Mindluster Platform

About the course Data structure types for beginners

Data structure types course, in this course we will learn about the Data Structure Types, essential for organizing and managing data efficiently in computer programs. Data structures are foundational to programming and software development, providing ways to store and manipulate data in a structured manner. We will explore a variety of data structures, starting with Arrays, which store elements in contiguous memory locations, allowing for efficient indexing and traversal. Next, we will cover Linked Lists, where elements are stored in nodes connected by pointers, providing flexibility in memory allocation. We will delve into Stacks and Queues, which are used for managing data with Last-In-First-Out (LIFO) and First-In-First-Out (FIFO) access patterns, respectively. The course will introduce Trees, including Binary Search Trees and Heaps, which facilitate hierarchical data representation and efficient searching and sorting operations. We will also examine Graphs, which model relationships between objects, and Hash Tables, which provide fast data retrieval using key-value pairs. Advanced data structures like Tries (Prefix Trees), AVL Trees, Red-Black Trees, and B-Trees will be discussed for specialized applications. Additionally, we will explore Sets and Maps for managing collections of unique elements and key-value pairs, respectively. Throughout the course, you will learn about the advantages, use cases, and implementation details of each data structure, along with practical coding examples and exercises. By the end of this course, you will have a comprehensive understanding of various data structure types, enabling you to choose and implement the most appropriate structures for different programming scenarios. Join us to master the art of data organization and elevate your programming skills.

Computer Science Category's Courses

Course Lesson(105)

Lesson 1 : **Definition of data structure what is data structure**

Lesson 2 : **Types of data structure**

Lesson 3 : **abstract data type ADT**

Lesson 4 : **List ADT list adt using array**

Lesson 5 : **Array Operations**

Lesson 6 : **C Program to insert an element into the array at begin position**

Lesson 7 : **C Program to insert an element into the array at specified position**

Lesson 8 : **C Program to insert an element into the array at end position**

Lesson 9 : **C Program to delete an element from the array at begin position**

Lesson 10 : **C Program to delete an element from the array at specified position**

Lesson 11 : **C Program to delete an element from the array at end position**

Lesson 12 : **C Program to update an array element at specified position**

Lesson 13 : **Limitations of arrays in C or Limitations of Static memory allocation**

Lesson 14 :
performance analysis of an algorithm Data Structures Design and analysis of Algorithms DAA

Lesson 15 :
frequency count method to calculate Time Complexity of an Algorithm step count method DS DAA

Lesson 16 :

asymptotic notations in data structure with example Design and analysis of Algorithms DAA

Lesson 17 : **Types of computing times data structures Design and analysis of Algorithms DAA**

Lesson 18 : **General rules or norms for calculating time complexity Data Structures DAA**

Lesson 19 :
Space Complexity of an Algorithm in Data Structures Design and analysis of Algorithms DAA

Lesson 20 : **LINKED LIST CREATION AND DISPLAY DATA STRUCTURES**

Lesson 21 :
LINKED LIST INSERTION AT BEGINNING ENDING SPECIFIED POSITION DATA STRUCTURES

Lesson 22 :
LINKED LIST DELETION FROM BEGINNING ENDING AND SPECIFIED POSITION DATA STRUCTURES

Lesson 23 : **counting number of nodes in a single linked list**

Lesson 24 : **searching a node in singly linked list**

Lesson 25 : **sorting a singly linked list**

Lesson 26 : **reversing a singly linked list in c**

Lesson 27 : **advantages and disadvantages of linked list differences between array and linked list**

Lesson 28 : **CIRCULAR LINKED LIST CREATE AND DISPLAY DATA STRUCTURES**

Lesson 29 :
CIRCULAR LINKED LIST INSERTION AT BEGINNING ENDING SPECIFIED POSITION DATA STRUCTURES

Lesson 30 :
CIRCULAR LINKED LIST DELETION FROM BEGINNING ENDING AND SPECIFIED POSITION DATA STRUCTURES

Lesson 31 : **advantages and disadvantages of circular linked list**

Lesson 32 : **DOUBLE LINKED LIST CREATE AND DISPLAY DATA STRUCTURES**

Lesson 33 :
DOUBLE LINKED LIST INSERTION AT BEGINNING ENDING SPECIFIED POSITION DATA STRUCTURES

Lesson 34 :
DOUBLE LINKED LIST DELETION FROM BEGINNING ENDING AND SPECIFIED POSITION DATA STRUCTURES

Lesson 35 : **Reversing Double Linked List**

Lesson 36 : **advantages and disadvantages of doubly linked list**

Lesson 37 : **Circular Double Linked List Create and Display**

Lesson 38 : **Circular Double Linked List Insertion Begin end and Specified Position**

Lesson 39 : **Circular Double Linked List Deletion Begin end and Specified Position**

