

# Table of Contents

|                                                    |    |
|----------------------------------------------------|----|
| <b>1 Introduction</b>                              | 1  |
| 1.1 The Role and Origins of Programming            | 1  |
| 1.2 The Evolution of Programming Languages         | 1  |
| 1.3 The “Module” Concept                           | 4  |
| 1.4 Lexical, Syntactic and Semantic Analysis       | 4  |
| 1.5 The MODULA-2 Vocabulary                        | 5  |
| 1.6 Syntactic Diagrams                             | 6  |
| 1.7 Structure of a MODULA-2 Program                | 7  |
| 1.8 A First Program: How to Write a String         | 9  |
| 1.9 Some Basic Considerations for Writing Programs | 9  |
| Summary                                            | 10 |
| <b>2 Numbers</b>                                   | 11 |
| 2.1 Integer and Real Numbers                       | 11 |
| 2.2 Operators                                      | 11 |
| 2.3 The MathLib0 Module                            | 13 |
| 2.4 Arithmetic Expressions                         | 15 |
| 2.5 How to Write Numbers                           | 16 |
| 2.6 Program Examples                               | 17 |
| Summary                                            | 18 |
| <b>3 Constants, Variables and Data Types</b>       | 19 |
| 3.1 Constants                                      | 19 |
| 3.2 Variables and Data Types                       | 20 |
| 3.3 INTEGER, CARDINAL and REAL Types               | 21 |
| 3.4 The Assignment Statement                       | 22 |
| 3.5 Type Conversion                                | 23 |
| 3.6 An Example: Exchange of Two Values             | 25 |
| Summary                                            | 26 |
| <b>4 The Boolean and CHAR Types</b>                | 27 |
| 4.1 Boolean Constants and Operations               | 27 |
| 4.2 Boolean Expressions                            | 27 |
| 4.3 Relational Operators                           | 30 |
| 4.4 Two Program Examples                           | 33 |
| 4.5 The Type CHAR                                  | 33 |
| Summary                                            | 39 |

|                                                   |            |
|---------------------------------------------------|------------|
| <b>5 Basic Control Statements</b> . . . . .       | <b>40</b>  |
| 5.1 The IF Statement . . . . .                    | 41         |
| 5.2 The WHILE Statement . . . . .                 | 44         |
| 5.3 The REPEAT Statement . . . . .                | 48         |
| 5.4 Program Examples . . . . .                    | 51         |
| Summary . . . . .                                 | 53         |
| <b>6 The Input/Output Modules</b> . . . . .       | <b>54</b>  |
| 6.1 The Role of the Read Operation . . . . .      | 54         |
| 6.2 Text Input . . . . .                          | 56         |
| 6.3 The InOut and RealInOut Modules . . . . .     | 59         |
| 6.4 Program Examples . . . . .                    | 63         |
| Summary . . . . .                                 | 63         |
| <b>7 Enumeration and Subrange Types</b> . . . . . | <b>64</b>  |
| 7.1 Enumeration Types . . . . .                   | 64         |
| 7.2 Order in Enumeration Types . . . . .          | 66         |
| 7.3 Subrange Types . . . . .                      | 68         |
| 7.4 A Program Example . . . . .                   | 71         |
| Summary . . . . .                                 | 71         |
| <b>8 Other Control Statements</b> . . . . .       | <b>72</b>  |
| 8.1 The LOOP Statement . . . . .                  | 72         |
| 8.2 The FOR Statement . . . . .                   | 75         |
| 8.3 The CASE Statement . . . . .                  | 79         |
| 8.4 Program Examples . . . . .                    | 84         |
| Summary . . . . .                                 | 85         |
| <b>9 Array Types</b> . . . . .                    | <b>86</b>  |
| 9.1 One-Dimensional Arrays . . . . .              | 86         |
| 9.2 Program Examples . . . . .                    | 91         |
| 9.3 String Types . . . . .                        | 92         |
| 9.4 Program Examples: Palindromes . . . . .       | 95         |
| 9.5 Multi-Dimensional Arrays . . . . .            | 95         |
| 9.6 Program Examples . . . . .                    | 101        |
| Summary . . . . .                                 | 101        |
| <b>10 Procedures and Locality</b> . . . . .       | <b>102</b> |
| 10.1 The Procedure Concept . . . . .              | 102        |
| 10.2 Locality . . . . .                           | 104        |
| 10.3 Parameters . . . . .                         | 113        |
| 10.4 A Program Example . . . . .                  | 121        |
| Summary . . . . .                                 | 121        |

