and thinking, the traditional problem disappears.”[277, summary]

Educators in computer science might find that what is fitting in the creation of algorithms to meet thoroughly specified performance constraints is external to the opinion of any human, because we define *thoroughly specified* to remove any dependence upon opinion. Whether this independence from human judgment confers “reality” onto a requirement is perhaps less interesting than the idea that we aim to teach our students reliable techniques for satisfying (solvable) problem statements. We wish the students’ conceptions of these techniques to be of sufficient depth that the students can generalize them to solve new problems.

## Re-presentation

Von Glasersfeld describes re-presentation referring to Kant’s Critique of Pure Reason[141] and the idea “Vorstellung”, which includes the idea of autonomous construction. An

example is the re-experiencing of events, as may happen in daydreams or dreams.

Cifarelli and Sevim[53, p. 361] wrote “Noting the various use of the term *representation*, von Glasersfeld saw the usefulness of viewing the process of mentally *re-presenting* a prior action as an active dynamic process that plays a fundamental role in the construction of conceptual structures ... action-based routines ‘that can be mentally called up and run’[278].”

## Reflection and Abstraction

The researcher von Glasersfeld considered both reflection and abstraction to be important to learning and knowing. He describes reflection quoting Locke and von Humboldt[279, p. 90] and classifies it as a kind of abstraction and observes that the delimitation of an item from the stream of sensory input in which it occurs is a representation[279, p. 91]. He connects the ideas of von Humboldt and Piaget, stating that this delimitation allows further abstraction, including *generalizing abstraction*. Von Glasersfeld deals with Berkeley’s exceptions to Locke’s writings on generalization by stating “we construct general ideas in order to classify particular things”[279, p. 92]. These ideas are not “images like picture postcards but operational recipes that produce them”[279, p. 92]. We can use these ideas to “recognize items that we have never seen before as exemplars of a familiar kind.”[279, p. 93]. Von Glasersfeld notes that such an operational recipe or pattern does not automatically turn into an image.[279]

Jackson[136] discusses levels of abstraction, including comparing their concision and usability. “Software is built on abstractions. Pick the right ones, and programming will flow naturally from design; modules will have small and simple in-

terfaces; and new functionality will more likely fit in without extensive reorganization. Pick the wrong ones, and programming will be a series of nasty surprises: interfaces will become baroque and clumsy as they are forced to accommodate unanticipated interactions, and even the simplest of changes will be hard to make. No amount of refactoring, bar starting again from scratch, can rescue a system built on flawed concepts.”[136, p. XIX] “The core of software development ... is the design of abstractions.”[136, p. XIX] “the environment of programming is so much more exacting than the environment of sketching design abstractions ... code is a poor medium for exploring abstractions ... code is clumsy and verbose ... design ... abstractions ... with a notation chosen for ease of expression and exploration ... precise and unambiguous ... (different from) mathematical syntax that makes them intimidating to software designers”[136, p. 2] “precise and expressive ... succinct”[136, p. 3] “the hardest, and most rewarding, challenge in software design is reducing a mass of complicated, incongruous details to a few simple generalities. Simplicity is the key to good software design.”[136, p. 28] “precise enough to share with others (or for us to recall for ourselves later)”[136, p. 30] “Diagrams are very useful representations, but they’re limited in their expressiveness. I often use a diagram to sketch the structure of a model, and then transcribe it into Alloy text ... Graphical output ... is indispensable. The Alloy Analyzer can display instances in graphical form, or in textual form, or as an expanding tree. ”[136, p. 32]

Concision can vary from verbose to cryptic, as these Alloy expressions demonstrate: “

```
all n: Name, d, d': Address |
n-> d in address and n->d' in address implies d = d'
```

” is predicate style.

“

```
all n: Name | lone n.address
```

” is navigation expression style, and

“

```
no ~address.address - iden
```

” is relational calculus style.[136, p. 34]

Coffey[57] wrote about improving an initially separated collection of data structures, discrete mathematics and analysis of algorithms courses, into a combination that used empirical studies to reinforce theory. Students benefited from the integration, in retention and grade point average.

Downs and Mamona-Downs[70] wrote about the difficulties students are seen to have with the exact form, the language in which formal proof is set. They highlight the idea of “proof language”, pointing out that students may not have prior experience of it. They argue that abstraction does not remove the utility of intuition, rather, the quote Piaget “intuition is essentially operational and the nature of operational structures is to dissociate form from content”[202, p. 87]. Downs and Mamona-Downs [70] wrote “The task might then be thought through ‘mentally’ via the representation. The use of representations are of critical importance in doing mathematics.” They discuss using representations as catalytically – as an aid during discussion which has disappeared from the final result. They characterize conversion between representations as often messy. They point out, quoting Moore[196, pp. 249-266] that “students can regard proofs as having a procedural character, in the sense

that a proof has to follow *a sequence of steps that one performs*". They suggest that students are, but should not be, constrained in the use of informal representations while thinking about proof. Simon[239] wrote about an informal reasoning style he called "transformational", consistent with the conceptualization of the same name, by Harel and Sowder[118]. Downs and Mamona-Downs[70, p. 1753] observed that "it is not rare for a student to mistake a definition for a proposal to be proved." They observed that there seemed to be "an abandoning of sense making once students are . . . handling symbolic systems" and that students cannot "build up symbolic frameworks in order to examine properties of sets". They mention Moore's[196] observation that if students are using both a mental image and a definition and have not thoroughly coordinated the two, they may be misled by using the concept image in preference to the definition.

Tall[259, p. 6] wrote "The long term development of mathematical thinking is consequently more subtle than adding new experiences to a fixed knowledge structure. It is a continual reconstruction of mental connections that evolve to build increasingly sophisticated knowledge structures over time."

**Analogy** Gentner and Smith[98] wrote "Analogy is often the most effective way for people to learn a new relational abstraction; this makes it highly valuable in education." According to Gentner and Smith, analogical reasoning has three parts, retrieval, mapping and evaluation. In a scheme based upon Tversky and Kahneman's[139] system 1 and system 2, and the need, described by Locke[169], to organize our concepts stored for use by system 1, we can see how the mapping part of analogy, which allows the sharing of an organization of information by new information organized in a simi-

lar way, would make this part of learning more efficient. Genter and Smith[98, p. 132] wrote “A possible outcome of structural alignment is abstraction of the common relational pattern.” Perhaps we can encourage abstraction by using analogies and making explicit the structures and their alignment. Genter and Smith[98, p. 132] refer to work by Holyoak[130] and Spellman[248], who studied the relationship of analogy-finding to the goals of the finder. From this work we might choose to motivate our students with a goal at the time we discuss specific analogies with them. Harel and Tall[120, p. 41] wrote about what they call a generic example. This should satisfy the principles of entification, necessity and parallel. To satisfy entification, the model should allow the student to perceive its parts as concepts. To satisfy necessity, the model should allow the student to perceive the rationale for operations related to the model. To satisfy parallel, a concrete model should support processes that are part of the more general concept the instructor is trying to construct, so that contemplation of the model induces structures in the mind that form the root of abstractions, so that subsequent mental advancement is expansive, rather than requiring reconstruction.

Retrieval of useful information is a separate but related problem in analogical reasoning (when useful analogies are available to be retrieved), and of course a generally interesting problem. Gick and Holyoak[100] studied retrieval in the setting of participants with analogical problem solutions in long term memory, with and without reminders. Genter and Smith[98, p. 133] observe this “is an example of what Alfred North Whitehead called ‘inert knowledge’ – knowledge that is not accessed when needed.”[300]



### 7.3.3 Neo-Piagetian Theory of Cognitive Development

According to Lister[166], “the principle difference in neo-Piagetian theory, is that people, *regardless of their age*, are thought to progress through increasing abstract forms of reasoning *as they gain expertise in a specific problem domain*.” The explanation for domain specific abstraction is that abstraction operates using limited (see Miller[193]) short term memory, and that the limitation applies to chunks of knowledge already learned by the participant, which can be expected to be subject-matter specific. Lister [166] wrote “From the neo-Piagetian perspective, the low level of abstraction in preoperational thinking, on a particular problem, is a consequence of the novice’s working memory being overwhelmed, since the novice has not yet learnt to chunk knowledge and information in that problem domain.” Lister [166] wrote “Since novice programmers who reason preoperationally tend not to abstract, and when they do abstract they tend to not be systematic, these novices struggle to make effective use of diagrammatic abstractions of code.”

Gluga et al.[104] wrote on mastering cognitive development theory in computer science education. They[104, p. 26] wrote: “Neo-Piagetian cognitive development theory deals directly with abstraction and reasoning ability.” They wrote: “Neo-Piagetian theory defines three main stages of cognitive development, which are . . . pre-operational, concrete operational and formal operational.” They explain that with pre-operational reasoning a student can focus on a single abstraction but tends not to see connections between abstractions; with concrete operational reasoning a student’s abstract thinking is restricted to familiar situations, restricted from hypothetical situations; formal operational reasoning can reason logically and systematically, reflect on reasoning and solve complex, unfamiliar problems.

Lister[166] classified problems used in and achievement levels reached by students of computing, using a neo-Piagetian theoretical framework.

### 7.3.4 Definitions

Edwards and Ward[79, p. 223] wrote: “Mathematical definitions are of fundamental importance in the axiomatic structure that characterizes mathematics. The enculturation of college mathematics students into the field of mathematics includes their acceptance and understanding of the role of mathematical definitions, that the words of the formal definition embody the essence of and completely specify the concept being defined.”

Alcock examines definitions as viewed by mathematics students: For students naïve about mathematical definitions, a category exists prior to its definition, it has members, and any properties of the category follow from its membership. Students need to invert that way of understanding, and to advance to the conceptualization that the defining property determines the category; to know that a definition is “precisely a set of necessary and sufficient conditions for category membership.”[6, p. 32]

Edwards and Ward[79, p. 223], using terminology from Landau[155], refer to *extracted* and *stipulated* definitions. Students need to know that mathematical definitions are stipulated. Rasslan and Vinner[214] reported that 68 percent of the students they tested could state the definition (of a specific term), only 36 percent applied the definition successfully and well.

Bills and Tall[29] wrote about “operable” definitions. They formulated a working definition: “A (mathematical) definition or theorem is said to be *formally operable* for a given individual if that individual is able to use it in creating or (meaningfully)

reproducing a formal argument.”

Harel and Sowder[119] wrote “University students’ distinctions among axioms, definitions, and theorems are not sharp.

Vinner[276] investigated appreciation of definitions among sophomore and junior mathematics majors (compared with freshman) at UC Berkeley and found that only half could correctly identify definitions from theorems, laws, facts about numbers, or axioms.

Other studies in the US and Israel of undergraduate mathematics majors show that the axiomatic proof scheme is not popular.

### 7.3.5 APOS Theory

Dubinsky and McDonald[74, p. 2] wrote “The theory we present begins with the hypothesis that mathematical knowledge consists in an individual’s tendency to deal with perceived mathematical problems situations by constructing mental *actions*, *processes* and *objects* and organizing them in *schemas* to make sense of the situations and solve the problems.” Their work is based on that of Piaget on reflective abstraction. An action is a transformation (of objects) involving steps. (As a Tversky and Kahneman system 2 operation would.) An action, repeated, and reflected upon, can produce a mental construction, a conception, called a process. The process can be viewed as an object, upon which transformations can act. A schema is a collection of actions, processes, object and other (sub) schemas (APOS). Thus a schema is similar to, but not the same as, what Tall and Vinner[262] “concept image”. The utility of APOS theory includes that “If there appear two students who agree in their performance up to a very specific mathematical point and then one student can take a

further step while the other cannot, the researcher tries to explain the difference by pointing to mental constructions of actions, processes, objects and/or schemas that the former student appears to have made but the other has not. Then theory then makes testable predictions that if a particular collection of actions, processes, object and schemas are constructed in a certain manner by a student, then this individual will likely be successful using certain mathematical concepts and in certain problem situations.” [74, p. 4]

## 7.4 Social Science Research

### 7.4.1 Psychology of Decision Making

Tversky and Kahneman [139, 267] describe how decisions, such as multiple choice questions for students, and categories for those performing inductive research, are influenced.

People creating proofs also make choices. A proof plan is designed, the plan is refined into steps of a size appropriate for the context in which the proof is being carried out.

Kahneman and Tversky [139, 267] use terms “system 1” and “system 2” to help distinguish between ways we make choices. “System 2 articulates judgments and makes choices, but it often endorses or rationalizes ideas and feelings that were generated by system 1. ...it also prevents many foolish thoughts and inappropriate impulses from overt expression. The investment of attention improves performance in numerous activities ...errors are not always due to intrusive and incorrect intuitions.” System 1 uses “the rich and detailed model of our world that is maintained

in associative memory: it ... automatically searches for some causal interpretation of surprises and of events as they take place. ... also holds the vast repository of skills we have acquired in a lifetime of practice”[139, p.416].

Use of an incompletely informed system 1 seems to correspond to what Herman et al.[127] (citing Chi et al.[52]) describe as “physics novices focus on surface features of problems rather than the underlying concepts and try to recall as quickly as possible any formula that seems to match the surface features of the problem they have encountered.” Herman et al.[127] reported “some subjects readily admitted that they relied only upon recall to succeed in the class.” The ability of system 2 to be lazy[139, p.39-49] and endorse system 1 when it could instead prevent foolish impulses seems quite relevant to Herman et al., when they state “Further research should investigate why students were so reluctant to rely on truth tables except in the simplest cases.”[127, p. 67]

Locke in 1690 distinguished these two types of decision making, stating “these maxims, and mathematical demonstrations are in this different; that the one has need of reason, using of proofs, to make them out, and to gain our assent: but the other, as soon as understood, are, without any the least reasoning, embraced and assented to.”[169, Book I §10]

Kahneman emphasizes that for system 1 to become skilled, “the acquisition of skills requires a regular environment, and adequate opportunity to practice, and rapid and unequivocal feedback about the correctness of thoughts and actions. When these conditions are fulfilled, skill eventually develops, and the intuitive judgments and choices that quickly come to mind will mostly be accurate.”[139, p.416]

## 7.4.2 Qualitative Research

Burnard[44] wrote about analyzing interview transcripts in qualitative research, organizing the work in stages, and showing examples of open coding.

Berglund et al.[27] wrote about qualitative research applied to computing education research, “Phenomenography has proved to be successful as a research approach in students of learning of computing concepts.” They listed programming and concepts of object and class as examples.

Glaser[101] wrote on conceptualization, in the context of grounded theory, a very useful qualitative research technique. Glaser’s method of later interview protocols being influenced by earlier interview protocols allows the researcher to explore ideas generated in response to earlier interviews. When Institutional Review Board review is needed for each interview protocol, an investigation with several protocols sequentially determined can be long term.

### Phenomenography and Variation Theory

Bussey et al.[45] provided a diagram distinguishing the intended object of learning, resident in the mind of the instructor, the enacted object of learning, available to both instructor and student, and the lived object of learning, experienced by the student in light of his or her prior knowledge. These distinctions are important in appreciating phenomenography and variation theory. Jones and Herbst[138] considered which theoretical frameworks might be most useful for studying student teacher interactions in the context of learning about proofs. Bussey et al.[45] illustrated student teacher interactions in the space of learning, and the objects of learning, in variation theory, modified from the model of Rundgren and Tibell[222]. Reid and Petocz[215]

used phenomenography to study students' conceptions of statistics. Their purposes included to "enable teachers to develop curricula that focus on enhancing the student learning environment and guiding student conceptions of statistics." They asked students to describe how they understood statistics and then organized student responses into a hierarchy of conceptions. They used interviews to understand individual students, and the group of interviews to show the variations they found. They found the students with the most superficial understanding to be carrying out steps without knowing their meaning.

**Origins** In 1970 Marton and Svensson began a tradition of a certain kind of educational research[253]. Marton is the founder of phenomenography, which he has characterized as investigating different understandings of reality[173]. We can argue, as did Locke, that in the domain of mathematics, which deals with complex mental concepts, the structures in the mind have more merit for argumentation about truth than any physical approximation to, for example, a triangle. Thus, by remembering that reality, in the domain of mathematics, proof, including proof applied to computer science, it is the mental ideal we address, we find no conflict between the mental constructions of radical constructivism and phenomenography's reality. This brings up the idea of a non-dualistic ontology, a feature of phenomenology as described by Åkerlind[3]. The variations among experiences students may have, of the concepts related to proof, are more or less complete, and more or less profound constructions, compared with a goal experience for learning; for the achievement of this goal by the student, evidence can be obtained by assessments. If the student's sense of having ascertained a theorem is to be communicated to another person, such as, to convince

that person, or in assessment, does not a dualistic situation arise? If so, we might apply phenomenography while a student is learning to ascertain, and decide to defer the convincing role of proof until after the student has learned proof. Marton and Booth[176] wrote about variation theory, giving its early development.

Boustedt wrote: “Phenomenography originated in educational questions of how learning comes about and how the learning process can be improved.”[37]

**Conception** Marton and Säljö conducted a sequence of articles[226, 227, 174] on the topic of conception, and category of description. To advance this discussion, Marton and Pong[179] wrote about the meaning of “conception”, or “conceptualization”. The conception refers to the “global meaning of the object conceptualized”, but consists of a “specific combination of features that have been discerned”.

Åkerlind [3] has written on variation and commonality in phenomenographic research methods. Åkerlind points out that in the analysis of the interview transcripts, a dynamic balance should be enacted, between finding categories of description and addressing the relationships among categories. Initially identification of categories is primary. This involves learning the features by which categories differ, which establishes the definition of each category. The categories are determined using student utterances that have been taken from their context. Subsequently the student utterances can be labeled with their category and observed within the context from which they were taken, as a check on whether the categorization seems to make sense. This can result in revisions to category membership and definition. Cycling between the category context and the transcript context is seen to have stabilized in practice.



**Critical Factors and Aspects** Critical factors have been defined by Marton[176, 181] and can be usefully compared with threshold concepts. Critical factors are elements of the subject matter that are necessary for students to advance their way of experiencing to a more advanced level. In 2015, Marton clarified the phrase, “To the extent that the learner has not already made a specific necessary aspect her own, it is a critical aspect for her. It is one of the things that she has to learn in order to meet the learning target (the education objective).”[175] Threshold concepts are characterized, by Meyer and Land[192], as transformative (changing the way a student perceives elements of the subject being learned), integrative (connecting previously separate concepts), irreversible (not likely to be forgotten), troublesome (difficult), and boundary markers (limiting a concept). Thus a threshold concept may be included in a conceptualization, and discernment of a critical factor helps a student reach a more advanced conceptualization. It appears that critical factors can be threshold concepts, but the set of critical factors is more comprehensive. Eckerdal et al.[76] wrote “threshold concepts are keystones, critical parts of the structure that hold the rest together”.

