

SEMESTER-I

Core I Introduction to Computer Science and Python Programming

Course Outcomes:

- To learn the fundamentals of computer and its working mechanism.
- To learn programming paradigms and design.
- To learn the basics of Python Programming for problem solving.

Learning Outcomes:

Upon completion of this course, students will be able to:

1. gain knowledge about the components of a computer with their functions
2. have an exposure to various computer networks
3. learn the steps for program development
4. learn the use of control structures, built-in functions, lists, recursions in Python programming

Unit-I:

Introduction to Computers: Generation of computers, types of computers. Computer System hardware: Block diagram, CPU, Memory unit, and Storage devices with their functions. Computer memory: Memory hierarchy, CPU Registers, Cache Memory, Primary Memory, Secondary memory, access methods.

Input & Output Units: Various types of Input and Output devices with their functions.

Computer Software: Introduction, Types of software: System software, Application software, Utility software, firmware and their usage. Computer Network: Importance of Networking, Network types: LAN, MAN, WAN, Internet and its applications, network devices: Repeater, bridge, hub, switch, router, and Gateway.

Unit-II:

Program Development: Problem Analysis, program design, development. Algorithm: Introduction, Pseudo code, Control Structures. Flowchart: Introduction, Symbols, preparing a flowchart. Programming Paradigms & Design: Structured programming, Object-oriented programming, Top-down and Bottom-up design approaches. Characteristics of a good program.

Programming Languages: Concept of Low-level & high-level languages, Different generations of Programming Languages. Translators: Assembler, Compiler, Interpreter. Linker, Loader, Editors, Integrated Development Environment. Programming Errors: Syntax, Semantic, Logical, Compile-time, run-time, link-time, environmental, Input/output errors. testing and debugging.

Unit-III:

Introduction to Python, getting started with Python, Python Basics: Identifiers, Keywords, Python types, basic types, mutable and immutable types, Integer & float ranges, Variable type & assignment, Arithmetic Operators, Precedence & Associativity, Conversions, built-in functions, modules, container types, comments & indentation, multi-lining, classes & Objects, Multiple Objects. Strings: Introduction, Accessing String elements, Properties, built-in functions, Methods, Conversions, Comparisons.

Console I/O: I/O operations, formatted printing. Decision Control Instruction: Logical operators, Conditional Expressions, all () & any (), receiving input, pass statement. Repetition Control Instruction: types, usage of loops, break & continue, else block of a loop.

Unit-IV:

Lists, Sets, Tuples, Dictionaries: creating, accessing, and looping-in each type. Applying basic operations, using built-in functions and methods on each type, possible data structure / mathematical operations on each type.

Comprehensions on List, Set, and dictionary. Functions: built-in and user-defined functions, invoking functions, unpacking arguments. recursive function, iteration vs recursion.

Text Books:

- ✓ *Computer Fundamentals by Anita Goel, Pearson Pub.*
- ✓ *Let us Python by Yashavant Kanetkar & Aditya Kanetkar, BPB Pub.*

Reference Books & e-Resources:

- ✓ *Fundamentals of Computers by Reema Thareja, Oxford University Press*
- ✓ *Python Programming: Using Problem Solving Approach by Reema Thareja, Oxford University Press*
- ✓ *Computer Fundamentals by D. P. Nagpal, S. Chand Pub.*
- ✓ <https://docs.python.org/3/>

Core-I- Lab: Programming using Python

1. Write a program to demonstrate the usage of various arithmetic operators.
2. Write a program that makes use of trigonometric functions available in math module.
3. Write a program that will convert various temperatures.
 - a. Fahrenheit to Centigrade
 - b. Centigrade to Fahrenheit
4. Write a program that will find the roots of a quadratic equation: $ax^2 + bx + c = 0$
5. Write a program that demonstrate the usage of various String functions.
6. Write a program that will ask you to enter your name, through keyboard, and perform following operations

- a. Find the middle name
 - b. Find the last name (using string slicing)
 - c. Re-write the name with surname first.
7. Write a program to find out whether the integer entered by the user, through the keyboard, is even or odd number.
 8. Find out the youngest among Shyam, Dugu and Ishan whose ages are entered by the user through keyboard.
 9. Given three points (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , write a program to check all the three points fall on one straight line.
 10. Write a program that will print the odd numbers from n_1 to n_2 where the values of n_1 and n_2 are entered by the user.
 11. Write a program to find the factorial value of a number entered by the user.
 12. Write a program to print all prime numbers between n_1 to n_2 where the values of n_1 and n_2 are entered by the user.
 13. Write a program to demonstrate basic operations on the list.
 14. Write a program to demonstrate stack and queue operations using a list of numbers.
 15. Write a program to ask the data of five students that contain name, roll number, age. Sort the list based on roll number of the Student. [Note: Use list of lists].
 16. Write a program that will add two square matrices. The dimension and elements of the matrices will be entered by the user.
 17. Write a program to demonstrate basic operations on the tuple.
 18. Store the data about the shares held by the user as tuples containing the following information about shares: share name, cost price, number of shares, selling price. Write a program to determine:
 - a. total cost of the portfolio
 - b. total amount gained or lost
 19. Write a program to demonstrate basic operations on the set.
 20. Create an empty set. Write a program that adds five student names to this set, modifies one existing name, and deletes two names existing in it. [ask the user which name to modify/delete].
 21. Write a program to demonstrate basic operations on the dictionary.
 22. Create a dictionary to store data (name, roll number) of N students. The key will be the roll number of the student and the value contains the data of the student (in a list). Write a program that asks the user to enter a name of a Student, search it in the dictionary and print the data of the Student if it is available otherwise display an appropriate message.
 23. Write a program to demonstrate basic comprehensions on list, set and dictionary.
 24. Write a program that will find x^n (x to the power of n) using a function. The function receives the value of x , n and should return the value of x^n . [don't use any mathematical function].