UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

1. Is it a triangle?

Aim

To check whether a triangle can be formed given the lengths of the sides.

Algorithm

Input: The lengths of the sides
Output: Conclusion of the triangle's existence

Pre-condition: The lengths must be positive.

Step 1: Start
Step 2: Define a function to check if a triangle is possible.
Step 3: The function checks if the sum of any two sides of the triangle is greater than the third.
Step 4: If yes, it prints that such a triangle can be formed.
Step 5: If no, it checks if the sum of any two sides is equal to the third.
Step 6: If yes, it prints that the triangle is degenerate. If no, it prints that such a triangle can never be formed.
Step 7: Read the lengths of the sides as input from the user.
Step 8: End

Program

```
# Python program to check if a triangle can exist or not provided the lengths of the sides.

# Function definition
def istriangle(a,b,c):
```

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

```
    if (a+b)>c and (b+c)>a and (c+a)>b:
        print("Yes, such a triangle can be formed.")
    elif (a+b)==c or (b+c)==a or (c+a)==b:
        print("The triangle is degenerate.")
    else:
        print("Such a triangle cannot be formed.")

# To take the lengths of the three sides as input from the user
print("Enter the lengths of the three sides of the triangle.")
p=float(input("Enter length of side p: "))
q=float(input("Enter length of side q: "))
r=float(input("Enter length of side r: "))
# Function call
istriangle(p,q,r)
```

Output

Input:

```
3,4,5
1,2,3
4,3,8
```

Output:

```
Enter the lengths of the three sides of the triangle.
Enter length of side p: 3
Enter length of side q: 4
Enter length of side r: 5
Yes, such a triangle can be formed.

Enter the lengths of the three sides of the triangle.
Enter length of side p: 1
Enter length of side q: 2
Enter length of side r: 3
```

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

```
The triangle is degenerate.

Enter the lengths of the three sides of the triangle.
Enter length of side p: 4
Enter length of side q: 3
Enter length of side r: 8
Such a triangle cannot be formed.
```

Results / Inferences

Program for checking the existence of a triangle given the lengths of its sides is written and executed.

UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

2. Factors of a number

Aim

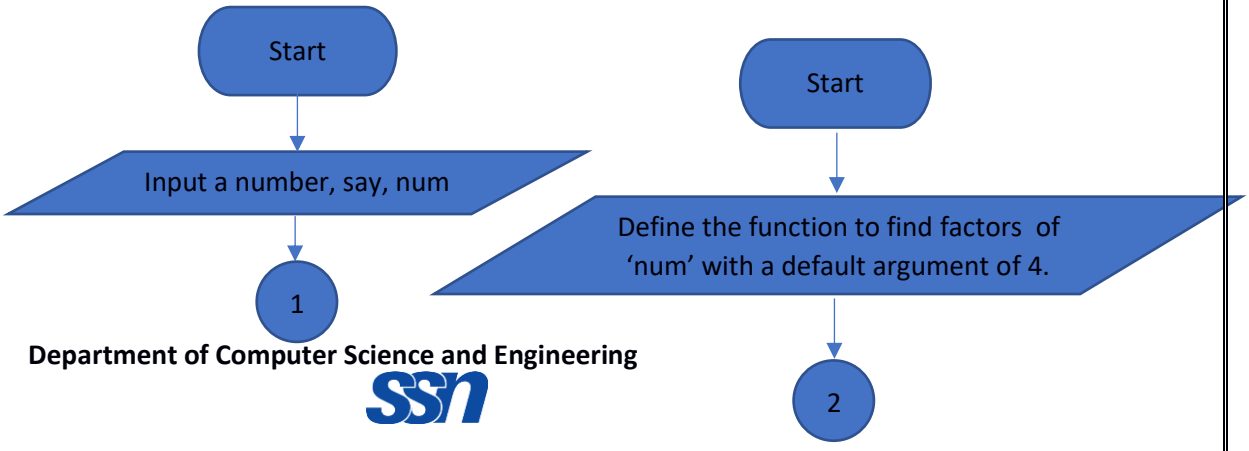
To print the factors of a number and to use a default value if the number is not given by the user.