Lesson 40 : **Applications of Linked List data structure**

Lesson 41 : **polynomial adt polynomial addition in data structure**

Lesson 42 : **stacks using arrays**

Lesson 43 : **Stacks using linked list**

Lesson 44 : **arithmetic expression in data structure applications of stacks**

Lesson 45 : **postfix expression evaluation algorithm example program**

Lesson 46 : **infix to postfix conversion algorithm examples program**

Lesson 47 :
balancing symbols using stack algorithm example program delimiter matching using stack

Lesson 48 : **queues using arrays**

Lesson 49 : **Queues using linked list**

Lesson 50 : **circular queues in data structures**

Lesson 51 : **double ended queue using array**

Lesson 52 : **double ended queue using linked list**

Lesson 53 : **Linear Search**

Lesson 54 : **Binary Search**

Lesson 55 : **bubble sort with example**

Lesson 56 : **Bubble Sort algorithm bubble sort program bubble sort example**

Lesson 57 : **selection sort with example**

Lesson 58 : **Selection Sort algorithm selection sort example selection sort program**

Lesson 59 : **insertion sort with example**

Lesson 60 : **Insertion Sort algorithm insertion sort program insertion sort example**

Lesson 61 : **merge sort algorithm merge sort algorithm with example merge sort program**

Lesson 62 : **MERGE SORT WITH EXAMPLE**

Lesson 63 : **analysis of merge sort merge sort time complexity analysis**

Lesson 64 : **quicksort algorithm quicksort program quicksort example step by step**

Lesson 65 : **quicksort example step by step quick sort example solved**

Lesson 66 : **analysis of quicksort algorithm quicksort time complexity analysis**

Lesson 67 : **radix sort algorithm bucket sort algorithm program example step by step**

Lesson 68 : **radix sort example step by step bucket sort example step by step**

Lesson 69 : **shell sort example step by step shell sort example in data structure**

Lesson 70 : **shell sort algorithm shell sort program in c shell sort with example**

Lesson 71 : **tree terminology in data structure**

Lesson 72 : **what is binary tree types of binary tree**

Lesson 73 : **representation of binary tree**

Lesson 74 : **binary tree traversals**

Lesson 75 : **C program to implement Binary Tree Traversals Data Structures**

Lesson 76 : **construct binary tree from inorder and preorder traversal examples**

Lesson 77 : **construct binary tree from inorder and postorder traversal examples**

Lesson 78 : **heap sort algorithm with example heap sort example step by step**

Lesson 79 : **Heap Sort Program explanation with an example**

Lesson 80 : **construction of expression tree using postfix expression Data Structures**

Lesson 81 : **what is binary search tree list of binary search tree operations**

Lesson 82 : **construction of binary search tree with example binary search tree example step by step**

Lesson 83 : **insertion of a node in binary search tree binary search tree insertion**

Lesson 84 : **deletion of a node in binary search tree binary search tree node deletion**

Lesson 85 : **searching a node in binary search tree binary search tree search operation**

Lesson 86 : **find smallest element in binary search tree Find min element in a binary search tree**

Lesson 87 : **find largest node in binary search tree**

Lesson 88 : **what is an avl tree avl tree rotations**

Lesson 89 : **Construct AVL tree for the elements 1 2 3 4 5 6 7 8 AVL Tree Insertion with example**

Lesson 90 : **construct AVL tree for 20 11 5 32 40 2 4 27 23 28 50 avl tree insertion with example**

Lesson 91 :
Construct AVL Tree for 7 14 2 5 10 33 56 30 15 25 66 70 4 AVL Tree insertion with example

Lesson 92 : **graph terminology in data structure**

Lesson 93 : **representation of graph in data structure**

Lesson 94 : **Graph Traversals Breadth First Search BFS**

Lesson 95 : **Graph traversal Depth first search DFS**

Lesson 96 : **what is minimum cost spanning tree**

Lesson 97 : **Prim s Algorithm for Minimum Cost Spanning Tree**

Lesson 98 : **Kruskal s Algorithm for Minimum Cost Spanning Tree**

Lesson 99 :
Hashing Hash Table Hash Function Types of hash functions Characteristics of a good hash function

Lesson 100 :
collision resolution techniques Separate Chaining open addressing linear probing Quadratic Double

Lesson 101 : **Sorting Techniques Algorithm Program Example Tracing Analysis Time Complexity**

Lesson 102 : **Algorithm Analysis Linked List Single Circular Double Circular Double Operations DS**

Lesson 103 : **Stacks and Queues Tutorial**

Lesson 104 : **Trees Graphs and Hashing in Data Structures**

Lesson 105 : **Rehashing in Data Structures Collision Resolution Technique**

Related courses

Linear Algebra for Computer Scientists

Insertion Sort

Bubble Sort



for Business Contact
business@mindluster.com