Techniques of

PROLOG PROGRAMMING

with implementation of
logical negation and quantified goals

T. Van Le, Ph.D.
University of Canberra
# Contents

## Preface

Chapter 1  Introduction to Prolog

1.1 What is Prolog? 1  
1.2 Prolog versus conventional programming 2  
1.3 Prolog programs 3  
1.4 Queries 7  
1.5 Working with Prolog 8  
1.6 Predefined functions and predicates 11  
1.7 Summary 13  
Solved problems 14  
Supplementary problems 17  

Chapter 2  Declarative Prolog programming

2.1 Writing a declarative Prolog program 20  
2.2 Recursive programming 23  
2.3 List processing 30  
2.4 List sorting 38  
2.5 Data structures 41  
2.6 Databases 44  
  2.6.1 Database as a set of records 45  
  2.6.2 Database as a set of attributes 46  
  2.6.3 Database as a collection of record lists 47  
  2.6.4 Database as a collection of sorted binary trees 48  
2.7 Fundamental techniques of declarative Prolog programming 50  
Solved problems 52  
Supplementary problems 68  

Chapter 3  Procedural Prolog programming

3.1 Procedural reading of Prolog programs 73  
3.2 Writing a procedural Prolog program 76  
3.3 Program execution: unification and resolution 80  

xiii
Chapter 4  Control and side-effect features of Prolog

4.1 Control of program execution 107
4.2 Prevention of backtracking: the cut predicate 108
  4.2.1 Correct use of cut 110
  4.2.2 Incorrect use of cut 112
  4.2.3 The once predicate 115
4.3 Forcing of backtracking: the fail predicate 119
4.4 Negation-as-failure: the not predicate 122
  4.4.1 Correct use of not 124
  4.4.2 Incorrect use of not 125
4.5 Repetition: the repeat predicate 126
4.6 Input-output predicates 128
4.7 Environment-manipulating predicates 132
  4.7.1 File-handling predicates 132
  4.7.2 Database-manipulating predicates 134
  4.7.3 System predicates 136
Solved problems 141
Supplementary problems 153

Chapter 5  Development of Prolog programs

5.1 Program development 155
5.2 Program style 157
5.3 Programming techniques 166
  5.3.1 Generate-and-test 166
  5.3.2 Divide-and-conquer 169
5.4 User interaction 172
  5.4.1 Menu-driven interaction 172
## Chapter 6  Advanced programming techniques and data structures

6.1 Nondeterministic programming 196
   6.1.1 Nondeterministic thinking 197
   6.1.2 Nondeterministic finite automata 198
6.2 Second-order programming 203
   6.2.1 Collection of data: the set-predicates 203
   6.2.2 Conversion of data: the =.. predicate 209
   6.2.3 Inspection of data: the functor and arg predicates 212
   6.2.4 Classification of data: type-predicates 213
   6.2.5 Definition of new functors: the op predicate 214
6.3 Representation of queues, frames, and arrays 219
   6.3.1 Queues 219
   6.3.2 Frames 223
   6.3.3 Arrays 225
6.4 Advanced tree structures 230
   6.4.1 AVL-trees 230
   6.4.2 B-trees 239
6.5 An example of program validation 248
Solved problems 258
Supplementary problems 269

## Chapter 7  Search techniques

7.1 Search: a problem-solving technique 272
   7.1.1 Representation of states 273
   7.1.2 Change of states 274
7.2 Search strategies 274
7.3 Depth-first search 281
7.4 Breadth-first search 285
Chapter 8  Meta-programming in Prolog

8.1 Meta-programs 341
8.2 Meta-programming techniques 343
8.2.1 Inspection of formulas 345
8.2.2 Modification of formulas 350
8.2.3 Generation of new formulas 351
8.2.4 Execution of goals 352
8.3 The problem of the meta-predicate not 355
8.4 LnProlog's negation evaluator 357
8.5 LnProlog's meta-preprocessor 362
8.6 Representation of quantified goals in LnProlog 368
8.7 Simulation of a Prolog interpreter for debugging 371
8.8 A Prolog debugger 381
Solved problems 387
Supplementary problems 395