Marton and Booth[176] have written on critical factors and aspects. Using the method of phenomenography, these can be determined for a cohort of students. Then, to the extent that this first cohort of students represents a second cohort, the critical factors and aspects can be used to help the students advance to the most inclusive and profound conceptualization.

Marton and Tsui[181] wrote “critical features are the features of an object of learning that are necessary to distinguish one way of thinking from another.”

**Discerning** Locke wrote about discerning: “clear discerning faculty of the mind, whereby it perceives two ideas to be the same, or different.”[169, Book II, chapter XI, §1] Locke also wrote about the inability to discern, as the nature of apple being different from that of red apples, when no other color of apple had entered awareness. Marton and Pang[177] wrote about discerning the critical aspects of the object of learning. In this context, “discerning the object of learning amounts to discerning its critical aspects.”[177, p. 193] Variation in the values taken on by a critical aspect help students discern that aspect. Locke[169] has observed that children who have seen apples, but only red ones, will be surprised the first time they see a yellow one; they have incorporated the red color into the definition of apple, and must now re-evaluate what are the characteristics of the general notion apple. Once the critical aspects are identified, testing of their effectiveness can be carried out.

Variation is a process that is used to help students discern the critical factors and aspects identified earlier. Variation includes four components, contrast, generalization, separation and fusion.

Bussey et al.[45] gave examples from chemistry of these four operations. For *contrast* “Once the student notices the difference in color of the two solutions, he or she could construct meaning for the concept of *acid* using both their prior knowledge and other information provided during the learning event. . . . *generalization* allows the individual to compare similar instances of the object of learning. . . . In the chemistry class, the student might experience strong acids, weak acids, Arrhenius acids, Brønsted-Lowry acids, Lewis acids, etc.” “*Separation* allows the individual to discern one feature of an object of learning from other features by varying only the feature of interest while holding all other features constant. . . . When learning about pH, a teacher might ask students to solve several problems in which the student must solve

for the pH of a solution when different volumes of a 1 molar strong acid are added to an acetic acid buffer. All other features of the buffer problems would be the same.”

“Lastly, *fusion* allows the individual to discern variation in several features of an object of learning simultaneously... In the classroom, the pH of a buffer system might be observed when the volume, concentration, and type of acid added are all varied. All the individual parts interact to form a specific whole. The students’ ability to perceive each components and its specific contribution to the whole can foster a more coherent conception.”

Fawcett, in the 1930’s, used a variation theory technique to help students discern the increase in probability that arose from an accumulation of evidence[88, p. 83]. A comparison against a similar group in the same school, and also on state-wide tests indicate his method was successful. Also, the parents were consulted by an independent investigator, and in summary, felt that the students’ critical thinking was improved[88, p. 108].

**Benefits** An idea of how phenomenography (including variation theory) benefits students is that identification of critical factors between more and less advanced conceptualizations (which are better and less good educational outcomes) helps educators know which elements of the material to be taught can most beneficially be dwelt upon, and assessed.

**Validity** Some researchers have split one cohort into two, obtaining critical factors and aspects for the first group, and seeing whether these are representative of the whole.

Åkerlind [3] observed that, though validity is considered, “Research outcomes may then be judged in terms of the insight ...into teaching and learning”, citing Entwistle[83], Marton and Booth[176].

**Applications** Bussey et al.[45] wrote on the application of variation theory for research in education in chemistry. They pointed out that variation in the critical feature of banana color allows the individual to create meaning related to the concept of banana ripeness.

Mun Ling Lo[165] wrote on the application of variation theory. Ärlebäck and Doerr[14] have undertaken to apply variation theory to the ability of students to learn modeling, in particular to apply mathematical models in the solution of problems.

Boustedt discussed[37] that within a group of students, processing a visual representation, there were exhibited several levels of understanding of the utility of these diagrams. The more profound understandings contained not only the abstraction (of code by diagram) and communication but that hierarchical relationships among classes can be designed (so, abstraction of class by superclass).

Runesson[223, 224] has applied variation theory to math.

Box applied phenomenography to understanding approaches to analysis and design[39]. After determining categories and characterizing them, Box discussed that the differences in the characteristics can be presented to students. Contrast between characteristics is instructive, and difference among artifacts serving the same purpose prompts generalization.

According to Marton and Pang[177], “this model has been used in many classes in Hong Kong and Sweden”. Lo, Marton, Pang and Pong[167] have reported on

the application of use of critical aspects in teaching. Marton and Pang conducted a study in which “the two teachers in the target group made use of their knowledge of the systematic variation framework and let only those aspects vary at the same time to which they want to draw the students’ attention while keeping other aspect invariant. They also made sure that all the relevant aspects were focused on by varying them one at a time and brought them together subsequently. The teachers in the comparison group, who were ignorant of the variation framework, did similar things but not consciously and systematically as the teachers in the target group. The lesson plan produced by the three teachers in the comparison group did not ensure that the necessary conditions for attending the relevant aspects of the phenomenon and bringing them together were present in the lessons.” [177, p. 206]

**Successful Results** The results show that the group of students taught by the target (variation theory-aware) teachers learned significantly better. “The differences occurred in how the object of learning was handled, structured, and presented. These differences were few in number and seemingly rather subtle . . . even when curricular materials are good and the conditions of implementation fidelity are met, many microfeatures can come into play and undermine the best of curricular intentions . . . the results from other studies point in the same direction.” [177, p. 216]

Åkerlind et al. [1, 2] have applied variation theory to curriculum in law and physics. They reported “overall the perceived benefits were profound, far exceeding the project leaders’ initial expectations at the commencement of the project.” A curriculum intervention was designed based upon ways of experiencing the status quo teaching. The impact of the intervention was assessed by observing student reactions, seeing

the students' learning.

**Future Growth** More recently, Marton[175] has written about a matured variation theory.

According to Rovio-Johansson and Ingerman[220], Marton's 2015 book *Necessary Conditions of Learning* may embody a shift in primary emphasis of phenomenography from methodological to theoretical concern, where variation theory is this theoretical area.

Pang and Ki[198] have recently revisited the idea of critical aspects.

### **Thematic Analysis**

Merriam[190] describes basic inductive qualitative analysis and several specializations.

Thematic analysis as used in psychology is described by Braun and Clarke[42].

Fereday and Muir-Cochrane[89] describe a hybrid approach of inductive and deductive coding and theme development. They present a staged approach by which data is coded and used to identify themes.

## **7.5 Mathematics Education Related to Proof**

The research questions are somewhat analogous to those that have been investigated among students in mathematics. In the literature of mathematics education, we found researchers [116] reporting quite similar conceptions of proof by mathematical induction in students of mathematics.

Research on teaching and learning about proof in mathematics education has

produced an extensive literature. Only a small sampling is mentioned below. Mathematics educators, including Keith Weber[285, 286, 287, 288], Harel and Sowder in 1998[118], and David Tall[255, 256, 260, 258] have studied students' learning of proof in the mathematics curriculum. Leron, in 1983, [160] has described the structural method for proof construction, attributing it to recent ideas from computer science.

According to Huang et al.[131], a large body of research suggests that an abstract cognitive processing style produces greater creativity.

According to Gray and Tall [109, p. 117], Hiebert and Lefevre observed "a connected web ... a network in which the linking relationships are as prominent as the discrete pieces of information ... a unit of conceptual knowledge cannot be an isolated piece of information; by definition is it part of conceptual knowledge only if the holder recognizes its relationship to other pieces of information." [128, p. 3-4]

Gibson[99] examined students' use of diagrams in proofs, and found that diagrams helped link students' ideas to mathematization, namely, to representation in symbols, and also to support variation, in the sense of the critical difference between what Harel and Sowder[118] call perceptual and transformational conceptualizations.

Healy and Hoyles[125] have reported on algebra students' preferences for the content of convincing arguments, and their distinction between preferences for ascertaining vs. preferences about what was likely to be well-received on assessments. İmamoğlu[135] has studied the conceptualizations of proof of students who were preparing to become mathematics and science teachers, in their freshman and senior years.

McGowan and Tall [186, p. 172] wrote "If learning defaults to the goal of learning how, it can be successful. However, if it is accompanied by a lack of conceptual meaning so that mistakes occur, it can become fragile and more likely to fail in the

longer term. At this stage the problems may proliferate as the student becomes confused as to which rule to use, where to use it, and how to interpret it.”

Tall and Mejia-Ramos [115, 261, p. 138] wrote “Here proof develops through generalized arithmetic and algebraic manipulation”, which were seen as different kinds of warrants for truth.

Pinto and Tall[206] built on Tall’s concept met-befores.

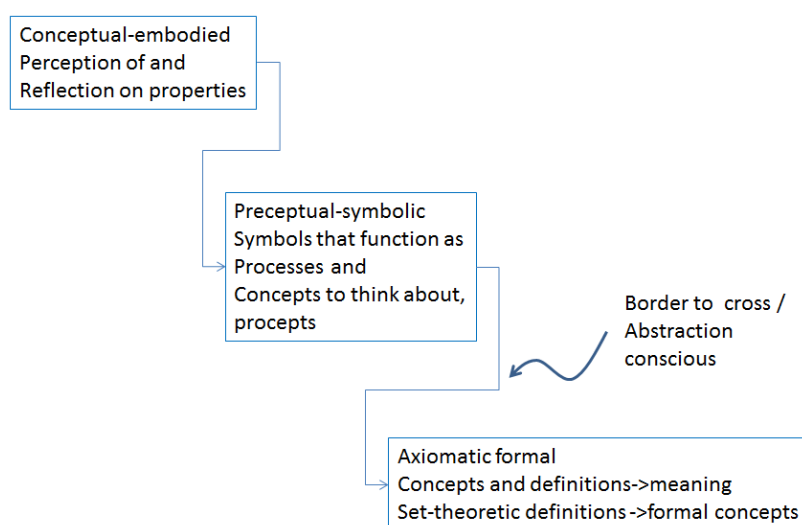


FIGURE 7.5.1: How proof develops, Tall Mejia-Ramos

Harel and Sowder [118, p. 237] have observed of teaching proof in mathematics: “Rather than gradually refining students’ conception of what constitutes evidence and justification in mathematics, we impose on them proof methods and implication rules that in many cases are utterly extraneous to what convinces them.” Schoenfeld et al.[142] editorialized that Harel and Sowder[118] “characterize students’ cognitive schemes of proof.”

The subdivisions in the 1998 version of categories of conceptualizations [118], specifically intuitive–axiomatic, structural and axiomatizing, matter much in com-



puter science, because intuitive–axiomatic could be thought to be less used in computer science than in math; a program’s content could be less intuitive than Euclidean geometry, and more subject to checking by assertion checking or debugger examination.

Harel and Sowder[118, p. 268] wrote of students and their contextual proof scheme: students have learned to work in a context, e.g.,  $\mathbb{R}^n$ , and so, interpret statements that have greater generality as restricted to be in the context they have learned “he or she has not yet abstracted the concept . . . beyond this specific context.”

Harel and Sowder[118, p. 274] wrote: “An important distinction between the structured proof scheme and the intuitive proof scheme is the ability to separate the abstract statements of mathematics (e.g.,  $1 + 1 = 2$ ) from their corresponding quantitative observations (e.g., 1 apple + 1 apple = 2 apples) or the axiomatically–based observations from their corresponding visual phenomena . . .”, “axiomatic proof scheme is epistemologically an extension of transformational proof scheme. One might mistakenly think of the axiomatic proof scheme is the ability to reason formally . . .”.

Baranchik and Cherkas[21] found three levels of understanding in a population taking algebra exams:

1. Early skills — arithmetic and elementary algebra
2. Later Skills — subsequent algebra and a variety of skills involving mathematical abstraction, and
3. Formalism — either devising a solution strategy or reformulating a problem into a standard form that permits a solution using early or later skills

Gibson[142, p. 289] wrote: “Students indicated that diagrams helped them understand information by appealing to their natural thinking. They said that diagrams

seemed to coincide with the way their ‘minds work’ and that information represented visually seemed easier or clearer than verbal/symbolic representations.”

Gibson[142, p. 291] quoted a student: “When I read the definitions you can’t think about the whole thing at once, but when you have a picture you can.” Gibson[142, p. 294] wrote “Because students did not usually think of their criteria in terms of formal definitions, their ability to decide whether their criteria had been met was hindered when they worked with information represented in only verbal/symbolic form. . . . They could obtain ideas more readily from diagrams than they could from verbal/symbolic representations.” Gibson[142, p. 297] wrote: “Why always keep the picture in your mind when you can have it on the paper, allowing you to focus more on how to get to the end of the proof instead of always having to recall the picture in each individual step?” Moore[196, p. 262] wrote: “The students’ ability to use the definitions in the proofs depended on their knowledge of the formal definitions, which in turn depended on their informal concept images. The students often needed to develop their concept images through examples, diagrams, graphs and others means before they could understand the formal verbal or symbolic definitions.”

Harel and Brown[117, p. 121] wrote “It is critical at this point, therefore, that the teacher facilitates the students’ formulation of these processes through whole class discussions and activities aimed at the refinement of justifications presented by the students.”

According to Fischbein[90, p. 47], intuition tends to survive even when contradicted by systematic formal instruction.

So we should keep track of the desired path of development, but should be aware also that there are mechanisms that can deflect students from our plan.

Fischbein [p.59] “Inferential affirmatory intuition may have an inductive or deduc-

tive structures. After one has found that a certain number of elements (objects, substances, individual, mathematical entities, etc.) have certain properties in common one tends *intuitively* to generalize and to affirm that the *whole* category of elements possesses that property. This is not a mere logical operation. The generalization appears more or less suddenly with a feeling of confidence. This is a fundamental source of hypotheses in science. According to Poincaré “generalization by induction copied, so to speak, from the procedures of experimental sciences” is one of the basic categories of intuition (Poincaré 1920 [p. 20]).

According to Fischbein[90, p. 67], Poincaré[207, p. 388] wrote “One morning walking on the bluff, the idea came to me with just the same characteristics of brevity, suddenness and immediate certainty that the arithmetical transformation of indeterminate ternary indefinite forms were identical to those of the non-Euclidean geometry.”

So we see that generalization happens apparently spontaneously (i.e., students may do this on their own), and can produce enduring beliefs.

As we have seen students resist learning definitions, an interest in helping them do so is informed by their difficulty, so we might devote time and attention to generalization/specialization hierarchies of definitions. Fischbein[90, p. 147] stated: “For many students the concepts of parallelogram, square and rectangle are not organized hierarchically. They represent classes of quadrilaterals of the same generality.”

Archavi et al.[13, p. 10] “To be successful, students must know both the appropriate heuristics and the mathematics required to solve the problem.”

The results have bridged papers in computer science education, by Professor Booth[34], and mathematics education, by Professor Harel[116].

Our students show many similarities, but some differences, shedding light on two things: The relevance of some curricular work done in mathematics, for use with our

students, but also some utility of the background of our students, namely practice of expressing algorithms in code, for students of mathematics.

Velleman, in 2006, has written software and a textbook[274] about proving with a structured approach. Weber has reported the success of several approaches to pedagogy[286, 287, 288, 289, 290, 295, 293]. Barnard [23] has commented upon students negating statements with quantifiers. Edwards and Ward[79, p.223] have discussed the role of definitions for undergraduate mathematics courses, stating “the enculturation of college mathematics students into the field of mathematics includes their acceptance and understanding of the role of mathematical definitions”. Bills and Tall [29] have distinguished student understanding of definitions that is sufficient that the student can use them in proofs. Harel and Sowder[118] and Harel and Brown[117] have conducted qualitative research on mathematics students’ conceptualization of proofs. They have developed three main categories, each with several subcategories. Evidence from our studies is consistent with the presence of these categories of conceptualizations in the population of CS(E) students.

Tall[255] has also categorized mathematics students’ understanding of proof. He has studied the development of cognitive abilities used in proof, starting, as did Piaget,[203] with abilities believed present at birth. Yang and Lin[304] have modeled reading comprehension. Leron[160] has written about encouraging students to attend to proof structure by teaching with generic proofs (proofs that use a generic particular). Mejia-Ramos et al.[187] have built a model for proof comprehension. They have observed that students who are assessed on appreciation of structural and other appropriate features of a proof, rather than on rote reproduction, are more likely to develop a deeper understanding of proof. Knipping[145] and Reid[216] have examined proof in mathematics education. Weber[286, 287, 288, 289, 290, 295, 292, 293, 294, 296, 297]

has investigated students' approaches to and difficulties with proof. When studying student proof attempts in group theory, Weber has found that some typical students' inability to construct proofs arise despite having adequate factual and procedural knowledge, the ability to apply that knowledge in a productive manner was lacking.[288] More specifically applying the knowledge was seen to include selecting among facts, guided by knowledge of which were important, for those most likely to be useful. Alcock and Weber[7] have studied students' understanding of warrants, the support for the use of a particular inference. Weber has published a framework for describing the processes that undergraduate students use to construct proofs.[291]

Harel and Brown wrote[117, p.120] "All our studies were based on the premise that it is only when students' current knowledge is recognized that teachers can devise and implement instruction that can bring about desirable outcomes in students' learning.... We found that ... (in current treatments of mathematical induction) ... the three most prevalent proof schemes among students are the authoritative proof scheme, the symbolic non-quantitative proofs scheme, and the result pattern generalization, which is a manifestation of the empirical proof scheme"

Hanna[113] wrote "in the classroom the key role of proof is the promotion of mathematical understanding". Hanna[113] listed eight functions of proof and proving: After verification and explanation is systematization (the organization of various results into a deductive system of axioms, major concepts and theorems). Hanna and deVilliers[114, p. 329] wrote "For mathematicians, proof varies according to the discipline involved, although one essential principle underlies all its varieties: To specify clearly the assumptions made and to provide an appropriate argument supported by valid reasoning so as to draw necessary conclusions." They[114, p. 330] wrote "proof is ... a sequence of ideas and insights with the goal of mathematical

understanding—specifically, understanding why a claim is true.”

Steffe[249], Cobb and Yackel[55, 56] are using constructivism to learn students’ conceptualization of concepts in mathematics. Dubinsky[74] argued for the use of reflective abstraction, to induce students to reflect, and developed APOS theory (actions-processes-objects-schema) to describe development of students’ conceptual understandings[15].

Simon and Tzur[240] attribute development of a new conception to a process involving learners’ goal-directed activity and natural processes of reflection.

