Kalyani Government Engineering College  
Department of Computer Application  
Python Programming Lab – MCAN191, Year: 2022-2023

Assignment: 3

# 1. Write a program to print numbers from 1 to 10.

for i in range(1,11):

  print(i)

o/p:-

1

2

3

4

5

6

7

8

9

10

# 2. Write a program that asks the user for a positive integer value. The program should

# calculate the sum of all the integers from 1 up to the number entered. For example, if the

# user enters 20, the loop will find the sum of 1, 2, 3, 4, ... 20.

n = int(input("Enter any integer: "))

sum = 0

for i in range(n+1):

    sum += i

print(f'Sum = {sum}')

o/p:-

Enter any integer: 6

Sum = 21

# 3. Write a program that prompts the user to input a number and prints its mulitiplication table.

n = int(input("Enter any integer: "))

for i in range(1, 11):

    print(f'{n} x {i} = {n\*i}')

o/p:-

Enter any integer: 7

7 x 1 = 7

7 x 2 = 14

7 x 3 = 21

7 x 4 = 28

7 x 5 = 35

7 x 6 = 42

7 x 7 = 49

7 x 8 = 56

7 x 9 = 63

7 x 10 = 70

# 4. Write a program that prompts the user to input a number and prints its factorial. The

# factorial of an integer n is defined as n! = 1 x 2 x 3 x ... x n; if n > 0 = 1; if n = 0 For instance,

# 6! can be calculated as 1 x 2 x 3 x 4 x 5 x 6.

n = int(input("Enter any integer: "))

f = 1

for i in range(2, n+1):

    f \*= i

print(f'Factorial = {f}')

o/p-

Enter any integer: 5

Factorial = 120

# 5. Two numbers are entered through the keyboard. Write a program to find the value of one

# number raised to the power of another

a,b = map(int, input("Enter two integer: ").split())

c = a

for i in range(1,b):

   c \*= a

print(f'{a}^{b} = {c}')

o/p-

Enter two integer: 3 4

3^4 = 81

# 6. Write a program that prompts the user to input a number and reverse its digits. For

# example, the reverse of 12345 is 54321; reverse of 5600 is 65.

n = int(input("Enter any integer: "))

temp = n

rev = 0

while n > 0:

    rev = rev\*10 + n%10

    n //= 10

print(f'Revesrse of {temp} is {rev}')

o/p-

Enter any integer: 12345

Revesrse of 12345 is 54321

# 7. Write a program that asks the user to input a positive integer. Your program should find

# and display the sum of digits of number. For example, sum of digits of number 32518 is

# 3+2+5+1+8 = 19.

n = int(input("Enter any integer: "))

temp = n

sum = 0

while n > 0:

    sum += n%10

    n //= 10

print(f'Sum of digits of {temp} is {sum}')

o/p-

Enter any integer: 32518

Sum of digits of 32518 is 19

# 8. A palindromic number is a number that remains the same when its digits are reversed.

# For example, 16461. Write a program that prompts the user to input a number and

# determine whether the number is palindrome or not.

n = int(input("Enter any integer: "))

temp = n

rev = 0

while n > 0:

    rev = rev\*10 + n%10

    n //= 10

if rev == temp:

    print(f'{temp} is palindrome')

else:

print(f'{temp} is not palindrome')

o/p-

Enter any integer: 121

121 is palindrome

# 9. Write a program that prompts the user to input a decimal integer and display its binary

# equivalent.

def decToBin(n):

  if n >= 1:

    decToBin(n // 2)

  print (n % 2, end='')

n = int(input("Enter a decimal number: "))

decToBin(n)

o/p-

Enter a decimal number: 9

01001

# 10. Write a program that prompts the user to input a binary number and display its decimal

# equivalent.

n = int(input("Enter a binary number: "))

temp = n

dec = 0

i = 0

while n > 0:

    a = 1

    for j in range(i):

        a \*= 2

    i += 1

    dec = dec + n%2\*a

    n //= 10

print(f'The decimal equivalent of ({temp})2 is ({dec})10')

