

B. Tech I Year I Semester

20CSE101 PROGRAMMING FOR PROBLEM SOLVING(PYTHON)

L	T	P	C
2	0	2	3

Course Prerequisite: None

Course Description:

Python is a language with a simple syntax, and a powerful set of libraries. It is an interpreted language, with a rich programming environment. While it is easy for beginners to learn, it is widely used in many scientific areas for data exploration. This course is an introduction to the Python programming language for students without prior programming experience. This course provides knowledge on how to implement programs in python language and to solve computational problems using the various programming constructs including data structures, functions, string handling mechanisms and file handling concepts.

Course Objectives:

1. Learn Python programming constructs.
2. Implement Python programs with conditional structures and loops.
3. Use functions for structuring Python programs.
4. Handle compound data using Python lists, tuples, and dictionaries.
5. Manipulate data using files handling in Python.

UNIT-I

Introduction: Algorithms, building blocks of algorithms (flow chart), History of Python, features of Python Programming, Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation. **Data Types** - Integers, Strings, Boolean.

- a) Develop a flowchart for the various arithmetic operations on numbers.
- b) Develop a flowchart to check whether the number is positive or negative.
- c) Develop a flowchart for finding whether a given number is even or odd.
- d) Develop a flowchart for finding biggest number among three numbers.
- e) Develop a flowchart for displaying reversal of a number.
- f) Develop a flowchart to print factorial of a number using function.
- g) Develop a flowchart to generate prime numbers series up to N using function.
- h) Develop a flowchart to check given number is palindrome or not using function.
- i) Alexa travelled 150 kms by train. How much distance in miles she actually covered?

(12)

UNIT-II

Operators and Expressions: Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations. **Control Flow** - if, if-elif-else, for, while, break, continue, pass.

- a) Swapping of two number with and without using temporary variable.
- b) If the age of Ram, Sam, and Khan are input through the keyboard, write a python program to determine the eldest and youngest of the three.

- c) Develop a program that performs arithmetic operations (Addition, Subtraction, Multiplication, and Division) on integers. Input the two integer values and operator for performing arithmetic operation through keyboard. The operator codes are as follows:
- For code '+', perform addition.
 - For code '-', perform subtraction.
 - For code '*', perform multiplication.
 - For code '/', perform division.
- d) Implement the python program to generate the multiplication table.
- e) Implement Python program to find sum of natural numbers
- f) If the first name of a student is input through the keyboard, write a program to display the vowels and consonants present in his/her name.
- g) The marks obtained by a student in 5 different subjects are input through the keyboard. Find the average and print the student grade as per the MITS examination policy as shown below.

% OBTAINED	GRADE
90 - 100	O (Outstanding)
80 - 89	A+ (Excellent)
70 - 79	A (Very Good)
60 - 69	B+ (Good)
50 - 59	B (Above)
45 - 49	C (Average)
40 - 44	P (Pass)
< 40	F (Fail)

- h) Implement Python Script to generate prime numbers series up to N.
- i) Given a number x, determine whether it is Armstrong number or not. Hint: For example, 371 is an Armstrong number since $3^3 + 7^3 + 1^3 = 371$. Write a program to find all Armstrong number in the range of 0 and 999.

(12)

UNIT-III

Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions. **Functions** - Defining Functions, Calling Functions, Passing Arguments, variable in python-Global and Local Variables.

- a) Write a Python script to
- create a list
 - access elements from a list
 - slice lists
 - change or add elements to a list
 - delete or remove elements from a list
- b) Write a Python script to read the values from a list and to display largest and smallest numbers from list.
- c) Write a Python script to compute the similarity between two lists.
- d) Write a Python script to read set of values from a Tuple to perform various operations.
- e) Write a Python script to perform basic dictionary operations like insert, delete and display.

- f) Write a Python program to count the occurrence of each word in a given sentence.
 g) Define a dictionary named population that contains the following data.

Keys	Values
Shanghai	17.8
Istanbul	13.3
Karachi	13.0
Mumbai	12.5

- h) Write a Python script to create Telephone Directory using dictionary and list to perform basic functions such as Add entry, Search, Delete entry, Update entry, View and Exit.
 i) Implement Python script to display power of given numbers using function.
 j) Implement a Python program that takes a list of words and returns the length of the longest one using function. **(12)**

UNIT-IV

String Handling -Modules: Creating modules, import statement, from.import statement, name spacing-**Files and Directories**

- a) Implement Python program to perform various operations on string using string libraries.
 b) Implement Python program to remove punctuations from a given string.
 c) Write a Python program to change the case of the given string (convert the string from lower case to upper case). If the entered string is “computer”, your program should output “COMPUTER” without using library functions.
 d) Implement Python program to capitalize each word in a string. For example, the entered sentence “god helps only people who work hard” to be converted as “God Helps Only People Who Work Hard”
 e) Write a Python script to display file contents.
 f) Write a Python script to copy file contents from one file to another.
 g) Write a Python script to combine two text files contents and print the number of lines, sentences, words, characters and file size.
 h) Write a Python commands to perform the following directory operations.
- List Directories and Files
 - Making a New Directory
 - Renaming a Directory or a File
 - Removing Directory or File

(12)

UNIT-V

Python packages, Introduction to PIP, Installing Packages via PIP(Numpy, Pandas etc..., Using PythonPackages. **Brief**

Tour of the Standard Library - Dates and Times, Data Compression, Turtle Graphics.

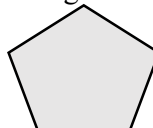
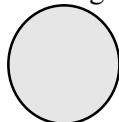
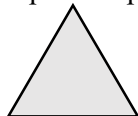
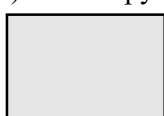
(10)

4a) Create a package named Cars and build three modules in it namely, BMW, Audi and Nissan. Illustrate the modules using class. Finally we create the `__init__.py` file. This file will be placed inside Cars directory and can be left blank or we can put the initialization code into it.

b) Create a class by name Student with instance variables such as `roll_no`, `name`, `year_of_study`, `branch`, `section`, and marks in any five subjects. The class should also contain one method for calculating the percentage of marks and the other method for printing a report as follows:

Roll No.	Name	Year	Section	Branch	M1	M2	M3	M4	M5	Percentage
101	abc	I	A	CSE	58	68	95	47	56	64.8

c) Write a python script to display following shapes using turtle.



(12)

Course Outcomes:

At the end of the course, students will be able to

1. Understand problem solving techniques and their applications
2. Understand the syntax and semantics of python.
3. Demonstrate the use of Python lists and dictionaries.
4. Demonstrate the use of Python File processing, directories.
5. Describe and apply object-oriented programming methodology and Standard Library.

Text Book:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016 (<http://greenteapress.com/wp/think-python/>)
2. Guido van Rossum and Fred L. Drake Jr, "An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

References:

1. Charles Dierbach, "Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
2. John V Guttag, "Introduction to Computation and Programming Using Python", Revised and expanded Edition, MIT Press , 2013.
3. Kenneth A. Lambert, "Fundamentals of Python: First Programs", CENGAGE Learning, 2012.
4. Paul Gries, Jennifer Campbell and Jason Montojo, "Practical Programming: An Introduction to Computer Science using Python 3", Second edition, Pragmatic Programmers,LLC,2013.
5. Robert Sedgewick, Kevin Wayne, Robert Dondero, "Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.

Mode of Evaluation: Model Lab Examinations, External Lab End Examination.

