Computer Programming Laboratory

B.Tech. 1st Semester

Name	**: SUBHENDU MAJI**
Roll Number	**: 18ETCS002121**
Department	**: Computer Science and Engineering**

Faculty of Engineering & Technology

Ramaiah University of Applied Sciences

Ramaiah University of Applied Sciences

Private University Established in Karnataka State by Act No. 15 of 2013

Faculty	Engineering & Technology
Programme	B. Tech. in Computer Science and Engineering
Year/Semester	1st Year / 1st Semester
Name of the Laboratory	Computer Programming Laboratory
Laboratory Code	18ESL109A

List of Experiments

1. Introduction to Python programming environment
2. Variables, data types, operators and expressions
3. Input output operations
4. Logic operations and decision making
5. Loop statements
6. Character and string operations
7. Functions
8. File handling
9. Data structures
10. Libraries

Name: SUBHENDU MAJI Roll Number: 18ETCS002121

Index Sheet

No.	Lab Experiment	Performing the experiment (7)	Document (7)	Viva (6)	Total Marks (20)
1	Introduction to Python programming environment				
2	Variables, data types, operators and expressions				
3	Input output operations				
4	Logic operations and decision making				
5	Loop statements				
6	Character and string operations				
7	Functions				
8	File handling				
9	Data structures				
10	Libraries				
11	Lab Internal Test conducted along the lines of SEE and valued for 50 Marks and reduced for 20 Marks				
	Total Marks				

Lab Internal Marks =

Signature of the Staff In-charge

Laboratory 7

Title of the Laboratory Exercise: Functions

1. Introduction and Purpose of Experiment

   A function serves as an abstraction mechanism to view many things as one thing. The function definition specifies the name of a new function and the sequence of statements that execute when the function is called. Once a function is defined, it can be called as many times as required. By solving these problems, students will be able to create user defined functions.

2. Aim and Objectives

   Aim

   - To develop programs using user defined functions

   Objectives

   At the end of this lab, the student will be able to

   - Apply user defined functions with proper definition

3. Experimental Procedure

   i. Analyse the problem statement
   ii. Design an algorithm for the given problem statement and develop a flowchart/pseudo-code
   iii. Implement the algorithm in Python language
   iv. Execute the Python program
   v. Test the implemented program
   vi. Document the Results
   vii. Analyse and discuss the outcomes of the experiment

4. Questions

   a. Write a Python program to find factorial of a number using
      i. User defined function without recursion
      ii. Recursive function

   b. Write a Python function to reverse a string. Use the returned result to check whether the given string is palindrome or not.

c. Write a Python function that takes a list and returns a new list with unique elements of the first list.

5. Calculations/Computations/Algorithms

5.1 program to find factorial with and without recursion.

```
Step1: start
Step2: make a function
Step3:for without recursive function , fact:=1
Step4: for i in range(1,a+1):
    Fact * := i
Step5: for with recursive function,
    If (n==0):
        Return 1
    Else:
        Return (n*fact(n-1))
Step6: input number for calculating factorial
Step7: stop
```

5.2 program to reverse a string. Use the returned result to check whether the given string is palindrome or not.

```
Step1: start
Step2: input the string from the user
Step3: x  := a[ : : - 1]
Step4: if a :== x
    Output “palindrome”
    Else:
    Output “not a palindrome”
Step5: stop
```

5.3 program to that takes a list and returns a new list with unique elements of the first list.

