MAHARSHI DAYANAND UNIVERSITY, ROHTAK

NAAC Accredited 'A' GRADE

(To be filled in by the Evaluator at the time of evaluation
To be stapled on the top of online print out of Answer Book)
(Checking Assistant to make sure No column is left Blank)

Roll No. 8017613

Name Bazgha Razi

Exam B.Tech (CSE) Subject Python Programming

Semester 3rd Syllabus Code PCC-CSE-207G(A)

Question Paper ID No. 3129 Date : 03 07 2021

TO BE FILLED BY EXAMINERS ONLY

Q.No	A	B	C	D	E	F	TOTAL
1							
2							
3							
4							
5							
6							
7							
8							
9							
10							

TOTAL IN FIGURES

TOTAL MARKS IN WORDS

Bag ID No.

Examiner ID

Full Signature of Examiner

Full Signature of Checking Assistant

C.O.E.
STAMP

a) University Roll No. (In figures): 8017613 In words: Eight Zero One Seven Six One Three

b) Name of the Student: Bazgha Razi

c) Class/Semester: B. Tech (CSE) / 3rd Semester

d) Name of the Paper: Python Programming

e) Question Paper ID: OLE-3129 f) Total No. of Pages Written by Candidate: 15

g) Date of Examination: 03/07/2021

h) Signature of Student: Bazgha Razi

Roll No. 8017613 Page No. 2 Date 03/07/21 Signature Razzia Razzi

Ans 1 a) Function: It is the relationship between one or more inputs and a set of outputs.

Arguments: It is the value that are sent to the function when it is called.

In functions, arguments may or may not be defined it is totally based on the program. If required then arguments are used in the function.
Example of function and arguments

```
def function (arguments):
```

Ans 1 b) The variable x & y refer to numbers. Write a code segment that prompts the user for an arithmetic operator and prints the value obtained by applying that operator to x and y

```
a = float (input ("Enter first no: "))
b = float (input ("Enter second no: "))
c = input ("Enter your operator:"))
```

```
if c == '+':
    print(a+b)
elif c == '-':
    print(a-b)
elif c == '*':
    print(a*b)
else c == '/':
    print(a/b)
```

Output:

Enter first ~~number~~ no : 3
Enter second no : 2
Enter your operator : +

5

Ans 1 c) Polymorphism.

In polymorphism, poly means many and morph means shape.
So, polymorphism is that one task can be performed in different ways.
Example: We have class of human. So, all humans can speak but differently, their languages may be differ.

Example:

```
class Bird:
    def intro(self):
        print("There are many types of birds.")

    def flight(self):
        print("Most of the birds can fly but some cannot")

class sparrow(Bird):
    def flight(self):
        print("Sparrow can fly")

class ostrich(Bird):
    def flight(self):
        print("Ostrich cannot fly")
object_sparrow = sparrow()
object_ostrich = ostrich()
obj_spr.flight() #Sparrows can fly.
```

Ans 2d) break statement

break is a keyword in python which is used to bring the program control out of the loop.
It breaks the loop one by one. It is used to break the execution of the program and control goes to the next line after the loop.

Syntax: loop statements
break;

Example: WAP to use break in the loop

```
s = 'python'
count = 0
for i in s:
    count = count + 1
    if i == 'h':
        break;
    print(i)
```

Output: pyt

Ans 1 e) What happens when the programmer forgets to update the loop control variable in a while loop?

If it happens then the loop runs for infinite times it is also known as infinite loop as updating the loop control variable is important to ensure that the variable count has been initialized. If it doesnot happen the the loop fails.

Ans 1 f) Code:

```
x = 55
print ("Before increment :", x)
x += 1
print ("After increment : ", x)
```

Output:

Before increment : 55

After increment : 56

Ans 2a) # WAP the math module includes a pow function that raises a number to a given power. The first argument is the number, and the second argument is the exponent. Write a code segment that import this function and calls it to print the values 82 and 54.

Example : 82^{54} =

Code :

```
import math
a = int(input("Enter the number : "))
b = int(input("\nEnter the exponent : "))
~~math.pow(a,b)~~
c = math.pow(a,b)
print(\nc)
```

Output : Enter the number : 82
Enter the exponent : 54

→ solution of $(82)^{54}$

Ans 25) # WAP to accept the length of three sides of a triangle as inputs. The program outputs should indicate whether or not the triangle is an equilateral triangle.

In an equilateral triangle all three sides of a triangle is equal.

Code:

```
print ("Enter the sides of a triangle")
a = int (input ("Enter the first side: "))
b = int (input ("Enter the second side: "))
c = int (input ("Enter the third side: "))

if a==b==c:
        print ("Triangle is equilateral")

else:
        print ("Triangle is not equilateral")
```

Output:

```
Enter the sides of a triangle
Enter the first side: 10
Enter the second side: 10
Enter the third side: 10
Triangle is equilateral
```

Ans 3a) # Assume that the variable myString refers to a string. Write a code segment that uses a loop to print the characters of the string in reverse order.

