

1. Record Nr.	UNINA9910861061603321
Autore	Campesato Oswald
Titolo	Data Structures in Java
Pubbl/distr/stampa	Bloomfield : , : Mercury Learning & Information, , 2023 ©2023
ISBN	1-68392-953-5 1-68392-954-3
Edizione	[1st ed.]
Descrizione fisica	1 online resource (248 pages)
Disciplina	005.133
Soggetti	Data structures (Computer science) Java (Computer program language) COMPUTERS / Programming Languages / Java
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	Cover -- Title Page -- Copyright -- Dedication -- Contents -- Preface -- Chapter 1: Introduction to Java -- A Very Brief Introduction to Java -- Downloading a Java Release (Short Version) -- Selecting a Version of Java (Detailed Version) -- Java 8 and Java 11 -- Java Version Numbers -- JRE Versus a JDK -- Java Distributions -- Java IDEs -- Data Types, Operators, and Their Precedence -- Java Comments -- Java Operators -- Creating and Compiling Java Classes -- "Hello World" and Working With Numbers -- The Java String Class -- Java Strings With Metacharacters -- The Java New Operator -- Equality of Strings -- Comparing Strings -- Searching for a Substring in Java -- Useful String Methods in Java -- Parsing Strings in Java -- Conditional Logic in Java -- Determining Leap Years -- Finding the Divisors of a Number -- Checking for Palindromes -- Working With Arrays of Strings -- Working With the StringBuilder Class -- Static Methods in Java -- Other Static Types in Java -- Summary -- Chapter 2: Recursion and Combinatorics -- What Is Recursion? -- Arithmetic Series -- Calculating Arithmetic Series (Iterative) -- Calculating Arithmetic Series (Recursive) -- Calculating Partial Arithmetic Series -- Geometric Series -- Calculating a Geometric Series (Iterative) -- Calculating Geometric Series (Recursive) -- Factorial Values -- Calculating Factorial Values (Iterative)

-- Calculating Factorial Values (Recursive) -- Calculating Factorial Values (Tail Recursion) -- Fibonacci Numbers -- Calculating Fibonacci Numbers (Recursive) -- Calculating Fibonacci Numbers (Iterative) -- Task: Reverse a String via Recursion -- Task: Check for Balanced Parentheses -- Task: Calculate the Number of Digits -- Task: Determine if a Positive Integer is Prime -- Task: Find the Prime Divisors of a Positive Integer -- Task: Goldbach's Conjecture. -- Task: Calculate the GCD (Greatest Common Divisor) -- Task: Calculate the LCM (Lowest Common Multiple) -- What Is Combinatorics? -- Working With Permutations -- Working With Combinations -- The Number of Subsets of a Finite Set -- Task: Subsets Containing a Value Larger Than k -- Summary -- Chapter 3: Strings and Arrays -- Time and Space Complexity -- Task: Maximum and Minimum Powers of an Integer -- Task: Binary Substrings of a Number -- Task: Common Substring of Two Binary Numbers -- Task: Multiply and Divide via Recursion -- Task: Sum of Prime and Composite Numbers -- Task: Count Word Frequencies -- Task: Check if a String Contains Unique Characters -- Task: Insert Characters in a String -- Task: String Permutations -- Task: Check for Palindromes -- Task: Check for Longest Palindrome -- Working With Sequences of Strings -- The Maximum Length of a Repeated Character in a String -- Find a Given Sequence of Characters in a String -- Task: Longest Sequences of Substrings -- The Longest Sequence of Unique Characters -- The Longest Repeated Substring -- Working With 1D Arrays -- Rotate an Array -- Task: Invert Adjacent Array Elements -- Task: Shift Nonzero Elements Leftward -- Task: Sort Array In-Place in O(n) Without a Sort Function -- Task: Generate 0 That Is Three Times More Likely Than a 1 -- Task: Invert Bits in Even and Odd Positions -- Task: Check for Adjacent Set Bits in a Binary Number -- Task: Count Bits in a Range of Numbers -- Task: Find the Right-Most Set Bit in a Number -- Task: The Number of Operations to Make All Characters Equal -- Task: Compute XOR without XOR for Two Binary Numbers -- Task: Swap Adjacent Bits in Two Binary Numbers -- Working With 2D Arrays -- The Transpose of a Matrix -- Summary -- Chapter 4: Search and Sort Algorithms -- Search Algorithms -- Linear Search -- Binary Search Walk-Through. -- Binary Search (Iterative Solution) -- Binary Search (Recursive Solution) -- Well-Known Sorting Algorithms -- Bubble Sort -- Find Anagrams in a List of Words -- Selection Sort -- Insertion Sort -- Comparison of Sort Algorithms -- Merge Sort -- Merge Sort With a Third Array -- Merge Sort Without a Third Array -- Merge Sort: Shift Elements From End of Lists -- How Does Quick Sort Work? -- Quick Sort Code Sample -- Shell Sort -- Task: Sorted Arrays and the Sum of Two Numbers -- Summary -- Chapter 5: Linked Lists (1) -- Types of Data Structures -- Linear Data Structures -- Nonlinear Data Structures -- Data Structures and Operations -- Operations on Data Structures -- What Are Singly Linked Lists? -- Tradeoffs for Linked Lists -- Singly Linked Lists: Create and Append Operations -- A Node Class for Singly Linked Lists -- Java Code for Appending a Node -- Finding a Node in a Linked List -- Appending a Node in a Linked List -- Finding a Node in a Linked List (Method 2) -- Singly Linked Lists: Update and Delete Operations -- Updating a Node in a Singly Linked List -- Java Code to Update a Node -- Deleting a Node in a Linked List -- Java Code for Deleting a Node -- Java Code for a Circular Linked List -- Java Code for Updating a Circular Linked List -- Working With Doubly Linked Lists (DLL) -- A Node Class for Doubly Linked Lists -- Appending a Node in a Doubly Linked List -- Java Code for Appending a Node -- Java Code for Inserting a New Root Node -- Java Code for Inserting an Intermediate Node -- Traversing the Nodes in a Doubly Linked List -- Updating a Node in a Doubly Linked

