

Spring 2018 Updates

Computing Technology for All

- New sections
 - 1.2 Historical figures in computing
 - 3.4 Cellular networks
- Modified sections
 - Type of computers
 - Common input devices

Data Structure Essentials

- New sections
 - 1.4 Algorithm analysis
 - 4.8 BST parent node pointers
 - 5.4 Red-black tree: A balanced tree
 - 5.5 Red-black tree: Rotations
 - 5.6 Red-black tree: Insertion
 - 5.7 Red-black tree: Removal
 - 7.1 B-trees
 - 7.2 2-3-4 tree search algorithm
 - 7.3 2-3-4 tree insert algorithm
 - 7.4 2-3-4 tree rotations and fusion
 - 7.5 2-3-4 tree removal

Digital Design

- New sections
 - 1.13 Top-down design
 - 1.15 Multiple outputs
- Modified sections
 - 1.1 Voltage, current, switches, and transistors (Combined 1.1 Voltage and current, 1.2 switches, and 1.3 CMOS transistors)
 - 1.2 Transistors and gates
 - 1.3 Boolean algebra and equations (Combined 1.5 Boolean algebra, 1.6 Boolean algebra and digital design)
 - 1.6 Equations to/from circuits
 - 1.8 Basic properties of Boolean algebra
 - 1.9 Sum-of-products form
 - 1.11 Sum-of-minterms form
 - 1.12 Truth tables
 - 2.1 Two-level combinational logic minimization
 - 2.2 K-maps: Introduction
 - 2.3 3- and 4-variable K-maps
 - 2.4 DeMorgan's Law
 - 2.5 XOR / XNOR gates
 - 2.7 Muxes
 - 2.8 Decoders
 - 2.9 Don't cares
 - 3.1 SR latches
 - 3.2 D flip-flops and registers (Combined 3.2 Flip-flops and 3.3 Basic registers)

Discrete Math

- New sections
 - 8.7 Loop invariants
 - 8.12 Analyzing the time complex of recursive algorithms
 - 8.13 Divide and conquer algorithms: Introduction and mergesort
 - 8.14 Divide and conquer algorithms: Binary search
 - 8.16 Solving linear non-homogeneous recurrence relations
 - 8.17 Divide and conquer recurrences
- New challenge activities in the following sections
 - 6.10 N-ary relations and relational databases
 - 7.2 Growth of functions and complexity
 - 7.5 Turing machines
 - 8.1 Sequences
 - 8.2 Recurrence relations
 - 8.3 Summations
 - 9.5 Greatest common divisor and Euclid's algorithm
 - 10.11 Inclusion-exclusion principle
 - 12.1 Probability of an event
 - 12.3 Conditional probability and independence
- Testbank

MATLAB, Introduction to

- Miscellaneous
 - Reorganized numerous sections into more focused chapters
- Modified sections
 - 4.8 Functions to create numeric row arrays
 - 4.9 Relational operators and row arrays
 - 4.10 Comparing floating-point numbers for equality
 - 4.11 Strings as arrays
 - 4.12 Column arrays
 - 5.1 1D element-wise arithmetic operators
 - 5.2 Logical operators and 1D arrays
 - 5.3 Combining relational and logical operators
 - 5.4 Functions and 1D arrays
 - 5.5 Saving and loading variables
 - 8.8 2D element-wise arithmetic operators (Renamed, was "2D Arrays: Arithmetic Operators")
 - 9.1 Simple plotting
 - 9.2 2D plotting
 - 9.10 Rectangular data grids (Renamed, was 'Meshgrid')
 - 14.1 Grouping data: Structure (Split and expanded, was "Structure")
 - 14.2 Defining a structure (Split and expanded, was "Structure")
 - 14.3 Accessing a structure (Split and expanded, was "Structure")
 - 14.4 MATLAB functions for structures (Split and expanded, was "Structure")
 - 14.6 Structure with array fields (Renamed, was "Structure of arrays")
 - 14.7 Array of structures
 - 15.1 Data classes (Renamed, was "Heterogenous data records")
 - 15.2 Grouping data: Cell (Renamed, was "Constructing cell arrays")
 - 15.3 Accessing a cell (Renamed, was "Indexing cell arrays")
 - 18.1 Interpolation
 - 18.2 Curve fitting: Least Squares Regression (Combined 6.14 Linear regression curve fitting and 6.15 Nonlinear regression)
 - 18.3 Numerical differentiation

