Experiment 1.1

Aim:

Program to calculate are of triangle

Theory:

Area of a triangle = (s*(s-a)*(s-b)*(s-c)) ** 0.5

Here is the semi-perimeter and a,b,and c are three side of the triangle.

Let’s we take the input as three side of a triangle and store them in three variable a,b, and c

Then, we have to calculated the semi-perimeter of the triangle and put this value in the triangle area formula

Program:

```
          # Three sides of the triangle is a, b and c:
a = float(input('Enter first side: '))
b = float(input('Enter second side: '))
c = float(input('Enter third side: '))

# calculate the semi-perimeter
s = (a + b + c) / 2

# calculate the area
area = (s*(s-a)*(s-b)*(s-c)) ** 0.5
print('The area of the triangle is %0.2f' %area)
```

Output:

```
Enter first side: 5
Enter second side: 6
Enter third side: 7
The area of the triangle is 14.70
```

Experiment 1.2

Aim:

Program to calculate are of rectangle

Theory:

Firstly, We will take input from the used for the length and breath using the input() function

Now, we will calculated the area of a rectangle by using the formula

Area = width * height

At last, point the area of rectangle to see the output.

Program:

```
            # Python Program to find Area of a Rectangle

width = float(input('Please Enter the Width of a Rectangle: '))

height = float(input('Please Enter the Height of a Rectangle: '))

# calculate the area

Area = width * height

print("\n Area of a Rectangle is: %.2f" %Area)
```

Output::

```
Please Enter the Width of a Rectangle:
Please Enter the Height of a Rectangle:
Area of a Rectangle is:
```

Experiment 1.3

Aim:

Program to calculate are of circle

Theory:

In this program I have defined a function as defindarea(r)
We can store the value Pi=3.14 which is fixed.
The function is returned as returnpi*(r*r)
A function is a named section of code that perform a specific task

Program:

```
        # Python program to find Area of a circle

def findArea(r):
   PI = 3.142
   return PI * (r*r);

# Driver method
print("Area of circle si %.6f" % findArea(5));
```

Output:

```
Area of circle is 78.550000
```

Experiment 2

Aim:

Program to find the union of two lists

Theory:

We can simply used Plus “+” operator inorder to combine two list into one
This will reflect the repetition

Program:

```
            # Python program to illustrate union
# Maintained repetition
def Union(lst1, lst2):
   final_list = lst1 + lst2
   return final_list

# Driver Code
lst1 = [23, 15, 2, 14, 14, 16, 20 ,52]
lst2 = [2, 48, 15, 12, 26, 32, 47, 54]
print("union of lst is:",Union(lst1, lst2))
```

Output::

```
union of lst is: [23, 15, 2, 14, 14, 16, 20, 52, 2, 48, 15,

12, 26, 32, 47, 54]
```

Experiment 3

Aim

Program to find the intersection of two lists

Theory:

We will use the built –in sets intersection method.

The intersection() Is a first class part of set

Program:

```
            def intersection_list(list1, list2):
  return list(set(list1) & set(list2))


list1 = [40, 90, 11, 58, 31, 66, 28, 54, 79]
list2 = [58, 90, 54, 31, 45, 11, 66, 28, 26]

print("Intersection of list is:",intersection_list(list1, list2))
```

Output:

```
Intersection of list is: [66, 90, 11, 54, 58, 28, 31]
```

Experiment 4

Aim:

Program to remove the –il th occurrence of the given word in a list where words repeat

Theory:

In this program we have created one list in that list placing element inside the square bracket [ ], separated by commas

A for loop I used for iterating over a sequence (That is either list,atule, a dictionary, aset, or a string)

With the for loop we can execute a set of statement once for each item in alsit

The remove method is used to remove the specified value from the list

Program:

```
        List2=[‘python’ , ‘java’, ‘Hadoop’ , ‘python’ , ‘c#’ , ‘python’]
For I in list2:
If I == ‘python’:
List2 remove(‘python’)
If ‘python’ not in list2:
Print(list2)
```

Output:

```
    [‘java’ , ‘hadoop’, ‘c#’,]
```

Experiment 5

Aim:

Program to count the occurrences of each word in given string sentence.

Theory:

In this program the count() method is used.

The count () method is the number of element with the specified value.

Syntax : list.count(value)

Program:

```
string="This is a sample Python program, Welcome to World Of Python Programming!"
word="Python"
list=[]
wordCount=0
list=string.split(" ")
for i in range(0,len(list)):
if(word==list[i]):
wordCount=wordCount+1
print("Number of occurrences found in the string:")
print(wordCount)
```

Output::

```
Number of occurrences found in the string:
2
```

Experiment 6

Aim:

Program to check if a substring is present in given string.

Theory::

input a string(python program) and a substring (Program) from the user and store it in separate variables.

Check if the substring is present in the string or not. To do this using find() in-built function.

Print the final Output:.

Exit.

