1914626

Nguyễn Trung Nguyên

**Lab 1**

##Bài 1: Hoàn thành 25/150 bài tập cơ bản ở trang: https://www.w3resource.com/python-exercises/python-basic-exercises.php

## 1/150 Write a Python program to print the following string in a specific format (see the output).

print("Twinkle, twinkle, little star, \n\tHow I wonder what you are! \n\t\tUp above the world so high, \n\t\tLike a diamond in the sky. \nTwinkle, twinkle, little star, \n\tHow I wonder what you are!")

print("\n 2:\n")

print("Twinkle, twinkle, little star,")

print("How I wonder what you are!".rjust(35))

print("Up above the world so high".rjust(45))

print("Like a diamond in the sky.".rjust(45))

print("Twinkle, twinkle, little star,")

print("How I wonder what you are!".rjust(35))

## 2/150 Write a Python program to get the Python version you are using.

import platform

import sys

print("Python version:", platform.python\_version())

print("Tuple:",platform.python\_version\_tuple())

print("Type:",type(platform.python\_version\_tuple()))

print("Python version:",sys.version,"\nVersion info:",sys.version\_info)

## 3/150 Write a Python program to display the current date and time.

import datetime

now = datetime.datetime.now()

print("Current date and time: ", now.strftime("%Y-%m-%d %H:%M:%S"))

## 4/150 Write a Python program which accepts the radius of a circle from the user and compute the area.

from math import pi

r = float(input ("Input the radius of the circle: "))

print("The area of the circle with radius = " + str(r) + " is:\n S = " + str(pi \* r \*\* 2))

## 5/150 Write a Python program which accepts the user's first and last name and print them in reverse order with a space between them.

fname = input("Input your first name: ")

lname = input("Input your last name: ")

print("Hello " + lname + " " + fname)

## 6/150 Write a Python program which accepts a sequence of comma-separated numbers from user and generate a list and a tuple with those numbers.

values = input("Input some comma seprated numbers: ")

*list* = values.split(",")

*tuple* = tuple(*list*)

print("List: ", *list*)

print("Tuple: ", *tuple*)

## 7/150 Write a Python program to accept a filename from the user and print the extension of that.

filename = input("Input the filename: ")

f\_exts = filename.split(".")

print("The extension of the file is: " + f\_exts[-1])

## 8/150 Write a Python program to display the first and last colors from the following list.

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

color\_list = []

if n < 2:

    print("Must enter at least 2 colors!")

    exit()

else:

    for i in range(n):

        color\_list.append(input("Color input %d:" %(i + 1)))

print("Your color is: ", color\_list)

print("First color: %s and Last color: %s" %(color\_list[0], color\_list[-1]))

## 9/150 Write a Python program to display the examination schedule. (Hien thi lich thi)

exam\_date = (23,8,2022)

print("The examination will start from: %i / %i / %i"%exam\_date)

## 10/150 Write a Python program that accepts an integer (n) and computes the value of n+nn+nnn.

a = int(input("Input an integer: "))

n1 = int( "%s" % a )

n2 = int( "%s%s" % (a,a) )

n3 = int( "%s%s%s" % (a,a,a) )

print("Values of n+nn+nnn is: ", n1 + n2 + n3)

## 11/150 Write a Python program to print the documents (syntax, description etc.) of Python built-in function(s).

## in các tài liệu (cú pháp, mô tả, v.v.) của (các) hàm tích hợp sẵn trong Python.

print("ABS: ",abs.\_\_doc\_\_)

print("\nType:\n",type.\_\_doc\_\_)

print("\nInt",int.\_\_doc\_\_)

## 12/150 Write a Python program to print the calendar of a given month and year.

import calendar

y = int(input("Input the year: "))

m = int(input("Input the month: "))

print(calendar.month(y,m))

## 13/150 Write a Python program to print the following 'here document'.

print(""" A string that you don't to escape

            This is a multiline

                Amazing """)

## 14/150 Write a Python program to calculate number of days between two dates.

from datetime import date

f\_date = date(2022, 8, 1)

l\_date = date(2022, 8, 23)

delta = l\_date - f\_date

print(delta.days)

## 15/150 Write a Python program to get the volume of a sphere with radius 6.

## để lấy thể tích của khối cầu có bán kính 6.

from math import pi

r = 6.0

V = 4.0 / 3.0 \* pi \* r \*\* 3

print("The volume of the sphere is: ", V)

## 16/150 Write a Python program to get the difference between a given number and 17, if the number is greater than 17 return double the absolute difference.