- 18.4 Numerical integration
- New sections
 - 9.11 Triangular data grids
 - 12.1 Scripts with local functions
 - 18.5 Nonlinear equations: Zeros
 - 15.5 Categorical arrays
 - 14.5 Structures as function arguments
 - 14.8 Structure example: Seat reservation system
 - 15.6 Table
- New chapter: Object Oriented Programming
 - 16.1 Objects: Introduction
 - 16.2 Classes: Introduction
 - 16.3 Class properties
 - 16.4 Class methods
 - 16.5 Class example: Seat reservation system
 - 16.6 Operator overloading
 - 16.8 Mutators, accessors, and private helpers
 - 16.7 Handle classes
 - 16.9 Handle class example: Ball drop
- New chapter: Linear Algebra I
 - 19.1 Vectors (Modified to replace section 6.7)
 - 19.2 Matrices
 - 19.3 Matrix transpose
 - 19.4 Matrix calculations
 - 19.5 Square matrices
 - 19.6 Elementary square matrices
 - 19.7 Linear systems
 - 19.8 Square matrices: Solving $Ax = b$
 - 19.9 Triangular matrix linear equation solvers
 - 19.10 Gaussian elimination I
 - 19.11 Gaussian elimination II
 - 19.12 Gauss-Jordan elimination
- New chapter: Linear Algebra II
 - 20.1 Matrix rank
 - 20.2 Properties of the row-echelon matrices of A
 - 20.3 Vector spaces I
 - 20.4 Vector spaces II
 - 20.5 Row vector space of A
 - 20.6 Column vector space of A
 - 20.7 Null vector space of A
 - 20.8 Null vector space of transpose A
 - 20.9 Solutions to $Ax = b$
 - 20.10 Least squares solutions to $Ax=b$
 - 20.11 Eigenvalues and eigenvectors

Materials and Energy Balances

- New sections
 - 9.5 Trendlines
- Modified sections
 - 1.2 Significant figures
 - 1.3 Units and unit conversions
 - 1.6 Pressure
 - 2.1 Process types and process units
 - 4.1 Solid, liquid, and gas
 - 4.3 Properties of steam
 - 4.6 Vapor pressure and Antoine equation
 - 5.2 Bubble and dew point
 - 6.5 Heat capacity
 - 6.6 Humidity
 - 7.1 Energy balances for reacting systems
 - 7.2 Heat of reaction and Hess's law
 - 7.3 Heat of formation method
 - 7.4 Combustion reactions and the energy balance

Programming in C

- New sections
 - 1.9 Why programming
 - 3.1 If-else branches (general)
 - 3.5 Detecting ranges (general)
 - 3.9 Example: Toll calculation
 - 4.6 Loops and strings
- Modified sections
 - 1.1 Programming (general)
 - 1.2 Programming basics (Combined 1.2 A first program, 1.3 Basic output, and 1.4 Basic input)
 - 1.3 Comments and whitespace
 - 1.4 Errors and warning
 - 2.1 Variables and assignments (general)
 - 2.2 Variables (int)
 - 2.3 Identifiers
 - 2.4 Arithmetic expressions (general) (Split 2.4 Arithmetic expressions (int))
 - 2.5 Arithmetic expressions (int) (Split 2.4 Arithmetic expressions (int))
 - 2.6 Example: Health data
 - 2.7 Floating-point numbers (double) (Split 2.5 Floating-point (double))
 - 2.8 Scientific notation for floating-point literals (Split 2.5 Floating-point(double))
 - 2.10 Using math functions
 - 2.11 Integer division and modulo (Split 2.4 Arithmetic expressions (int))
 - 2.12 Type conversions
 - 2.14 Characters
 - 2.15 Strings
 - 3.2 If-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.3 More if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.4 Equality and relational operators
 - 3.6 Detecting ranges with if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.7 Logical operators (Split 3.4 Logical operators)
 - 3.8 Order of evaluation (Split 3.4 Logical operators)
 - 3.11 Boolean data types