### 7.5.1 Mathematical Problem Solving

Catherine Ulrich[268, p. 370] pointed out that problem solving “can be interpreted throughout as referring to her drawing of a diagram, her translation of quantitative relationships represented in the diagram into algebraic equations, and manipulation of her algebraic notation to arrive at a numerical answer.” Ulrich[268, p. 371] went on to say “moves from recognition to a nascent anticipation of the results of her actions and of her ability to run through her actions using mental representations both serve as a nice elaboration of what comes between Piaget’s third and fourth levels of projections ... the fourth level of projection corresponds to *reflected abstraction*.”

Cai et al.[47] wrote summaries of six empirical papers on problem solving. Cai and Lester[46] “found a strong correlation between the representations teachers use and the representations their students use, ... Chinese teachers in this study overwhelmingly used symbolic representations ... U.S. teachers relied almost exclusively on verbal explanations and pictorial representations.” Lowrie[171] commented that “most traditional problem-solving tasks in mathematical classrooms are disconnected

from students' out-of-school experiences. In fact, students learn to disregard their real-world knowledge when solving mathematical problems in a classroom environment." (This dovetails nicely with the results described by Lasry and Mazur[156], in which it was noted that a common sense physics problem was incorrectly solved only by the physics students.) Weber[293] argued that "attempting to construct a proof can be viewed as a problem-solving task in which the prover is asked to build a logically valid argument that concludes with the statement to be proven." He illustrated three modes of reasoning (procedural, syntactic, and semantic) that point to some of the qualitatively distinct ways in which undergraduates successfully construct proofs in their advanced mathematical courses. He then analyzed what students can learn through constructing proofs in each mode of reasoning. Grugnetti and Jaquet[110] referred to situation theory "...encourages ...students with successive tasks that reflect increasingly challenging variants on the same situation ...in the context of problem solving and proof."

## **Words in Mathematical Learning**

Amalric and Dehaene[9] have shown, using magnetic resonance imaging, that brain areas involved in mathematic reasoning are distinct from language areas. They cite Einstein, who, according to Hadamard[111] said "Words and language, whether written or spoken, do not seem to play any part in my thought processes." Eide and Eide[80] write of Einstein "thinking in words was not natural for him, and that his usual mode of thinking was nonverbal. To communicate verbally he needed first to 'translate' his almost entirely nonverbal thoughts into words. Einstein described the process this way: '[C]ombinatory play [with nonverbal symbols] seems to be the es-

sential feature in productive thought – before there is any connection with logical construction in words or other kinds of signs which can be communicated to others ... Conventional words or other signs have to be sought for laboriously only in a secondary stage.’ ” [80]

While assignments that are reflections, designed to promote reflective abstraction, seem desirable as a means of encouraging the re-presentation (in the sense of von Glasersfeld) of mathematical ideas, it might be that stressing the use of language in writing reflections diminishes their effectiveness. Perhaps a stress upon diagrams would be helpful.

### 7.5.2 Proof

Hanna and deVilliers [114, p. 334] wrote “Frequently, students do not see a connection between argumentation in empirical situations and mathematical proofs. They consider proof a mathematical ritual that does not have any relevance to giving reasons and arguments in other circumstances or disciplines.”

Edwards and Ward [79, p. 223] wrote “One of the earliest subjects of undergraduate mathematics education research was students’ difficulties in writing formal mathematical proofs.”

Epp [84, p. 886] wrote “Over the past several decades, many of those involved with mathematics education at the college and university level have turned their attention to the difficulties students experience in mathematics courses requiring them to write proofs. ” and “Often their efforts consisted of little more than a few disconnected calculations and imprecisely or incorrectly used words and phrases that did not even advance the substance of their cases. My students seemed to live in a different logical



and linguistic world.”

Epp[84, p. 888] wrote “Work by Thompson[263] and Goetting[105] indicates that statements requiring proof by contradiction or proof by contraposition are particular sources of difficulty, even for students who have taken advanced mathematics courses.”

Alcock and Inglis[5, 4] and Weber[296] wrote a series of articles discussing representation systems in proof production by students. One definition of the difference between syntactic and semantic proofs is: Proofs produced mainly within a representation system are thought of as syntactic, whereas those for which the work performed is mainly outside the representation system, are semantic [5, p. 209]. Another is: Proofs produced entirely within the representation system of the to-be-proved statement are thought of as syntactic, whereas those for which the work performed is in any part outside the particular representation system of the to-be-proved statement, are semantic [5, p. 209]. Alcock and Inglis[5, p. 210] stated: “complex proof productions . . . might involve multiple translations between representation systems”. They wrote “the non-trivial ability to accurately translate back and forth between representation systems” Alcock and Simpson[6] stated: “However, they (students) must now (in university) also work with entire categories of objects . . . showing that a whole category is contained within another category.” Alcock and Simpson[6] described three strategies, generalization, property abstraction and working from definitions, that develop in university students of mathematics.

Moore[196] performed a grounded-theory study of students of mathematics at the university level, who were taking a course on group theory or on proof-reading and proof-making. He concluded that the major sources of the students’ problems with proofs were: concept understanding, mathematical language and notation and getting started. Concept understanding involves definitions and images, as well as usage of

the concept. He found that some students do not readily shift from “show that  $x$  is  $X$ ” to “show that  $x$  satisfies the definition of  $X$ ”. Alcock and Simpson[6] point out that some students do not make a distinction between a dictionary definition that describes a pre-existing collection of objects and a mathematical definition that establishes a collection and determines its members. Alcock and Simpson[6, p. 33] seem to have been describing some of our participants when they stated “If the lecturer also provides examples, intending them to be illustrative, the student may even construct their prototype on the basis of experience with them, ignoring the definition altogether.”

A 2011 issue of Mathematics Education was devoted to exemplification in mathematical thinking. In this issue, according to Antonini et al.[12], Bardelle and Ferrari[22] “analyze the interplay between definitions and the use of examples in the particular case of the definition of the monotonicity of functions. In an introductory mathematics course for undergraduate students, they asked students to explain the reasons why some examples do or do not satisfy a given definition.”

Bardelle and Ferrari[22, p.233] wrote “What makes examples so important is that they are the link between the general and the particular.”

Dubinsky et al.[73, p. 44] find definitions with multiple mixed quantifiers logically complicated for students of mathematics. We found this for students studying the pumping lemma for regular languages, and for context free languages. They state “Quantification is, on the other hand, one of the least often acquired and most rarely understood concepts at all levels, from secondary school on up – even, in many cases, into graduate school.” Ralston, according to Dubinsky et al., has expressed the opinion that quantification is too difficult for students in the first two years of university. Dubinsky et al.[73] recommended staging the students’ understanding, ensuring back-

ground, introducing a single level of quantification, and considering multiple levels. Dubinsky et al.[73, p. 47] wrote “In general, students appear to have considerable difficulty with negation.” This matches our participants. Dubinsky et al.[73] found that some students erred while negating statements containing both at least one quantifier and an implication. These students negated the quantifier correctly but became confused about the implication. Moreover, each of these erring students used a memorized rule. Each student who described negating the implication first, and then attending to the rest of the expression, got the negation correct.

The teaching method these researchers found that helped students is described as constructing the quantification schema.

1. There are simple declarations. These are then linked with the standard logical connectors (AND, OR, IMPLIES, with NOT).
2. There are variables, that have domains and are iterated through their domains, resulting in truth or falsity for functions of these variable.
3. Connect logical expressions (boolean functions) by means of all conjunction or all disjunction.
4. Replace with universal or existential quantifier.
5. Acquire the idea of iterating a variable through its domain to obtain a set of propositions and apply a quantifier to form a single proposition.
6. Parse two-level quantifications with two variables into an inner and an outer quantification, such that the inner quantification uses on value of the variable that is iterated in the outer quantification.

7. Recognize higher level quantifications, group two inner quantifications and apply the two-level schema. See that nesting to any level is workable.

Dubinsky[72] wrote that in implementing instruction based on the above principles, they found that “computer experience can be an effective way of not only helping student to construct reasonable schemas, but also to get them to reconstruct erroneous or incomplete conceptions. The basic principle is that anytime you construct something on a computer then, whether you are aware of it or not, you construct something in your head.” He does not appear to cite Papert for Constructionism. ISETL has a web page, <http://homepages.ohiodominican.edu/~cottrilj/datastore/isetl/>, which says it was last updated in 2015. The computer exercises did appear to help the students.

Barnard[23] wrote on students’ ability to negate statements, and the impact of meaning on this, reporting “even after two years at university, one in three students could not negate apparently simple statements.”

Lin, Lee and Wu Yu, in [162], wrote about students’ understanding of proof by contradiction, and reported that negating “some”, “all” and “only one” presented students with decreasing difficulty in that order.

Alcock and Weber[7] wrote that validity in proof requires that there is “good reason to believe that each statement follows from the preceding statements or other accepted knowledge, i.e., that there is a valid warrant for making that statement in the context of this argument.” They stated[7, p. 133] “Our results are consistent with Selden and Selden’s (2002) finding that students’ performance ... can improve considerably when they are encourage to reflect upon their critique of a proof. This suggests that the ability to validate proofs may be in many students’ zone of proximal

development and that students' abilities in this regard might improve substantially with relatively little instruction." They suggested "explicit classroom activities be devoted to inferring and evaluating the often implicit warrants used in mathematical proofs" because otherwise "students may fail to see the difference between evaluating the logical *truth* of an implication and using a warrant to determine whether a statement is *legitimate* to assert in a proof."

Bates and Constable[24] wrote about using proofs to generate programs. Use of their language PRL beyond what was probably the intended purpose, could give students another representation of a proof, one they could check, by using programming, which may be a relatively strong skill for them. A PRL website <http://www.nuprl.org/> exists, which has been updated in 2015.

Blanton and Stylianou[31] used scaffolding with whole class discussion to help students learn to construct proof. They wrote "the extent of one's development within the ZPD is predicated in part upon how the more knowing other organizes, or *scaffolds*, the task at hand." They wrote "Our results suggest that students who engage in whole-class discussions that include metacognitive acts as well as transactive discussions about metacognitive acts make gains in their ability to construct mathematical proofs. . . .students can internalize public argumentation in ways that facilitate private proof construction if instructional scaffolding is appropriately designed to support this."

Furinghetti and Morselli[95] address a subject whose importance has been indicated by noted mathematicians including Hadamard and Poincaré, namely that emotions are involved in doing mathematics, including proof. They relate emotions to beliefs. They mention a belief (held by students) reported by Schoenfeld, that "students who have understood . . . will be able to solve any assigned problem in five

minutes ”[229, p. 359]. Schoenfeld wrote that students can be significantly affected by their beliefs. Furinghetti and Morselli sought subtle emotions, e.g., curiosity, frustration, confidence. It may be that helpful instructors should be concerned about management of affect in the students.

Healy and Hoyles[124, 125] asked students in 9th grade what proof means and what it is for. Over a quarter of these able students had little or no sense of the purpose of proof or its meaning[125, p. 417-418].

Sowder[247] found that “Exposure to the same material twice allows the student, on the second exposure, to focus on proof methods.”

Harel and Sowder[119] wrote “Overall, it appears that least some of the deficiencies in students’ acquisition of more sophisticated proof schemes may stem from the lack of opportunity to engage in proof-fostering activities.”

### **Proof by Mathematic Induction**

Dubinsky[71], in Teaching Mathematical Induction I, taught mathematical induction to a small class, using computer experiences to attempt to guide students to make appropriate reflective abstractions. Dubinsky wrote: “Mathematical induction is very hard for those undergraduates who, although reasonably successful in other subjects, do not have special talents for mathematics. It seems to be generally agreed by college mathematics teacher that, in general, our students do not come to understand this concept. Indeed, if you question students – even those who have had several mathematics courses – almost all of them will have heard of induction, not many of them will be able to say anything intelligent about what it is, much less actually use it to solve a problem. ... Very few students can actually construct an induction

proof and not many can understand one.”[71, p. 305]. Dubinsky wrote: “In any case, it seems clear that even after a learner understands a concept at the level of being able to explain it, further development is necessary” (to use it in solving problems)[71, p. 308] and “even though they may be given practice with a large number of examples, lacking a schema with which to assimilate and organize these ‘experiences’, they do not develop any autonomous skills relative to induction.”[71, p. 308]. Experience writing code that would evaluate a proposition that was a function of input ( $P(N)$ ), together with writing code that would accept an integer ( $N$ ) and evaluate the implication  $P(N) \rightarrow P(N+1)$  was helpful to the students. The experiment was assessed by interviews, and student success on a problem.

Harel and Brown wrote[117, p.112] “Many students do not possess a *deductive proof scheme—a scheme by which assertions are proved by the rules of deduction*—and cannot appreciate, or have difficulty answering . . . skepticism; hence, they see no need to look for a deductive proof.” They wrote [117, p.114] “Students experience major difficulties understanding the statement of the principle of mathematical induction, when the principle is applicable, and how to apply it.” This corresponds to our students, who do not think to apply mathematical induction to discover whether a given problem is suitable for a solution by recursion. They[117, p.115] wrote, of some students, “the student views mathematical induction as a mere procedure for manipulating symbols – a manifestation of the symbolic non-referential proof scheme rather than a deductive process.” They report an incident when an instructor asked a class about mathematic induction, and they replied only “It is a proof with steps.” A similar incident has been reported by Dubinsky[71]. Harel and Brown wrote[117, p.115] “In sum, our interpretation of students’ difficulties understanding the statement of the principle of mathematical induction, when the principle is applicable, and

how to apply it, is that these difficulties are, for the most part, a result of employing instructional techniques that (a) do not facilitate a need – an intellectual need – for mathematical induction and, (b) do not allow mathematical induction to arise as a proof technique.” Harel and Brown[117, p.117] conducted experiments in instruction, in which they gave problems implicit in their use of recursion, then explicitly formulated as recursion. Students are then asked to reflect upon similarities between the two sets of problems and their solutions, to help them discern (abstract) a method. Students are then asked how they might apply this method, i.e., what kind of problems are amenable to this method. Afterwards they are instructed in the principle of mathematical induction. Harel and Brown[117, p.118] gave an illustration of the difference between inferring a pattern from a sequence of results, and inferring a pattern from the process that a problem describes. They wrote “Recognizing and understanding the use of an inductive hypothesis is nontrivial for many students. ...students found their use of an inductive hypothesis problematic in that they viewed the inductive hypothesis as something to be proved”. (This is as opposed to the inductive step, which of course must be proved.) Even students of mathematics are not seeing the linkage formed from one basis and any natural number of connecting links as the structure of the proof.

Krantz [8] describes proof by induction, giving several examples in this book of proof techniques for computer science.

Harel and Brown[117] wrote “Despite its intuitive appeal and its central role within mathematics, research in mathematics education has documented that students have major difficulties understanding mathematical induction”, citing Dubinsky, Fischbein, Engel, Movshovitz-Hadar, Reid, Robert and Schwarzenberger.



## 7.6 Computer Science Education on Proof

Given that our interest is fostering intellectual growth in the domain of proof, it makes sense to enquire how most effectively to interact with the construction of ideas, particularly relevant to proof, being carried out by the student.

Lamport, in 1995, [154] in work on proof construction, has given one approach that computer science students might find compatible with their background.

This section mentions related work on logic, translation among representations, abstraction, abstraction and representation.

### 7.6.1 Logic

Almstrum[8] showed that students taking the AP exam in computer science had a more difficult time with questions involving logic than with other questions.

Almstrum[8] has investigated the understanding of undergraduate computer science students of problems related to logic, compared to problems only weakly related to logic, and has shown that some students have trouble with the notion of truth or falsity.

Goldman et al.[107] showed that a difficulty with logic could also be detected in students of computer science when studying discrete math. In this context, students had difficulty with proof techniques, with proofs by mathematic induction and indirect proofs being more difficult; they also had difficulty proving correctness, and converting natural language into logic formulas, and understanding the halting problem. This study also showed students having difficulty converting a verbal specification into a state diagram, relating a timing diagram and a state machine, understanding race conditions, converting algorithms to register transfer logic, and debugging, testing

and troubleshooting.

Herman et al.[127] focused on the difficulty students (who have taken discrete math and have just taken digital logic) have with propositional logic, and in translating natural language expressions into Boolean formulas. One possibly surprising result from Herman et al., though consistent with what this dissertation found, was that students have difficulty negating, to such an extent that a NAND with two operands was confusing. Herman et al. reported “One student seemed to even struggle with what ‘not both’ meant.” Herman et al. also reported that students had difficulty with “if and only if” by itself and XNOR by itself. Herman et al. give evidence that students sometimes choose to attempt memorization when we might hope that they would have understanding that would make such memorization unnecessary, reporting “if A implies B then you get A OR NOT B, so I can’t remember if that it’s exactly what it is, but it’s something like A OR NOT B.” Another helpful finding by Herman et al. is that some students omit negated variables from Boolean formulas, so, when the correct formula is  $ab\bar{c}$  some students write  $ab$ . Yet another useful insight found by Herman et al. is that students have difficulty relating the use of logic to their understanding of programs: “The most ambiguous construction for the subjects was ‘if - then.’ On several occasions subjects specifically mentioned that they knew that there is a difference between how ‘if - then’ is used in programming or colloquial English and how ‘if-then’ is used in Boolean logic. Despite this knowledge many subjects were unable to articulate the difference between the contexts.”

From Montfort et al.[194], we learn that “students reason using simplified causal relationships”.

Epp[84, p. 890] observes that use in natural language of some of the same words (if) we use in the context of mathematical proof can be confusing, because the mean-

ings in mathematics are not only precise but also different from the natural language understandings. Moreover, these words as used in natural language phrases can be interpreted to mean more than they would mean in mathematical context, for example, saying “some” in a non-mathematical context can denote  $\exists x \wedge \neg \forall x$ , but in a mathematical context denotes  $\exists x$ .

Epp[84, p. 891] wrote “Emphasizing general principles probably leads to deeper understanding, but if time is limited, weaker students may fail to connect the principles with specific problems ... Focusing on narrow problem-solving strategies may lead to greater success for a larger fraction of students ... but it may ... provide an inadequate basis for more advanced work”

Epp [84, p. 892] wrote that Cheng et al.[51] “indicates that (1) when training in the abstract rules of logic is combined with training using concrete, discursive examples, improvement in students’ reasoning performance is significantly greater than when either abstract training or examples training is administered alone, and (2) explaining logical principles by reference to analogous ‘pragmatic reasoning schemas,’ such as are used in everyday discourse about permission and obligation, increases the likelihood that students will apply the principles in more abstract contexts.”

Epp [84, p. 895] wrote “it is helpful to introduce each principle with examples of sentences whose ‘natural’ interpretation agrees with the one used in standard logic. One can even suggest that students learn a few of these sentences to refer to as prototypes when they are unsure about how to interpret formal mathematical statements. It is also helpful, however, to acknowledge explicitly some of the differences between mathematical logic and ... everyday life.”