Algorithm

Input: A number and a choice to input a number
Output: Conclusion of the triangle's existence

Step 1: Start
Step 2: Define a function to find the factors of a number with the default argument being 4. The function checks if the number is divisible by any natural number less than the number and prints those divisors as the factors of the number.
Step 3: Ask if the user wants to input a number. If no, go to Step 5.
Step 4: Read a number as input from the user.
Step 5: Call the function to find the factors and use the default value if the number has not been input by the user.
Step 6: End

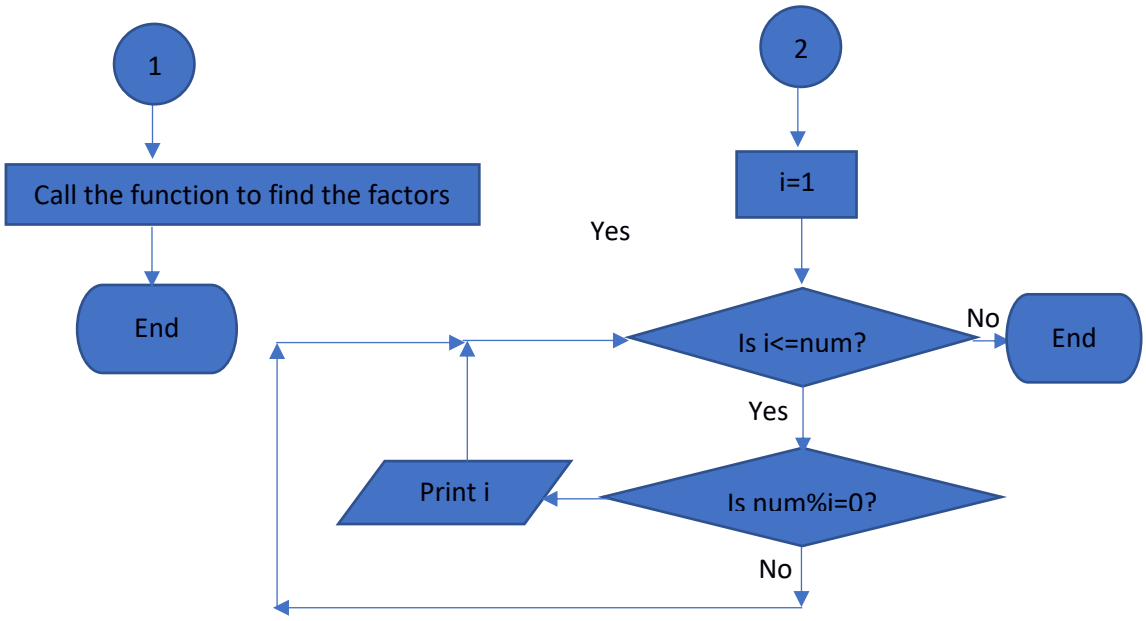
Flowchart



UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5



Program

```
# Python program to print the factors of a number and use a default
input if the number is not entered by the user.

# Function definition
def factors(num=4):
    print("The factors of",num,"are: ")
    for i in range(1,num+1):
        if num%i==0:
            print(i,end=" ")
    print()
ch=int(input("Want to enter a number?(1-yes,2-no): "))
if (ch==1):
    n=int(input("Enter a number: "))
    factors(n)     # Function call
elif (ch==2):
    factors()      # Function call with default value
```

UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

Output

Input:

36, NIL

Output:

```
Want to enter a number?(1-yes,2-no): 1
Enter a number: 36
The factors of 36 are:
1 2 3 4 6 9 12 18 36

Want to enter a number?(1-yes,2-no): 2
The factors of 4 are:
1 2 4
```

Results / Inferences

Program for printing the factors of a given number using a default argument is written and executed.

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

3. Powers of a number

Aim

To find the power of a number given the base and the exponent.