Program:

```
def check(str1, sstr):
  if (str1.find(sstr) == -1):
    print(sstr,"IS NOT PRESENT IN THE GIVEN STRING")
  else:
    print(sstr,"IS PRESENT IN THE GIVEN STRING")
# Driver code
str1 = input("Enter the string ::>")
sstr=input("Enter Substring ::>")
check(str1, sstr)
```

Output::

```
Enter the string ::> python program
Enter Substring ::> program
program IS PRESENT IN THE GIVEN STRING
Enter the string ::> python program
Enter Substring ::> programming
programming IS NOT PRESENT IN THE GIVEN STRING
```

Experiment 7

Aim:

Program to map two lists into dictionary

Theory::

Dictionary is an ordered collection of data value, whose value are accessible by key

The dictionary is mapping of key and value

The zip () function of python makes a herator that aggregate element from at list two listes. To map two lists, we can use the python zip() function. This function allows us to combine two lists together. We can use one list as the key for dictionary and the other as the values.

In the given example, we have 2 list, 1 containing a list of students and other containing their marks. The requirement I sto create a single dictionary that store the name of the student along with their marks.

Program:

```
            students = ['Shubham', 'piyush', 'sonu', 'Abhi']
marks = [89, 53, 92, 86]

students_dict = dict(zip(students, marks))
print(students_dict)
```

Output::

```
        {'Shubham': 89, 'piyush': 53, 'sonu': 92, 'Abhi': 86}
```

Experiment 8

Aim

Program to count the frequency of word appearing in a string using a dictionary

Theory:

When it is required to count the frequency of words that appear in a string with the help of dictionary, the ‘split’ method is used to split the values, and a list comprehension is used.

The list comprehension is a shorthand to iterate through the list and perform operations on it.

A list can be used to store heterogeneous values (i.e data of any data type like integer, floating point, strings, and so on).

Program:

```
my_string = input("Enter the string :")
my_list=[]
my_list=my_string.split()
word_freq=[my_list.count(p) for p in my_list]
print("The frequency of words is ...")
print(dict(zip(my_list,word_freq)))
```

Output::

```
Enter the string : my name is shubham shubham
The frequency of words is ...
{'my': 1, 'name': 1, 'is': 1, 'shubham': 2}
```

Experiment 9

Aim:

Program to create a dictionary with key as first character and value as words starting with that character

Theory:

When it is required to create a dictionary with the key as the first character and the associated value as the word which is the starting of that character, the s'plit' method, a dictionary and simple 'if' condition is used.

Program

```
my_string=input("Enter the string :")
split_string = my_string.split()
my_dict={}
for elem in split_string:
  if(elem[0] not in my_dict.keys()):
    my_dict[elem[0]]=[]
    my_dict[elem[0]].append(elem)
  else:
    if(elem not in my_dict[elem[0]]):
      my_dict[elem[0]].append(elem)
print("The dictionary created is")
for k,v in my_dict.items():
  print(k,":",v)
```

Output:

```
        Enter the string :my name is shubham shu
The dictionary created is
m : ['my']
n : ['name']
i : ['is']
s : ['shubham', 'shu']
```

Experiment 10

Aim:

Program to find the length of a list using recursion

Theory::

When it is required to find the length of a list with the help of recursion technique, a user defined method is used, and simple indexing technique is used.

A list can be used to store heterogeneous values (i.e data of any data type like integer, floating point, strings, and so on).

The recursion computes output of small bits of the bigger problem, and combines these bits to give the solution to the bigger problem.

Program:

```
def list_length(my_list):
if not my_list:
return 0
return 1 + list_length(my_list[1::2]) + list_length(my_list[2::2])
my_list = [26, 7, 20, 22, 40, 52, 78]
print("The list is :")
print(my_list)
print("The length of the string is : ")
print(list_length(my_list))
```

Output::

```
The list is :
[26, 7, 20, 22, 40, 52, 78]
The length of the string is :
7
```

Experiment 11

Aim:

Compute the diameter,circumference,and volume of a sphere using class

Theory:

For finding the diameter we required formula that is 2*radius

For finding circumference we need Pi =3.14 & the formula is pi * diameter

For finding

Program:

```
import math

pi = math.pi

sphere = float(input("Enter the sphere's radius : "))

radius = sphere
diameter = 2 * radius
circumference = pi * diameter
surfaceArea = 4 * pi * radius * radius
volume = 4/3 * pi * radius * radius * radius

print("Diameter : ", diameter)
print("Circumference : ", circumference)
print("surface area : ", surfaceArea)
print("volume : ", volume)
```

Output:

```
Enter the sphere's radius : 60
Diameter :  120.0
Circumference :  376.99111843077515
surface area :  45238.93421169302
volume :  904778.6842338603
```

Experiment 12

Aim:

Program to read file and capitalize the first letter of every word in the file

Theory:

To capitalize the first letter of the every word in the file and print letter we will use the title () function in python.

The title function in python is the python string method which is used to convert the first character in each word to uppercase & the remaining character to lowercase in the string and return the new string

We will take the contain as input.

We will open file & save its contain by using open() function in python

Program:

```
 File-gfg= open (‘python.tx’, ‘t’)
For line in file –gfg:
Output= line.title()
Print(output)
```

Output:

```
List Is Mutable
```