## để nhận sự khác biệt giữa một số nhất định và 17, nếu số lớn hơn 17 trả về gấp đôi sự khác biệt tuyệt đối.

n = int(input("Input a number: "))

def difference(*n*):

    if *n* <= 17:

        return 17 - *n*

    else:

        return (*n* - 17) \* 2

print(difference(n))

## 17/150  Write a Python program to test whether a number is within 100 of 1000 or 2000

n = int(input("Input a number: "))

def near\_thousand(*n*):

    return ((abs(1000 - *n*) <= 100) or (abs(2000 - *n*) <= 100))

print(near\_thousand(n))

## 18/150 Write a Python program to calculate the sum of three given numbers, if the values are equal then return thrice of their sum.

## nếu các giá trị bằng nhau thì trả về gấp ba lần tổng của chúng.

x = int(input("Input a number First: "))

y = int(input("Input a number Second: "))

z = int(input("Input a number Third: "))

def sum\_thrice(*x*,*y*,*z*):

    sum = *x* + *y* + *z*

    if *x* == *y* == *z*:

        sum = sum \* 3

    return sum

print(sum\_thrice(x,y,z))

## 19/150 Write a Python program to get a new string from a given string where "Is" has been added to the front. If the given string already begins with "Is" then return the string unchanged.

*str* = input("Input a string: ")

def new\_string(*str*):

    if *str*[:2] == "Is":

        return *str*

    else:

        return "Is" + *str*

print(new\_string(*str*))

## 20/150 Write a Python program to get a string which is n (non-negative integer) copies of a given string.

## lấy một chuỗi là n (số nguyên không âm) nhân bản của một chuỗi đã cho.

*str* = input("Input a string: ")

n = int(input("Input the number of copies: "))

def larger\_string(*str*, *n*):

    result = ""

    for i in range(*n*):

        result = result + *str*

    return result

print(larger\_string(*str*, n))

## 21/150 Write a Python program to find whether a given number (accept from the user) is even or odd, print out an appropriate message to the user.

num = int(input("Enter a number: "))

mod = num % 2

if mod > 0:

    print(num, "is an odd number")

else:

    print(num, "is an even number")

## 22/150 Write a Python program to count the number 4 in a given list.

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

if n < 0:

    exit()

a = []

for i in range(n):

    a.append(int(input("Nhap so thu %d:" % (i + 1))))

print("Your array is:", a)

def list\_count\_4(*a*):

    count = 0

    for num in *a*:

        if num == 4:

            count = count + 1

    return count

print("Number of occurrences of 4 is:",list\_count\_4(a))

## 23/150 Write a Python program to get the n (non-negative integer) copies of the first 2 characters of a given string.

# Return the n copies of the whole string if the length is less than 2.

*str* = input("Input a string: ")

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

def substring\_copy(*str*, *n*):

    flen = 2

    if flen > len(*str*):

        flen = len(*str*)

    substr = *str*[:flen]

    result = ""

    for i in range(*n*):

        result = result + substr

    return result

print(substring\_copy(*str*, n))

## 24/150 Write a Python program to test whether a passed letter is a vowel or not.

n = input("Input a char: ")

def is\_vowel(*char*):

    all\_vowels = "ueoai"

    return *char* in all\_vowels

print(is\_vowel(n))

## 25/150 Write a Python program to check whether a specified value is contained in a group of values.

n = int(input("Input the number of arrays: "))

if n < 0:

    exit()

a=[]

for i in range(n):

    a.append(int(input("Number input %d" %(i + 1))))

print("Your array is:", a)

n2 = int(input("Input the number to search: "))

def is\_group\_member(*a*,*n2*):

    for value in *a*:

        if *n2* == value:

            return print("Values is contained in a group")

        else:

            return print("Values isn't contained in a group")

print("Value is: ",n2)

is\_group\_member(a,n2)

"""

Bai 2: Viet ham thuc hien cac chuong trinh sau

1. Tính:

a) (a + b),

b) a/b,

c) a^b.