o/p-

Enter a binary number: 1001

The decimal equivalent of (1001)2 is (9)10

# 11. Write a program that prompts the user to input a positive integer. It should then output a

# message indicating whether the number is a prime number. A prime number is a number

# that is evenly divisible only by itself and 1. For example, the number 5 is prime because

# it can be evenly divided only by 1 and 5. The number 6, however, is not prime because it

# can be divided evenly by 1, 2, 3, and 6.

n = int(input("Enter a number: "))

is\_prime = True

for i in range(2,n//2):

    if n%i == 0:

        is\_prime = False

        break

if is\_prime:

    print(f'{n} is prime')

else:

    print(f'{n} is not prime')

o/p-

Enter a number: 7

7 is prime

# 12. Write a program that prompts the user to input two numbers and display its HCF. The

# Highest Common Factor (HCF) also called the Greatest Common Divisor (GCD) of two

# whole numbers, is the largest whole number that's a factor of both of them.

a,b = map(int, input("Enter two integer: ").split())

min = a

if min > b:

    min = b

hcf = 1

for i in range(1,min+1):

    if a%i == 0 and b%i == 0:

        hcf = i

print(f'HCF of {a} {b} = {hcf}')

o/p-

Enter two integer: 8 12

HCF of 8 12 = 4

# 13. Write a program to enter the numbers till the user wants and at the end it should display

# the count of positive, negative and zeros entered.

total = 0

pos = 0

neg = 0

zeros = 0

stop = False

while not stop:

    n = int(input(f'Enter number {total+1}: '))

    total += 1

    if n < 0:

        neg += 1

    elif n > 0:

        pos += 1

    else:

        zeros += 1

    flag = int(input("press 1 to continue 0 to exit: "))

    if flag == 0:

        stop = True

print(f'Total numbers = {total}\nPositives = {pos}\nNegatives = {neg}\nZeros = {zeros}')

o/p-

Enter number 1: 5

press 1 to continue 0 to exit: 1

Enter number 2: -7

press 1 to continue 0 to exit: 0

Total numbers = 2

Positives = 1

Negatives = 1

Zeros = 0

# 14. Write a program to enter the numbers till the user wants and at the end the program

# should display the largest and smallest numbers entered

total = 0

max = None

min = None

stop = False

num = []

while not stop:

    n = int(input(f'Enter number {total+1}: '))

    num.append(n)

    total += 1

    if max == None or max < n:

        max = n

    if min == None or min > n:

        min = n

    flag = int(input("press 1 to continue 0 to exit"))

    if flag == 0:

        stop = True

print(f'Total numbers = {num}\nMaximum = {max}\nMinimum = {min}')

o/p-

Enter number 1: 45

press 1 to continue 0 to exit: 1

Enter number 2: 21

press 1 to continue 0 to exit: 1

Enter number 3: -7

press 1 to continue 0 to exit: 1

Enter number 4: 5

press 1 to continue 0 to exit: 0

Total numbers = [45, 21, -7, 5]

Maximum = 45

Minimum = -7

# 15. An Armstrong number of three digits is an integer such that the sum of the cubes of its

# digits is equal to the number itself. For example, 371 is an Armstrong number since 33 +

# 73 + 13 = 371. Write a program to find all Armstrong number in the range of 0 and 999

def isArmstrong(num):

  arm = False

  temp = num

  sum = 0

  while num:

    sum += (num%10)\*\*3

    num //= 10

  if sum == temp:

    arm = True

  return arm

for i in range(1000):

  if(isArmstrong(i)):

    print(i)

o/p-

0

1

153

370

371

407

# 16. Write a program to obtain the first 25 numbers of a Fibonacci sequence. In a Fibonacci

# sequence the sum of two successive terms gives the third term. Following are the first

# few terms of the Fibonacci sequence:

# 0 1 1 2 3 5 8 13 21 34 55 89...

def fibo(n):

  a = 0

  b = 1

  for i in range(n):

    print(a, end=' ')

    sum = a + b

    a = b

    b = sum

fibo(25)

o/p-

0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765 10946 17711 28657 46368