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| <b>11 Transmission of Parameters and Scope</b> . . . . .                  | 122 |
| 11.1 Variable and Value Parameters . . . . .                              | 122 |
| 11.2 Open Array Parameters . . . . .                                      | 125 |
| 11.3 Rules of Scope . . . . .                                             | 127 |
| Summary . . . . .                                                         | 131 |
| <b>12 Function Procedures and Procedure Types</b> . . . . .               | 132 |
| 12.1 The Function Procedure Concept . . . . .                             | 132 |
| 12.2 A Complete Example . . . . .                                         | 137 |
| 12.3 Procedure Types . . . . .                                            | 137 |
| 12.4 A Complete Example . . . . .                                         | 142 |
| Summary . . . . .                                                         | 144 |
| <b>13 Recursion</b> . . . . .                                             | 145 |
| 13.1 The Concept of Recursion . . . . .                                   | 145 |
| 13.2 The Towers of Hanoi . . . . .                                        | 147 |
| 13.3 Quicksort and Permutations . . . . .                                 | 150 |
| 13.4 Indirect Recursion and Recursive Descent . . . . .                   | 151 |
| Summary . . . . .                                                         | 158 |
| <b>14 Modules</b> . . . . .                                               | 159 |
| 14.1 The Module Concept . . . . .                                         | 159 |
| 14.2 Definition Modules . . . . .                                         | 160 |
| 14.3 Implementation Modules . . . . .                                     | 162 |
| 14.4 Separate Compilation . . . . .                                       | 163 |
| 14.5 A Complete Example: The Stack . . . . .                              | 165 |
| Summary . . . . .                                                         | 167 |
| <b>15 Local Modules</b> . . . . .                                         | 169 |
| 15.1 The Concept of Local Modules . . . . .                               | 169 |
| 15.2 Import/Export Lists . . . . .                                        | 170 |
| 15.3 Rules of Scope . . . . .                                             | 171 |
| 15.4 Local Modules Defined Side by Side . . . . .                         | 173 |
| 15.5 Modules Defined Local to a Procedure . . . . .                       | 174 |
| 15.6 A Complete Example . . . . .                                         | 179 |
| Summary . . . . .                                                         | 180 |
| <b>16 Records</b> . . . . .                                               | 181 |
| 16.1 RECORD Types . . . . .                                               | 181 |
| 16.2 Manipulation of Records and Fields . . . . .                         | 183 |
| 16.3 The WITH Statement . . . . .                                         | 186 |
| 16.4 An Example of Data Abstraction: The Use of Multiple Stacks . . . . . | 188 |
| 16.5 Records with Variant Parts . . . . .                                 | 191 |
| Summary . . . . .                                                         | 196 |

|                                                                |     |
|----------------------------------------------------------------|-----|
| <b>17 Sets</b>                                                 | 197 |
| 17.1 SET Types and the BITSET Type                             | 197 |
| 17.2 Set Constants and Variables                               | 198 |
| 17.3 Set Operators                                             | 200 |
| 17.4 The Standard Procedures INCL and EXCL                     | 204 |
| 17.5 An Example of a Module for Processing Bitsets             | 205 |
| Summary                                                        | 209 |
| <b>18 Streams</b>                                              | 210 |
| 18.1 The Stream Concept                                        | 210 |
| 18.2 Access to a Stream Component                              | 211 |
| 18.3 Text Streams                                              | 211 |
| 18.4 A Program Example: Bold Text                              | 216 |
| Summary                                                        | 216 |
| <b>19 Dynamic Data Structures</b>                              | 217 |
| 19.1 Pointers                                                  | 217 |
| 19.2 Dynamic Allocation                                        | 218 |
| 19.3 Lists                                                     | 221 |
| 19.4 A Complete Example of List Processing with a Local Module | 226 |
| 19.5 Stacks and fifo Queues: Two List Processing Modules       | 226 |
| 19.6 Binary Trees                                              | 234 |
| Summary                                                        | 241 |
| <b>20 Processes</b>                                            | 243 |
| 20.1 Concurrency                                               | 243 |
| 20.2 Processes                                                 | 244 |
| 20.3 Monitors                                                  | 244 |
| 20.4 Signals                                                   | 245 |
| 20.5 A Classical Example: A Shared Buffer                      | 249 |
| Summary                                                        | 249 |
| <b>21 Low-Level Features</b>                                   | 250 |
| 21.1 Why Low-Level Programming?                                | 250 |
| 21.2 The Type WORD                                             | 250 |
| 21.3 The Type ADDRESS                                          | 253 |
| 21.4 Type Transfer Functions                                   | 255 |
| 21.5 Coroutines                                                | 256 |
| 21.6 An Application of Coroutines                              | 259 |
| 21.7 Dynamic Storage                                           | 259 |
| 21.8 Device Handling, Interrupts and Fixed Addresses           | 260 |
| 21.9 Low-Level File Operations                                 | 262 |
| 21.10 Generic Modules                                          | 264 |
| Summary                                                        | 268 |

|                                                         |     |
|---------------------------------------------------------|-----|
| Table of Contents                                       | XI  |
| References . . . . .                                    | 269 |
| Appendix 1: The Syntax of MODULA-2 . . . . .            | 270 |
| Appendix 2: Keywords and Standard Identifiers . . . . . | 291 |