- 3.12 String comparisons
 - 3.13 Character operations
 - 3.14 String access operations
 - 4.1 Loops (general)
 - 4.2 While loops
 - 4.3 More while examples
 - 4.4 For loops (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 4.5 More for loop examples (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 5.1 Array concept (general)
 - 5.2 Arrays
 - 5.5 Multiple arrays
 - 5.7 Loop-modifying or copying/comparing arrays
 - 6.1 Function basics (Combined 6.1 Function basics and 6.2 Parameters)
 - 6.2 Return
 - 6.3 Reasons for functions
 - 6.4 Functions with branches/loops
 - 7.2 Structs and functions
 - 7.3 Structs and arrays
- Miscellaneous
 - Code style updated in Ch 1 - 5, 6.1 - 6.4 (latter sections/chapters will be updated in a future release)
 - Modified code examples to remove "return;" from end of functions where return is void
 - Modified code examples that initialize variables upon declaration

Programming in C++

- New sections
 - 1.9 Why programming
 - 3.1 If-else branches (general)
 - 3.5 Detecting ranges (general)
 - 3.9 Example: Toll calculation
 - 4.6 Loops and strings
 - 14.1 Range-based for loop
 - 14.2 List
 - 14.3 Pair
 - 14.4 Map
 - 14.5 Set
 - 14.6 Queue
 - 14.7 Deque
- Modified sections
 - 1.1 Programming (general)
 - 1.2 Programming basics (Combined 1.2 A first program, 1.3 Basic output, and 1.4 Basic input)
 - 1.3 Comments and whitespace
 - 1.4 Errors and warning
 - 2.1 Variables and assignments (general)
 - 2.2 Variables (int)
 - 2.3 Identifiers
 - 2.4 Arithmetic expressions (general) (Split 2.4 Arithmetic expressions (int))
 - 2.5 Arithmetic expressions (int) (Split 2.4 Arithmetic expressions (int))
 - 2.6 Example: Health data
 - 2.7 Floating-point numbers (double) (Split 2.5 Floating-point(double))
 - 2.8 Scientific notation for floating-point literals (Split 2.5 Floating-point(double))
 - 2.10 Using math functions
 - 2.11 Integer division and modulo (Split 2.4 Arithmetic expressions (int))

- 2.12 Type conversions
 - 2.14 Characters
 - 2.15 Strings
 - 3.2 If-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.3 More if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.4 Equality and relational operators
 - 3.6 Detecting ranges with if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.7 Logical operators (Split 3.4 Logical operators)
 - 3.8 Order of evaluation (Split 3.4 Logical operators)
 - 3.11 Boolean data types
 - 3.12 String comparisons
 - 3.13 Character operations
 - 3.14 String access operations
 - 3.15 More string operations (Renamed, was: "String modify operations")
 - 4.1 Loops (general)
 - 4.2 While loops
 - 4.3 More while examples
 - 4.4 For loops (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 4.5 More for loop examples (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 5.1 Array/vector concept (general)
 - 5.2 Vectors
 - 5.6 Vector resize
 - 5.7 Vector push_back
 - 5.8 Loop-modifying or copying/comparing vectors
 - 5.8 Multiple vectors
 - 6.1 Function basics (Combined 6.1 Function basics and 6.2 Parameters)
 - 6.2 Return
 - 6.3 Reasons for functions
 - 6.4 Functions with branches/loops
 - 7.5 Initialization and constructors (Updates for C++11)
 - 7.7 Constructor initialization lists (Updates for C++11)
 - 15.2 Arrays
 - 15.4 Multiple arrays
 - 15.5 Loop-modifying or copying/comparing arrays
 - 15.12 Structs and functions
 - 15.13 Structs and vectors
- Miscellaneous
 - Code style updated in Ch 1 - 5, 6.1 - 6.4 (latter sections/chapters will be updated in a future release)
 - Modified code examples to remove "return;" from end of functions where return is void
 - Modified code examples that initialize variables upon declaration

Programming in Java

- New Sections
 - 1.9 Why programming
 - 3.1 If-else branches (general)
 - 3.5 Detecting ranges (general)
 - 3.9 Example: Toll calculation
 - 4.6 Loops and strings
 - 6.11 Using Scanner in methods
 - 6.12 Perfect size arrays
 - 6.13 Oversize arrays
 - 6.14 Methods with oversize arrays
 - 6.15 Comparing perfect size and oversize arrays

