

Contents

Acknowledgments	ix
About the Authors	xi
Authors' Note	xiii

1	What This Book Is About	1
1.1	Programming and Mathematics	2
1.2	A Historical Perspective	2
1.3	Prerequisites	3
1.4	Roadmap	4
2	The First Algorithm	7
2.1	Egyptian Multiplication	8
2.2	Improving the Algorithm	11
2.3	Thoughts on the Chapter	15
3	Ancient Greek Number Theory	17
3.1	Geometric Properties of Integers	17
3.2	Sifting Primes	20
3.3	Implementing and Optimizing the Code	23
3.4	Perfect Numbers	28
3.5	The Pythagorean Program	32
3.6	A Fatal Flaw in the Program	34
3.7	Thoughts on the Chapter	38
4	Euclid's Algorithm	41
4.1	Athens and Alexandria	41
4.2	Euclid's Greatest Common Measure Algorithm	45
4.3	A Millennium without Mathematics	50
4.4	The Strange History of Zero	51
4.5	Remainder and Quotient Algorithms	53
4.6	Sharing the Code	57
4.7	Validating the Algorithm	59
4.8	Thoughts on the Chapter	61

5	The Emergence of Modern Number Theory	63
5.1	Mersenne Primes and Fermat Primes	63
5.2	Fermat's Little Theorem	69
5.3	Cancellation	72
5.4	Proving Fermat's Little Theorem	77
5.5	Euler's Theorem	79
5.6	Applying Modular Arithmetic	83
5.7	Thoughts on the Chapter	84
6	Abstraction in Mathematics	85
6.1	Groups	85
6.2	Monoids and Semigroups	89
6.3	Some Theorems about Groups	92
6.4	Subgroups and Cyclic Groups	95
6.5	Lagrange's Theorem	97
6.6	Theories and Models	102
6.7	Examples of Categorical and Non-categorical Theories	104
6.8	Thoughts on the Chapter	107
7	Deriving a Generic Algorithm	111
7.1	Untangling Algorithm Requirements	111
7.2	Requirements on A	113
7.3	Requirements on N	116
7.4	New Requirements	118
7.5	Turning Multiply into Power	119
7.6	Generalizing the Operation	121
7.7	Computing Fibonacci Numbers	124
7.8	Thoughts on the Chapter	127
8	More Algebraic Structures	129
8.1	Stevin, Polynomials, and GCD	129
8.2	Göttingen and German Mathematics	135
8.3	Noether and the Birth of Abstract Algebra	140
8.4	Rings	142
8.5	Matrix Multiplication and Semirings	145
8.6	Application: Social Networks and Shortest Paths	147
8.7	Euclidean Domains	150
8.8	Fields and Other Algebraic Structures	151
8.9	Thoughts on the Chapter	152
9	Organizing Mathematical Knowledge	155
9.1	Proofs	155
9.2	The First Theorem	159

9.3	Euclid and the Axiomatic Method	161
9.4	Alternatives to Euclidean Geometry	164
9.5	Hilbert's Formalist Approach	167
9.6	Peano and His Axioms	169
9.7	Building Arithmetic	173
9.8	Thoughts on the Chapter	176
10	Fundamental Programming Concepts	177
10.1	Aristotle and Abstraction	177
10.2	Values and Types	180
10.3	Concepts	181
10.4	Iterators	184
10.5	Iterator Categories, Operations, and Traits	185
10.6	Ranges	188
10.7	Linear Search	190
10.8	Binary Search	191
10.9	Thoughts on the Chapter	196
11	Permutation Algorithms	197
11.1	Permutations and Transpositions	197
11.2	Swapping Ranges	201
11.3	Rotation	204
11.4	Using Cycles	207
11.5	Reverse	212
11.6	Space Complexity	215
11.7	Memory-Adaptive Algorithms	216
11.8	Thoughts on the Chapter	217
12	Extensions of GCD	219
12.1	Hardware Constraints and a More Efficient Algorithm	219
12.2	Generalizing Stein's Algorithm	222
12.3	Bézout's Identity	225
12.4	Extended GCD	229
12.5	Applications of GCD	234
12.6	Thoughts on the Chapter	234
13	A Real-World Application	237
13.1	Cryptology	237
13.2	Primality Testing	240
13.3	The Miller-Rabin Test	243
13.4	The RSA Algorithm: How and Why It Works	245
13.5	Thoughts on the Chapter	248

14 Conclusions 249**Further Reading 251****A Notation 257****B Common Proof Techniques 261****B.1 Proof by Contradiction 261****B.2 Proof by Induction 262****B.3 The Pigeonhole Principle 263****C C++ for Non-C++ Programmers 265****C.1 Template Functions 265****C.2 Concepts 266****C.3 Declaration Syntax and Typed Constants 267****C.4 Function Objects 268****C.5 Preconditions, Postconditions, and Assertions 269****C.6 STL Algorithms and Data Structures 269****C.7 Iterators and Ranges 270****C.8 Type Aliases and Type Functions with using in C++11 272****C.9 Initializer Lists in C++11 272****C.10 Lambda Functions in C++11 272****C.11 A Note about inline 273****Bibliography 275****Index 281**