

8.1	9.3. Eigenvalues of Symmetric Triangular Matrices by Bisection Method.....	369
8.2	Problem Set 8.2.....	368
8.3	9.5. Other Methods.....	373
8.4	10.1. Introduction to Optimization.....	374
8.5	10.2. Minimization Along a Line.....	374
8.6	10.3. Powell's Method.....	376
8.7	10.4. Downhill Simplex Method.....	378
8.8	Problem Set 10.1.....	399
8.9	Appendices.....	407
8.10	A.1. Taylor Series.....	407
8.11	A.2. Matrix Algebra.....	410
8.12	Preface.....	ix

9.1	List of Program Modules.....	411
9.2	<b>1 Introduction to Python .....</b>	1
9.3	1.1 General Information .....	1
9.4	1.2 Core Python .....	4
9.5	1.3 Functions and Modules .....	16
9.6	1.4 Mathematics Modules .....	18
9.7	1.5 numpy Module .....	20
9.8	1.6 Plotting with matplotlib.pyplot .....	25
9.9	1.7 Scoping of Variables .....	28
9.10	1.8 Writing and Running Programs .....	29
10.1	<b>2 Systems of Linear Algebraic Equations .....</b>	31
10.2	2.1 Introduction .....	31
10.3	2.2 Gauss Elimination Method .....	37
10.4	2.3 LU Decomposition Methods .....	44
10.5	Problem Set 2.1 .....	55
10.6	2.4 Symmetric and Banded Coefficient Matrices .....	59
10.7	2.5 Pivoting .....	69
10.8	Problem Set 2.2 .....	78
10.9	*2.6 Matrix Inversion .....	84
10.10	*2.7 Iterative Methods .....	87
10.11	Problem Set 2.3 .....	98
10.12	2.8 Other Methods .....	102
11.1	<b>3 Interpolation and Curve Fitting .....</b>	104
11.2	3.1 Introduction .....	104
11.3	3.2 Polynomial Interpolation .....	105
11.4	3.3 Interpolation with Cubic Spline .....	120
11.5	Problem Set 3.1 .....	126
11.6	3.4 Least-Squares Fit .....	129
11.7	Problem Set 3.2 .....	141
12.1	<b>4 Roots of Equations .....</b>	145
12.2	4.1 Introduction .....	145
12.3	4.2 Incremental Search Method .....	146

4.3 Method of Bisection.....	148
4.4 Methods Based on Linear Interpolation.....	151
4.5 Newton-Raphson Method.....	156
4.6 Systems of Equations.....	161
<i>Problem Set 4.1</i> .....	166
*4.7 Zeros of Polynomials .....	173
<i>Problem Set 4.2</i> .....	180
4.8 Other Methods .....	182
<b>5 Numerical Differentiation.....</b>	<b>183</b>
5.1 Introduction.....	183
5.2 Finite Difference Approximations.....	183
5.3 Richardson Extrapolation.....	188
5.4 Derivatives by Interpolation.....	191
<i>Problem Set 5.1</i> .....	195
<b>6 Numerical Integration.....</b>	<b>199</b>
6.1 Introduction.....	199
6.2 Newton-Cotes Formulas .....	200
6.3 Romberg Integration.....	207
<i>Problem Set 6.1</i> .....	212
6.4 Gaussian Integration .....	216
<i>Problem Set 6.2</i> .....	230
*6.5 Multiple Integrals.....	232
<i>Problem Set 6.3</i> .....	243
<b>7 Initial Value Problems.....</b>	<b>246</b>
7.1 Introduction.....	246
7.2 Euler's Method.....	247
7.3 Runge-Kutta Methods .....	252
<i>Problem Set 7.1</i> .....	263
7.4 Stability and Stiffness .....	268
7.5 Adaptive Runge-Kutta Method .....	271
7.6 Bulirsch-Stoer Method .....	280
<i>Problem Set 7.2</i> .....	287
7.7 Other Methods .....	292
<b>8 Two-Point Boundary Value Problems.....</b>	<b>293</b>
8.1 Introduction.....	293
8.2 Shooting Method.....	294
<i>Problem Set 8.1</i> .....	304
8.3 Finite Difference Method.....	307
<i>Problem Set 8.2</i> .....	317
<b>9 Symmetric Matrix Eigenvalue Problems .....</b>	<b>321</b>
9.1 Introduction.....	321
9.2 Jacobi Method .....	324
9.3 Power and Inverse Power Methods.....	336
<i>Problem Set 9.1</i> .....	345
9.4 Householder Reduction to Tridiagonal Form .....	351

9.5 Eigenvalues of Symmetric Tridiagonal Matrices .....	359
<i>Problem Set 9.2</i> .....	368
9.6 Other Methods .....	373
<b>10 Introduction to Optimization .....</b>	<b>374</b>
10.1 Introduction .....	374
10.2 Minimization Along a Line .....	376
10.3 Powell's Method .....	382
10.4 Downhill Simplex Method .....	392
<i>Problem Set 10.1</i> .....	399
<b>Appendices .....</b>	<b>407</b>
A1 Taylor Series .....	407
A2 Matrix Algebra .....	410

<i>This book is designed for engineers and engineering students of advanced standing who have knowledge of engineering mechanics, statics, dynamics, and mechanics of materials.</i>	
<b>List of Program Modules (by Chapter) .....</b>	<b>417</b>
<b>Index .....</b>	<b>421</b>

The primary purpose of the text is to teach numerical methods. It is not a primer on Python programming. We introduce just enough Python to implement the numerical algorithms. That leaves the vast majority of the language unexplored.

Most engineers are not programmers, but problem solvers. They want to know what methods can be applied to a given problem, what their strengths and pitfalls are, and how to implement them. Engineers are not expected to write computer code for basic tasks from scratch; they are more likely to use functions and subroutines that have been already written and tested. Thus, programming by engineers is largely confined to assembling existing bits of code into a coherent package that solves the problem at hand.

The “bit” of code is usually a function that implements a specific task. For the user the details of the code are unimportant. What matters are the interface (what goes in and what comes out) and an understanding of the method on which the algorithm is based. Because no numerical algorithm is infallible, the importance of understanding the underlying method cannot be overemphasized. It is, in fact, the rationale behind learning numerical methods.

This book attempts to conform to the views outlined earlier. Each numerical method is explained in detail and its shortcomings are pointed out. The examples that follow individual topics fall into two categories: hand computations that illustrate the inner workings of the method, and small programs that show how the computer code is utilized in solving a problem. Problems that require programming are marked with .

The material consists of the usual topics covered in an engineering course on numerical methods: solution of equations, interpolation and data fitting, numerical differentiation and integration, solution of ordinary differential equations, and eigenvalue problems. The choice of methods within each topic is tilted toward relevance to engineering problems. For example, there is an extensive discussion of symmetric, sparsely populated coefficient matrices in the solution of simultaneous equations.