Chapter 9  Building expert systems in Prolog

9.1 What is an expert system? 397
9.2 Components of an expert system 398
9.3 Building an expert system 400
9.4 Representation of uncertainty 404
9.5 Generation of explanations 407
9.6 A meta-interpreter for expert systems 417
9.7 Expert system shells 422
## CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.8</td>
<td>ESSLN: An expert system shell with logical negation</td>
<td>423</td>
</tr>
<tr>
<td>9.9</td>
<td>Quantified goals in ESSLN</td>
<td>425</td>
</tr>
<tr>
<td>9.10</td>
<td>ESSLN's user interface</td>
<td>427</td>
</tr>
<tr>
<td>9.11</td>
<td>Subtypes in ESSLN</td>
<td>457</td>
</tr>
<tr>
<td></td>
<td>Solved problems</td>
<td>462</td>
</tr>
<tr>
<td></td>
<td>Supplementary problems</td>
<td>474</td>
</tr>
</tbody>
</table>

### Chapter 10  Natural language processing in Prolog

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1</td>
<td>Natural language processing</td>
<td>476</td>
</tr>
<tr>
<td>10.2</td>
<td>Context-free grammars</td>
<td>477</td>
</tr>
<tr>
<td>10.3</td>
<td>Definite clause grammars</td>
<td>480</td>
</tr>
<tr>
<td>10.4</td>
<td>Logical representation of sentences</td>
<td>484</td>
</tr>
<tr>
<td>10.5</td>
<td>Interpretation of logical formulas</td>
<td>488</td>
</tr>
<tr>
<td>10.6</td>
<td>A database query-answering system</td>
<td>493</td>
</tr>
<tr>
<td></td>
<td>Solved problems</td>
<td>499</td>
</tr>
<tr>
<td></td>
<td>Supplementary problems</td>
<td>510</td>
</tr>
</tbody>
</table>

### Chapter 11  System simulation in Prolog

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1</td>
<td>What is simulation?</td>
<td>512</td>
</tr>
<tr>
<td>11.2</td>
<td>Object-oriented simulation models</td>
<td>513</td>
</tr>
<tr>
<td>11.3</td>
<td>Prolog simulation models</td>
<td>515</td>
</tr>
<tr>
<td>11.4</td>
<td>Realisation of nondeterminism</td>
<td>519</td>
</tr>
<tr>
<td>11.5</td>
<td>Multiobject requirement</td>
<td>525</td>
</tr>
<tr>
<td>11.6</td>
<td>Implementation of rendez-vous</td>
<td>530</td>
</tr>
<tr>
<td>11.7</td>
<td>POSS: A Prolog-based object-oriented simulation system</td>
<td>534</td>
</tr>
<tr>
<td>11.8</td>
<td>Simulation programming with POSS</td>
<td>539</td>
</tr>
<tr>
<td></td>
<td>Solved problems</td>
<td>543</td>
</tr>
<tr>
<td></td>
<td>Supplementary problems</td>
<td>557</td>
</tr>
</tbody>
</table>

### Appendices

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Theory of unification and resolution</td>
<td>561</td>
</tr>
<tr>
<td>A.1</td>
<td>Unification</td>
<td>561</td>
</tr>
<tr>
<td>A.2</td>
<td>SLD-resolution</td>
<td>563</td>
</tr>
<tr>
<td>B</td>
<td>Pure-Prolog interpreter</td>
<td>567</td>
</tr>
</tbody>
</table>
C  Prolog predefined functions and predicates       569
D  The ASCII character set                           576
E  Prolog operators precedence                      577
F  List of defined procedures                       578
G  The diskettes                                     582

Bibliography                                        585
Index                                               594