List -- Java Code to Update a Node -- Deleting a Node in a Doubly Linked List -- Java Code to Delete a Node -- Summary -- Chapter 6: Linked Lists (2) -- Task: Adding Numbers in a Linked List (1) -- Task: Adding Numbers in a Linked List (2) -- Task: Adding Numbers in a Linked List (3).
Task: Display the First k Nodes -- Task: Display the Last k Nodes -- Reverse a Singly Linked List via Recursion -- Task: Remove Duplicates -- Task: Concatenate Two Lists -- Task: Merge Two Ordered Linked Lists -- Task: Split an Ordered List Into Two Lists -- Task: Remove a Given Node from a List -- Task: Find the Middle Element in a List -- Task: Reverse a Linked List -- Task: Check for Palindrome in a Linked List -- Summary -- Chapter 7: Queues and Stacks -- What Is a Queue? -- Types of Queues -- Creating a Queue Using an Array List -- Creating a Queue Using an Array List -- Other Types of Queues -- What Is a Stack? -- Use Cases for Stacks -- Operations With Stacks -- Working With Stacks -- Task: Reverse and Print Stack Values -- Task: Display the Min and Max Stack Values (1) -- Task: Reverse a String Using a Stack -- Task: Find Stack Palindromes -- Task: Balanced Parentheses -- Task: Tokenize Arithmetic Expressions -- Task: Classify Tokens in Arithmetic Expressions -- Infix, Prefix, and Postfix Notations -- Summary -- Index.

Sommario/riassunto

This book is a fast-paced introduction to using data structures with Java. Numerous code samples and listings are included to support myriad topics. The first chapter contains a quick introduction to Java, along with Java code samples to check for leap years, find divisors of a number, and work with arrays of strings. The second chapter introduces recursion and uses code samples to check if a positive number is prime, to find the prime divisors of a positive integer, to calculate the GCD (greatest common divisor) and LCM (lowest common multiple) of a pair of positive integers. The third chapter contains Java code samples involving strings and arrays, such as finding binary substrings of a number, checking if strings contain unique characters, counting bits in a range of numbers, and how to compute XOR without using the XOR function. Chapters 4 through 6 include Java code samples involving search algorithms, concepts in linked lists, and tasks involving linked lists. Finally, Chapter 7 discusses data structures called queues and stacks, along with additional Java code samples. FEATURES: Extensive topics, code samples, and scripts related to data structures Covers strings, arrays, queues, and stacks, linked lists, computing the XOR function, checking for unique characters, and more Includes companion files with code samples from the book (available for downloading from the publisher)
