

Fundamentals of Computer Programming (Theory)

Teaching Scheme

Lectures: 3 hours/week

Practical: 3 hours/week

Examination Scheme

Test (I and II) and Mid-sem: 50

End-sem: 50

Objectives

1. Introduction to computer technology with emphasis on computer programming
2. Building logical approach to solve practical problems using computers

Unit I

(4 hrs)

Introduction

1. Introduction to computers

- Brief history of computers
- Computer hardware: various parts such as CPU, input/output devices etc.
- Types of computers: single user (PC), multi-user/client-server etc.
- General concepts related to software
 - System software, application software
 - Free and Open Source vs. Proprietary software
 - World-wide Web, e-mail, office applications, newsgroups etc.

2. Overview of programming environments

- Example of a popular proprietary operating system such as Microsoft Windows
- Example of a popular open source operating system such as GNU/Linux

Unit II

(10 hrs)

'C' language syntax and basic problem solving

1. Introduction to following 'C' language constructs:

- Basic program structure of a typical 'C' program
- I/O statements
- Variables and Constants
- Operators
- Control flow statements and blocks: if-else, switch
- Loop statements: for, while, do-while, break and continue

2. Application of the above constructs to solve the following elementary programming problems

- Exchanging the values of two variables
- Summation of a set of numbers
- Sine function computation

- Generation of the Fibonacci sequence
- Finding the square root of a number
- The greatest common divisor of two integers
- Generating prime numbers
- Raising a number to large power

Unit III

(8 hrs)

Array Techniques, Functions and Problem solving using Arrays and Functions

1. 'C' language syntax related to Arrays and Functions

- Defining array variables (one, two and multi-dimensional)
- Defining and calling functions
- Global, local and static variables

2. Solving problems using Arrays and Functions

- Finding the maximum number in a set
- Array order reversal
- Removal of duplicates from an ordered array
- Selection sort
- Bubble sort

Unit IV

(10 hrs)

Introduction to Data Structures

1. 'C' syntax for following constructs/features

- Pointers and addresses
- Pointers and dynamic arrays
- Structures and Unions
- File handling

2. Programs using data structures

- Create a linked list
- Insert element into an existing linked list
- Delete element(s) from a linked list
- Traverse a linked list
- Reading (writing) structures from(to) a file

Unit V

(4 hrs)

Recursive Algorithms

- Concept of Recursion
- Factorial computation
- Tower of Hanoi problem
- Finding (n, r) (Combinations of n things taken r at a time)
- Finding maximum number from an array of numbers

Unit VI

(4 hrs)

Introduction to Object Oriented Programming paradigm

1. Concepts related to Object Oriented Programming paradigm
 - Classes, Objects
 - Private and Public variables and methods
 - Inheritance
2. Sample programs using OO Programming language (C++/Java)

Text Books

1. How to solve it by Computer by R.G. Dromey, Pearson Education
2. Programming in ANSI C by E. Balguruswamy, Tata Mc-Graw Hill
3. Object-Oriented Programming With C++ By E. Balaguruswamy, Tata Mc-Graw Hill

Reference Books

1. The 'C' programming language by Kernighan and Ritchie, Prentice Hall
2. Computer Programming in 'C' by V. Rajaraman , Prentice Hall

Fundamentals of Computer Programming (Laboratory)

List of Experiments

Instructors will have flexibility in framing assignments so as to cover all the topics discussed in the class. The programs mentioned in the theory syllabus are listed below for ready reference:

- Exchanging the values of two variables
- Summation of a set of numbers
- Sine function computation
- Generation of the Fibonacci sequence
- Finding the square root of a number
- The greatest common divisor of two integers
- Generating prime numbers
- Raising a number to large power
- Finding the maximum number in a set
- Array order reversal
- Removal of duplicates from an ordered array
- Selection sort
- Bubble sort
- Create a linked list
- Insert element into an existing linked list
- Delete element(s) from a linked list
- Traverse a linked list
- Reading (writing) structures from(to) a file
- Factorial computation
- Tower of Hanoi problem
- Finding (n, r) (Combinations of n things taken r at a time)
- Finding maximum number from an array of numbers
- Sample programs using OO Programming language (C++/Java)