2. Tính diện tích hình chữ nhật khi biết bán kính

3. Xuất tất cả các số nguyên tố trong 1 khoảng cho trước

4. Kiểm tra 1 số nguyên n có phải là số Fibonacci hay không

5. Tìm số Fibonacci thứ n (dùng đệ quy và không đệ quy)

6. Tính tổng n số Fibonacci đầu tiên (dùng đệ quy và không đệ quy)

7. Tính tổng căn bậc 2 của n số nguyên đầu tiên

8. Giải phương trình bậc 2: ax2 + bx + c=0

"""

## 1. Tinh

a = int(input("Nhap a: "))

b = int(input("Nhap b: "))

print("a + b = ",a + b)

print("a / b = ",a / b)

print("a ^ b = ",a \*\* b)

## 2. Tinh S hcn khi biet a va b

a = int(input("Nhap a: "))

b = int(input("Nhap b: "))

print("Dien tich hinh chu nhat co a = %d va b = %d la S =" %(a,b), a \* b)

## 3. Xuất tất cả các số nguyên tố trong 1 khoảng cho trước

import math

a = int(input("Nhap a: "))

b = int(input("Nhap b: "))

def isPrimeNumber(*n*):

    if *n* < 2:

        return False

    for i in range(2, int(math.sqrt(*n*)) + 1):

        if *n* % i == 0:

            return False

    return True

def printPrimeNumber(*a*, *b*):

    for i in range(*a*, *b* + 1):

        if isPrimeNumber(i):

            print(i, *end*="  ")

print("Số nguyên tố từ %d đến %d " %(a,b))

printPrimeNumber(a,b)

## 4. Kiểm tra 1 số nguyên n có phải là số Fibonacci hay không

import math

n = int(input("Nhập n: "))

def check(*x*):

    n = int(math.sqrt(*x*))

    return n \* n == *x*

def check\_fibonacci(*a*):

    return check(5 \* *a* \* *a* - 4) or check(5 \* *a* \* *a* + 4)

if check\_fibonacci(n) == True:

    print("%d is Fibonacci" %n)

else:

    print("%d Not Fibonacci" %n)

## 5. Tìm số Fibonacci thứ n (dùng đệ quy)

## 6. Tính tổng n số Fibonacci đầu tiên (dùng đệ quy)

import math

def findFibonacci(*n*):

    if *n* < 0:

        print("Không hợp lệ")

    elif *n* == 0:

        return 0

    elif *n* == 1 or *n* == 2:

        return 1

    else:

        return findFibonacci(*n* - 1) + findFibonacci(*n* - 2)

n = int(input("Nhập n: "))

print("Số Fibonacci thứ %d là:" %n, findFibonacci(n))

def sumFibonacci(*m*):

    sum = 0

    for i in range(*m* + 1):

        sum += findFibonacci(i)

    return sum

m = int(input("Nhập m số Fibonacci: "))

print("Tổng %d số Fibonacci đầu tiên là:" %m, sumFibonacci(m))

## 5. Tìm số Fibonacci thứ n (không đệ quy)

## 6. Tính tổng n số Fibonacci đầu tiên (không đệ quy)

def findFibonacci2(*n*):

    a = 0

    b = 1

    if *n* < 0:

        print("Không hợp lệ")

    elif *n* == 0:

        return 0

    elif *n* == 1:

        return b

    else:

        for i in range(1, *n*):

            c = a + b

            a = b

            b = c

        return b

n = int(input("Nhập n: "))

print("Số Fibonacci thứ %d là:" %n, findFibonacci2(n))

def sumFibonacci2(*m*):

    a = 0

    b = 1

    if n < 0:

        print("Không hợp lệ")

    elif n == 0 or n == 1:

        return n

    else:

        s = 0

        while b <= n:

            s += b

            a, b = b, a + b

        return s

m = int(input("Nhập m số Fibonacci:"))

print("Tổng %d số Fibonacci đầu tiên là:" %m, sumFibonacci2(m))

## 7. Tính tổng căn bậc 2 của n số nguyên đầu tiên

from cmath import sqrt

def sumFirstOfInteger(*n*):

    sum = 0

    for i in range(*n*):

        sum += sqrt(i)

    return sum

n = int(input("Nhập n số nguyên đầu tiên: "))

print("Tổng căn bậc 2 của %d số nguyên đầu tiên là:" %n, sumFirstOfInteger(n))

## 8. Giải phương trình bậc 2: ax2 + bx + c=0

import math

a = float(input("Nhập a: "))

b = float(input("Nhập b: "))

c = float(input("Nhập c: "))

def giaiPTB2(*a*, *b*, *c*):

    if *a* == 0 and *b* ==0:

        print("a hoặc b phải khác 0")

    delta = *b* \* *b* - 4 \* *a* \* *c*

    if delta > 0:

        x1 = (float)((-*b* + math.sqrt(delta)) / (2 \* *a*))

        x2 = (float)((-*b* - math.sqrt(delta)) / (2 \* *a*))

        print("Phương trình có 2 nghiệm là: x1 = ", x1, "x2 = ", x2)

    elif delta == 0:

        x1 = (-*b* / (2 \* *a*))

        print("Phương trình có nghiệm kép: x1 = x2 = ", x1)

    else:

        print("Phương trình vô nghiệm")

giaiPTB2(a, b, c)