Algorithm

Input: The base and the exponent
Output: The required power of the number

Pre-condition: The exponent must be a positive integer
Post-condition: The result must be a positive real number

Step 1: Start
Step 2: Define a function to calculate the power of a number taking the base and exponent as parameters. The function multiplies the base with itself repeatedly for as many times as is the value of the exponent and returns the result.
Step 3: Read the base and exponent values as input from the user.
Step 4: Call the function and print the return value.
Step 5: End

Program

```
# Python program to calculate the required power of a number.

# Function definition
def calc_exp(base,exp):
    res=1
    for i in range(0,exp):
      res*=base
```

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

```
    return res
# To get the input from the user
print("Enter the required base and exponent values: ")
b=eval(input("Base: "))
e=int(input("Exponent (integer>0 only): "))
if e<=0:
    print("Exponent must be an integer greater than 0.")
    exit()
# Function call
print("Result:",calc_exp(b,e))
```

Output

Input:

2 7

Output:

```
Enter the required base and exponent values:
Base: 2
Exponent (integer>0 only): 7
Result: 128
```

Results / Inferences

Program for calculating the required power of a number is written and executed.

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

4. Sum of digits and Reverse of a number

Aim

To find the reverse of a number and calculate the sum of its digits.

Algorithm

Input: Any number
Output: The reverse of the number and the sum of its digits

Pre-condition: The number must be a positive integer
Post-condition: The reverse and the sum of the digits must also be integers.

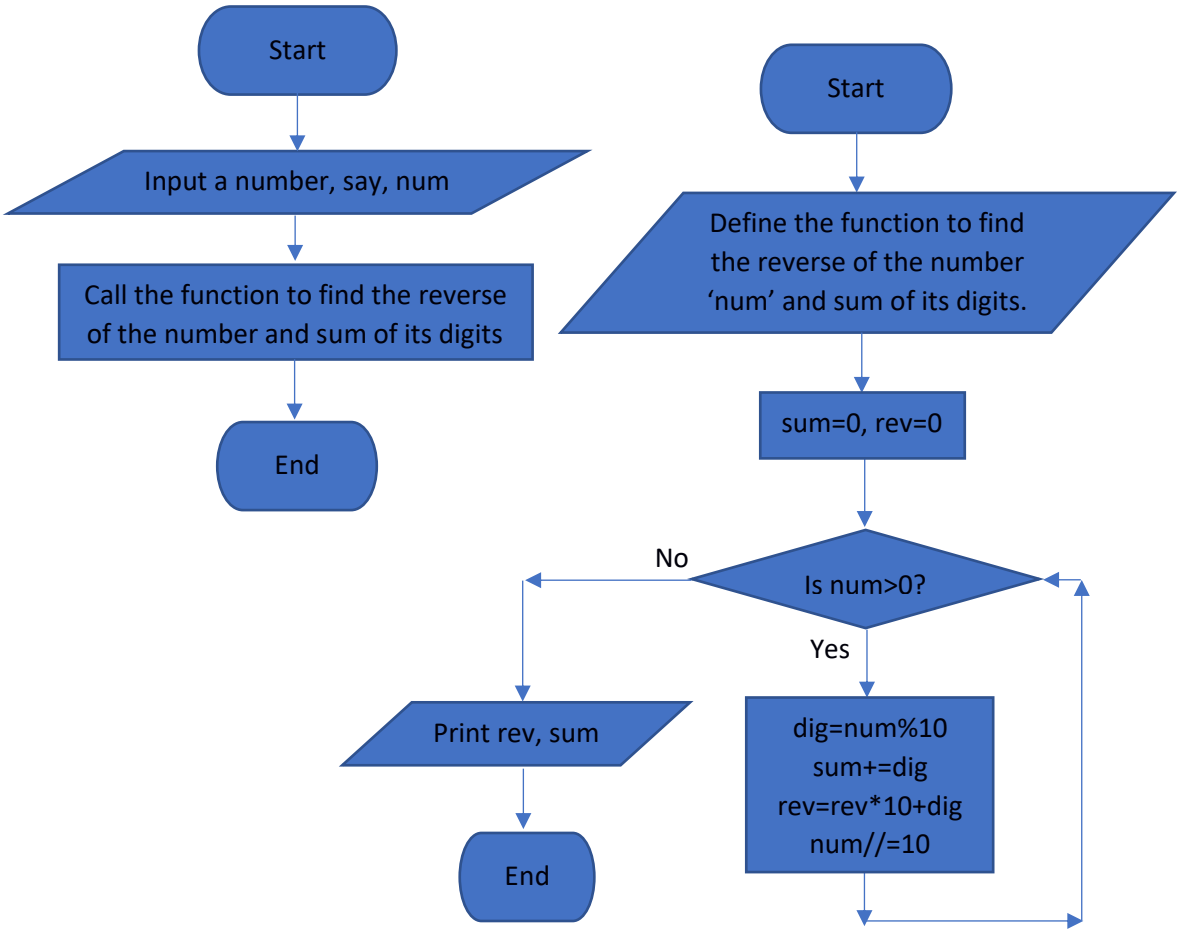
Step 1: Start
Step 2: Define a function to find the reverse of a number and the sum of its digits. The function extracts each digit from the number and sums them to get the digits. It also reverses the number by adding every extracted digit successively to the previous one multiplied by 10. The reverse and sum are printed.
Step 3: Read a number as input from the user.
Step 4: Pass the number as the argument and call the function.
Step 5: End

UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

Flowchart



Program

```
# Python program to find the reverse of a given number and also
calculate the sum of its digits.

# Function definition
def reversum(num):
```

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

```
    print("The reverse of",num,"is",end=" ")
    sum,rev=0,0
    while(num>0):
        dig=num%10
        sum+=dig
        rev=rev*10+dig
        num//=10
    print(rev,".")
    print("The sum of its digits is",sum,".")
    return rev, sum

# To get the number as input from the user
n=int(input("Enter a number: "))
# Function call
reversum(n)
```

Output

Input:

425

Output:

```
Enter a number: 425
The reverse of 425 is 524 .
The sum of its digits is 11 .
```

Results / Inferences

Program for finding the reverse of a number and calculating the sum of its digits is written and executed.

UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

5. Area of Circle and Sphere

Aim

To calculate the area of a circle and a sphere when the radius is known.

Algorithm

Input: The radius of the circle and the sphere
Output: The area of the circle and the sphere

Pre-condition: The radius should be a positive real number
Post-condition: The area must always be positive.

Step 1: Start
Step 2: Define a function that accepts the radius as the parameter and calculates the area of a circle. The function returns the area, i.e., $pi*r^2$.
Step 3: Define a function that accepts the radius as the parameter and calculates the area of a sphere. The function returns the area, i.e., $4*pi*r^2$.
Step 4: Read the value of the radius as input from the user.
Step 5: Give the user a choice to find the area of the associated circle or sphere.
Step 6: If the choice is circle, call the function for the circle, passing the radius as the argument and print the return value.
Step 7: If the choice is sphere, call the function for the sphere, passing the radius as the argument and print the return value.
Step 8: End

UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

Program

```
# Python program to calculate the area of a circle and a sphere given
the radius.

# Importing math module
import math
# Function definition
def circle(radius):
    return math.pi*(radius**2)
def sphere(radius):
    return 4*math.pi*(radius**2)

# To get the radius as input
r=float(input("Enter the radius: "))
choice=int(input("1-Circle or 2-Sphere?: "))
if (choice==1):
  print("Area of circle:",circle(r))       # Function call
elif (choice==2):
  print("Area of sphere:",sphere(r))       # Function call
else:
  print("Invalid choice!")
```

Output

Input:

1 (choice 1), 1 (choice 2)

Output:

```
Enter the radius: 1
1-Circle or 2-Sphere?: 1
Area of circle: 3.141592653589793
```

UGE1197
Programming in Python Lab
AY: 2020-2021

Krithika Swaminathan
Date: 25/01/2021
S03018

Ex. No.: 5

```
Enter the radius: 1
1-Circle or 2-Sphere?: 2
Area of sphere: 12.566370614359172
```

Results / Inferences

Program for calculating the area of a circle and sphere given the radius is written and executed.