- Modified sections
 - 1.1 Programming (general)
 - 1.2 Programming basics (Combined 1.2 A first program, 1.3 Basic output, and 1.4 Basic input)
 - 1.3 Comments and whitespace
 - 1.4 Errors and warning
 - 2.1 Variables and assignments (general)
 - 2.2 Variables (int)
 - 2.3 Identifiers
 - 2.4 Arithmetic expressions (general) (Split 2.4 Arithmetic expressions (int))
 - 2.5 Arithmetic expressions (int) (Split 2.4 Arithmetic expressions (int))
 - 2.6 Example: Health data
 - 2.7 Floating-point numbers (double) (Split 2.5 Floating-point (double))
 - 2.8 Scientific notation for floating-point literals (Split 2.5 Floating-point (double))
 - 2.10 Using math methods
 - 2.11 Integer division and modulo (Split 2.4 Arithmetic expressions (int))
 - 2.12 Type conversions
 - 2.14 Characters
 - 2.15 Strings
 - 3.2 If-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.3 More if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.4 Equality and relational operators
 - 3.6 Detecting ranges with if-else (Split 3.3 Multiple if-else branches)
 - 3.7 Logical operators (Split 3.4 Logical operators)
 - 3.8 Order of evaluation (Split 3.4 Logical operators)
 - 3.11 Boolean data types
 - 3.12 String comparisons
 - 3.13 Character operations
 - 3.14 String access operations
 - 3.15 More string operations (Renamed, was: "String modify operations")
 - 4.1 Loops (general)
 - 4.2 While loops
 - 4.3 More while examples
 - 4.4 For loops (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 4.5 More for loop examples (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 5.1 Array concept (general)
 - 5.2 Arrays
 - 5.5 Multiple arrays
 - 5.7 Loop-modifying or copying/comparing arrays
 - 6.1 Method basics (Combined 6.1 Method basics and 6.2 Parameters)
 - 6.2 Return
 - 6.3 Reasons for methods
 - 6.4 Methods with branches/loops
 - 7.2 Classes: Introduction
 - 15.2 List: LinkedList

- Miscellaneous
 - Code style updated in Ch 1 - 5, 6.1 - 6.4 (latter sections/chapters will be updated in a future release)
 - Modified code examples to remove "return;" from end of methods where return is void
 - Modified code examples that initialize variables upon declaration

AP Java

- New Sections
 - 1.9 Why programming
 - 3.1 If-else branches (general)
 - 3.5 Detecting ranges (general)
 - 3.9 Example: Toll calculation
 - 4.6 Loops and strings
 - 6.11 Using Scanner in methods
 - 6.12 Perfect size arrays
 - 6.13 Oversize arrays
 - 6.14 Methods with oversize arrays
 - 6.15 Comparing perfect size and oversize arrays
- Modified sections
 - 1.1 Programming (general)
 - 1.2 Programming basics (Combined 1.2 A first program, 1.3 Basic output, and 1.4 Basic input)
 - 1.3 Comments and whitespace
 - 1.4 Errors and warning
 - 2.1 Variables and assignments (general)
 - 2.2 Variables (int)
 - 2.3 Identifiers
 - 2.4 Arithmetic expressions (general) (Split 2.4 Arithmetic expressions (int))
 - 2.5 Arithmetic expressions (int) (Split 2.4 Arithmetic expressions (int))
 - 2.6 Example: Health data
 - 2.7 Floating-point numbers (double) (Split 2.5 Floating-point (double))
 - 2.8 Scientific notation for floating-point literals (Split 2.5 Floating-point (double))
 - 2.10 Using math methods
 - 2.11 Integer division and modulo (Split 2.4 Arithmetic expressions (int))
 - 2.12 Type conversions
 - 2.14 Characters
 - 2.15 Strings
 - 3.2 If-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.3 More if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.4 Equality and relational operators
 - 3.6 Detecting ranges with if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.7 Logical operators (Split 3.4 Logical operators)
 - 3.8 Order of evaluation (Split 3.4 Logical operators)
 - 3.10 Boolean data types
 - 3.11 String comparisons
 - 3.12 Character operations
 - 3.13 String access operations
 - 3.14 More string operations (Renamed, was: "String modify operations")
 - 4.1 Loops (general)
 - 4.2 While loops
 - 4.3 More while examples
 - 4.4 For loops (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 4.5 More for loop examples (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 5.1 Array concept (general)
 - 5.2 Arrays
 - 5.5 Multiple arrays
 - 5.7 Loop-modifying or copying/comparing arrays
 - 6.1 Method basics (Combined 6.1 Method basics and 6.2 Parameters)
 - 6.2 Return
 - 6.3 Reasons for methods
 - 6.4 Methods with branches/loops
 - 7.2 Classes: Introduction