Step1: start

Step2: input element from user in an empty list

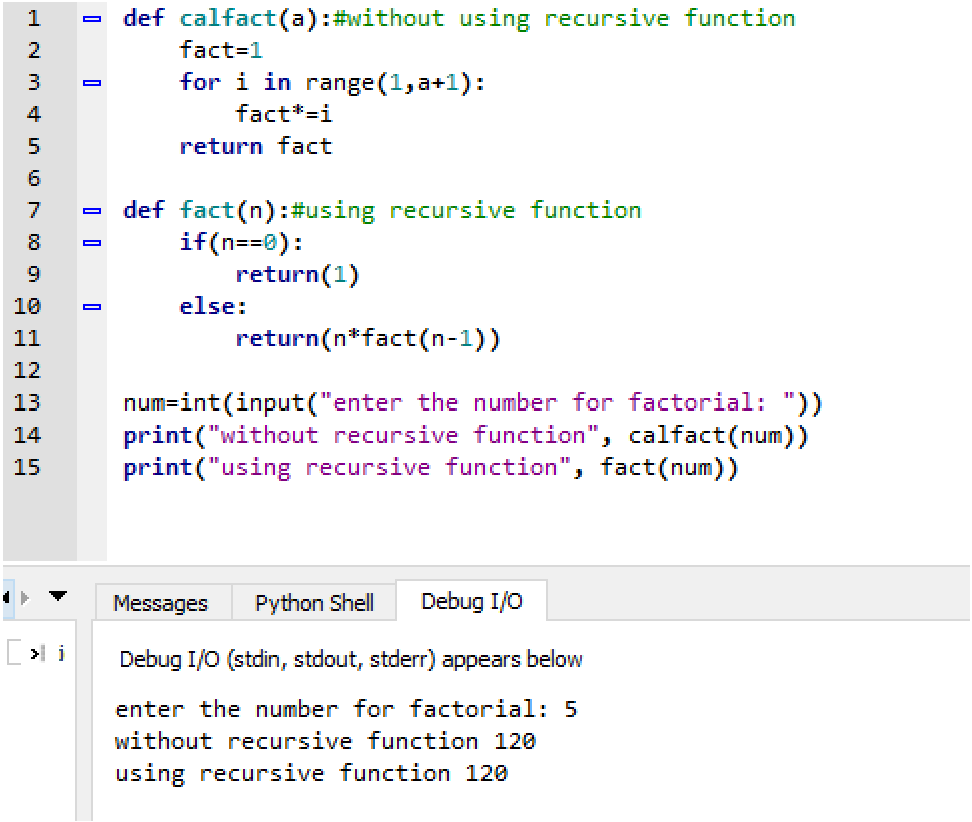
Step3: using a for-loop check whether each element is repeating or not

Step4: if repeating then print once

Step5: stop

6. Presentation of Results

6.1 program to find factorial with and without recursion



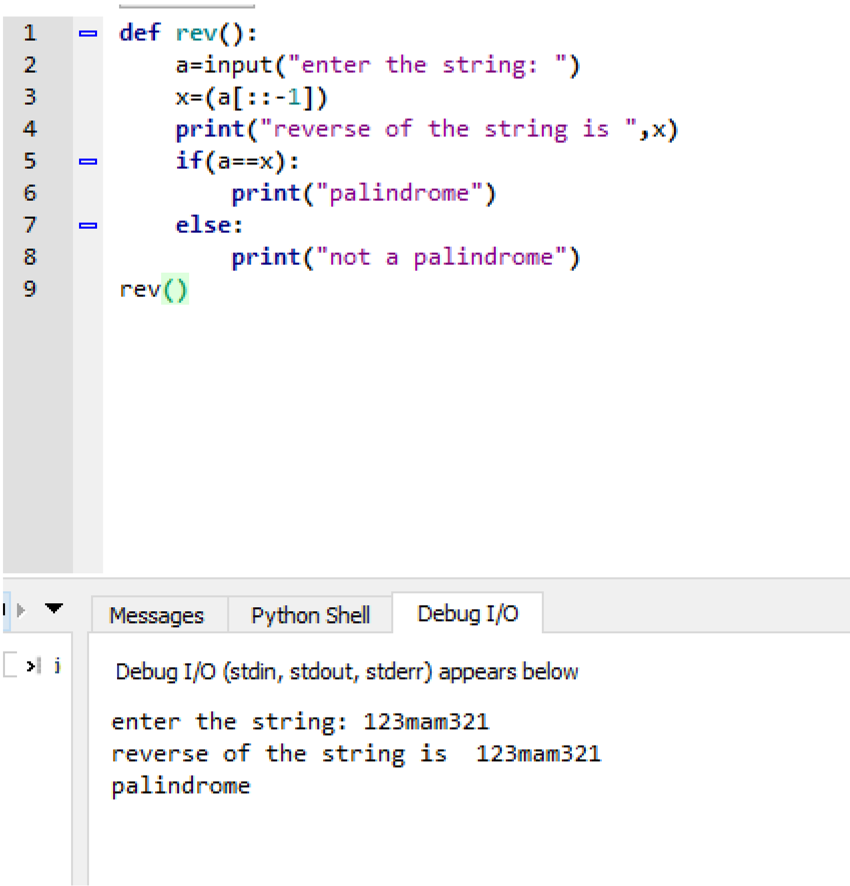
```
def calfact(a):#without using recursive function
    fact=1
    for i in range(1,a+1):
        fact*=i
    return fact

def fact(n):#using recursive function
    if(n==0):
        return(1)
    else:
        return(n*fact(n-1))

num=int(input("enter the number for factorial: "))
print("without recursive function", calfact(num))
print("using recursive function", fact(num))
```

