GeeksforGeeks

MASTER C PROGRAMMING

Complete Beginner to Advanced

Detailed
Course Syllabus

CONTENTS

C - BASICS TO ADVANCED

INTRODUCTION

- Analysis of Algorithms
- Asymptotic Notation
- Big 0 notation
- Omega notation
- Theta notation
- Analysis of loops
- Time Complexity and Space Complexity

RECURSION

- Introduction
- Application
- Recursion practices
- Tail Recursion
- Writing Base Cases
- Print 1 to n and n to 1 using recursion

ARRAYS

- Introduction
- Operations on array
- Average of an array
- Maximum in array
- Second largest in array
- Check if array is sorted
- Reverse an array
- Rotate an array

SEARCHING

- Linear Search and its analysis
- Binary Search and its analysis
- Index of first and last occurrence
- Count occurrences in a sorted array

CONTENTS

OPERATORS

- Arithmetic Operators
- Unary Arithmetic Operators
- Comparison Operators
- Assignment Operators
- Bitwise Operator inC (AND, OR and XOR)
- Bitwise Operator inC (Left Shift, Right Shift and NOT)
- Signed Number Representation and Bitwise Not
- Operator Precedence &Associativity
- Day before Ndays
- Sum of N Natural Numbers
- Last Digit of a Number

FLOW CONTROL

- If Else in C
- If Else Examples in C
- Else if with Example in C
- Switch inC
- Even Odd Game
- Largest of ThreeNumbers
- Leap Year
- Simple Calculator

FUNCTION IN C

- Functions inC
- Application of Functions
- Function Declaration & Definition
- How FunctionWorks
- Inline Function
- Practice Problems on C Functions
- Recursion Practice Questions
- First Digit of a Number
- Prime Factorization

LOOPS

- While Loop in C
- For Loop in C

CONTENTS

- Do While Loop in C
- Break inC
- Continue inC
- Nested Loop in C
- Pattern
- Square Pattern
- Triangle Pattern
- Inverted Triangle Pattern
- Factor of Number
- Check for Prime
- Next Prime Number
- All Divisors of a Number
- GCD & LCM of Two numbers
- Fibonacci Numbers
- Count Digits of a Number
- Table of a Number

ARRAY

- Introduction to Array in C
- Declaring and Initializing Arrays
- Accessing Array Element
- Size of an Array in C
- Array Traversal in C
- Different Types of Arrays
- Check if Array is Sorted
- Count Distinct in an array
- Sum of an Array
- Average of and Array
- Maximum in an Array

POINTERS

- Address and DereferenceOperators in C
- Introduction to Pointers in C
- Application of Pointers in C
- Function Parameters and Pointers
- Array Parameters andPointers
- Pointer Arithmetic
- Void Pointer in C
- NULL inC
- Pointer vsArrays
- Pointer to Pointer in C

CONTENTS

DYNAMIC MEMORY ALLOCATION

- Memory Structure of a Program
- malloc(), calloc() and free()
- Memory Leak

STRING IN C

- String in C (Introduction)
- String Syntax, Size and Length in C
- String Comparison inC
- String Copy inC
- String Concatenation in C
- Pattern Searching
- Strncat(), strncmp(), and strncpy()
- Substring search inC
- String Tokenization inC
- Reverse aString
- Check for palindrome
- String Binary to Decimal
- String Decimal to Binary

MULTI DIMENSIONAL ARRAY IN C

- Multi Dimensional Array in C
- Passing 2D Arrays as Argument To Functions
- Transpose of a Matrix
- Matrix Multiplication

STRUCTURE AND UNION

- Struct inC
- Structure Variable Initialization
- Structure Arrays
- Structure Pointer
- Structure Alignment
- Reason for Structure Alignment in C
- Union inC
- Application of Union

CONTENTS

ADVANCED

- Function Pointers in C
- Passing Function to Pointers
- File Handling in C
- Read froma File
- Write to aFile

DATA STRUCTURES IN C

INTRODUCTION

- Analysis of Algorithms
- Asymptotic Notation
- Big 0 notation
- Omega notation
- Theta notation
- Analysis of loops
- Time Complexity and Space Complexity

RECURSION

- Introduction
- Application
- Recursion practices
- Tail Recursion
- Writing Base Cases
- Print 1 to n and n to 1 using recursion

ARRAYS

- Introduction
- Operations on array
- Average of an array
- Maximum in array
- Second largest in array
- Check if array is sorted
- Reverse an array
- Rotate an array

CONTENTS

SEARCHING

- Linear Search and its analysis
- Binary Search and its analysis
- Index of first and last occurrence
- Count occurrences in a sorted array

SORTING

- Bubble Sort
- Selection Sort
- Insertion Sort
- Merging two sorted arrays
- Introduction of merge sort, algorithm and analysis
- Partitioning an array
- Lomuto Partitioning and Haore’s Partitioning
- Implementation of Quick Sort and analysis

MATRIX

- Introduction and Passing 2D arrays as arguments
- Matrix boundary traversal
- Matrix in snake pattern
- Transpose of a matrix
- Spiral traversal of matrix
- Searching in row-wise and column-wise sorted matrix

HASHING

- Concept of hashing
- Direct Address Table
- Collision Handling
- Chaining
- Open addressing
- Double Hashing

STRINGS

- Introduction
- Escape sequences
- Reverse a string
- String Comparisons
- Operations on String
- Pattern Searching
- Check for Anagram
- Check for Palindrome

CONTENTS

LINKED LIST

- Introduction
- Implementation and Applications
- Traversal of Linked List
- Insertion at beginning and end in Linked List
- Sorted insert in Linked List
- Delete first and last node of Linked List
- Reverse a linked list.

DOUBLY LINKED LIST

- Introduction
- Advantages and Disadvantages
- Insertion at beginning and end in Doubly Linked List
- Delete first and last node of Doubly Linked List
- Reverse a Doubly Linked List

CIRCULAR LINKED LIST

- Introduction
- Advantages and Disadvantages
- Insertion at beginning and end in Circular Linked List
- Delete head and Kth node of Circular Linked List.

STACK

- Introduction
- Array implementation
- Linked List implementation
- Prefix, Infix and Postfix expressions, their conversion and evaluation.

QUEUE

- Introduction
- Implementation using linked list
- Insertion in queues, Deletion in queues
- Implementing stack using queues and vice versa
- Circular queues Introduction and applications
- Implementing using array and linked list.

CONTENTS

DEQUE

- Introduction
- Applications and array implementation.

TREES

- Introduction of Trees
- Applications
- Binary Tree
- Traversal of Tree
- Implementation of Preorder, Inoreder and Postorder traversal
- Iterative Inorder and Preorder

BINARY SEARCH TREES

- Introduction
- Insert, Ceil, Floor and Search in BST.

HEAP

- Introduction
- Implementation of Heap
- Binary Heap(Heapify and Extract)
- Binary Heap(Decrease Key, Build Heap and Delete)

GRAPH

- Introduction
- Representation – Adjacency List and Adjacency Matrix
- Implementation of Adjacency List
- Application of BFS and DFS