Mindluster Platform

About the course Data Structures

Data structure is a storage that is used to store and organize data. It is a way of arranging data on a computer so that it can be accessed and updated efficiently. Depending on your requirement and project, it is important to choose the right data structure for your project .

Programming Category's Courses

Course Lesson(145)

Lesson 1 : **Data Structure Getting Started**

Lesson 2 : **Definition Of Data Structure**

Lesson 3 : **Classification Of Data Structure**

Lesson 4 : **What Is An Algorithm**

Lesson 5 : **Space and Time Complexity of an Algorithm**

Lesson 6 : **Time Complexity Big Oh Notation**

Lesson 7 : **Time Complexity Big Omega Notation**

Lesson 8 : **Time Complexity Big Theta Notation**

Lesson 9 : **Asymptotic Notations GATE Exercise 1**

Lesson 10 : **Asymptotic Notations GATE Exercise 2**

Lesson 11 : **Time Complexity GATE Exercise**

Lesson 12 : **Algorithm Classification Introduction**

Lesson 13 : **Algorithm Classification Simple Recursive Algorithm**

Lesson 14 : **Algorithm Classification Back Tracking Algorithm**

Lesson 15 : **Algorithm Classification Divide And Conquer**

Lesson 16 : **Algorithm Classification Dynamic Programming**

Lesson 17 : **Algorithm Classification Greedy Algorithm**

Lesson 18 : **Algorithm Classification Branch And Bound**

Lesson 19 : **Algorithm Classification Brute Force**

Lesson 20 : **Algorithm Classification Randomized Algorithm**

Lesson 21 : **String Introduction**

Lesson 22 : **String Operations**

Lesson 23 : **String Processing Insert**

Lesson 24 : **String Processing Delete**

Lesson 25 : **String Processing Replace**

Lesson 26 : **Pattern Matching Algorithm Brute Force**

Lesson 27 : **Knuth Morris Patterson KMP Pattern Matching Algorithm**

Lesson 28 : **Example On KMP Pattern Matching Algorithm**

Lesson 29 : **Definition Of Linear Array Data Structure**

Lesson 30 : **Insert Operation On An Array**

Lesson 31 : **Delete Operation On An Array**

Lesson 32 : **Memory Representation Of One Dimensional Array**

Lesson 33 : **Memory Representation Of Two Dimensional Array In Row Major**

Lesson 34 : **Memory Representation Of Two Dimensional Array In Column Major**

Lesson 35 : **Saddle Point Of A Matrix**

Lesson 36 : **Saddle Point Implementation Algorithm**

Lesson 37 : **Lower Triangular Sparse Matrix**

Lesson 38 : **Tridiagonal Sparse Matrix**

Lesson 39 : **Sparse Matrix Another Example**

Lesson 40 : **Array GATE Exercise**

Lesson 41 : **Definition Of Recursion**

Lesson 42 : **Recursion Variations Direct**

Lesson 43 : **Recursion Variations Indirect**

Lesson 44 : **Recursion Variations Nested**

Lesson 45 : **Recursion Variations Excessive**

Lesson 46 : **Recursion Variations Tail**

Lesson 47 : **Recursion Overheads**

Lesson 48 : **Towers of Hanoi as an Example of Recursion**

Lesson 49 : **Recursion GATE Exercise 1**

Lesson 50 : **Recursion GATE Exercise 2**

Lesson 51 : **Linked List Introduction**

Lesson 52 : **Linked List Traversing And Processing**

Lesson 53 : **Linked List Finding Maximum And Minimum Node Info**

Lesson 54 : **Linked List Searching On Unsorted Linked List**

Lesson 55 : **Linked List Searching On Sorted Linked List**

Lesson 56 : **Linked List Insert A New Node As The First Node**

Lesson 57 : **Linked List Insert A New Node After Search Node**

Lesson 58 : **Linked List Deletion Of A Node With Given Info**

Lesson 59 : **Header Linked List Introduction**

Lesson 60 : **Header Linked List Insertion As The First Node**

Lesson 61 : **Circular Header Linked List Introduction**

Lesson 62 : **Two Way Linked List Introduction**

Lesson 63 : **Two Way Linked List Insertion**

Lesson 64 : **Two Way Linked List Deletion**

Lesson 65 : **Reverse Of A Linked List**

Lesson 66 : **Polynomial Representation Using A Linked List**

Lesson 67 : **Polynomial Addition Using Linked List Example**

