Contents

Preface — V

1 Introduction to computer mathematics languages — 1
1.1 Introduction to solving mathematical problems — 1
1.1.1 Why learn a computer mathematics language? — 1
1.1.2 Analytical and numerical solutions — 5
1.1.3 Development of mathematical packages — 6
1.1.4 Limitations of conventional computer languages — 8
1.2 History of computer mathematics languages — 10
1.2.1 The early days of computer mathematics languages — 10
1.2.2 Representative modern computer mathematics languages — 11
1.3 Three-phase solution of scientific computing problems — 12
1.4 Problems — 14

2 Fundamentals of MATLAB programming — 17
2.1 Command windows and fundamental commands — 18
2.1.1 Regulations in variable names — 18
2.1.2 Reserved constants — 19
2.1.3 Setting of display formats — 20
2.1.4 Low-level operating system commands — 21
2.1.5 Setting of MATLAB working environment — 21
2.1.6 MATLAB workspace and management — 23
2.1.7 Other supporting facilities — 23
2.2 Commonly used data types — 24
2.2.1 Numeric data types — 24
2.2.2 Symbolic data — 26
2.2.3 Generation of arbitrary symbolic matrices — 28
2.2.4 Symbolic functions — 29
2.2.5 Integer and logic variables — 29
2.2.6 Recognition of data types — 29
2.2.7 Sizes and lengths of matrices — 30
2.3 String data type — 30
2.3.1 Expression of string variables — 30
2.3.2 String processing methods — 32
2.3.3 Conversion of string variables — 33
2.3.4 Executions of string commands — 34
2.3.5 Interface of MuPAD language — 35
2.4 Other commonly used data types — 36
2.4.1 Multidimensional arrays — 36
2.4.2 Cell arrays — 37

Bibliografische Informationen
http://d-nb.info/119075018X

digitalisiert durch DEUTSCHE NATIONAL BIBLIOTHEK

2.4.3 Tables — 38
2.4.4 Structured variables — 41
2.4.5 Other data types — 42
2.5 Fundamental statement structures — 42
2.5.1 Direct assignment statements — 42
2.5.2 Function call statements — 43
2.5.3 Functions with different syntaxes — 43
2.5.4 Colon expressions — 44
2.5.5 Submatrix extractions — 45
2.5.6 Generation of equally spaced row vectors — 46
2.6 Reading and writing of different data types — 46
2.6.1 Reading and writing of data files — 46
2.6.2 Low-level reading and writing commands — 47
2.6.3 Reading and writing of Excel files — 48
2.7 Problems — 50

3 Fundamental mathematical computations — 53
3.1 Algebraic computation of matrices — 53
3.1.1 Transposing, flipping and rotating matrices — 53
3.1.2 Arithmetic operations — 55
3.1.3 Complex matrices and transformations — 56
3.1.4 Powers and roots of matrices — 57
3.1.5 Dot operations — 59
3.2 Logic and comparison operations — 59
3.2.1 Logic operations with matrices — 59
3.2.2 Comparisons of matrices — 60
3.2.3 Searching commands in matrix elements — 60
3.2.4 Attribute judgement — 62
3.3 Computation of transcendental functions — 62
3.3.1 Exponentials and logarithmic functions — 63
3.3.2 Trigonometric functions — 63
3.3.3 Inverse trigonometric functions — 65
3.3.4 Transcendental functions of matrices — 66
3.4 Simplifications and conversions of symbolic expressions — 68
3.4.1 Polynomial operations — 68
3.4.2 Conversions and simplifications of trigonometric functions — 69
3.4.3 Simplification of symbolic expressions — 70
3.4.4 Variable substitution of symbolic expressions — 71
3.4.5 Conversions of symbolic expressions — 72
3.5 Fundamental computations with data — 72
3.5.1 Integer rounding and rationalization of data — 73
3.5.2 Sorting and finding maximum and minimum of vectors — 74

3.5.3 Mean, variance and standard deviation — 75
3.5.4 Prime factors and polynomials — 76
3.5.5 Permutations and combinations — 78
3.6 Problems — 79

4 Flow control structures of MATLAB language — 83
4.1 Loop structures — 83
4.1.1 The `for` loop structure — 83
4.1.2 The `while` loop structure — 86
4.1.3 Loop implementation of iterations — 87
4.1.4 Assistant statements of loop structures — 90
4.1.5 Vectorized implementation of loops — 90
4.2 Conditional structures — 93
4.2.1 Simple conditional structures — 93
4.2.2 General form of conditional structures — 94
4.2.3 Vectorized expressions of piecewise functions — 96
4.3 Switch structures — 98
4.4 Trial structure — 100
4.5 Problems — 101

5 Function programming and debugging — 105
5.1 MATLAB scripts — 105
5.2 Fundamental structures of MATLAB functions — 106
5.2.1 Fundamental function structures — 106
5.2.2 Regulations in function names — 108
5.2.3 Examples of function programming — 108
5.3 Skills of MATLAB function programming — 112
5.3.1 Recursive structures — 112
5.3.2 Functions with variable numbers of inputs and outputs — 114
5.3.3 Fault tolerance manipulation — 116
5.3.4 Global variables — 117
5.3.5 Reading and writing of MATLAB workspace — 118
5.3.6 Anonymous and inline functions — 119
5.3.7 Subfunctions and private functions — 121
5.4 MATLAB function debugging — 122
5.4.1 Debugging of MATLAB functions — 122
5.4.2 Pseudocode and code protection — 125
5.5 MATLAB live editor — 125
5.5.1 Live editor interface — 126
5.5.2 Creating a live document — 126
5.5.3 Execution of embedded code — 128
5.5.4 Embed other objects in live editor — 128