Epp [84, p. 896] wrote “Moore reported that students in a transition-to-higher-mathematics course made good progress in learning to perform such translations

(between formal and informal modes of expression) and that a number cited the experience as contributing to their ability to do proofs.”

Epp [84, p. 896] wrote “Exercises that mix logic, language and mathematics help sensitize them to the importance” of symbols and small but significant words, such as “if”, “and” and “or”.

According to Hanna and deVilliers, writing in 2008[114], “Some educators hold the traditional assumption that teaching students elements of formal logic, such as first-order logic with quantifiers, would easily translate into helping them to understand the deductive structure of mathematics and to write proofs. However, research has shown that this transfer does not happen automatically. It remains unclear what benefit comes from teaching formal logic . . . Hence, we need more research to support or disconfirm the notion that teaching students formal logic increases their ability to prove or to understand proofs.” One can wonder whether it is the case that something more is needed, to exercise the utility of the formal logic, for students needing to prove and/or understand proofs.

Harel and Sowder[119] reported: “It is interesting to note that the first National Assessment of Educational Progress (1972-1973) included a multiple-choice item on logic that involved recognizing the logical equivalent of ‘All good drivers are alert.’ Only about one-half of the 17-year-olds, 11th graders, chose the correct alternative (‘A person who is not alert is not a good driver’).”

DeRidder, Silver and Kenney[66] wrote about students in 9th grade, that the time teachers spent “developing reasoning and analytical ability” corresponded to better student achievement.

## 7.6.2 Abstraction and Representation

Hazzan[122] wrote on mathematics students and also computing students, carrying out a reduction of abstraction that was not necessarily beneficial.

Harel and Sowder[119] wrote that remarks (including some by Balacheff [19, p. 177], a world recognized pioneer in the area of the learning and teaching of proof) “call attention to the move from concrete to abstract”.

Boustedt[37] wrote “People who discuss design of software need to represent their thoughts and ideas with some sort of model. The Unified Modeling Language (UML)[221] is a visual model language that fits object-oriented (OO) analysis and design well.”

Marton and Säljö[180] have shown the significance of letting students know the type of test they may expect, for the type of study the students employ. Students who know they will be tested via performance utilizing concepts, for example, by essay or oral exam, adjust their learning to gain principles and main points, compared with students who expect, for example, multiple choice tests, and adjust for retaining details.

Montfort et al.[194] mentioned the importance of abstraction to difficulties students experienced: “The first theme of inappropriate groupings emphasizes the importance of the organization of knowledge in students’ acquisition of new concepts in accordance with theories that focus on the coherence of students’ knowledge organization (e.g., knowledge-in-pieces or naïve theory).” They go on to mention that multiple representations are used: “Second, these inappropriate groupings extend beyond the use of proper terms to the use of symbols and diagrams.” Montfort et al. suggest that “As with the theme of inappropriate groupings, students’ use of direct simplified

causal narratives reveals a commitment to simple rather than complex organizations of knowledge.” Locke has written[169, Book III, chapter II, §6] “since all thinks that exist, are only particulars, how come we by general terms, or where find we those general names they are supposed to stand for? Words become general, by begin made the signs of general ideas: and ideas become general, be separating from them the circumstances of time, or place, or any other ideas that may determine them to this or that particular existence. By this way of abstraction, they are made capable of representing more individuals than one; each of which, having in it a conformity to that abstract idea, is (as we call it) of that sort.” Later, Locke[169, Book III, chapter XI, §25] “words standing for things which are known, and distinguished by their outward shapes, should be expressed by little draughts and prints made of them. A vocabulary made after this fashion, would, perhaps, with more ease, and in less time, teach the true signification of many terms, . . . and settle true ideas in mens minds, of several things, . . . than all the large and laborious comments of learned critics.”

This work of organizing many ideas into, for example, a hierarchical network of defined ideas, is a respectable work: consider Carolus Linneaus’ work of categorizing plants, or Dewey’s work of categorizing subjects for libraries.

Locke wrote[169, Book IV, chapter III, §30] “Where we have adequate ideas, and where there is a certain and discoverable connexion between them, yet we are often *ignorant, for want of tracing these ideas we have* or may have, and finding out those intermediate ideas . . . thus many are ignorant of mathematical truths, not out of any imperfection of their faculties, or uncertainty in the things themselves; but for want of application in acquiring, examining, and by due ways comparing those ideas”.

Much respect has been accorded to those who have developed especially helpful visualizations and mathematizations. The IEEE Spectrum article “Modern Maestro”,

about David Forney[10], reports that “The result was a landmark 1973 paper in the Proceedings of the IEEE that popularized the Viterbi algorithm by introducing a visualization technique called the ‘trellis diagram’.” Anderson, in “Modern Maestro” quotes Forney as saying “After a discovery has been made, he’ll add to it.” Anderson goes on to say “In a sentence, he (Forney) sums up these insights: ‘You know, the right mathematical language to talk about this invention is *this*.’”[10, p. 47].

Kaiser, of Richard Feynman, wrote: “For all of Feynman’s many contributions to modern physics, his diagrams have had the widest and longest-lasting influence. Feynman diagrams have revolutionized nearly every aspect of theoretical physics since the middle of the twentieth century.”[140, 3,4] Kaiser goes on to write “With the diagrams’ aid, entire new calculational vistas opened for physicists. . . .Feynman diagrams helped to transform the way physicists saw the world.”

Moström et al.[197] and Boustedt[37] have written about abstraction by computer science students working with class diagrams. Moström et al.[197] reported that abstraction *per se* is not a threshold, which is to say, not an idea the discernment of which changed the way they experienced computing. Abstraction was relevant, however, in ideas the students did judge to be transformative.

Aspinwall et al.[18] wrote about the controversy on use of visual representation. They wrote “It is quite clear in some of the literature (e.g., Janvier, 1987, Lesh et al. 1987, National Council of Teachers of Mathematics, 1989) that having multiple ways to represent knowledge is beneficial.” They go on to report that some students suffer from “uncontrollable mental imagery”, and for these students visual representations can constitute a hindrance.

### 7.6.3 Quantifiers

Dubinsky and Yiparaki[75] studied understanding of quantifiers, including after intervention. They wrote “we present results that suggest that what understandings students do appear to have of quantified statements in everyday situations may not transfer very easily to mathematical situations. ...we need to help our students learn how the language of mathematics works in order to communicate with them. ...instead of trying to make everyday life analogies between ordinary English statements and mathematical statements, perhaps we should remain in the mathematical contexts and concentrate our efforts directly on helping students understand mathematical statements in their natural mathematical habitats.”

In 2010 Pillay[205] asserted that “there has been no research into the actual learning difficulties experienced by students with the different topics” in formal languages and automata theory. Of the pumping lemmas, Pillay stated “A majority of the students made logical errors when proving that a language is regular and using the Pumping Lemma to show that a language is nonregular. These could be attributed to a lack of problem-solving skills and an understanding of the Pumping Lemma.” Devlin[68] observed that quantifiers can appear daunting to the uninitiated, and that statements containing multiple quantifiers can be difficult to understand.

### 7.6.4 Symbols

Hüttel and Nørmark[134] described a successful method for improving both student activity level in the course and final grades, which combines peer assessment with creation of notes that can be used during the exam. (“The incentive was that their answers to text questions would be available for them to use at the written exam.



No other textual aids would be allowed at the exam.”[p. 4]) The better performance on the exam is welcome; whether it is due to having notes compared to closed book, or having performed the review might not be certain. According to Arnoux and Finkel[16], it is not unusual for students to acquire mathematical knowledge without attaching meaning to it, and leaving them unable to solve some problems. They go on to report that Paivio proved that “double coding (verbal and visual)” facilitated remembering. They also report that different parts of the brain are used to process verbal and visual information, and therefore more of the brain is involved when both verbal and pictorial communication is used. They prefer multi-modal representations. Xing[303] writes about aiding students comprehension of proofs being aided by graphs. She reports “students feel that Pumping Lemma(PL) is so abstract to grasp that using it to prove that a language is non-regular is a daunting task.” She shows a graphically laid out proof that a given language is not regular. This graph has the advantage over a traditional proof, i.e., a sequence of statements, that the dependencies of states on axioms or intermediate results are plainly shown by graph edges. Simon et al.[237] ask “Is it possible that students plug and chug in computing, not really understanding the concepts as we would like them to?” and go on to say “We posit that the need exists for computing instructors to design assessments more directly targeting understanding, not just doing, computing. And, of course, to adopt teaching approaches that support student development of these skills.” Mazur[185] developed peer instruction to address students’ propensity to practice a plug-and-chug approach to problems. This approach has been applied to computer science teaching, including theory of computation, by several researchers including Simon, Zingaro, Porter, Bailey-Lee and others[237, 211, 212, 238].

Dubinsky[72], guided by constructionist ideas of Piaget, applied programming in

his language ISETL to teaching mathematics students about quantification. The results were that this method was more effective than its predecessor – students developed “some understanding of quantification and the ability to work with it, even when the particular problems ... were difficult.”

### 7.6.5 Mathematic Induction and Recursion

In the literature of computer science education we found research [35] on a different topic, but with similar results. Booth reported categories of conceptions of recursion similar to our categories of conception of proof by mathematical induction.

Both mathematics and computer science teach students about proof by mathematic induction. Kinnunen and Simon[144] describe an example applying phenomenography to computing education research, listing several recent examples, and also providing a detailed description of a mainly data- but also theory-driven refinement of categories. Berglund, Eckerdal and Thuné [78, 27, 266] have applied phenomenography to computing education research, obtaining classifications by judicious grouping of student conceptions derived from interview data. Eckerdal et al. [77] describe how the results using phenomenography showed additional insights beyond other methods.

### 7.6.6 Teaching Pumping Lemmas

In 2003 Weidmann[298] wrote a dissertation on teaching Automata Theory to students at the college level. She found that past performance in prerequisite theory courses was a statistically significant indicator for success in their college level course. She described a theoretical framework called “pedagogical positivism”, a stance between

logical positivism and constructivism, allowing the notion of a teaching method best suited to a group of students to learn Automata Theory. She interviewed a teacher with “several” years of experience teaching this course (p. 5), who “admitted that she did not have a better way to teach abstract thinking other than repeated exposure” (p. 98). In chapter 5, Discussion, Conclusions and Implications, of this dissertation[298], the suggestion “Instead of simply providing the solution to a problem in class, or stating the intuitive leap that makes the problem easy to solve, the students should be exposed to the iterative thought process that lead to the intuition that created the solution.”(p. 201) appears. One suggestion is “Learning objectives should be set to focus on familiarity with formalisms and rigorous mathematical notations (p. 224) and another suggestion is Include programming projects as part of the required coursework”(p. 224). The combination of these brings to mind the suggestion of Harel and Papert[121]: “constructing personally designed pieces of instructional software”, and the thought that the students might dwell more effectively on the notion of abstraction as they tried to teach someone else about it.

## Chapter 8

# Conclusion

Harel and Sowder [118, p. 277] stated:

by their natures, teaching experiments and interview studies do not give definitive conclusions. They can, however, offer indications of the state of affairs and a framework in which to interpret other work.

In this chapter there are conclusions associated with each research question, and arising from the combination of the investigation into the individual questions. In summary, the conclusion is that students of computer science in this study do not differ greatly from those studied by Harel and Sowder[119] and by Healy and Hoyles[124, 125]. Healy and Hoyles asked students to describe what proof means and what it is for, and this quantitative study reported that “over a quarter of these able students had little or no sense of the purpose of proof or its meaning.” Some of our students could suggest many purposes of proof. Though asked, none of the students reported making use of proof other than when assigned.

## 8.1 What Students Think Proofs Are

Students' thoughts about what proofs are evolve during the curriculum. Before they have seen proofs used in analysis of algorithms, some students think that the discrete mathematics course serves to acquaint them with the specific theorems proved in the class.

Even after seeing proofs applied in analysis of algorithms, some students do not reflect on their previous experience in discrete structures to the extent that they realize its purpose was to acquaint them with proof. For example, some students do not make a connection between proof by mathematic induction and the correctness of a recursive algorithm, even when they have earned good grades in both courses.

Some students, even when they are taking theory of computation, do not understand that computer scientists use proof to ascertain for themselves, and to convince others, of the correctness of what they assert. Instead, these students use extrinsic conviction (some authoritative source informs them), or incorrect intrinsic conviction (they are convinced by examples).

Those students who were dual majors of mathematics and computer science and engineering did appreciate the purpose of proof and the role of mathematical definitions. Of those interviewed, no student who was not a dual major in math appreciated the value of definitions in understanding warranting transformative proof steps.

A summary statement offered by a student was: "Don't care about why, as long as it works."

## 8.2 How Students Attempt to Understand Proofs

Some students do not attempt to understand proofs. Of those who do attempt to understand, some students attempt pattern matching with patterns they have learned, such as proof by mathematic induction, or proof by contradiction. Some students look for warrants from one line to the next. None of the students we interviewed mentioned divide and conquer, or seeing the role of lemmas.

## 8.3 What Students Think Proof is For

Students offered many reasons, including good reasons, for proof, such as:

- in algorithms class the professor can calculate and state confidently the resource utilization of algorithms
- communicating mathematical ideas

Absent from the reasons offered was: ascertaining whether a context suits an algorithm. When asked whether they ever used recursion, students said they did not know how to ascertain that recursion was suited to the context of a problem. They used recursion when they were instructed to for problems they were assigned.

Some undergraduates say that, for them, proof has no purpose outside of assignment completion.

## 8.4 How Students Attempt to Apply Proof, When Assigned, and When Not Assigned

The students we interviewed attempted to apply proof only when assigned. For undergraduates, these situations included homework and tests. For some graduate students, this included a manuscript, in which a proof was expected.

Some students attempt to use proofs they have seen in class, even when the claim to be proved does not match the claim proved in class. Some students attempt to use proof techniques according to how familiar they are with the technique, rather than by examining the claim to be proved. Some students modify proof techniques, possibly by errors in memory and possibly by errors in understanding, but proof-attempts reveal that understanding is lacking. Some students seem to rely on extrinsic conviction; they “don’t care why it works, as long as it does”.

## 8.5 Whether Students Exhibit Consequences of Inability With Proof

We conclude from student interview testimony, that some students never use some algorithms, or algorithmic approaches, including recursion. Though the students are comfortable using the technique in problems on homework or on tests, they do not feel confident in their judgment about contexts in which recursion is applicable.

They obtain their conviction, not from proof, which they do not attempt in situations in which they are contemplating algorithmic approaches. Instead they rely on supposedly authoritative commentary: “My uncle works at Microsoft and he says they never use recursion.”

## 8.6 Student Familiarity with Specific Proof Techniques

Some students are quite familiar with proof by mathematic induction.

Some students are most familiar with proof techniques that they find are like processes. “First I identify a base case and prove it, which is easy. Then I write the induction step and prove it, though it does not convince me.”

In theory of computation, there are several proofs following the same pattern which includes modus tollens. Even though students have seen the modus tollens expressed in variables, knowing this general formulation does not immediately help the students to specialize that formulation to the proofs about languages in which modus tollens is used.

Some students sometimes attempt variations upon proof techniques they have seen. When the variation is no longer equivalent to the proof technique, the students can err.

## 8.7 How Students Use Structure in Proof

Because students are familiar with program structure, such as method calls and specialized contexts (scopes) within switch/case statements, one might hope for transfer of this aptitude for decomposition and synthesis to be generalized to application in comprehending and constructing proofs. However, the situation seems to be that some students are:

- not sure what lemmas are



- not sure about how to use contexts/scopes in which assumptions are made, as in proof by contradiction or categorizing a problem space into cases

Thus when we discuss the use of structure, instead of recognizing it as a powerful generalization from structure in algorithms, students may not see the analogy.

## 8.8 What Students Think Makes a Proof Valid

Some students think that if they follow the procedure they have learned for constructing a proof, it will be valid.

Some students are aware that a (valid) proof can be constructed by using rules of inference in sequence.

For the students who were not dual majors with mathematics, none mentioned using the mathematical definition for providing a warrant for a transformation step.

## 8.9 Conclusions from the Combined Data

We proposed several conclusions:

C1: Outside of using known proofs to demonstrate known resource consumption properties of known algorithms, some students do not grasp the idea that within a defined context a proof provides universal, unequivocal assurance. As such they do not see how proof could be used to establish that a context is suitable for the application of an algorithm. Some students do not know why they might want to use proof. Some students, given difficulties with transfer, may

not recognize that proof is helpful in determining situations to which knowledge can be transferred.

Some students' difficulty with generalization could easily contribute to their difficulties with transfer, may not recognize that proof is helpful in determining situations to which knowledge can be transferred.

C2: When applied to proof, some students' difficulty with generalization could easily contribute to their difficulty with transfer. Mathematical formulation, much used in the application of proof to situations students may encounter outside of class, is a form of generalization. The perception that proof techniques exercised on number theoretic ideas are about number theoretic ideas rather than about proof techniques might discourage transfer of proof techniques.

C3: Some students have a difficult time recognizing that the same thing is being expressed when this thing is expressed using multiple representations. Some widely recognized especially good students can do this, reporting it is challenging but fun to switch between representations. These representations are of algorithms, data structures and logical transformation steps. Some students report not noticing similar information being taught in more than one class, due to a change in the representation forms being used from one class to the next. Some students, lacking ability to translate between representations, are hindered by the use of different representations they encounter in proofs. Example is seeing recursion in code, seeing data structures in graphs, seeing proof by mathematic induction in mathematical formulation.

C4: Some students do not see that we are teaching them argumentation because we

believe they will need to be able to understand and sometimes use argumentation in their work as programmers.

- C5: Some students do not have a sense of direction about the transformations they are trying to achieve, reporting that they spend a lot of time attempting to try all rules of inference. Some students do not match the granularity of progress by applying a small number of transformation steps with the granularity of progress they are trying to achieve.

For those students for whom there is room for improvement, it would help them to know that they are constructing an argument, of the form, we can assert  $x$  because  $y$ , and that the overall argument is built from smaller arguments, and the smallest arguments are rules of inference. Moreover, the structure of that statement the proof is to show give valuable guidance to the structure to the proof. Moreover, the definitions of the entities appearing in that statement the proof is to show are important for finding reasons, for justifying transformations.

- C6: students' attention to an item depends upon their (possibly naive) assessment of the importance of that item ("when definitions are given, most students tune out. I try to listen, but I stop taking notes . . . i really prefer examples.")