- 23.2 List: LinkedList
- Miscellaneous
 - Code style updated in Ch 1 - 5, 6.1 - 6.4 (latter sections/chapters will be updated in a future release)
 - Modified all relevant examples to remove "return;" from end of methods where return is void
 - Modified all relevant examples that initialize variables upon declaration

Java Early Objects

- New Sections
 - 1.8 Why programming
 - 5.1 If-else branches (general)
 - 5.5 Detecting ranges (general)
 - 5.9 Example: A SimpleCar class
 - 5.10 Example: Toll calculation
 - 6.6 Loops and strings
 - 8.3 Using Scanner in methods
 - 20.2 Example: A Dice Game
 - 20.7 Perfect size arrays
 - 20.8 Oversize arrays
 - 20.9 Methods with oversize arrays
 - 20.10 Comparing perfect size and oversize arrays
- Modified sections
 - 1.1 Programming (general)
 - 1.2 Programming basics (Combined 1.2 A first program, 1.3 Basic output, and 1.4 Basic input)
 - 1.3 Errors and warning
 - 2.2 Variables and assignments (general)
 - 2.3 Variables (int)
 - 2.4 Identifiers
 - 2.5 Arithmetic expressions (general) (Split 2.4 Arithmetic expressions (int))
 - 2.6 Arithmetic expressions (int) (Split 2.4 Arithmetic expressions (int))
 - 2.7 Example: Health data
 - 2.8 Floating-point numbers (double) (Split 2.5 Floating-point (double))
 - 2.9 Scientific notation for floating-point literals (Split 2.5 Floating-point (double))
 - 2.12 Constructing Objects
 - 3.1 Classes: Introduction
 - 3.2 User-defined method basics (EO)
 - 3.3 Return (EO)
 - 3.4 Reasons for defining methods (EO)
 - 4.2 Using math methods
 - 4.3 Integer division and modulo (Split 2.4 Arithmetic expressions (int))
 - 4.4 Type conversions
 - 4.6 Characters
 - 4.7 Strings
 - 5.2 If-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 5.3 More if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 5.4 Equality and relational operators
 - 5.6 Detecting ranges with if-else (Reorganized 3.1 If-else and 3.3 Multiple if-else branches)
 - 3.7 Logical operators (Split 3.4 Logical operators)
 - 3.8 Order of evaluation (Split 3.4 Logical operators)
 - 5.12 Boolean data types
 - 5.13 String comparisons
 - 5.14 Character operations
 - 5.15 String access operations
 - 5.16 More string operations (Renamed, was: "String modify operations")

- 6.1 Loops (general)
 - 6.2 While loops
 - 6.3 More while examples
 - 6.4 For loops (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 6.5 More for loop examples (Reorganized 4.4 For loops and 4.5 More for loop examples)
 - 7.1 Array concept (general)
 - 7.2 Arrays
 - 7.5 Multiple arrays
 - 7.7 Loop-modifying or copying/comparing arrays
 - 8.1 Methods with branches/loops (EO)
 - 8.2 Unit testing methods (EO)
 - 8.4 Methods: Common errors (EO)
 - 8.5 Array parameters (EO)
 - 8.6 Scope of variable/method definitions (EO)
 - 8.7 Method name overloading (EO)
 - 9.8 Classes, ArrayLists, and methods: A seat reservation example (EO)
 - 9.10 ArrayList ADT (EO)
 - 9.11 Parameters of reference types (EO)
 - 9.13 Java example: Domain name availability with classes (EO)
 - 13.2 List: LinkedList
 - 20.4 Engineering examples using Methods (EO)
 - 20.10 Comments and whitespace
- Miscellaneous
 - Code style updated in Ch 1 - 5, 6.1 - 6.4 (latter sections/chapters will be updated in a future release)
 - Modified all relevant examples to remove "return;" from end of methods where return is void
 - Modified all relevant examples that initialize variables upon declaration