|  |  |
| --- | --- |
| **Ex. 6** | **EXPLORING USER-DEFINED FUNCTIONS** |
| **Date:26/02/24** |  |

Aim:

To explore user-defined functions in Python by writing programs for the following:

1. Write a function that can generate Pythagorean triplets up to N. Obtain N from the user and pass it as an argument to your function.
2. Write a function to generate the sum of the sine series, sin(𝑥) and another to generate the sum of the cosine series cos(𝑥) up to N terms. These functions must have two arguments, x and N, where N is an optional argument. Here x is the angle in radians. Get the angle in degrees from the user and convert to radians. Pass this angle to the functions. The computation of the sum of the sine series also requires the computation of the factorial of a number, so write a function to compute factorial as well.
3. Write a menu-driven program to perform the following tasks. Write a separate function for each task.
   * Check if a given integer is an Armstrong number.
   * Check if a given integer is a Deficient number.
   * Check if a given integer is a Palindrome.
   * Exit the program.

Algorithm:

(a)

Step 1: Prompt the user for the upper limit of the triplet values

Step 2: Define a function named triplets that takes an integer n as input

Step 3: Use nested loops with variables m and n to generate sides a, b, and c where a = m^2 - n^2, b = 2mn, c = m^2 + n^2

Step 4: Check if a, b, and c form a Pythagorean triplet and c is less than the input limit

Step 5: Print valid triplets

Step 6: Call the triplets function with the user's input value

(b)

Step 1: Import the math module for any necessary mathematical operations

Step 2: Prompt the user for an angle in degrees and the number of terms in the series

Step 3: Convert the angle from degrees to radians

Step 4: Define a recursive function fact to calculate the factorial of a number

Step 5: Define the function sine that computes the sine of an angle using its Taylor series expansion

- Initialize sum to 0

- Use a loop to add terms of the series up to n, adjusting the sign and factorial for each term

- Print the sum after each term is added

Step 6: Define the function cos that computes the cosine of an angle using its Taylor series expansion

- Similar to sine but adjust the power and factorial for cosine terms

- Print the sum after each term is added

Step 7: Call the sine and cos functions with the number of terms and the angle in radians

(c)

Step 1: Import the math module

Step 2: Define the function armstrong to determine if a number is an Armstrong number

Step 3: Define the function deficient to check if a number is deficient

Step 4: Define the function palindrome to verify if a number is a palindrome

Step 5: Display a menu with options to check for Armstrong number, deficient number, palindrome, or exit the program

Step 6: Continuously prompt the user for their choice until they decide to exit

- If choice is 1, prompt for a number and call the armstrong function

- If choice is 2, prompt for a number and call the deficient function

- If choice is 3, prompt for a number and call the palindrome function

- If choice is 4, exit the loop and terminate the program

- If the input is invalid, display an error message

Program:

(a)

num=int(input("Enter the value of n:"))

def triplets(n):

   a,b,c=0,0,0

   for m in range(1,num+1):

      for n in range(1,m):

        a=(m\*m)-(n\*n)

        b=(2\*m\*n)

        c=(m\*m)+(n\*n)

        if((a\*a)+(b\*b)==(c\*c) and c<num):

            print(a,b,c)

        else:

          break

triplets(num)

(b)

import math

deg=int(input("Enter the angle in degrees:"))

n=int(input("Enter the number of inputs:"))

x=deg\*3.14/180

def fact(num):

    if(num==0):

      return 1

    else:

      return num\*fact(num-1)

def sine(n,x):

    sum=0

    for i in range(n):

        sum+= pow(-1,i)\*pow(x,2\*i+1)/fact(2\*i+1)

        print("The sum of the sine series is:",sum)

def cos(n,x):

    sum=0

    for i in range(n):

         sum+=pow(-1,i)\*pow(x,2\*i)/fact(2\*i)

         print("The sum of the cosine series is:",sum)

sine(n,x)

cos(n,x)

(c)

import math

def armstrong(num):

    ognum=num

    sum=0

    n=0

    while(num!=0):

        num=num//10

        n+=1

    num=ognum

    while ognum!=0:

         number=ognum%10

         sum+=math.pow(number,n)

         ognum=ognum//10

         if(sum==num):

            print("It's an armstrong number")

         else:

            print("It's not an armstrong number")

def deficient(num):

   sum=0

   for i in range(1,(num//2)+1):

       if(num%i==0):

         sum+=i

       if(sum<num):

         print("It's a deficient number")

       else:

          print("It's not a deficient number")

def palindrome(num):

    string=str(num)

    k=string[::-1]

    if(k==string):

      print("It's a palindrome")

    else:

     print("It's not a palindrome")

while(1):

  print("(1) Armstrong number:")

  print("(2) Deficient number:")

  print("(3) Palindrom:")

  print("(4) EXit")

  choice=int(input("Enter a choice:"))

if(choice==1):

   num=int(input("Enter a number:"))

   armstrong(num)

elif(choice==2):

    num=int(input("Enter a number:"))

    deficient(num)

elif(choice==3):

    num=int(input("Enter a number:"))

    palindrome(num)

elif(choice==4):

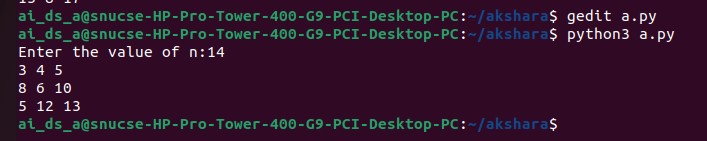
    break

else:

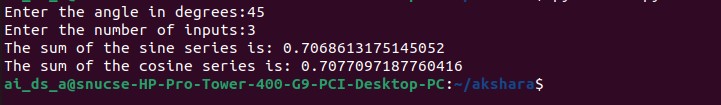
   print("Invalid choice")

Screenshot of Output:

(a)



(b)



**A screenshot of a computer program

Description automatically generated**

(c)

Result:

Thus, programs have been written and executed to explore user-defined functions in Python.