## **Representation**

Some students have made clear that the choice of representation matters. Not only are some methods of representation more engaging and communicative than others, some students do not translate from the representation in which they learn a concept, to a more general representation of that concept. When they see the same concept

taught in another class using a different representation, some students do not make the connection.

### **Structure**

Some students notice a similarity between structures, such as modules, switch case constructs and recursion, used in programming and structures or patterns used in proof techniques.

Some students do not find this similarity helpful. Some students do not feel comfortable creating proofs that are not patterns that can be produced by following a procedure. Some students do not notice that additional premises provide a point of departure for logical deduction.

Some students, familiar with if-then-else constructs, may overlook that an implication may have a false truth value.

## **8.10 Recognizing an Endpoint**

A qualitative study is thought to be finished when an internally consistent narrative, compatible with the data, both situating the data and explaining them, has been produced. For our research questions, a model, accompanied by the report (see section 5.11), which is a narrative combining the information obtained from inquiry about these topics, completes the work.

It is not unusual for qualitative studies in education to generate implications for pedagogy. A product based on this model takes the form of a brochure (see Appendix A) in which the critical aspects inferred during the analysis are emphasized within

the curriculum.

Data from our extended student body, that provide a persuasive model containing categories of conceptualizations, and that are closely enough related that some insight about concepts differentiating adjacent categories can be inferred, are thought sufficient to generate this narrative. The proposed differentiating concepts are thought to have the potential to become material for a larger survey, thereby providing a starting point for new work.

## **8.11 Application of Findings / Implications for Pedagogy**

These are proposals that would be applications of the findings.

In the phenomenographic /variation theoretic paradigm, the implications for pedagogy are that the identified critical factors/aspects (henceforward, factor) are to be discussed in class, in a way that incorporates variation. In particular, properties associated with the critical factor are assigned values along a dimension of variation, and the consequences of the different values are discussed.

Given Sowder's observation[247], that "Exposure to the same material twice allows the student, on the second exposure, to focus on proof methods.", the implication is that in a course after discrete structures, such as software engineering, or algorithms, we could show the application of proof to correctness or resource consumption, and illustrate the dimension of variation, varying the value assigned to the property associated with the critical factor.

Some examples of this are given in the brochure, see Appendix A.

## Chapter 9

### Future Work

A cluster of related problems exists. Some of these are derived from pursuing the conclusions about the research questions, checking and, where appropriate, extending these ideas. Other topics of inquiry grew out of the original investigation.

#### 9.1 Pursuit of Conclusions

These future work items are keyed to the conclusions items.

- FW1: Maybe we might want to try exhibiting the utility of proof for ascertaining whether a newly posed situation is one in which a learned technique is applicable. Clicker questions offering in opposition several situations should encourage specialization.
- FW2: Develop some exercises in generalization in a context of proof.
- FW3: Give students practice in translating between different representations,

set up some experiment to see whether this practice helps them when they are posed with the problem of converting graphs in data structures into code in data structures.

- FW4: Try interspersing teaching about proof with applications to provability-driven development; try to see whether a change in students' appreciation in the use of warrants results. Promoting extrinsic "Maybe this will be good enough for the grader" to intrinsic "I can tell whether this is good enough for the specific application of proving this code fragment always works." would be the goal.
- FW5: multiple representations: Work on problems like those in Appendix B.
- FW6: One conclusion from the qualitative study is that students' attention to an item depends upon their (possibly naive) assessment of the importance of that item ("when definitions are given, most students tune out. I try to listen, but I stop taking notes . . . I really prefer examples.")

From a (social) constructivist perspective, it is easy to appreciate that students find it much easier to build concepts upon a foundation. We are often urged to define terms before using them. In this order, students' assessment of the utility of a definition might not be high. If, by contrast, we follow the students' stated preference for examples, and offer tutorial form proof attempts, and point out that a definition is used to provide a warrant for a transformation of an expression, then it could be that students' attention is better kept.

To emphasize the role of definitions in proofs, exercises that require the warrants to be explicitly stated, perhaps along with the name of the transformation step being taken, maybe even offer the template: "The definition that allows us to

justify this step is: ”

## 9.2 Helping Students Discern Derivation for Proof of Correctness

Recall that variation theory holds that students cannot discern a thing unless contrast is provided. Pang[177] has pointed out that, for persons aware of only one language, “speaking” and “speaking their language” are conflated. Only when the existence of a second language is known, does the idea of speaking become separated from the idea of speaking a specific language. As Locke[169] pointed out, a child will include the property red with the idea of apple, until an apple that is not red is introduced. We may wish to alert students to the ability to derive code from requirements mathematically. We could show them, for example, an even function. We could show them how to construct a function that is even. We could show them how to construct a function that is odd. This might help them see the difference between writing code and testing it afterwards, and deriving the code to be correct by construction. Use what has to be true to constrain what can be written.

Phenomenography in its larger sense, including variation theory, suggests that the ways to help students discern the factors critical for them to advance their conceptualization are *contrast, separation, generalization and fusion*[177].



# Bibliography

- [1] Gerlese Åkerlind, Susan Carr-Gregg, Rachael Field, Leanne Houston, Judith Jones, Mandy Lupton, Jo McKenzie, and Cheryl Treloar, *A threshold concepts focus to first year law curriculum design: supporting student learning using variation theory*, 13th pacific rim first year in higher education conference, Adelaide, Australia, 2010.
- [2] Gerlese Åkerlind, JA McKenzie, and Mandy Lupton, *A threshold concepts focus to curriculum design: supporting student learning through application of variation theory*, Verified OK (2011).
- [3] Gerlese S Åkerlind, *Variation and commonality in phenomenographic research methods*, Higher Education Research & Development **31** (2012), no. 1, 115–127.
- [4] Lara Alcock and Matthew Inglis, *Doctoral students use of examples in evaluating and proving conjectures*, Educational Studies in Mathematics **69** (2008), no. 2, 111–129.
- [5] ———, *Representation systems and undergraduate proof production: A comment on weber*, The Journal of Mathematical Behavior **28** (2009), no. 4, 209–211.

- [6] Lara Alcock and AP Simpson, *Definitions: dealing with categories mathematically.*, For the learning of mathematics. **22** (2002), no. 2, 28–34.
- [7] Lara Alcock and Keith Weber, *Proof validation in real analysis: Inferring and checking warrants*, The Journal of Mathematical Behavior **24** (2005), no. 2, 125–134.
- [8] Vicki L Almstrum, *Investigating student difficulties with mathematical logic*, Teaching and Learning Formal Methods (1996), 131–160.
- [9] Marie Amalric and Stanislas Dehaene, *Origins of the brain networks for advanced mathematics in expert mathematicians*, Proceedings of the National Academy of Sciences **113** (2016), no. 18, 4909–4917.
- [10] Mark Anderson, *Modem maestro: G. david forney jr., the 2016 ieee medal of honor recipient, turns information theory into practice*, IEEE Spectrum **53** (2016), no. 5, 44–47.
- [11] Daniel Ansari and Ian M Lyons, *Cognitive neuroscience and mathematics learning: how far have we come? where do we need to go?*, ZDM **48** (2016), no. 3, 379–383.
- [12] Samuele Antonini, Norma Presmeg, Maria Alessandra Mariotti, and Orit Zaslavsky, *On examples in mathematical thinking and learning*, ZDM **43** (2011), no. 2, 191–194.
- [13] Abraham Arcavi, Cathy Kessel, Luciano Meira, and John P Smith, *Teaching mathematical problem solving: An analysis of an emergent classroom community*, Research in collegiate mathematics education, III **7** (1998), 1–70.

- [14] Jonas B Ärleback and Helen M Doerr, *Moving beyond a single modelling activity*, Mathematical Modelling in Education Research and Practice, Springer, 2015, pp. 293–303.
- [15] Ilana Arnon, Jim Cottrill, Ed Dubinsky, Asuman Oktaç, Solange Roa Fuentes, Maria Trigueros, and Kirk Weller, *Apos theory: A framework for research and curriculum development in mathematics education*, Springer Science & Business Media, 2013.
- [16] P. Arnoux and A. Finkel, *Using mental imagery processes for teaching and research in mathematics and computer science*, International Journal of Mathematical Education in Science and Technology **41** (2010), no. 2, 229–242.
- [17] David Aspinall, *Proof general: A generic tool for proof development*, Tools and Algorithms for the Construction and Analysis of Systems, Springer, 2000, pp. 38–43.
- [18] Leslie Aspinwall, Kenneth L Shaw, and Norma C Presmeg, *Uncontrollable mental imagery: Graphical connections between a function and its derivative*, Educational Studies in Mathematics **33** (1997), no. 3, 301–317.
- [19] Nicolas Balacheff, *The benefits and limits of social interaction: The case of mathematical proof*, Mathematical knowledge: Its growth through teaching, Springer, 1991, pp. 173–192.
- [20] ———, *The researcher epistemology: a deadlock from educational research on proof*, 2002 International Conference on Mathematics-” Understanding proving and proving to understand”. Taipei: NSC and NTNU, Citeseer, 2002, pp. 23–44.

- [21] Alvin Baranchik and Barry Cherkas, *Supplementary methods for assessing student performance on a standardized test in elementary algebra*, Research in Collegiate Mathematics Education **3** (1998), 216–233.
- [22] Cristina Bardelle and Pier Luigi Ferrari, *Definitions and examples in elementary calculus: the case of monotonicity of functions*, ZDM **43** (2011), no. 2, 233–246.
- [23] Tony Barnard, *The impact of meaning on students' ability to negate statements*, PME conference, vol. 2, The Program Committee of the 18th PME Conference, 1995, pp. 2–3.
- [24] Joseph L Bates and Robert L Constable, *Proofs as programs*, ACM Transactions on Programming Languages and Systems (TOPLAS) **7** (1985), no. 1, 113–136.
- [25] R Barker Bausell, William B Moody, and F Neil Walzl, *A factorial study of tutoring versus classroom instruction*, American Educational Research Journal (1972), 591–597.
- [26] Mordechai Ben-Ari, *Constructivism in computer science education*, Acm sigcse bulletin, vol. 30, ACM, 1998, pp. 257–261.
- [27] Anders Berglund, Mats Daniels, and Arnold Pears, *Qualitative research projects in computing education research: an overview*, Proceedings of the 8th Australasian Conference on Computing Education-Volume 52, Australian Computer Society, Inc., 2006, pp. 25–33.
- [28] EW Beth and J Piaget, *Mathematical epistemology and psychology, (translated by w. mays), d*, Reidel. Dordrecht, The Netherlands (originally published 1965) (1966).

- [29] Liz Bills and David Tall, *Operable definitions in advanced mathematics: The case of the least upper bound*, Proceedings of the Conference of the International Group for, ERIC, 1998, p. 111.
- [30] Paul Black and Dylan Wiliam, *Inside the black box: Raising standards through classroom assessment*, Granada Learning, 1998.
- [31] Maria L Blanton, Despina A Stylianou, and Maria Manuela David, *The nature of scaffolding in undergraduate students' transition to mathematical proof*, International Group for the Psychology of Mathematics Education **2** (2003), 113–120.
- [32] Benjamin S Bloom et al., *Taxonomy of educational objectives. vol. 1: Cognitive domain*, New York: McKay (1956).
- [33] David N Boote and Penny Beile, *Scholars before researchers: On the centrality of the dissertation literature review in research preparation*, Educational researcher **34** (2005), no. 6, 3–15.
- [34] S Booth, *The experience of learning to program. example: Recursion*, Proceedings of the 5th Workshop of the Psychology of Programming Interest Group, 1992, pp. 122–145.
- [35] Shirley Booth, *On phenomenography, learning and teaching*, Higher education research & development **16** (1997), no. 2, 135–158.
- [36] ———, *Learning computer science and engineering in context*, Computer Science Education **11** (2001), no. 3, 169–188.

- [37] Jonas Boustedt, *Students' different understandings of class diagrams*, Computer Science Education **22** (2012), no. 1, 29–62.
- [38] John Bowden and Ference Marton, *The university of learning*, Psychology Press, 2004.
- [39] Ilona Box, *Toward an understanding of the variation in approaches to analysis and design*, Computer Science Education **19** (2009), no. 2, 93–109.
- [40] Richard E Boyatzis, *Transforming qualitative information: Thematic analysis and code development*, Sage, 1998.
- [41] John Bransford, Linda Zech, D Schwarz, Bngid Barron, and Nancy Vye, *Designs for environments that invite and sustain mathematical thinking*, Symbolizing and Communicating in Mathematics Classrooms (2000), 275–324.
- [42] Virginia Braun and Victoria Clarke, *Using thematic analysis in psychology*, Qualitative research in psychology **3** (2006), no. 2, 77–101.
- [43] Jacqueline Grennon Brooks and Martin G Brooks, *In search of understanding: The case for constructivist classrooms*, ASCD, 1999.
- [44] Philip Burnard, *A method of analysing interview transcripts in qualitative research*, Nurse education today **11** (1991), no. 6, 461–466.
- [45] Thomas J Bussey, MaryKay Orgill, and Kent J Crippen, *Variation theory: A theory of learning and a useful theoretical framework for chemical education research*, Chemistry Education Research and Practice **14** (2013), no. 1, 9–22.

- [46] Jinfa Cai and Frank K Lester, *Solution representations and pedagogical representations in Chinese and US classrooms*, The Journal of Mathematical Behavior **24** (2005), no. 3, 221–237.
- [47] Jinfa Cai, Joanna Mamona-Downs, and Keith Weber, *Mathematical problem solving: What we know and where we are going*, The Journal of Mathematical Behavior **24** (2005), no. 3, 217–220.
- [48] Rudolf Carnap, *Introduction to symbolic logic and its applications*, Dover Books, 1958.
- [49] Gary Chartrand, Albert D Polimeni, and Ping Zhang, *Mathematical proofs: A transition to advanced mathematics*, Boston, 2008.
- [50] Daniel Chazan, *High school geometry students' justification for their views of empirical evidence and mathematical proof*, Educational studies in mathematics **24** (1993), no. 4, 359–387.
- [51] Patricia W Cheng, Keith J Holyoak, Richard E Nisbett, and Lindsay M Oliver, *Pragmatic versus syntactic approaches to training deductive reasoning*, Cognitive Psychology **18** (1986), no. 3, 293 – 328.
- [52] Michelene TH Chi, Paul J Feltovich, and Robert Glaser, *Categorization and representation of physics problems by experts and novices\**, Cognitive science **5** (1981), no. 2, 121–152.
- [53] Victor V Cifarelli and Volkan Sevim, *Examining the role of re-presentation in mathematical problem solving: An application of ernst von glasersfelds conceptual analysis*, Constructivist Foundations **9** (2014), no. 3, 360–369.

- [54] Victor Vincent Cifarelli, *The role of abstraction as a learning process in mathematical problem-solving*, (1988).
- [55] Paul Cobb, *Conducting teaching experiments in collaboration with teachers.*, (2000).
- [56] Paul Cobb, Erna Yackel, and Kay McClain, *Symbolizing and communicating in mathematics classrooms: Perspectives on discourse, tools, and instructional design*, Routledge, 2012.
- [57] John W Coffey, *Integrating theoretical and empirical computer science in a data structures course*, Proceeding of the 44th ACM technical symposium on Computer science education, ACM, 2013, pp. 23–28.
- [58] Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*, 2 ed., Thousand Oaks, CA: Sage, 2003.
- [59] ———, *Qualitative inquiry and research design, choosing among five approaches*, 2 ed., Thousand Oaks, CA: Sage, 2007.
- [60] Catherine H Crouch, Jessica Watkins, Adam P Fagen, and Eric Mazur, *Peer instruction: Engaging students one-on-one, all at once*, Research-Based Reform of University Physics **1** (2007), no. 1, 40–95.
- [61] Bo Dahlin, *Enriching the theoretical horizons of phenomenography, variation theory and learning studies*, Scandinavian Journal of Educational Research **51** (2007), no. 4, 327–346.



- [62] Philip M Davidson, *Piagets category-theoretic interpretation of cognitive development: A neglected contribution*, Human Development **31** (1988), no. 4, 225–244.
- [63] C Nevalle Dean and Michael G Hinchey, *Formal methods and modeling in context*, Teaching and Learning Formal Methods (1996), 99–116.
- [64] Norman K Denzin, *The research act: A theoretical introduction to sociological methods*, Transaction publishers, 1973.
- [65] Norman K. Denzin and Yvonna S. Lincoln, *The Sage handbook of qualitative research*, Sage, 2005.
- [66] Charleen M DeRidder, Edward A Silver, and Patricia Ann Kenney, *Results from the seventh mathematics assessment of the national assessment of educational progress*, 2003.
- [67] Keith Devlin, *Why universities require computer science students to take math*, Association for Computing Machinery. Communications of the ACM **46** (2003), no. 9.
- [68] Keith Devlin, *Introduction to mathematical thinking*, Keith Devlin, 2012.
- [69] Willi Dörfler, *Means for meaning*, Symbolizing and communicating in mathematics classrooms (2000), 99–131.
- [70] Martin Downs and Joanna Mamona-Downs, *The proof language as a regulator of rigor in proof, and its effect on student behavior*, Proceedings of the Fourth Congress of the European Society for Research in Mathematics Education, 2005, pp. 1748–1757.

- [71] Ed Dubinsky, *Teaching mathematical induction: I.*, The Journal of Mathematical Behavior (1986).
- [72] ———, *On learning quantification*, Journal of Computers in Mathematics and Science Teaching **16** (1997), 335–362.
- [73] Ed Dubinsky, Flor Elterman, and Cathy Gong, *The student's construction of quantification*, For the learning of mathematics **8** (1988), no. 2, 44–51.
- [74] Ed Dubinsky and Michael A McDonald, *Apos: A constructivist theory of learning in undergraduate mathematics education research*, The teaching and learning of mathematics at university level, Springer, 2002, pp. 275–282.
- [75] Ed Dubinsky and Olga Yiparaki, *On student understanding of AE and EA quantification*, Research in Collegiate Mathematics IV (2000), 239–289.
- [76] Anna Eckerdal, Robert McCartney, Jan Erik Moström, Kate Sanders, Lynda Thomas, and Carol Zander, *From limen to lumen: computing students in liminal spaces*, Proceedings of the third international workshop on Computing education research, ACM, 2007, pp. 123–132.
- [77] Anna Eckerdal and Michael Thuné, *Novice Java programmers' conceptions of object and class, and variation theory*, ACM SIGCSE Bulletin, vol. 37, ACM, 2005, pp. 89–93.
- [78] Anna Eckerdal, Michael Thuné, and Anders Berglund, *What does it take to learn 'programming thinking'?*, Proc. of the first international workshop on Computing education research, ACM, 2005, pp. 135–142.