Debug I/O (stdin, stdout, stderr) appears below

```
enter the number for factorial: 5
without recursive function 120
using recursive function 120
```

Figure 1 program to find factorial with and without recursion

6.2 program to check palindrome or not

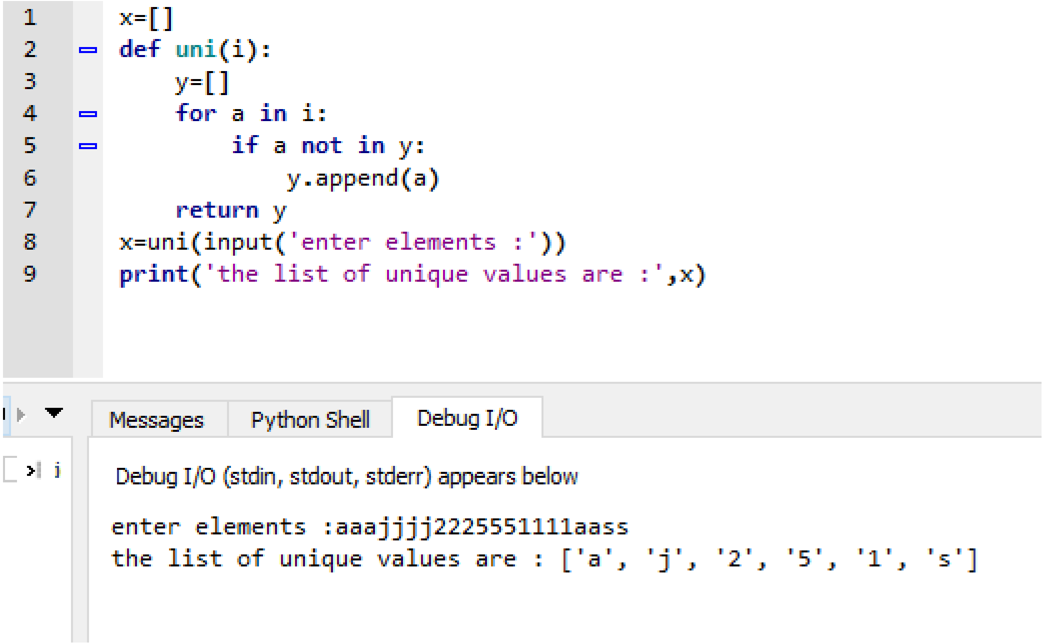

```python
def rev():
    a=input("enter the string: ")
    x=(a[::-1])
    print("reverse of the string is ",x)
    if(a==x):
        print("palindrome")
    else:
        print("not a palindrome")
rev()
```

Debug I/O (stdin, stdout, stderr) appears below

```
enter the string: 123mam321
reverse of the string is  123mam321
palindrome
```

Figure 2 program to check palindrome or not

6.3 program to make list of unique elements


```python
x=[]
def uni(i):
    y=[]
    for a in i:
        if a not in y:
            y.append(a)
    return y
x=uni(input('enter elements :'))
print('the list of unique values are :',x)
```

Debug I/O (stdin, stdout, stderr) appears below

```
enter elements :aaajjjj2225551111aass
the list of unique values are : ['a', 'j', '2', '5', '1', 's']
```

Figure 3 program to make list of unique elements

7. Analysis and Discussions

User defined function are used to make code easier to understand and access to different part of the program easily.

Using functions, the error finding and testing becomes easy. code reusability increases. And most importantly program readability increases.

8. Conclusions

User defined function help to make code more readable and easier to understand and to work on it.it simplifies the complexity of program to an extent

9. Comments

    i. Limitations of Results

The programs can be made more user friendly by applying graphics.