POSSESSION OF MOBILES IN EXAM IS UFM PRACTICE

Name of student _____ Enrollment No. _____

BENNETT UNIVERSITY, GREATER NOIDA
B.TECH
TEST – End-Term
FALL(ODD) SEMESTER-I, A.Y. 2017- 2018

COURSE CODE: ECSE101L
COURSE NAME : Computational Thinking with Python
MAX . MARKS: 60

MAX. TIME: 2 hours
COURSE CREDIT : 5
TOTAL NO. OF PAGES=12

Signature of Invigilator:_____

Signature of Examiner: _____

Instructions:

* Attempt all questions and marks for each question are written against them.
* Use the last two pages of your answer sheet for rough work.

1. (a) **[Marks 4]** Write a program in Python to input an integer y and print each digit of y in the reverse order.

Example: I/P= 6787

O/P= 7

8

7

6

(b) **[Marks 6]** Modify the above program to determine whether the given integer y is palindrome.
Example: I/P: y=6776

o/p: The given no. is palindrome

2. **[Marks 8]** Write a Python program that counts the no. of occurrences of each string in a given string. **Example:** I/P= "you can not end a sentence with because because because is a conjunction"

o/p= "you" appeared 1 times

"can" appeared 1 times

"not" appeared 1 times

"end" appeared 1 times

"a" appeared 2 times

"sentence" appeared 1 times

"with" appeared 1 times

"because" appeared 3 times

"is" appeared 1 times

"conjunction" appeared 1 times

3. **[Marks 8]** Assume that there is one file named "test_file.txt" in your current working directory. Write a Python program to read the file and count the number of sentences inside the file. A sentence is counted when a word ends with character ".". If there is any exception raised at the time of reading the file, your program should handle the exception and display the message "File reading errors!!!".

4. **[Marks 6]** Write a Python program to count the number of digits in a given integer literal using **recursion.**

5. **[Marks 6]** Write a Python program that accepts four integer literals N, M, D1 and D2 (where, N is less than M, D1>=2 and D2>=2) and calculates the sum of integers between N and M inclusive which are divisible by D1 but not divisible by D2. Values of N, M, D1 and D2 must be taken from the user.

Test Data: I/P: N=9, M=300, D1=7 and D2=63

O/P: Sum of integers between 9 & 300 that are divisible by 7 but not by 63 is 5684

6. **[Marks 8]** Write a Python program to find the list in a list of lists whose sum of elements is the highest.
Example:
I/P: [[1,2,3], [4,5,6], [10,11,12], [7,8,9]]
Expected O/P: [10, 11, 12]

7. **[Marks 6]** Create a class named "Circle" and initialize it with radius. Make two methods getArea and getCircumference inside this class in order to calculate the area and circumference of the a circle having a radius (r).

8. **[Marks 8 (2 X 4)]** Write the output of the following Python programs:

(a)
a = True
b = False
c = False
 	
if a or b and c:
print("Condition is True")
else:
print("Condition is False")
	O/P:

(b)
for i in range(1, 5):
for j in range(i):
print(i, end=' ')
print()
	O/P:

(c) tuple1 = (0, 1, 2, 3) tuple1[0] = 4 print(tuple1)	**O/P:**
(d) lst = [0 ,1, 2, 3] print(lst[1:]) print(lst[::-1]) print(lst[2:4])	**O/P:**

Rough Work

Rough Work