

Unique Paper Code : 42341102-OC
Name of the Course : B.Sc. Prog. / Mathematical Science (CBCS)
Name of the Paper : Problem Solving using Computers
Semester : I
Year of Admission : 2015, 2016, 2017, 2018

Duration: 3 Hours

Maximum Marks: 75

Instructions for the candidates:

1. All questions carry equal marks.
2. Attempt any four questions.

1. (a) Define a class Rectangle that has the two data members Length and Breadth and three methods given below:
 - Constructor to initialize the data members Length and Breadth.
 - Area – This computes the area of the rectangle.
 - Perimeter- This computes the perimeter of the rectangle.
 - `__str__` function to generate a string that may be used to display the data members of the rectangle.

Also, write Python statements to perform the following:

- Create an object of the class Rectangle with suitable values.
- Invoke the method Area
- Print all the data members of the class

(b) Find and correct the errors in the code snippet given below:

```
DEF execmain()  
    x = input("Enter a number:")  
    if (abs(x) = x):  
        print("You entered a positive number")  
    else
```

```

x=*- 1
print "Number made positive:"x

```

2. (a) Check the validity of variable names with respect to the naming conventions of Python. Justify your answer.

- new_var
- _newvar
- lvar
- new%var
- new var

- (b) Give an algorithm for each of the following problems:

- To compute whether a number is a prime number or not.
- To determine whether a string is a palindrome.

- (c) Give the output corresponding to the statements mentioned in the following code segment.

```

a = 35
b = 17
c = a * b
print(c) #Statement 1
c = a | b
print(c) #Statement 2
c = a ^ b
print(c) #Statement 3
c = ~b
print(c) #Statement 4
print(b<<3) #Statement 5
print(a>>2) #Statement 6

```

3. a) Apply binary search to search key value 89 in the following list of numbers. Show all the intermediate steps required to do the search.

```
[8, 12, 39, 45, 57, 67, 78, 89, 91]
```

- b) Apply the selection sort to arrange the following list of elements in the ascending order. Show all the intermediate steps of each pass. Determine the

number of passes to sort the entire list.

[21, 71, 43, 10, 60, 37, 91, 2, 55, 42]

c) Evaluate the following expressions and justify your answers:

I. $9 * 2 - 3 ** 2 // 6 + 5$

II. $10 < 5$ or $7 < 12$ and not $1 > 3$

III. `'hello' < 'hi'` or `'Iamfine' > 'I am not'`

4. a) Write a program to print the sum of the first n terms of the following series:

$1/3 + 1/6 - 1/9 + 1/12 + \dots$

b) Define a function that accepts a number n and a string `mystr` as input arguments and returns a list `lst` containing n elements. Each element of the list must be a string repeated i times where i ranges from 1 to n . For example, if the arguments are 3 and 'Hi' then the function should return ['Hi', 'HiHi', 'HiHiHi'].

c) Define a function `count` that accepts a string `s` as an input and counts the vowels, consonants, and spaces that occur in `s` and print their counts in the result.

d) Define a recursive function `fibonacci` that accepts i as an input and returns the i th term of the Fibonacci series. Define a function `main` that accepts a number n and invokes the function `fibonacci` to display the fibonacci series upto n terms.

5 (a) What will be the output produced on execution of the following code segments? Justify your answers.

```
i) def changelist():
    L1=[]
    L2=[]
    L3=[]
    for i in range(1,10):
        L1.append(i)
    print(L3)          #Statement 1
    for i in range(10,1,-2):
```

```

        L1.append(i)
        print(L3)          #Statement 2
        for i in range(len(L2)):
            L3.append(L2[i] + L1[i])
            L3.append(len(L1)-len(L2))
        print(L3)          #Statement 3
    changelist()
ii) list1 = ['english','geography','history',2014,
2021]
        print(list1[1][-3:-9:-1])    #Statement 4
        list1[3] = 2001
        del list1[3]
        print(list1[2])    #Statement 5
        print(list1)        #Statement 6
        print(list1[2:4])    #Statement 7
iii) txt='Hi! Good morning! Have a great day! Enjoy.'
        txt = txt.split('!')
        print(txt)          #Statement 8
        print('*'.join(txt)) #Statement 9
        txt = txt.partition('!')
        print(txt)         #Statement 9
        print(txt.find('good')) #Statement 10
        print(s1.count('o'))    #Statement 11

```

- (b) Define a Python function `triangle` that accepts a positive integer `n` as an argument and produces the pattern given below as output. In the pattern, for example, when 4 is entered as the value of `n`, the output will be as follows.

```

1
2 2
3 3 3
4 4 4 4
3 3 3
2 2
1

```

- 6
- a) Show the contents of the stack on execution of each of the following sequence of operation: `push('A')`, `isempty()`, `push('B')`, `pop()`, `push('C')`, `pop()`, `pop()`. Assume that the stack is initially empty. What will be the effect of another `pop()` operation invoked after the execution of the above sequence of operations?
- b) Enumerate any four types of Exception. Explain under what conditions these exceptions are raised by the Python interpreter. How are they handled?
- c) Perform the following operations on a queue in the sequence given below (assume queue is initially empty)
- Enqueue 3
 - Enqueue 23
 - Enqueue 5
 - Dequeue
 - Dequeue
 - Enqueue 11
 - Dequeue
 - Dequeue
- d) Define a function `prime_value_remove` that accepts a string `s` as an input and returns a string `res` with the characters removed at indexes (starting with 0) corresponding to prime numbers. Example: for a string `s="Helloworld"`, it should return `res="Heoold"`.