Example: Hello World # Original String
dlroW olleH # Reversed String

Code:

```
myString = input("Enter the string : ")
# now take an empty string
String = ("")
for i in myString:
    String = i + string
print("Reversed String : ", String)
```

Output:

Enter the string : Hello World
Reversed String : dlroW olleH

Ans 3b) // WAP that prompts the user for a filename. If the file exists, the program should print its content on the terminal. Otherwise it should print an error message.

Code :

```
import os
from os.path import exists

file = input ("Enter the file name:")
if exists (file):
    inputfile = open (file, 'r')
    print (inputfile.read())

else:
    print ("Error!")
    print ("This file does not exist.")
```

Output :

```
Enter the file name : file.txt

Error!
This file does not exist.
```

Roll No. 8017613 Page No. 11 Date 03/07/21 Signature Razgha Razi

Ans 4 a) Anne complains that defining functions to use in her program is a lot of extra work. She says she can finish her programs much more quickly if she just writes them using the basic operators & control statements. State 3 reasons why her view is shortsighted.

First, I'll explain what is function. So,

Function: It is a relationship between one or more inputs and a set of outputs.

In python, function is a self-contained block of code that encapsulates a specific task or related group tasks.

Syntax: def function ( ):
    statement
    code
    :

Now, 3 reasons why her view is shortsighted.

1) Abstraction and Reusability
Suppose she write some code that does something useful. As she continue

development, she find that the task performed by that code is one she need often, in many places within the program. Then she has to write the code again and again but using functions she can use the given function again which is easier to write & understand

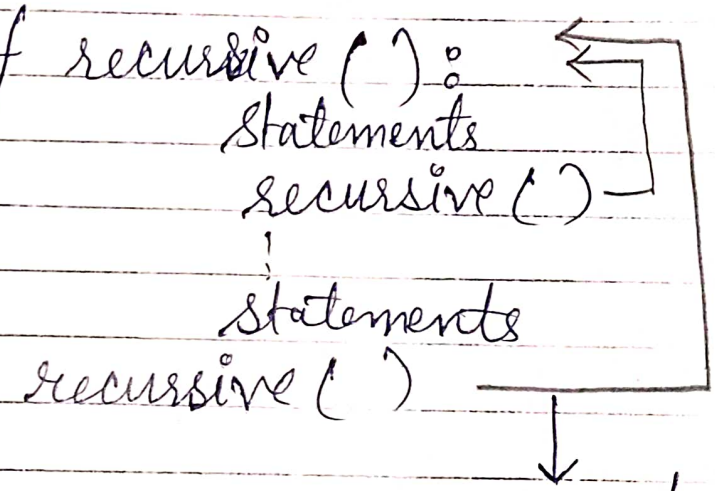
ii) Modularity:
Functions allow complex processes to be broken up into smaller steps. Instead of writing all the code together, then it's broken into separate function each of which perform or focus on specific task.

iii) By using function her program is more organised and clean.

4b) Recursive function fact that returns the factorial of a given +ve integer. In what way is a recursive design different from top down design?

Recursive function: In python, a function that calls itself is known as recursive function.

Syntax: def recursive():
    Statements
    recursive()
    Statements
recursive()



Same function call itself i.e known as recursive function.

WAP to print factorial of a give +ve integer using recursive function

Code:
```
def fact(a):
    if a==1:
        return 1
    else:
        return (a* fact(a-1))
a=3
print(fact(a))
```

Ans5a) Parameters: It is used to pass the information into a function.

It is a named variable passed into a function. It is also used to import arguments into functions. Parameters are initialized to the value of the arguments supplied.

Return Statement: It is a key component of functions. It is used to make our functions send python objects back to the caller code. These objects are also known as function's return value.

WAP to define a no. is even or not.

Code:

```
a = int(input("Enter a number:"))
def even(a):
    if (a%2==0):
        print("True")
    else:
        print("False")
even(a)
```

Output: Enter a number: 4
True

Ques5b) # WAP that takes as an input a text file. Your program should print all the unique words in the file in alphabetical order.

Code:

```
def file(filename, outputfile):
    filename = open(filename, 'r')
    filecontent = filename.read()
    filename.close()
    a = []
    words = filecontent.split()
    b = open(outputfile, 'w')
    for word in words:
        if word not in a:
            a.append(word)
            b.write(str(word)+'\n')
    file.close()

file('file.txt', 'outputfile.txt')
for line in sorted(open('outputfile.txt')):
    print(line, end='')
```