Lesson 68 : **Polynomial Addition Using Linked List Algorithm**

Lesson 69 : **Stack Introduction**

Lesson 70 : **Stack Operations Introduction**

Lesson 71 : **Stack Operations PUSH POP**

Lesson 72 : **Implementing Two Stacks In One Array**

Lesson 73 : **Implementing Four Stacks In One Array**

Lesson 74 : **Stack Application Evaluation Of Post Fix Expression**

Lesson 75 : **Stack Application Evaluation Of Post Fix Expression Example**

Lesson 76 : **Stack Application Infix to Postfix Conversion**

Lesson 77 : **Stack Application Infix To Postfix Conversion Example**

Lesson 78 : **Stack GATE Exercise**

Lesson 79 : **Queue Operations Introduction**

Lesson 80 : **Queue Operations Insert**

Lesson 81 : **Queue Operations Delete**

Lesson 82 : **Priority Queue**

Lesson 83 : **Binary Tree And Terminologies**

Lesson 84 : **Binary Tree Representation In Memory**

Lesson 85 : **Binary Tree Traversals**

Lesson 86 : **Binary Tree Traversals GATE Exercise 1**

Lesson 87 : **Binary Tree Traversals GATE Exercise 2**

Lesson 88 : **Algorithm Of In Order Traversal**

Lesson 89 : **Algorithm Of Pre Order Traversal**

Lesson 90 : **Algorithm Of Post Order Traversal**

Lesson 91 : **Binary Search Tree Introduction**

Lesson 92 : **Searching An Item In BST**

Lesson 93 : **Insertion Of An Item In BST**

Lesson 94 : **Deletion Of A Node From BST With One Or Zero Child Node**

Lesson 95 : **Deletion of a Node from BST With Two Children**

Lesson 96 : **Deleting A Node From BST**

Lesson 97 : **Ranked Binary Search Tree**

Lesson 98 : **Extended Binary Tree Or 2 Tree**

Lesson 99 : **Huffman s Algorithm With Example**

Lesson 100 : **Huffman s Coding**

Lesson 101 : **Exercise On Huffman Code**

Lesson 102 : **Binary Tree GATE Exercise**

Lesson 103 : **Binary Search Tree GATE Exercise 1**

Lesson 104 : **Binary Search Tree GATE Exercise 2**

Lesson 105 : **AVL Tree A Height Balanced Tree**

Lesson 106 : **AVL Search Tree LL And RR Rotation**

Lesson 107 : **AVL Search Tree RL And LR Rotation**

Lesson 108 : **AVL Search Tree GATE Exercise**

Lesson 109 : **B Tree Definition And Properties**

Lesson 110 : **B Tree An Example**

Lesson 111 : **Heap Tree**

Lesson 112 : **Heap Tree Insertion Operation**

Lesson 113 : **Heap Tree Insertion Operation Example**

Lesson 114 : **Heap Tree Deletion Operation**

Lesson 115 : **Heap Tree Deletion Operation Example**

Lesson 116 : **Heap Sort**

Lesson 117 : **Exercise On Heap Sort**

Lesson 118 : **Bubble Sort**

Lesson 119 : **Modified Bubble Sort**

Lesson 120 : **Insertion Sort**

Lesson 121 : **Selection Sort**

Lesson 122 : **Quick Sort**

Lesson 123 : **Merge Sort**

Lesson 124 : **Execution Tracing Of Merge Sort**

Lesson 125 : **Radix Sort Example**

Lesson 126 : **Radix Sort Algorithm**

Lesson 127 : **Sorting GATE Exercise 1**

Lesson 128 : **Sorting GATE Exercise 2**

Lesson 129 : **External Sorting Introduction**

Lesson 130 : **External Sorting Sample Implementation**

Lesson 131 : **Linear Search**

Lesson 132 : **Linear Search On Sorted Array**

Lesson 133 : **Binary Search**

Lesson 134 : **Trie Structure Algorithm**

Lesson 135 : **Trie Structure Example**

Lesson 136 : **Hashing**

Lesson 137 : **Hashing Collision and Collision Resolution**

Lesson 138 : **Hashing Chaining**

Lesson 139 : **Hashing Chaining Example**

Lesson 140 : **Hashing With Example**

Lesson 141 : **Hashing GATE Exercise**

Lesson 142 : **Tutorix Simply Easy Learning Steps**

Lesson 143 : **Tutorix Brings Simply Easy Learning**

Lesson 144 : **Tutorix Brings Simply Easy Learning**

Lesson 145 : **Tutorix Brings Simply Easy Learning**

Related courses

React Projects Workshops

Node js back end JavaScript

Python programming language

Social Network Theme UI With Sass

Full Stack React Django

Build a Node js App With Sequelize

for Business Contact
business@mindluster.com