5.5.5 Output of live files — 131
5.6 Problems — 131

6 Two-dimensional graphics — 135
6.1 Drawing two-dimensional plots — 135
6.1.1 Plotting data — 135
6.1.2 Plots of mathematical functions — 139
6.1.3 Plots of piecewise functions — 139
6.1.4 Titles in plots — 141
6.1.5 Plots with multiple vertical axes — 143
6.2 Decoration of plots — 145
6.2.1 Plot decoration with interface tools — 145
6.2.2 LaTeX support commands — 146
6.2.3 Superimposing formulas in plots — 148
6.3 Other two-dimensional plotting functions — 149
6.3.1 Polar plots — 150
6.3.2 Plots of discrete samples — 151
6.3.3 Histograms and pie charts — 152
6.3.4 Filled plots — 155
6.3.5 Logarithmic plots — 156
6.3.6 Error bar plots — 157
6.3.7 Dynamic trajectories — 157
6.3.8 Two-dimensional animation — 158
6.4 Plot window partitioning — 159
6.4.1 Regular partitioning — 159
6.4.2 Arbitrary segmentation — 161
6.5 Implicit functions — 162
6.6 Displaying and simple manipulation of images — 165
6.6.1 Input images — 165
6.6.2 Editing and displaying images — 166
6.6.3 Color space conversion — 167
6.6.4 Edge detection — 167
6.6.5 Histogram equalization — 168
6.7 Output of MATLAB graphs — 170
6.7.1 Output menus and applications — 170
6.7.2 Output commands of plots — 171
6.8 Problems — 171

7 Three-dimensional graphics — 175
7.1 Three-dimensional curves — 175
7.1.1 Drawing three-dimensional plots from data — 175
7.1.2 Three-dimensional plots of mathematical functions — 176

7.1.3 Filled plots — 177
7.1.4 Bar and pie charts — 178
7.1.5 Ribbon plots — 180
7.2 Three-dimensional surfaces — 182
7.2.1 Mesh grids and surfaces — 182
7.2.2 Shading and lights — 186
7.2.3 Three-dimensional surface from images — 188
7.2.4 Representation of functions — 189
7.2.5 Surfaces from scattered data — 190
7.3 Viewpoint setting in three-dimensional plots — 191
7.3.1 Definition of viewpoints — 192
7.3.2 Orthographic views — 193
7.3.3 Setting of arbitrary viewpoints — 193
7.4 Other three-dimensional plots — 194
7.4.1 Contour lines — 194
7.4.2 Quiver plots — 196
7.4.3 Three-dimensional implicit plots — 197
7.4.4 Surfaces of parametric equations — 199
7.4.5 Surfaces of complex functions — 199
7.4.6 Spheres and cylinders — 200
7.4.7 Voronoi diagrams and Delaunay triangulation — 203
7.5 Special treatment of three-dimensional plots — 205
7.5.1 Rotation of surfaces — 205
7.5.2 Axis specification for surfaces — 207
7.5.3 Cutting of surfaces — 208
7.5.4 Patches in surfaces — 208
7.6 Four-dimensional plots — 210
7.6.1 Slices — 210
7.6.2 A volume visualization interface — 212
7.6.3 Creating and playing of three-dimensional animations — 213
7.7 Problems — 214

8 MATLAB and its interface to other languages — 217
8.1 Introduction to C interfaces with MATLAB — 218
8.1.1 Environment setting of compilers — 218
8.1.2 Data types in Mex — 218
8.1.3 Mex file structures — 220
8.1.4 Mex file programming and procedures — 223
8.2 Mex manipulation of different data types — 225
8.2.1 Processing of various input and output data types — 225
8.2.2 Reading and writing of string variables — 226
8.2.3 Processing of multidimensional arrays — 228

8.2.4 Processing of cells — 229
8.2.5 Reading and writing of MAT files — 231
8.3 Direct calling of MATLAB functions from C programs — 233
8.4 Standalone program conversion from MATLAB functions — 238
8.5 Problems — 239

9 Fundamentals in object-oriented programming — 241
9.1 Concepts of object oriented programming — 241
9.1.1 Classes and objects — 241
9.1.2 Data type of classes and objects — 242
9.2 Design of classes — 243
9.2.1 The design of a class — 244
9.2.2 Design and input of classes — 245
9.2.3 Class display — 246
9.3 Programming of overload functions — 247
9.3.1 Overload addition functions — 247
9.3.2 Simplification functions via like-term collection — 248
9.3.3 Overload subtraction functions — 249
9.3.4 Overload multiplication functions — 250
9.3.5 Overload power functions — 252
9.3.6 Assignment and extraction of fields — 253
9.4 Inheritance and extension of classes — 254
9.4.1 Definition and display of extended classes — 254
9.4.2 Overload functions for ftf objects — 256
9.4.3 Frequency domain analysis of fractional-order transfer functions — 258
9.5 Problems — 259

10 Graphical user interface design using MATLAB — 261
10.1 Essentials in graphical user interface design — 261
10.1.1 The relationships of objects in MATLAB interface — 261
10.1.2 Window objects and properties — 262
10.1.3 Commonly used properties in window objects — 262
10.1.4 Extraction and modification of object properties — 265
10.1.5 Easy dialog boxes — 267
10.1.6 Standard dialog boxes — 269
10.2 Fundamental controls in interface design — 272
10.2.1 Commonly used controls supported — 273
10.2.2 Commonly used properties in controls — 274
10.2.3 Getting the handles — 275
10.3 Graphical user interface design tool – Guide — 276
10.4 Advanced techniques in interface design — 286

10.4.1 Design of menu systems — 287
10.4.2 Design of toolbars — 287
10.4.3 Embedding ActiveX controls — 289
10.5 APP packaging and publication — 291
10.6 Problems — 291

Bibliography — 293

MATLAB function index — 295

Index — 301