- [79] Barbara Edwards and Michael B. Ward, *Undergraduate mathematics courses, Making the connection: Research and teaching in undergraduate mathematics education* (2008), no. 73, 223.
- [80] Brock Eide and Fernet Eide, *The dyslexic advantage: Unlocking the hidden potential of the dyslexic brain*, Hay House, Inc, 2011.
- [81] Jordan Ellenberg, *How not to be wrong: The power of mathematical thinking*, Penguin, 2014.
- [82] Emma Enström, *On difficult topics in theoretical computer science education*, (2014).
- [83] Noel Entwistle, *Introduction: Phenomenography in higher education*, Higher Education Research & Development **16** (1997), no. 2, 127–134.
- [84] Susanna S Epp, *The role of logic in teaching proof*, The American Mathematical Monthly **110** (2003), no. 10, 886–899.
- [85] ———, *Discrete mathematics with applications*, Cengage Learning, 2011.
- [86] Paul Ernest, *Constructing mathematical knowledge: Epistemology and mathematics education*, vol. 4, Taylor & Francis, 1994.
- [87] ———, *Constructivism: Which form provides the most adequate theory of mathematics learning?*, Journal für Mathematik-Didaktik **15** (1994), no. 3-4, 327–342.
- [88] Harold Pascoe Fawcett, *The nature of proof; a description and evaluation of certain procedures used in a senior high school to develop an understanding of the nature of proof.*, (1938).

- [89] Jennifer Fereday and Eimear Muir-Cochrane, *Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development*, International journal of qualitative methods **5** (2008), no. 1, 80–92.
- [90] H Fischbein, *Intuition in science and mathematics: An educational approach*, vol. 5, Springer Science & Business Media, 1987.
- [91] J. Franklin and A. Daoud, *Proof in mathematics: An introduction*, [web.maths.unsw.edu.au/jim/proofsch2.pdf](http://web.maths.unsw.edu.au/jim/proofsch2.pdf), 2011.
- [92] William Frawley, *Vygotsky and cognitive science: Language and the unification of the social and computational mind.*, ERIC, 1997.
- [93] Itzhak Fried, Ueli Rutishauser, Moran Cerf, and Gabriel Kreiman, *Single neuron studies of the human brain: Probing cognition*, MIT Press, 2014.
- [94] Timothy Fukawa-Connelly, *Classroom sociomathematical norms for proof presentation in undergraduate in abstract algebra*, The Journal of Mathematical Behavior **31** (2012), no. 3, 401–416.
- [95] Fulvia Furinghetti and Francesca Morselli, *Every unsuccessful problem solver is unsuccessful in his or her own way: affective and cognitive factors in proving*, Educational Studies in Mathematics **70** (2009), no. 1, 71–90.
- [96] David Fuys, Dorothy Geddes, and Rosamond Tischler, *The van hiele model of thinking in geometry among adolescents*, Journal for Research in Mathematics Education. Monograph (1988), i–196.

- [97] Dedre Gentner and Arthur B Markman, *Structure mapping in analogy and similarity.*, American psychologist **52** (1997), no. 1, 45.
- [98] Dedre Gentner and Linsey Smith, *Analogical reasoning*, Encyclopedia of human behavior (2012), 130–136.
- [99] David Gibson, *Students use of diagrams to develop proofs in an introductory analysis course*, CBMS issues in mathematics education **7** (1998), 284–307.
- [100] Mary L Gick and Keith J Holyoak, *Analogical problem solving*, Cognitive psychology **12** (1980), no. 3, 306–355.
- [101] Barney G Glaser, *Conceptualization: On theory and theorizing using grounded theory*, International Journal of Qualitative Methods **1** (2008), no. 2, 23–38.
- [102] Barney G Glaser and Anselm L Strauss, *Theoretical sampling*, Sociological methods. A sourcebook (1970), 105–114.
- [103] Barney G Glaser, Anselm L Strauss, and Elizabeth Strutzel, *The discovery of grounded theory; strategies for qualitative research.*, Nursing Research **17** (1968), no. 4, 364.
- [104] Richard Gluga, Judy Kay, Raymond Lister, and Sabina Kleitman, *Mastering cognitive development theory in computer science education*, Computer Science Education **23** (2013), no. 1, 24–57.
- [105] Mary Margaret Goetting, *The college student’s understanding of mathematical proof*, Ph.D. thesis, College Park, 1995.
- [106] Robert Goldblatt, *Topoi: the categorial analysis of logic*, vol. 98, Elsevier, 2014.

- [107] Ken Goldman, Paul Gross, Cinda Heeren, Geoffrey Herman, Lisa Kaczmarczyk, Michael C Loui, and Craig Zilles, *Identifying important and difficult concepts in introductory computing courses using a delphi process*, ACM SIGCSE Bulletin **40** (2008), no. 1, 256–260.
- [108] Tracy Goodson-Espy, *Reflective abstraction as an individual and collective learning mechanism*, Constructivist Foundations **9** (2014), no. 3, 381–383.
- [109] Eddie M Gray and David O Tall, *Duality, ambiguity, and flexibility: A "proceptual" view of simple arithmetic*, Journal for research in Mathematics Education (1994), 116–140.
- [110] Lucia Grugnetti and François Jaquet, *A mathematical competition as a problem solving and a mathematical education experience*, The Journal of Mathematical Behavior **24** (2005), no. 3, 373–384.
- [111] Jacques Hadamard, *An essay on the psychology of invention in the mathematical field*, Courier Corporation, 1954.
- [112] Richard H Hammack, *Book of proof*, Virginia Commonwealth University, 2013.
- [113] Gila Hanna, *Proof, explanation and exploration: An overview*, Educational studies in mathematics **44** (2000), no. 1-2, 5–23.
- [114] Gila Hanna and Michael de Villiers, *Icmi study 19: Proof and proving in mathematics education*, ZDM **40** (2008), no. 2, 329–336.
- [115] Gila Hanna, Hans Niels Jahnke, and Helmut Pulte, *Explanation and proof in mathematics*, Springer, 2010.

- [116] Guershon Harel, *The development of mathematical induction as a proof scheme: A model for dnr-based instruction*'23, *Learning and teaching number theory: Research in cognition and instruction* **2** (2002), 185.
- [117] Guershon Harel and Stacy Brown, *Mathematical induction: Cognitive and instructional considerations*, *Making the connection: Research and practice in undergraduate mathematics* (2008), 111–123.
- [118] Guershon Harel and Larry Sowder, *Students proof schemes: Results from exploratory studies*, *Research in collegiate mathematics education III* **7** (1998), 234–282.
- [119] ———, *Toward comprehensive perspectives on the learning and teaching of proof*, *Second handbook of research on mathematics teaching and learning* **2** (2007), 805–842.
- [120] Guershon Harel and David Tall, *The general, the abstract, and the generic in advanced mathematics*, *For the learning of mathematics* **11** (1991), no. 1, 38–42.
- [121] Idit Ed Harel and Seymour Ed Papert, *Constructionism.*, Ablex Publishing, 1991.
- [122] Orit Hazzan, *How students attempt to reduce abstraction in the learning of mathematics and in the learning of computer science*, *Computer Science Education* **13** (2003), no. 2, 95–122.
- [123] Orit Hazzan and Yael Dubinsky, *Abstraction*, *Agile Anywhere*, Springer, 2014, pp. 53–55.

- [124] Lulu Healy and Celia Hoyles, *Justifying and proving in school mathematics: Technical report on the nationwide survey*, Institute of Education, University of London, 1998.
- [125] ———, *A study of proof conceptions in algebra*, Journal for research in mathematics education (2000), 396–428.
- [126] Vincent F Hendricks, Stig Andur Pedersen, and Klaus Frovin Jørgensen, *Proof theory: History and philosophical significance*, vol. 292, Springer Science & Business Media, 2013.
- [127] Geoffrey L Herman, Lisa Kaczmarczyk, Michael C Loui, and Craig Zilles, *Proof by incomplete enumeration and other logical misconceptions*, Proceedings of the Fourth international Workshop on Computing Education Research, ACM, 2008, pp. 59–70.
- [128] James Hiebert, *Conceptual and procedural knowledge: The case of mathematics*, Routledge, 2013.
- [129] David Hilbert, *Axiomatisches denken*, Mathematische Annalen **78** (1917), no. 1, 405–415.
- [130] Keith James Holyoak and Paul Thagard, *Mental leaps: Analogy in creative thought*, MIT press, 1996.
- [131] Li Huang, Francesca Gino, and Adam D Galinsky, *The highest form of intelligence: Sarcasm increases creativity for both expressers and recipients*, Organizational Behavior and Human Decision Processes (2015).



- [132] Paul Hutchison and Andrew Elby, *Evidence of epistemological framing in survey question misinterpretation*, 2012 Physics Education Research Conference, vol. 1513, AIP Publishing, 2013, pp. 194–197.
- [133] Michael Huth and Mark Ryan, *Logic in computer science: Modelling and reasoning about systems*, Cambridge University Press, 2004.
- [134] H. Hüttel and K. Nørmark, *Experiences with web-based peer assessment of coursework*, Proceedings of the 4th International Conference on Computer Supported Education-CSEDU, 2012.
- [135] Fatma Imamoglu, Thorsten Kahnt, Christof Koch, and John-Dylan Haynes, *Changes in functional connectivity support conscious object recognition*, *Neuroimage* **63** (2012), no. 4, 1909–1917.
- [136] Daniel Jackson, *Software abstractions: logic, language, and analysis*, MIT press, 2012.
- [137] F Burton Jones, *The moore method*, *American Mathematical Monthly* (1977), 273–278.
- [138] Keith Jones and Patricio Herbst, *Proof, proving, and teacher-student interaction: Theories and contexts*, *Proof and proving in mathematics education*, Springer, 2011, pp. 261–277.
- [139] Daniel Kahneman, *Thinking, fast and slow*, Macmillan, 2011.
- [140] David Kaiser, *Drawing theories apart: The dispersion of feynman diagrams in postwar physics*, University of Chicago Press, 2009.

- [141] Immanuel Kant and Paul Guyer, *Critique of pure reason*, Cambridge University Press, 1998.
- [142] James J Kaput, Alan H Schoenfeld, and Ed Dubinsky, *Research in collegiate mathematics education iii*, vol. 4, American Mathematical Soc., 1998.
- [143] Judy Kendall, *Axial coding and the grounded theory controversy*, Western journal of nursing research **21** (1999), no. 6, 743–757.
- [144] Päivi Kinnunen and Beth Simon, *Phenomenography and grounded theory as research methods in computing education research field*, Computer Science Education **22** (2012), no. 2, 199–218.
- [145] Christine Knipping and David Reid, *Revealing structures of argumentations in classroom proving processes*, The argument of mathematics, Springer, 2013, pp. 119–146.
- [146] Maria Knobelsdorf, Christoph Kreitz, and Sebastian Böhne, *Teaching theoretical computer science using a cognitive apprenticeship approach*, Proceedings of the 45th ACM technical symposium on Computer science education, ACM, 2014, pp. 67–72.
- [147] Jeff Kramer, *Is abstraction the key to computing?*, Communications of the ACM **50** (2007), no. 4, 36–42.
- [148] Steven G Krantz, *Handbook of logic and proof techniques for computer science*, Springer Science & Business Media, 2002.
- [149] ———, *The proof is in the pudding: the changing nature of mathematical proof*, Springer Science & Business Media, 2011.

- [150] Steven George Krantz, *How to teach mathematics*, American Mathematical Soc., 1999.
- [151] Laura Krefting, *Rigor in qualitative research: The assessment of trustworthiness*, American journal of occupational therapy **45** (1991), no. 3, 214–222.
- [152] Deanna Kuhn, Jonas Langer, Lawrence Kohlberg, and Norma S Haan, *The development of formal operations in logical and moral judgment.*, Genetic Psychology Monographs (1977).
- [153] George Lakoff, *Women, fire, and dangerous things: What categories reveal about the mind*, Cambridge Univ Press, 1990.
- [154] Leslie Lamport, *How to write a proof*, The American mathematical monthly **102** (1995), no. 7, 600–608.
- [155] Sidney Landau, *1. dictionaries: The art and craft of lexicography*, 2001.
- [156] Nathaniel Lasry, Jonathan Guillemette, and Eric Mazur, *Two steps forward, one step back*, Nature Physics **10** (2014), no. 6, 402–403.
- [157] F William Lawvere, *The category of categories as a foundation for mathematics*, Proceedings of the conference on categorical algebra, Springer, 1966, pp. 1–20.
- [158] Richard Lehrer, Leona Schauble, Susan Carpenter, and David Penner, *The inter-related development of inscriptions and conceptual understanding*, Symbolizing and communicating in mathematics classrooms: Perspectives on discourse, tools, and instructional design (2000), 325–360.
- [159] Stephen Lerman, *Articulating theories of mathematics learning*, Paul Ernest, Studies in Mathematics Education Series **4** (2012), 41–49.

- [160] Uri Leron, *Structuring mathematical proofs*, American Mathematical Monthly (1983), 174–185.
- [161] Richard Lesh and Guershon Harel, *Problem solving, modeling, and local conceptual development*, Mathematical thinking and learning **5** (2003), no. 2-3, 157–189.
- [162] F-L Lin, Y-S. Lee, and J-Y. Wu Yu, *Students' understandin of proof by contradiction*, Proceedings of the 27th International Group for the Psychology of Mathematics Education Conference Held Jointly with the 25th PME-NA Conference (Honolulu, Hawaii, July 13-18, 2003). Volume 4., ERIC, 2003.
- [163] Yvonna S Lincoln and Egon G Guba, *Naturalistic inquiry*, vol. 75, Sage, 1985.
- [164] ———, *But is it rigorous? trustworthiness and authenticity in naturalistic evaluation*, New directions for program evaluation **1986** (1986), no. 30, 73–84.
- [165] Mun Ling Lo, *Variation theory and the improvement of teaching and learning*, Göteborg: Acta Universitatis Gothoburgensis, 2012.
- [166] Raymond Lister, *Concrete and other neo-piagetian forms of reasoning in the novice programmer*, Proceedings of the Thirteenth Australasian Computing Education Conference-Volume 114, Australian Computer Society, Inc., 2011, pp. 9–18.
- [167] Mun Ling Lo, Ference Marton, Ming Fai Pang, and Wing Yan Pong, *Toward a pedagogy of learning*, Classroom discourse and the space of learning (2004), 189–225.

- [168] Joanne Lobato, *The actor-oriented transfer perspective and its contributions to educational research and practice*, Educational Psychologist **47** (2012), no. 3, 232–247.
- [169] John Locke, *An essay concerning humane understanding, first edition*, Thoemmes Press, 1690.
- [170] Deborah Loewenberg et al., *Mathematical proficiency for all students: Toward a strategic research and development program in mathematics education*, Rand Corporation, 2003.
- [171] Tom Lowrie, *Problem solving in technology rich contexts: Mathematics sense making in out-of-school environments*, The Journal of Mathematical Behavior **24** (2005), no. 3, 275–286.
- [172] Ference Marton, *Phenomenography – describing conceptions of the world around us*, Instructional science **10** (1981), no. 2, 177–200.
- [173] ———, *Phenomenography: A research approach to investigating different understandings of reality*, Qualitative research in education: Focus and methods **21** (1988), 143–161.
- [174] ———, *Cognosco ergo sum. reflections on reflections*, Reflections on phenomenography: Toward a methodology **109** (1996), 163–89.
- [175] ———, *Necessary conditions of learning*, Routledge, 2014.
- [176] Ference Marton and Shirley A Booth, *Learning and awareness*, Routledge, 1997.
- [177] Ference Marton and Ming Fai Pang, *On some necessary conditions of learning*, The Journal of the Learning sciences **15** (2006), no. 2, 193–220.

- [178] ———, *Meanings are acquired from experiencing differences against a background of sameness, rather than from experiencing sameness against a background of difference: Putting a conjecture to the test by embedding it in a pedagogical tool*, Frontline Learning Research **1** (2013), no. 1, 24–41.
- [179] Ference Marton and Wing Yan Pong, *On the unit of description in phenomenography*, Higher education research & development **24** (2005), no. 4, 335–348.
- [180] Ference Marton and R Säljö, *On qualitative differences in learning outcome as a function of the learner's conception of the task*, british Journal of educational Psychology **46** (1976), no. 2, 115–127.
- [181] Ference Marton, Amy BM Tsui, Pakey PM Chik, Po Yuk Ko, and Mun Ling Lo, *Classroom discourse and the space of learning*, Routledge, 2004.
- [182] Ference Marton, David Watkins, and Catherine Tang, *Discontinuities and continuities in the experience of learning: An interview study of high-school students in hong kong*, Learning and instruction **7** (1997), no. 1, 21–48.
- [183] John Mason, *Mathematical abstraction as the result of a delicate shift of attention*, For the learning of mathematics (1989), 2–8.
- [184] Arthur Mattuck, *Introduction to analysis*, Prentice Hall, 1999.
- [185] Eric Mazur, *Confessions of a converted lecturer*, Waterloo University, [www.youtube.com/watch](http://www.youtube.com/watch) (2009).
- [186] Mercedes A McGowen and David O Tall, *Metaphor or met-before? the effects of previous experience on practice and theory of learning mathematics*, The Journal of Mathematical Behavior **29** (2010), no. 3, 169–179.

- [187] Juan Pablo Mejia-Ramos, Evan Fuller, Keith Weber, Kathryn Rhoads, and Aron Samkoff, *An assessment model for proof comprehension in undergraduate mathematics*, Educational Studies in Mathematics **79** (2012), no. 1, 3–18.
- [188] S Merriam, *What can you tell from an n ofl?: Issues of validity and reliability in qualitative research*, PAACE Journal of Lifelong Learning **4** (1995), 50–60.
- [189] Sharan B. Merriam, *Qualitative research in practice*, Jossey-Bass, 2002.
- [190] Sharan B Merriam, *Qualitative research: A guide to design and implementation*, John Wiley & Sons, 2009.
- [191] Sharan B. Merriam, *Qualitative research a guide to design and implementation*, Jossey-Bass, 2009.
- [192] Jan Meyer and Ray Land, *Threshold concepts and troublesome knowledge: Linkages to ways of thinking and practising within the disciplines*, University of Edinburgh Edinburgh, 2003.
- [193] George A Miller, *The magical number seven, plus or minus two: some limits on our capacity for processing information.*, Psychological review **63** (1956), no. 2, 81.
- [194] Devlin Montfort, Geoffrey L Herman, Shane Brown, Holly M Matusovich, Ruth A Streveler, and Olusola Adesope, *Patterns of student conceptual understanding across engineering content areas*, learning **5** (2015), 6.
- [195] Kevin C Moore, *Re-presentations and conceptual structures of what?*, Constructivist Foundations **9** (2014), no. 3, 371–373.

