MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

NATIONAL TECHNICAL UNIVERSITY

«KHARKIV POLYTECHNIC INSTITUTE»

Department of Software Engineering and Management Information Technologies

Report from lab № 4

discipline «Fundamentals of python»

Executed by: Student of group 1.КН.201.8e1

CHUKWU IRELE AMORIN

Checked by: Senior lecturer

Kharkiv

2019

**Laboratory work №4**

**Loop “for”**

*1. "Row - 1"*

Two integers *A* and *B* are given (with *A* ≤ *B*). Print all numbers from *A* to *B* inclusive.

*2. "Row - 2"*

Given two integers A and B. Print all numbers from A to B inclusive, in ascending order if A <B, or in decreasing order otherwise.

*3. "Row - 3"*

Two integers *A* and *B* are given, *A*> *B*. Print all odd numbers from *A* to *B* inclusive, in descending order. In this task, you can do without an if statement.

*4. "The sum of ten numbers"*

Given 10 integers. Calculate their amount. Write a program that uses the least number of variables.

*5. "The sum of N numbers"*

Given a few numbers. Calculate their amount. First enter the number of numbers *N*, then enter exactly *N* integers. What is the smallest number of variables needed to solve this problem?

*6. "Sum of cubes"*

For a given positive integer *n*, calculate the sum 13 + 23 + 33 + ... + n3.

*7. "Factorial"*

The factorial of n is the product 1 × 2 × ... × n. Designation: *n* !.

Given a positive integer *n*, calculate the value of *n* !. Using the math library in this task is prohibited.

*8. “The sum of factorials”*

For this natural number *n*, calculate the sum 1! +2! +3! + ... + n! 1! +2! +3! + ... + *n* !. In solving this problem, you can use only one cycle. Using the math math library in this task is prohibited.

*9. "Number of zeros"*

Given *N* numbers: first enter the number *N*, then enter exactly *N* integers. Count the number of zeros among the entered numbers and print this number. You need to count the number of numbers equal to zero, not the number of digits.

*10. "Ladder"*

Given a positive integer *n* ≤ 9, output a ladder from *n* steps, the *i*-th step consists of numbers from 1 to *i* without spaces.

*11. Lost Card*

For a board game, cards with numbers from 1 to *N* are used. One card is lost. Find her, knowing the numbers of the remaining cards.

Given the number *N*, then *N* - 1 number of the remaining cards (various numbers from 1 to *N*). The program should display the number of the lost card.

Arrays and similar data structures cannot be used.

Solution:

import sys

#If you want an inclusive range then pass stop argument value as stop+step.

#for i in range(start, stop, step):

#1

def row1():

    A = int(input("input the value of A : "))

    B = int(input(" input the value  B : "))

    for num in range(B,A):

        print(num)

#2

def row2():

    A = int(input("input the value of A : "))

    B = int(input(" input the value  B : "))

    if A < B :

        step = 1

    elif B < A:

        step = -1

    start = A

    stop = B + step

    for num in range(start,stop,step):

         print(num)

#3

def row3():

    A = int(input("input the value of A : "))

    B = int(input(" input the value  B : "))

    step = -1

    start = A

    stop = B + step

    for num in range(start,stop,step):

        while (num % 2) != 0 :

            print(num)

            break

#4

def sumr():

    count = 1

    numbers =[]

    sumT = 0

    while count <= 10 :

        numbers.append(int(input("input thr value :  ")))

        count += 1

    for \_ in numbers :

        sumT += \_

    print("the  sum of numbers are : ", sumT)

#5

def suumN():

    count = 1

    numbers =[]

    sumT = 0

    N = int (input("enter value of N : "))

    while count <= N :

        numbers.append(int(input("input thr value :  ")))

        count += 1

    for \_ in numbers :

        sumT += \_

    print("the  sum of numbers are : {0} and the least num of variable is : {1}  ".format(sumT,4))

#6

def cubes():

    count = 1

    numbers =[]

    sumT = 0

    N = int (input("enter value of N : "))

    while count <= N :

        numbers.append(int(input("input thr value :  ")))

        count += 1

    for \_ in numbers :

        cube  = \_ \*\* 3

        sumT += \_

    print("the  sum of cubes are :  ", sumT)

#7

def factorial():

     n = int(input("input the value of  N : "))

     fact = 1

     for i in range(1,n+1):

         fact = fact \* i

     print ("The factorial of {0} is : ".format(n))

     print (fact)

#8

def factorial\_sum():

     n = int(input("input the value of  N : "))

     sumT = 0

     for i in range(1, n+1):

         fact = 1

         for j in range(1,i+1):

             fact = fact \* j

         sumT += fact

     print ("The factorial of {0} is : ".format(n))

     print (sumT)

#9

def zeros():

    n = int(input("enter the value for n : "))

    numbers = []

    count  = 1

    while count <= n :

        numbers.append(int(input("input  value :  ")))

        count += 1

    zero = 0

    for \_ in numbers:

        if \_ is 0 :

            zero += 1

    print("the number of zeros is : " , zero)

#10

def ladder():

    n = int(input("enter the number "))

    for \_ in range(1,n+1):

        a = []

        for i in range(1,\_+1):

            print(i,end ="")

        print()

#11

def card():

    n = int(input("enter the number of cards : "))

    words = ""

    for \_ in range(1,n+1):

        words += str(\_)

    i = 1

    while i < n:

        j  = input("enter remaining number : ")

        words = words.replace(j,"")

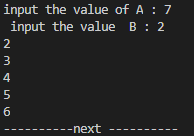
        i += 1

    print("the missing number is : ", words )

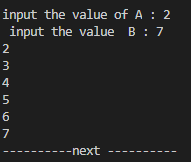
if \_\_name\_\_ == "\_\_main\_\_":

    card()

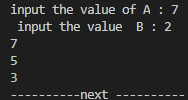
No1:



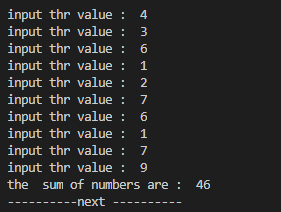
No2:



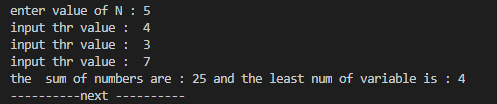
No3:



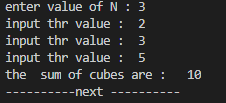
No4:



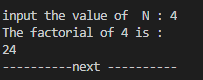
No5:



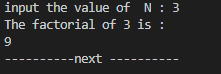
No6:



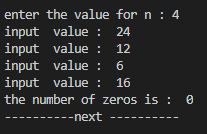
No7:



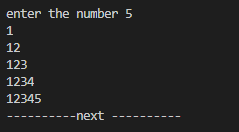
No8:



No9:



No10:



No11:

