Guidelines for Problem Solving using Computers (BSCS01)
B. Sc. Prog. Computer Science

Topics	Chapters	References
Unit I Computer Fundamentals and Problem Solving: Basic Computer Organization: CPU, memory, I/O Units. Problem solving using computer, notion of an algorithm.	Chapter 1 [1.1,1.2], Chapter 2[2.2]	Reference [2]
Unit II Introduction to Python Programming: Python interpreter, using python as calculator, python shell, indentation, identifiers and keywords, literals, strings, arithmetic, relational and logical operators. **Unit III Creating Python Programs:** Creating Python Programs: Input and output statements, defining functions, control statements default arguments, errors and exceptions	Chapter1 [pg 10-19], Chapter 2, Chapter 3 ,Chapter 4[till pg 123] Chapter 8 Chapter 2[2.4]	Reference [1] Reference [3]
Unit IV Inbuilt Data Structures: strings, lists, sets, tuples, nested lists, built-in functions, dictionary and associated operation	Chapter 6,7	Reference [3]
Unit V Object Oriented Programming: Introduction to Classes, Objects and Methods, Standard Libraries, File handling through libraries	Chapter 7 [till pg 217] Chapter 14[pg 368-379]	Reference [1]
Unit VI Sorting and Searching: Iterative and Recursive methods for searching and sorting	Chapter –13[till pg355] Chapter 12[12.1.1, 12.1.3,12.2]	Reference [1] Reference [3]

* Python 3.6 recommended

References

1. Urban, M. & Murach, J. (2018). Python Programming. Shroff.

2. Liang, Y. D. (2013). Introduction to Programming using Python. Pearson.

3. Taneja, S. & Kumar, N. (2018). Python Programming - A modular Approach. Pearson.

List of Practicals:

Use of functions and exception handling to be encouraged wherever applicable.

1. Execution of expressions involving arithmetic, relational, logical operators

2. Write a Python function to produce the following outputs.

(a)
```
*
**
***
****
*****
******
*******
```

(b)
```
$ $ $ $ $
$       $
$       $
$       $
$ $ $ $ $
```

3. Write a Python program to calculate the roots of a quadratic equation. Use suitable functions from math.h

4. Write a Python program to generate a table of sins, cosines and tangents. Make a variable x in range from 0 to 2π in steps of $(\pi/4)$. For each value of x, print the value of sin(x), cos(x) and tan(x).

5. Write a menu driven program to invoke functions to calculate the area of square, rectangle, circle and triangle. Use suitable assertions.

6. Write a Python function that takes a number as an input from the user and computes its factorial (using recursion). Then find the sum of the n terms of the following series:

$$1 - 1/1! + 1/2! - 1/3! + ... + 1/n$$

7. Write both iterative and recursive implementation of Python function to return n^{th} terms of Fibonacci sequence.

8. Write a function that takes a number as an input and finds its reverse and computes the sum of its digits.

9. Write a function that takes two numbers as input parameters and returns their least common multiple.

10. Write a function that takes a number as an input and determine whether it is prime or not. Use this function to display all prime numbers till the provided number n.

11. Write a menu driven program using in-built string functions to do the following tasks. Menu should be displayed and user must be prompted to enter choice. Repeat this sequence till user enters exit option.

MENU

1. Look for a substring in the given string and returns its position

2. Replace substring 'good' with 'best' in the given string

3. Find all the substring in the string which are separated by delimiter ;

4. Convert the given text into title form

5. Convert lowercase to uppercase and uppercase to lowercase in the given string

6. Exit

12. Define a class STUDENT to store his/her name, rollno and marks in three subjects and a class variable to store maximum of average marks of the class. Use constructor/destructor to initialize/delete the values to data members and define following user-defined function members:

   i. Display the member's values with average mark

   ii. Display the maximum average marks of the students

13. Write a function that reads a text file and calculates the frequency of vowels. Use a variable of dictionary type to maintain the count.

14. Write a Python function that prints a dictionary where the keys are numbers between 1 and 5 and the values are cubes of the keys.

15. Consider a tuple t1=(1,2,5,7,9,2,4,6,8,10). Write a program to perform following operations:

   a.   Print half the values of tuple in one line and the other half in the next line.

   b.   Print another tuple whose values are even numbers in the given tuple.

   c.   Concatenate a tuple t2= (11, 13, 15) with t1.

   d.   Return maximum and minimum value from this tuple.

16. Write a menu driven program to do following using functions:

   • Selection sort

   • Insertion sort

17. Write a menu driven program to perform the following using functions:

   • Linear search

   • Binary search

18. Write a program to implement a class for finding area and perimeter of a rectangular playground. Write constructor, destructor, and functions for calculating area and perimeter.

19. Write a program to perform the following functionality using csv files

   a. Create a csv file for maintain student records containing Name and total marks obtained

   b. Read the file created above to display the details of every third student

20. Perform the following functions using lists:

   1. Find whether all elements in list are numbers or not

   2. If numeric list then count number of odd values in it

   3. if string list then display the largest string in the list

4. if all elements are string then count numeric string and string with alphabets only

21. Write a program that accepts two strings and perform the following using sets

    a. Convert each string into separate set
    b. Identify and display the common characters between the two sets
    c. Identify and display the distinct characters between the two sets