- [196] Robert C Moore, *Making the transition to formal proof*, Educational Studies in mathematics **27** (1994), no. 3, 249–266.
- [197] Jan Erik Moström, Jonas Boustedt, Anna Eckerdal, Robert McCartney, Kate Sanders, Lynda Thomas, and Carol Zander, *Concrete examples of abstraction as manifested in students' transformative experiences*, Proceedings of the Fourth international Workshop on Computing Education Research, ACM, 2008, pp. 125–136.
- [198] Ming Fai Pang and Wing Wah Ki, *Revisiting the idea of critical aspects*, Scandinavian Journal of Educational Research (2016), 1–14.
- [199] Michael Quinn Patton, *Qualitative evaluation and research methods .*, SAGE Publications, inc, 1990.
- [200] Danijela Petrovic, *Using small-step refinement for algorithm verification in computer science education.*, CICM Workshops, 2014.
- [201] Jean Piaget, *Experiments in contradiction*, University of Chicago Press, 1974.
- [202] ———, *Comments on mathematical education*, Contemporary Education **47** (1975), no. 1, 5.
- [203] Jean Piaget, Margaret Cook, and WW Norton, *The origins of intelligence in children*, vol. 8, International Universities Press New York, 1952.
- [204] Jean Piaget and Leonard DiLisio, *Correspondences and transformations*, 1979.
- [205] N. Pillay, *Learning difficulties experienced by students in a course on formal languages and automata theory*, ACM SIGCSE Bulletin **41** (2010), no. 4, 48–52.



- [206] Márcia Maria Fusaro Pinto, David Tall, and O Zaslavsky, *Student constructions of formal theory: giving and extracting meaning*, Proceedings of the Conference of the International Group for, vol. 100, ERIC, 1999, p. 1230.
- [207] J Poincaré, *Dernières pensées, trans. by jw bolduc as mathematics and science: Last essays*, Flammarion & Kessinger Pub., Paris & NY (1913).
- [208] George Polya, *Mathematics and plausible reasoning: vol 1: Induction and analogy in mathematics.*, Oxford University Press, 1954.
- [209] ———, *Mathematics and plausible reasoning: Patterns of plausible inference*, vol. 2, Princeton University Press, 1968.
- [210] ———, *How to solve it: A new aspect of mathematical method*, Princeton university press, 2014.
- [211] L. Porter, C. Bailey Lee, B. Simon, Q. Cutts, and D. Zingaro, *Experience report: a multi-classroom report on the value of peer instruction*, Proceedings of the 16th annual joint conference on Innovation and technology in computer science education, ACM, 2011, pp. 138–142.
- [212] L. Porter, C.B. Lee, B. Simon, and D. Zingaro, *Peer instruction: do students really learn from peer discussion in computing*, Proceedings of the Seventh International Workshop on Computing Education Research. ACM, 2011, pp. 45–52.
- [213] Laurel Puchner and Linda Markowitz, *Using an adult development model to help explain pre-service teacher resistance to learning about race*, Race Ethnicity and Education (2014), 1–20.

- [214] S Rasslan and S Vinner, *Images and definitions for the concept of increasing/decreasing function*, PME Conference, vol. 4, 1998, pp. 4–33.
- [215] Anna Reid and Peter Petocz, *Students' conceptions of statistics: A phenomenographic study*, Journal of Statistics Education **10** (2002), no. 2, 1–28.
- [216] David A Reid and Christine Knipping, *Proof in mathematics education*, Research, learning and teaching. Rotterdam: Sense Publisher (2010).
- [217] Frederick Reif, *Applying cognitive science to education*, 2008.
- [218] Lyn Richards, *Handling qualitative data: A practical guide*, Sage, 2014.
- [219] Gian-Carlo Rota, *The phenomenology of mathematical beauty*, Synthese **111** (1997), no. 2, 171–182.
- [220] Airi Rovio-Johansson and Åke Ingerman, *Continuity and development in the phenomenography and variation theory tradition*, Scandinavian Journal of Educational Research **60** (2016), no. 3, 257–271.
- [221] James Rumbaugh, Ivar Jacobson, and Grady Booch, *Unified modeling language reference manual, the*, Pearson Higher Education, 2004.
- [222] Carl-Johan Rundgren and Lena AE Tibell, *Critical features of visualizations of transport through the cell membrane – an empirical study of upper secondary and tertiary students' meaning-making of a still image and an animation*, International Journal of Science and Mathematics Education **8** (2010), no. 2, 223–246.

- [223] Ulla Runesson, *Beyond discourse and interaction. variation: a critical aspect for teaching and learning mathematics*, Cambridge journal of education **35** (2005), no. 1, 69–87.
- [224] ———, *What is it possible to learn? on variation as a necessary condition for learning*, Scandinavian journal of educational research **50** (2006), no. 4, 397–410.
- [225] Roger Säljö, *Learning in the learner's perspective. I. some common-sense conceptions*, ERIC Clearinghouse, 1979.
- [226] ———, *Minding action: Conceiving of the world versus participating in cultural practices.*, Nordisk pedagogik **14** (1994), no. 2, 71–80.
- [227] ———, *Talk as data and practicea critical look at phenomenographic inquiry and the appeal to experience*, Higher Education Research & Development **16** (1997), no. 2, 173–190.
- [228] Andreas Schäfer, Jan Holz, Thiemo Leonhardt, Ulrik Schroeder, Philipp Brauner, and Martina Ziefle, *From boring to scoring—a collaborative serious game for learning and practicing mathematical logic for computer science education*, Computer Science Education **23** (2013), no. 2, 87–111.
- [229] Alan H Schoenfeld, *Learning to think mathematically: Problem solving, metacognition, and sense making in mathematics*, Handbook of research on mathematics teaching and learning (1992), 334–370.
- [230] ———, *Reflections on a course in mathematical problem solving*, Research in collegiate mathematics education III (1998), 81–113.

- [231] Thomas A Schwandt, Yvonna S Lincoln, and Egon G Guba, *Judging interpretations: But is it rigorous? trustworthiness and authenticity in naturalistic evaluation*, New directions for evaluation **2007** (2007), no. 114, 11–25.
- [232] Andreas Schwill, *Computer science education based on*, Information Technology: Supporting change through teacher education (2016), 285.
- [233] Andreas Schwill et al., *Fundamental ideas of computer science*, Bulletin-European Association for Theoretical Computer Science **53** (1994), 274–274.
- [234] Bruce Sherin, *On conceptual metaphor and the flora and fauna of mind: Commentary on brookes and etkina; and jeppsson, haglund, and amin*, International Journal of Science Education **37** (2015), no. 5-6, 806–811.
- [235] Jonathan Sillito, *Saturate application*.
- [236] Danijela Simic, *Using small-step refinement for algorithm verification in computer science education*, The International Journal for Technology in Mathematics Education (2015).
- [237] B. Simon, M. Kohanfars, J. Lee, K. Tamayo, and Q. Cutts, *Experience report: peer instruction in introductory computing*, Proceedings of the 41st ACM technical symposium on Computer science education, ACM, 2010, pp. 341–345.
- [238] Beth Simon and Quintin Cutts, *Peer instruction: a teaching method to foster deep understanding*, Commun. ACM **55** (2012), no. 2, 27–29.
- [239] Martin A Simon, *Beyond inductive and deductive reasoning: The search for a sense of knowing*, Educational Studies in mathematics **30** (1996), no. 2, 197–210.

- [240] Martin A Simon and Ron Tzur, *Explicating the role of mathematical tasks in conceptual learning: An elaboration of the hypothetical learning trajectory*, Mathematical thinking and learning **6** (2004), no. 2, 91–104.
- [241] Michael Sipser, *Introduction to the theory of computation*, Cengage Learning, 2012.
- [242] Jan Smedslund, *Circular relation between understanding and logic*, Scandinavian Journal of Psychology **11** (1970), no. 1, 217–219.
- [243] Jennifer Christian Smith, *A sense-making approach to proof: Strategies of students in traditional and problem-based number theory courses*, The Journal of Mathematical Behavior **25** (2006), no. 1, 73–90.
- [244] T. Smith and R. McCartney, *Mathematization in teaching pumping lemmas*, 2013 IEEE Frontiers in Education Conference (FIE), Oct 2013, pp. 1671–1677.
- [245] Thérèse Smith and Robert McCartney, *Computer science students' concepts of proof by induction*, Proceedings of the 14th Koli Calling International Conference on Computing Education Research, ACM, 2014, pp. 51–60.
- [246] CD Smock and E von Glasersfeld, *Epistemology and education: The implications of radical constructivism for knowledge acquisition*, Report# 14 **28** (1974), no. 4, 635–652.
- [247] Larry Sowder, *University faculty views about mathematics majors understanding of proof*, Unpublished manuscript, San Diego State University (2004).
- [248] Barbara A Spellman and Keith J Holyoak, *Pragmatics in analogical mapping*, Cognitive psychology **31** (1996), no. 3, 307–346.

- [249] B. Steffen, F. Howar, and M. Merten, *Introduction to active automata learning from a practical perspective*, Formal Methods for Eternal Networked Software Systems (2011), 256–296.
- [250] Anselm Leonard Strauss, Juliet M Corbin, et al., *Basics of qualitative research*, vol. 15, Sage Newbury Park, CA, 1990.
- [251] Andreas J Stylianides and Deborah L Ball, *Understanding and describing mathematical knowledge for teaching: Knowledge about proof for engaging students in the activity of proving*, Journal of Mathematics Teacher Education **11** (2008), no. 4, 307–332.
- [252] Despina A Stylianou, Maria L Blanton, and Eric J Knuth, *Teaching and learning proof across the grades: A k-16 perspective*, Routledge, 2010.
- [253] Lennart Svensson, *Theoretical foundations of phenomenography*, Higher Education Research & Development **16** (1997), no. 2, 159–171.
- [254] D. Tall, O. Yevdokimov, B. Koichu, W. Whiteley, M. Kondratieva, and Y.H. Cheng, *Cognitive development of proof*, Proof and Proving in Mathematics Education (2012), 13–49.
- [255] David Tall, *The nature of mathematical proof*, Mathematics Teaching **127** (1989), 28–32.
- [256] ———, *The psychology of advanced mathematical thinking*, Advanced mathematical thinking, Springer, 1991, pp. 3–21.
- [257] David Tall, *The cognitive development of proof: is mathematical proof for all or for some.*, (1998).

- [258] David Tall, *The transition to formal thinking in mathematics*, Mathematics Education Research Journal **20** (2008), no. 2, 5–24.
- [259] ———, *How humans learn to think mathematically: exploring the three worlds of mathematics*, Cambridge University Press, 2013.
- [260] David Tall, Eddie Gray, Maselan Bin Ali, Lillie Crowley, Phil DeMarois, Mercedes McGowen, Demetra Pitta, Marcia Pinto, Michael Thomas, and Yudariah Yusof, *Symbols and the bifurcation between procedural and conceptual thinking*, Canadian Journal of Math, Science & Technology Education **1** (2001), no. 1, 81–104.
- [261] David Tall and Juan Pablo Mejia-Ramos, *The long-term cognitive development of reasoning and proof*, Springer, 2010.
- [262] David Tall and Shlomo Vinner, *Concept image and concept definition in mathematics with particular reference to limits and continuity*, Educational studies in mathematics **12** (1981), no. 2, 151–169.
- [263] Denisse R Thompson, *Learning and teaching indirect proof*, The Mathematics Teacher **89** (1996), no. 6, 474.
- [264] Patrick W Thompson, *Students, functions, and the undergraduate curriculum*, Research in collegiate mathematics education **1** (1994), 21–44.
- [265] Neena Thota, *Programming course design: Phenomenographic approach to learning and teaching*, Teaching and Learning in Computing and Engineering (LaTiCE), 2014 International Conference on, IEEE, 2014, pp. 125–132.

- [266] Michael Thuné and Anna Eckerdal, *Variation theory applied to students conceptions of computer programming*, European Journal of Engineering Education **34** (2009), no. 4, 339–347.
- [267] Amos Tversky and Daniel Kahneman, *The framing of decisions and the psychology of choice*, Science **211** (1981), no. 4481, 453–458.
- [268] Catherine Ulrich, *Issues around reflective abstraction in mathematics education*, Constructivist Foundations **9** (2014), no. 3, 370–371.
- [269] Lars Ulriksen, Henriette Tolstrup Holmegaard, and Lene Møller Madsen, *Weaving a bridge of sense: students' narrative constructions as a lens for understanding students' coping with the gap between expectancies and experiences when entering higher education*, European Educational Research Journal **12** (2013), no. 3, 310–319.
- [270] Leslie G Valiant, *Circuits of the mind*, Oxford University Press, 2000.
- [271] JP Van Haneghan, Linda Barron, MF Young, SM Williams, NJ Vye, and JD Bransford, *The jasper series: An experiment with new ways to enhance mathematical thinking*, Enhancing thinking skills in the sciences and mathematics (1992), 15–38.
- [272] Pierre M Van Hiele, *Structure and insight: A theory of mathematics education*, 1986.
- [273] EJ Van Someren, *Slow brain oscillations of sleep, resting state and vigilance*, vol. 193, Elsevier, 2011.
- [274] Daniel J. Velleman, *How to prove it*, Cambridge University Press, 1998.



- [275] G Vico, *De antiquissima italorum sapientia*.(1710). naples, italy: Stamperia de' classici latini. von glasersfeld, e. 1988. environment and communication, ICME-6, Budapest, Hungary (1858).
- [276] Shlomo Vinner, *The concept of exponentiation at the undergraduate level and the definitional approach*, Educational Studies in Mathematics **8** (1977), no. 1, 17–26.
- [277] Ernst Von Glasersfeld, *An introduction to radical constructivism*, The invented reality (1984), 17–40.
- [278] ———, *Learning as a constructive activity*, In C. Janvier (Ed.), Problems of representation in the, Citeseer, 1987.
- [279] ———, *Radical constructivism: A way of knowing and learning. studies in mathematics education series: 6.*, ERIC, 1995.
- [280] ———, *Radical constructivism*, Routledge, 2013.
- [281] Lev Semenovič Vygotskij, Robert W Rieber, and Aaron S Carton, *The collected works of LS Vygotsky. vol. 1, Problems of general psychology including the volume Thinking and speech*, Plenum P., 1987.
- [282] Lev Semenovič Vygotsky, *Language and thought*, Massachusetts Institute of Technology Press, Ontario, Canada (1962).
- [283] ———, *Mind in society*, (1978).
- [284] ———, *Zone of proximal development*, Mind in society: The development of higher psychological processes (1987), 52–91.

- [285] weber, <http://www.maa.org/programs/faculty-and-departments/curriculum-department-guidelines-recommendations/teaching-and-learning/research-sampler-8-students-difficulties-with-proof#sthash.SMs7qEx9.dpuf>.
- [286] Keith Weber, *Student difficulty in constructing proofs: The need for strategic knowledge*, Educational Studies in Mathematics **48** (2001), no. 1, 101–119.
- [287] ———, *Beyond proving and explaining: Proofs that justify the use of definitions and axiomatic structures and proofs that illustrate technique*, For the Learning of Mathematics (2002), 14–17.
- [288] ———, *The role of instrumental and relational understanding in proofs about group isomorphisms*, Proceedings from the 2nd International Conference for the Teaching of Mathematics, 2002.
- [289] ———, *Research sampler 8: students difficulties with proof*, The Mathematical Association of America: Online. Mathematical Association of America. Retrieved on October 1 (2003), 2009.
- [290] ———, *Students difficulties with proof*, Retrieved September 14 (2003), 2005.
- [291] ———, *A framework for describing the processes that undergraduates use to construct proofs*, Proceedings of the 28th Conference of the International Group for the Psychology of Mathematics Education, vol. 4, 2004, pp. 425–432.
- [292] ———, *Traditional instruction in advanced mathematics courses: A case study of one professor's lectures and proofs in an introductory real analysis course*, The Journal of Mathematical Behavior **23** (2004), no. 2, 115–133.

- [293] ———, *Problem-solving, proving, and learning: The relationship between problem-solving processes and learning opportunities in the activity of proof construction*, The Journal of Mathematical Behavior **24** (2005), no. 3, 351–360.
- [294] ———, *How mathematicians determine if an argument is a valid proof*, Journal for Research in Mathematics Education (2008), 431–459.
- [295] Keith Weber and Lara Alcock, *Semantic and syntactic proof productions*, Educational Studies in Mathematics **56** (2004), no. 2-3, 209–234.
- [296] ———, *Proof in advanced mathematics classes: Semantic and syntactic reasoning in the representation system of proof*, (2009).
- [297] Keith Weber and Juan Pablo Mejia-Ramos, *Why and how mathematicians read proofs: An exploratory study*, Educational Studies in Mathematics **76** (2011), no. 3, 329–344.
- [298] P.K. Weidmann, *Science education perceptions of teaching and learning automata theory in a college-level computer science course a thesis*, Ph.D. thesis, The University of Texas at Austin, 2003.
- [299] Etienne Wenger, *Communities of practice: Learning, meaning, and identity*, Cambridge university press, 1999.
- [300] Alfred North Whitehead, *The aims of education*, Daedalus **88** (1959), no. 1, 192–205.
- [301] Ludwig Wittgenstein, RG Bosanquet, N Malcolm, R Hrees, Y Symthies, and Cora Diamond, *Wittgenstein's lectures on the foundations of mathematics, cambridge, 1939*, Cornell University Press, 1976.

- [302] Harry F Wolcott, *Transforming qualitative data: Description, analysis, and interpretation*, Sage, 1994.
- [303] C.C. Xing, *A graphical framework for assisting proofs*, Journal of Computing Sciences in Colleges **25** (2010), no. 5, 48–57.
- [304] Kai-Lin Yang and Fou-Lai Lin, *A model of reading comprehension of geometry proof*, Educational Studies in Mathematics **67** (2008), no. 1, 59–76.
- [305] Andreas Zendler and Christian Spannagel, *Empirical foundation of central concepts for computer science education*, Journal on Educational Resources in Computing (JERIC) **8** (2008), no. 2, 6.

## Appendices

# Appendix A

## Brochure for Critical Aspects Integrated into Curriculum

The critical features / aspects (henceforth, features) have been collected from chapter 4. In some cases, multiple research questions revealed similar critical features. When this occurred, the critical features were combined. The critical features have been reordered for the convenience of exposition.

