

Q1. Write a function that takes the lengths of the two shorter sides of a right triangle as its parameters. Return the hypotenuse of the triangle, computed using Pythagorean theorem, as the function's result.

```
In [1]: import math
a = float(input('Enter a side of a: '))
b = float(input('Enter a side of b: '))
c = math.sqrt(a ** 2 + b ** 2)
print('The Length of Hypotenuse is :',c)

Enter a side of a: 25
Enter a side of b: 12
The Length of Hypotenuse is : 27.730849247724095
```

Q2. An online retailer provides express shipping for many of its items at a rate of \$10.95 for the first item, and \$2.95 for each subsequent item. Write a function that takes the number of items in the order as its only parameter. Return the shipping charge for the order as the function's result. Include a main program that reads the number of items purchased from the user and displays the shipping charge.

```
In [2]: def shipping_charge(n):
    if (n == 1):
        return 10.95

    else:
        return 10.95 + (n - 1) * (2.95)
n = int(input("Enter the number of orders :"))
print("The total shipping charges for",n,"orders is :",shipping_charge(n))

Enter the number of orders :30
The total shipping charges for 30 orders is : 96.50000000000001
```

Q3. Write a function that takes three numbers as parameters, and returns the median value of those parameters as its result. Include a main program that reads three values from the user and displays their median.

```
In [3]: a = float(input("Input first parameter value : "))
b = float(input("Input second parameters value : "))
c = float(input("Input third parameters value : "))
if a > b:
    if a < c:
        median = a
    elif b > c:
        median = b
    else:
        median = c
else:
    if a > c:
        median = a
    elif b < c:
        median = b
    else:
        median = c
print("The median is", median)

Input first parameter value : 25
Input second parameters value : 10
Input third parameters value : 85
The median is 63.0
```

Q4. Write a function that determines whether or not three lengths can form a triangle. The function will take 3 parameters and return a Boolean result. In addition, write a program that reads 3 lengths from the user and demonstrates the behavior of this function. In general, if any one length is greater than or equal to the sum of the other two then the lengths cannot be used to form a triangle. Otherwise they can form a triangle.

```
In [4]: def triangle(A,B,C):
    if A+B >= C and B+C >= A and C+A >= B:
        return True
    else:
        return False
A = int(input('Enter the value of length 1 :'))
B = int(input('Enter the value of length 2 :'))
C = int(input('Enter the value of length 3 :'))
triangle(A,B,C)

Enter the value of length 1 :5
Enter the value of length 2 :4
Enter the value of length 3 :6
True
```

Q5. In this exercise you will write a function named isInteger that determines whether or not the characters in a string represent a valid integer. When determining if a string represents an integer you should ignore any leading or trailing white space. Once this white space is ignored, a string represents an integer if its length is at least 1 and it only contains digits, or if its first character is either + or - and the first character is followed by one or more characters, all of which are digits. Write a main program that reads a string from the user and reports whether or not it represents an integer.

```
In [5]: str1=input('Enter a value :')
def isinteger(str1):
    n=str1.replace(' ','')
    x=n.lstrip('-+')
    return x.isdigit()
print(isinteger(str1))

Enter a value :6
True
```

Q6. In this exercise you will create a function named nextPrime that finds and returns the first prime number larger than some integer, n. The value of n will be passed to the function as its only parameter. Include a main program that reads an integer from the user and displays the first prime number larger than the entered value.

```
In [6]: def nextprime(n):
    p=n+1
    for i in range(2,p):
        if(p%i==0):
            p=p+1
    else:
        print(p)
n=int(input('Enter a number to find next prime number :'))
nextprime(n)

Enter a number to find next prime number :4
5
```

Q7. In this exercise you will write a function that determines whether or not a password is good. We will define a good password to be a one that is at least 8 characters long and contains at least one uppercase letter, at least one lowercase letter, and at least one number. Your function should return true if the password passed to it as its only parameter is good. Otherwise it should return false. Include a main program that reads a password from the user and reports whether or not it is good.

```
In [8]: def password(pwd):
    length = lower = upper = digit = False
    if len(pwd) >= 8:
        length = True
        for letters in pwd:
            if letters.islower():
                lower = True
            if letters.isupper():
                upper = True
            if letters.isdigit():
                digit = True

        if length and upper and lower and digit:
            return True
        else:
            return False
pwd = input('Enter the password: ')
password(pwd)

Enter the password: Aravindh@253
True
```

Q8. Write a program that reads values from the user until a blank line is entered. Display the total of all of the values entered by the user (or 0.0 if the first value entered is a blank line). Complete this task using recursion. Your program may not use any loops. The body of your recursive function will need to read one value from the user, and then determine whether or not to make a recursive call. Your function does not need to take any parameters, but it will need to return a numeric result.

```
In [10]: def fun():
    n = input("If you want to exit enter a blank line otherwise enter a value : ")
    if n == " ":
        return 0.0
    else:
        n = int(n)
        return (n + fun())
print("Total of all the values = ", fun())

If you want to exit enter a blank line otherwise enter a value : 8
If you want to exit enter a blank line otherwise enter a value : 8
If you want to exit enter a blank line otherwise enter a value :
Total of all the values = 16.0
```

Q9. In this exercise you will write a recursive function that determines whether or not a string is a palindrome. The empty string is a palindrome, as is any string containing only one character. Any longer string is a palindrome if its first and last characters match, and if the string formed by removing the first and last characters is also a palindrome. Write a main program that reads a string from the user. Use your recursive function to determine whether or not the string is a palindrome. Then display an appropriate message for the user.

```
In [11]: def is_palindrome(s):
    if len(s) < 1:
        return True
    else:
        if s[0] == s[-1]:
            return is_palindrome(s[1:-1])
        else:
            return False
str1 = input("Enter a string:")
if(is_palindrome(str1)==True):
    print("Entered String is a palindrome!")
else:
    print("Entered String is not a palindrome!")

Enter a string:civic
Entered String is a palindrome!
```

Q10. Write a function that takes a list of strings as its only parameter. Your function should return a string that contains all of the items in the list formatted in the manner described previously as its only result. While the examples shown previously only include lists containing four elements or less, your function should behave correctly for lists of any length. Include a main program that reads several items from the user, formats them by calling your function, and then displays the result returned by the function. Consider the following four lists:

apples

apples and oranges

apples, oranges and bananas

apples, oranges, bananas and lemons

```
In [12]: def formatting (lst2):
    s = ''
    s += lst2[0]

    for i in (lst2[1:-1]):
        s = s + ' , ' + i
    s = s + ' and ' + lst2[-1]
    return s
a = input("Enter the elements seperated with space :")
b= list(a.split(" "))
print(formatting(b))

Enter the elements seperated with space :apples oranges bananas lemons
apples, oranges, bananas and lemons
```