# 17. Write a program to add first seven terms of the following series using a for loop:

# 1/1! + 2/2! + 3/3! +...

def fact(n):

  if n == 1:

    return 1

  return n\*fact(n-1)

def sumOfSeries(terms):

  sum = 0

  for i in range(1, terms+1):

    sum += i / fact(i)

  return sum

print(sumOfSeries(7))

o/p-

2.7180555555555554

# 18. Compute the sum up to n terms in the series

# 1 - 1/2 + 1/3 - 1/4 + 1/5 -... 1/n

# where n is a positive integer and input by user.

def sumOfSeries(n):

  sum = 0

  for i in range(1, n+1):

    if i%2 == 0:

      sum -= 1 / i

    else:

      sum += 1 / i

  return sum

n = int(input("Enter n: "))

print(sumOfSeries(n))

o/p-

Enter n: 5

0.7833333333333332

# 19. Floyd's triangle is a right-angled triangular array of natural numbers as shown below:

# Write a program to print the Floy'd triangle.

# 1

# 2  3

# 4  5  6

# 7  8  9  10

# 11 12 13 14 15

def floydTriangle(n):

  a = 1

  for i in range(n):

    for j in range(i+1):

      print("%2d"%a, end='  ')

      a += 1

    print()

n = int(input("Enter n: "))

floydTriangle(n)

o/p-

Enter n: 6

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

16 17 18 19 20 21

# 20. Write a program to compute sin x for given x. The user should supply x and a positive

# integer n. We compute the sine of x using the series and the computation should use all

# terms in the series up through the term involving xn

# sin x = x - x^3/3! + x^5/5! - x^7/7! + x^9/9!...

def fact(n):

  if n <= 1:

    return 1

  return n\*fact(n-1)

def sin(x,n):

  sum = 0

  a = 1

  for i in range(1, n+1):

    if i%2 == 0:

      sum -= x\*\*a / fact(a)

    else:

      sum += x\*\*a / fact(a)

    a += 2

  return sum

x,n = map(int, input("Enter x and n: ").split())

print(sin(x,n))

o/p-

Enter x and n: 2 3

0.9333333333333333

# 21. Write a program to compute cosine of x. The user should supply x and a positive integer

# n. We compute the cosine of x using the series and the computation should use all terms

# in the series up through the term involving xn

# cos x = 1 - x^2/2! + x^4/4! - x^6/6! .....

def fact(n):

  if n <= 1:

    return 1

  return n\*fact(n-1)

def sin(x,n):

  sum = 0

  a = 0

  for i in range(1, n+1):

    if i%2 == 0:

      sum -= x\*\*a / fact(a)

    else:

      sum += x\*\*a / fact(a)

    a += 2

  return sum

x,n = map(int, input("Enter x and n: ").split())

print(sin(x,n))

o/p-

Enter x and n: 2 3

-0.33333333333333337

# 22. Write a program that generates a random number and asks the user to guess what the

# number is. If the user's guess is higher than the random number, the program should

# display "Too high, try again." If the user's guess is lower than the random number, the

# program should display "Too low, try again." The program should use a loop that repeats

# until the user correctly guesses the random number. Program should count and display

# number of tries to win the game.

import random

win = False

n = random.randint(1,10)

attempts = 0

while not win:

  attempts += 1

  print(f"Chance number: {attempts}")

  guess = int(input("Enter your number: "))

  if guess > n:

    print("Too high, try again.")

  elif guess < n:

    print("Too low, try again.")

  else:

    win = True

    print("Congratulations, you won.\nNo of attempts:", attempts)

o/p-

Chance number: 1

Enter your number: 4

Too low, try again.

Chance number: 2

Enter your number: 5

Too low, try again.

Chance number: 3

Enter your number: 6

Too low, try again.

Chance number: 4

Enter your number: 7

Congratulations, you won.

No of attempts: 4