CHAPTER - IV
RECURSIVE FUNCTIONS

RECURSION

A recursive function is a function that calls itself until a “base condition” is true, and execution stops. While false

Recursion in computer science is a method of solving a problem where the solution depends on solutions to smaller instances of the same problem (as opposed to iteration). ...

Most computer programming languages support recursion by allowing a function to call itself from within its own code.

ADVANTAGES OF RECURSION

1. **Recursive functions make the code look clean and elegant.**
2. **A complex task can be broken down into simpler sub-problems using recursion.**
3. **Sequence generation is easier with recursion than using some nested iteration**.

DISADVANTAGES OF RECURSION

1. **Sometimes the logic behind recursion is hard to follow through.**
2. **Recursive calls are expensive (inefficient) as they take up a lot of memory and time.**
3. **Recursive functions are hard to debug**.

RECURSION - PROGRAM

Factorial of given number using recursion:

```
def factorial(n):
        if n == 1:
                return 1
        else:
                return n * factorial(n-1)
```

RECURSION - PROGRAM

We can track the recursive function:

```
 def factorial(n):
    print("factorial has been called with n = " + str(n))
    if n == 1:
        return 1
    else:
        res = n * factorial(n-1)
        print("intermediate  result  for  ",  n,  "  *  factorial("
,n-1, "): ",res)
        return res

print(factorial(5))
```

RECURSION - PROGRAM

```
factorial has been called with n = 5
factorial has been called with n = 4
factorial has been called with n = 3
factorial has been called with n = 2
factorial has been called with n = 1
intermediate result for  2  * factorial( 1 ):  2
intermediate result for  3  * factorial( 2 ):  6
intermediate result for  4  * factorial( 3 ):  24
intermediate result for  5  * factorial( 4 ):  120
120
```

RECURSION - PROGRAM

Fibonacci series using recursive function:

```
def fib(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else:
        return fib(n-1) + fib(n-2)
```

RECURSION - PROGRAM

```
# Python program to find the H.C.F of two input number

# define a function
def computeHCF(x, y):

# choose the smaller number
    if x > y:
        smaller = y
    else:
        smaller = x
    for i in range(1, smaller+1):
        if((x % i == 0) and (y % i == 0)):
            hcf = i

    return hcf
```

```
def gcd(a,b):
if(b==0):
return a
else:
return gcd(b,a%b)
a=int(input("Enter first
number:"))
b=int(input("Enter second
number:"))
GCD=gcd(a,b)
print("GCD is: ") print(GCD)
```

RECURSION - PROGRAM

```
# take input from the user
# num1 = int(input("Enter first number: "))
# num2 = int(input("Enter second number: "))

print("The H.C.F. of", num1,"and", num2,"is",
    computeHCF(num1, num2))
```

RECURSION - PROGRAM

```
def sum_recursive(current_number, accumulated_sum):
    # Base case
    # Return the final state
    if current_number == 11:
        return accumulated_sum

    # Recursive case
    # Thread the state through the recursive call
    else:
        return sum_recursive(current_number + 1,
      accumulated_sum + current_number)
```

RECURSION - PROGRAM

1 Recursive Python function to find sum of natural numbers.

2. Recursive Python function to find sum of even numbers.

3. Recursive Python function to find sum of odd numbers.

4. Recursive Python function to find sum of fib series.

5. Recursive Python function to find given number is prime or not.

Class Test

1. Which of the following is the use of function in python?

   a) Functions are reusable pieces of programs
   b) Functions don’t provide better modularity for your application
   c) you can’t also create your own functions
   d) All of the mentioned

2. Which keyword is use for function?

   a) Fun b) Define
   c) def d) Function

Class Test

3. What is the output of the below program?

```
def sayHello():
    print('Hello World!')
sayHello()
sayHello()
```

a) Hello World!
Hello World!
b) ‘Hello World!’
‘Hello World!’
c) Hello
Hello

Class Test

4. What is the output of the below program?

```
def printMax(a, b):
    if a > b:
        print(a, 'is maximum')
    elif a == b:
        print(a, 'is equal to', b)
    else:
        print(b, 'is maximum')
printMax(3, 4)
```

a) 3 b) 4

c) 4 is maximum d) None of the mentioned

Class Test

5. What is the output of the below program ?

```
x = 50
def func(x):
print('x is', x)
x = 2
print('Changed local x to', x)
func(x)
print('x is now', x)
```

a) x is now 50 b) x is now 2

c) x is now 100 d) None of the mentioned

Class Test

6. What is the output of the below program?

```
x = 50
def func():
global x
print('x is', x)
x = 2
print('Changed global x to', x)
func()
print('Value of x is', x)
```

a) x is 50
Changed global x to 2
Value of x is 50

b) x is 50
Changed global x to 2
Value of x is 2

c) x is 50
Changed global x to 50
Value of x is 50

d) None of the mentioned

Class Test

7. What is the output of below program?

```
def say(message, times = 1):
print(message * times)
say('Hello')
say('World', 5)
```

a) Hello
WorldWorldWorldWorldWorld

b) Hello
World 5

c) Hello
World,World,World,World,
World

d) Hello
HelloHelloHelloHelloHello

Class Test

8. What is the output of the below program?

```
def func(a, b=5, c=10):
print('a is', a, 'and b is', b, 'and c is', c)

func(3, 7)
func(25, c = 24)
func(c = 50, a = 100)
```

a) a is 7 and b is 3 and c is 10
a is 25 and b is 5 and c is 24
a is 5 and b is 100 and c is 50

b) a is 3 and b is 7 and c is 10
a is 5 and b is 25 and c is 24
a is 50 and b is 100 and c is 5

c) a is 3 and b is 7 and c is 10
a is 25 and b is 5 and c is 24
a is 100 and b is 5 and c is 50

d) None of the mentioned

Class Test

9. What is the output of below program?

```
def maximum(x, y):
if x > y:
return x
elif x == y:
return 'The numbers are equal'
else:
return y

print(maximum(2, 3))
```

a) 2 b) 3 c) The numbers are equal

d) None of the mentioned

Class Test

10. Which of the following is a features of DocString?

a) Provide a convenient way of associating documentation with Python modules, functions, classes, and methods

b) All functions should have a docstring

c) Docstrings can be accessed by the __doc__ attribute on objects

d) All of the mentioned

Class Test

1	A
2	C
3	A
4	C
5	A
6	B
7	A
8	C
9	B
10	D

Thank You