1. mathematical statements about algorithm properties (RQ 3)
2. provability of desired property determines whether algorithm satisfies requirements (RQ 3)
3. analogies, abstraction, generalization (RQ 2)
4. step of abstraction (RQ 1)
5. use of special tokens (RQ 1)
6. special purpose language (RQ 1)

7. limited role for emotion, not anxiety (RQ 2)
8. Ascertaining and convincing are derived from logical, deductive reasoning. (RQ 10)
9. mathematical definitions are important (RQ 2)
10. Justifications for logical deductive steps, warrants, can depend on definitions. (RQ 10)
11. make the truth of the consequence evident (RQ 1)
12. multiple representations and preferences among them (RQ 2)
13. consider using diagrams (RQ 2)
14. consider visualizing with dynamics (RQ 2)
15. commonly used combinations of statements (RQ 1)
16. proofs can have component parts (RQ 2)
17. criteria for composing combinations of statements (RQ 1)
18. compare problem statements look for similarity (RQ 2)
19. re-use of proof techniques (RQ 3)
20. For clues to beginning a proof, examine the structure of the claim. (RQ 4)
21. Consider rearranging a statement's parts. (RQ 7)
22. Sequences of statements perform transformations (RQ 8)

- 23. View transformation from premise to claim as a trajectory. (RQ 4)
- 24. Patterns contain some generally useful sequences. (RQ 4,8)
- 25. Applying patterns is good, but be aware of the reasoning. (RQ 9,10)
- 26. Reasoning has been captured in patterns, reasoning should always be applied.  
(RQ 8,9)
- 27. Use rules of inference to do this reasoning (RQ 7)
- 28. A subset of the permissible transformations makes progress toward the goal  
statement. (RQ 9)

## A.1 Motivation

Given a context, i.e., a problem, in which algorithms are to be applied, not all algorithms are equally good solutions. Mathematics provides a language for describing some important properties of individual algorithms, by which one may be compared to another, for suitability to a given application context. When an algorithm is created, it is useful to characterize it according to their resource utilization, as it increases with problem instance size, for which we use big-O (actually big-Omicron) notation. These characterizations are guaranteed by using proof.

## A.2 Step of Abstraction

Here we can use the students' well-known desire to work from examples by beginning with an example. The example should be something familiar to at least almost all of



the students. Next, we change properties of this example one by one, to show that the example gets changed into a non-example. This was done earlier (see section 2.1.3). Then, we have a second example, coordinated with the first so that an analogy can be drawn. Some class participation should be used to be sure the students are following this. Perhaps they can be cajoled into supplying a third example, and describing the shared relationships among their parts that are shared among the three examples. Once the relationship has been well identified, the abstraction process, which takes note of the shared relationships, and discounts the specifics from the interrelated components, has been illustrated.

Sipser[241] and Polya[209] have provided problems that we can use in this way. The problem given by Polya is to find the error in a proof-attempt of a claim that all women are blue-eyed blonds. The problem given by Sipser is to find the error in a proof-attempt of a claim that all horses are of the same color. We can discuss the analogy between the two problems. We can help the students discern the analogy by noting that it is lost when we change the claim about a unique value for the coloration property. These are designed to help students understand the reasoning in proof by mathematic induction. We can introduce a mathematical formulation. We can imagine a single woman, who is a blue-eyed blond. Already we have an erroneous base case, because a single example cannot prove a universal claim. Likewise, we can imagine a single horse, and claim it is of a single color, which is readily proven false by horses such as the Appaloosa.

We can imagine a set of women, or horses, and there are  $k + 1$  of them. From this set, we can take one out, and we can claim, because it is the premise of the induction step, that the set of  $k$  individuals has the coloring under discussion, which we might denote  $c_1$ . We can claim that, once we put that individual back into the set and take

another individual out, we can likewise assume a set of  $k$  of one coloration, denoted  $c_2$ . We can claim that, due to individuals present in both of the sets of  $k$  individuals, that  $c_1 = c_2$ . We can also hope that the students will notice on their own that there is no element in both subsets, when  $k = 1$ ; if they do not, we can lead them to this discovery.

The point here is that the notation works just as well for people or horses, it serves the purpose of abstraction of the problems, so that more such problems are solved at once.

## A.3 Language of Abstraction

Here, exercise mathematical formulation. We want students to see that they can abstract from example situations that they understand, to a mathematical notation, enabling them to work with concision. For example, we might work with implication. We can present the idea of implication by contrasting it with if-then statements in code.

```
if(locationOfHit == locationOfShip){
  remove(ship);}
```

We can point out that this code removes the ship when the location of the hit matches the location of the ship, and that the code does not remove the ship when the location of the hit does not match the location of the ship. We can denote these  $h = s$  and  $r$ , we can formulate  $(h = s) \rightarrow r$  and  $r \rightarrow (h = s)$ . We can point out  $(h = s) \iff r$ . We can remark upon the different usage of “if” in programming and in mathematical formulation.

To help students discern the difference between “if and only if” from “if”, it may

help to ask them to formulate several natural language descriptions, of situations of “if and only if”, and of situations where simply “if” is appropriate, into symbolic notation.

## A.4 Logical Deduction, from Definitions, without Anxiety

We make clear that the reason we believe things, the only kind of argument that is accepted, is logical deduction. Qualitative study of our student population has shown that not all students accept that becoming convinced about claims of resource utilization is achieved specifically by a logical deduction. Measurements are made and reported, so we should allocate some time to the thoroughness of coverage of a proof compared with examples. Here, we combat anxiety with practice, which develops confidence. Here, we explicitly use definitions to make logical deductions. We develop sequences of logical deductions to show transformations.

Our students encounter the definition of a finite automaton in theory of computation. We can prove (using the pigeon-hole principle) that finite automata that accept strings of infinite length must reuse at least one state. We can show that this proof depends upon the definition of finite automaton, and the definition of a string’s having been accepted.

## A.5 Representations

Here we apply multiple representations, and practice moving from one to another. These include mathematical formulation, picking up from abstraction, and also di-

agrams, and also code. We use animation, including animated diagrams, including UML executions.

Finite automata are a good subject for practice with multiple representations, because we routinely use both graphs and mathematical formulation. Moreover, the execution of a finite automaton has dynamics and can be illustrated with animation.

We can represent the automaton, and its execution in diagrams, in mathematical formulation, and in code.

## A.6 Transformation Trajectories

Here we study sequences of logical deductions, and component parts of proofs.

We can vary the sequence of statements in a proof. This gives the opportunity to discuss warrants. We do not necessarily expect to obtain a proof, if we start with a proof and change the sequence of statements.

We can recall that mathematical formulation allows us to prove more than one concrete situation at a time. We can look at a proof that proves one such situation, and propose an analogous situation, and recall that the proof also serves in the analogous situation. This suggests the idea of proof technique, which can be seen as a pattern in the sequence of statements in a proof.

Proof techniques that can be applied as patterns, with attention to the reasoning, including the warrants, being used.

## A.7 Recognizing When Techniques Apply

Here we practice applying the techniques, with emphasis on how to match the technique to the problem. We also consider modifying the problem to fit an idea of when a technique applies.

As we have previously shown analogous problems being solved by a single proof, we can expand upon that to finding a boundary between problems that are similar enough, and those that are different. This is a good example of variation, to help the students discern what it is that allows a technique to be applicable.

Dynamic programming and greedy programming are fine subjects for considering the context in which an algorithm applies, and may also yield situations when proof techniques (such as proof of correctness) apply.

## A.8 Proofs from Algorithms, from Theory of Computation

Examine some proofs, such as the dynamic array time complexity, and the non-deterministic pushdown automaton language class matching the context free grammar language class, in terms developed previously.

## Appendix B

# Examination of Multiple Representations

Several figures have been reprinted with permission.

©Mathematical Association of America, 2016. All rights reserved.

## B.1

$$\pi_1 = \beta_1, \pi_2 = (1 - \beta_1) \beta_2, \dots, \pi_c = \beta_c \prod_{j=1}^{c-1} (1 - \beta_j), \dots$$

FIGURE B.1.1: Equation (1)

$$\sum_{i=0}^k \frac{1}{2^i}$$

FIGURE B.1.2: Equation (2)

```

1: num = rand()*10;
2: stock = 1;
3: amount = 0;
4: frac = 1/2;
   getRandomFraction()
5: while num > 0 do
6:   stock = stock*frac;
7:   amount = stock+amount;
8:   num = num-1;
9: end while
   return amount;
}

```

ALGORITHM 1: Code

```

1: num = rand()*10;
2: stock = 1;
3: amount = 0;
4: frac = 1/2;
   getRandomFraction()
5: frac=rand();
6: amount = stock*frac;
7: stock = stock - amount;
   return amount;

```

ALGORITHM 2: Code

Circle your choice:

Equation (1) goes with Algorithm 1? Yes, No

Equation (1) goes with Algorithm 2? Yes, No

Equation (2) goes with Algorithm 1? Yes, No

Equation (2) goes with Algorithm 2? Yes, No



## B.2

Show which equation goes with which diagram, if they can be matched.

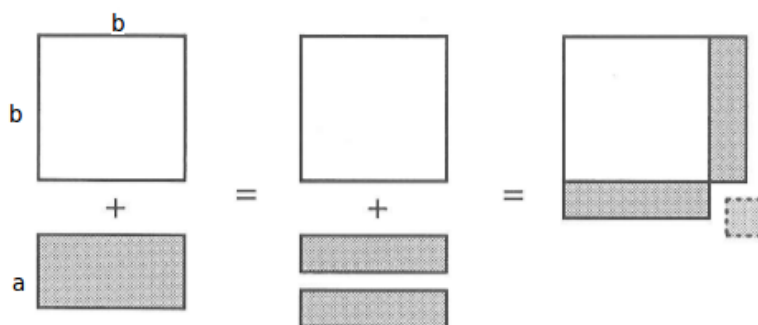


FIGURE B.2.1: Three part diagram

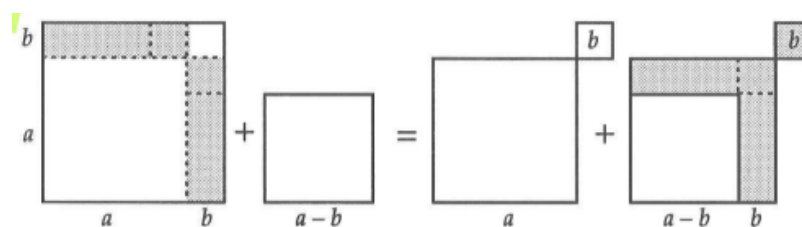


FIGURE B.2.2: Four part diagram

Equation 1.  $b^2 + ab = (b + a/2)^2 - (a/2)^2$

Equation 2.  $(a + b)^2 + (a - b)^2 = 2(a^2 + b^2)$

Circle your choice:

Equation 1 goes with Figure B.2.1? Yes, No

Equation 2 goes with Figure B.2.1? Yes, No

Equation 1 goes with Figure B.2.2? Yes, No

Equation 2 goes with Figure B.2.2? Yes, No

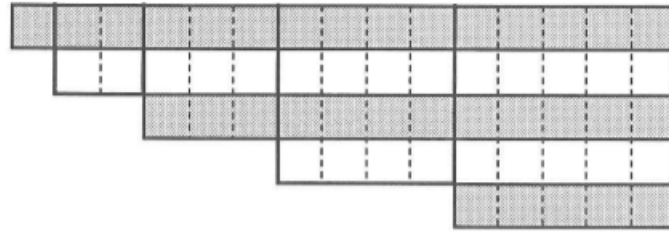


FIGURE B.3.1: First

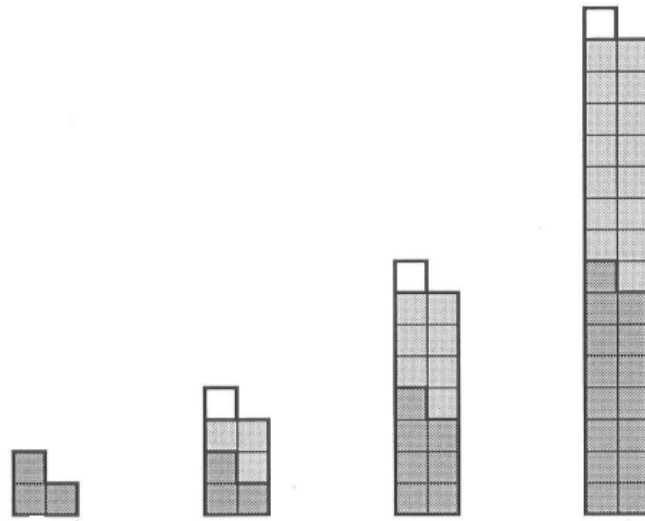


FIGURE B.3.2: Second

## B.3

```

1: amount = 1;
   getAmount(n){
2: amount = 0;
3: for  $i = 0$  to  $n$  do
4:
5:   for  $j = 0$  to  $i$  do
6:     amount = amount +j;
7:   end for
8: end for
   return amount;}

```

ALGORITHM 3: Code

```

1: amount = 1;
   getAmount(n){
2: if  $n = 0$  then
3:   return (1)
4: else
5:   return(2*getAmount(n-1)+1)
6: end if
   }

```

ALGORITHM 4

Circle your choice:

Figure B.3.1 goes with Algorithm 3? Yes, No

Figure B.3.1 goes with Algorithm 4? Yes, No

Figure B.3.2 goes with Algorithm 3? Yes, No

Figure B.3.2 goes with Algorithm 4? Yes, No

**B.4**

Describe in text what this symbolic statement means:

$$\sum_{k \in \mathbb{N}}^{\infty} (2k + 1)$$

## B.5

Describe in text, what Algorithm 5 is doing:

Challenge question: What is the significance of the process described by Algorithm 5?

```

1: done = false;
2: m = 100; an arbitrary bound
3: while !done do
4:   n = 2;
5:   while  $n \geq 2$  do
6:     n=n+1;
7:     a = 0;
8:     b = 0;
9:     c = 0;
10:    while  $0 \leq a \&a < m$  do
11:      a=a+1;
12:      while  $0 \leq b \&b < a$  do
13:        b=b+1;
14:        while  $0 \leq c \&c < (a + b)$  do
15:          c=c+1;
16:          if  $a^n + b^n = c^n$  then
17:            done = true;
18:          end if
19:        end while
20:        c = 0;
21:      end while
22:      b = 0;
23:    end while
24:    a = 0;
25:  end while
26: end while

```

## B.6

Applying symbolic representation to figure

Label the parts of these figures with mathematical symbols:

Challenge question: What is the significance of these figures taken together?

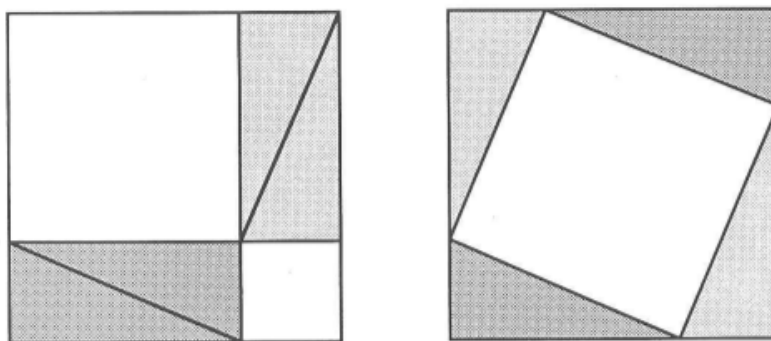


FIGURE B.6.1: Puzzle pieces

## B.7

Applying figure to symbolic representation

Draw a figure that expresses:

Through any three points that are not collinear, two can be used to identify a line, and the third can be used, combined with that line, to identify a line parallel to the first line.



## B.8

Applying pseudocode to figure

Write pseudocode (e.g., as has been seen earlier in these questions) to count the little squares, according to the method suggested by the shading in the figure:

Hint: the bottom row could be row 1.

$n$  and  $k$  could be parameters.

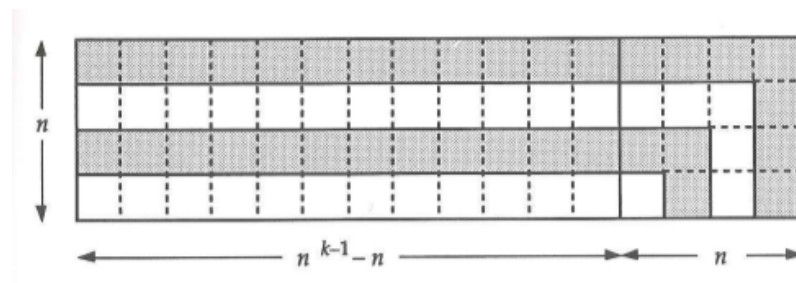


FIGURE B.8.1: Counting by expanding boundary

## B.9

Applying pseudocode to symbolic representation

Write pseudocode for the calculation below:

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

Hint:  $\binom{n}{k}$  means, from a set of size  $n$  choose (without replacement)  $k$  elements.

For example, from a list of friends of size  $n$ , choose a list of party guests of size  $k$ .

(Without replacement means, you do not send two invitations to the same friend.)

Please say whether the concrete example (friends, guest list) aided your thinking.

## B.10

Synthesis: word problems to figures

Draw a figure for this word problem:

Pick an angle between 0 and  $\pi/2$  radians, call it  $\theta$ .

A right triangle can be drawn, the height of which is 1 plus the tangent of  $\theta$ , the base of which is 1 plus the cotangent of  $\theta$ . The angle  $\theta$  is adjacent to the base. It will be the case that the hypotenuse of this triangle is the cosecant of  $\theta$  plus the secant of  $\theta$ .

**B.11**

Synthesis: word problems to symbolic representation

Express in symbols this word problem:

The number of moves in a game of size  $n$  is given by twice the number of moves in a game of size  $(n-1)$ , plus one more move.

**B.12**

Synthesis: word problems to pseudocode

Write pseudocode for this word problem:

Two trains, initially 40 miles apart on the same track, going opposite directions, are getting closer to each other at the rate of 40 miles per hour. An insect flies from one train to the other and back, at a constant speed of  $v$ , repeatedly. How far does the insect fly, before the trains collide?

## Appendix C

### Interview Protocol

1. When you took course “x”, as far as you remember, was there any discussion about proofs?
2. Do you remember whether you had to furnish proofs of any kind?
3. What kinds?
4. Do you remember any example proofs?
5. Why do you think course “x” treats proofs?
6. Why do you think course “x” uses the examples it does?
7. Have you noticed any use that you make, of the material related to proofs, from a previous course supporting work in a later course?
8. If you had to choose mathematical symbolic formulation, or software code, *to articulate carefully* a concept you were trying to express, would you find one easier than another? Which?

9. Does it ever arise that you wish to convince yourself, or someone else, of a technical point?
10. If so, what techniques do you employ?