## 9. Tính n!

def tinhGiaiThua(*n*):

    kq = 1

    if *n* == 0 or *n* == 1:

        return 1

    else:

        for i in range(2, *n* + 1):

            kq = kq \* i

        return kq

n = int(input("Nhập số nguyên dương n: "))

print("Giai thừa của", n, "là", tinhGiaiThua(n))

## 10.In \* dạng tam giác dưới như hình bên, đầu vào là số hàng(cột)

def draw(*r*):

    for row in range(1, *r* + 1):

        for col in range(1, row + 1):

            print("\*", *end*=' ')

        print()

r = int(input("Nhập: "))

draw(r)

## 11. Đổi giờ - phút – giây: thời gian đầu vào là giây được đổi thành giờ, phút, giây.

##Xuất kết quả ra màn hình dưới dạng: giờ:phút:giây. Ví dụ: soGiay = 3770 thì xuất

##ra màn hình 1:2:50.

time = int(input("Nhập số giây: "))

hour = (time//3600)%24

minute = (time%3600)//60

second = (time%3600)%60

print("Số giây nhập vào là: %d giây" %time)

print(hour,":",minute,":",second)

## 12.Cho một mảng số nguyên: (nên viết 2-3 cách)

array = [5,7,6,9,14,26,21,17,44,8,10,7,7]

## a) Xuât tất cả các số lẻ không chia hết cho 5

listSoLekhongChiaHet5 = (*list*(filter(lambda *a*:(*a*&1 == 1) and (*a* % 5)!= 0, array)))

print("Danh sách số lẻ không chia hết cho 5\n",listSoLekhongChiaHet5)

## b) Xuất tất cả các số Fibonacci

listFibonacci = []

def check(*x*):

    n = int(math.sqrt(*x*))

    return n \* n == *x*

def check\_fibonacci(*a*):

    return check(5 \* *a* \* *a* - 4) or check(5 \* *a* \* *a* + 4)

def printFibonacci():

    for i in array:

        if check\_fibonacci(i) == True:

            listFibonacci.append(i)

    print("Số Fibonacci có trong mảng\n",listFibonacci, *end*=' ')

printFibonacci()

## c) Tìm số nguyên tố lớn nhất

listPrime = []

def checkPrimeNumber(*n*):

    flag = 1

    if *n* < 2:

        flag = 0

        return flag

    for i in range(2, *n*):

        if *n* % i == 0:

            flag = 0

            break

    return flag

def checkPrimeList():

    for i in array:

        if checkPrimeNumber(i) == 1:

            listPrime.append(i)

    print("Danh sách số nguyên tố trong mảng\n", listPrime)

def findPrimeMax():

    print("Số nguyên tố lớn nhất trong mảng:", max(listPrime))

checkPrimeList()

findPrimeMax()

## d) Tìm số Fibonacci bé nhất

def findPrimeMin():

    print("Số Fibonacci bé nhất trong mảng:", min(listFibonacci))

findPrimeMin()

## e) Tính trung bình các số lẻ

listSoLe = (*list*(filter(lambda *a*:(*a*&1 == 1), array)))

print(listSoLe)

def TBSoLe(*listSoLe*):

    SumLe = 0

    for i in *listSoLe*:

        SumLe += i

    return SumLe / len(*listSoLe*)

print("Trung bình các số lẻ là: ",TBSoLe(listSoLe))

## f) Tính tích các phần tử là số lẻ không chia hết cho 3 trong mảng

listKhongChiaHet3 = (*list*(filter(lambda *a*:(*a*&1 == 1) and (*a* % 3)!= 0, array)))

print(listKhongChiaHet3)

def TichKhongChiaHet3(*listKhongChiaHet3*):

    tich = 1

    for i in *listKhongChiaHet3*:

        tich \*= i

    return tich

TichKhongChiaHet3(listKhongChiaHet3)

## g) Đổi chỗ 2 phần tử của danh sách, đầu vào là 2 vị trí cần đổi chỗ

def doiCho(*pos1*, *pos2*):

    array[*pos1*], array[*pos2*] = array[*pos2*], array[*pos1*]

    return array

pos1 = int(input('Nhập ví trí 1: '))

pos2 = int(input('Nhập vị trí 2: '))

print("Danh sách mảng lúc đầu\n",array)

print("\nDanh sách mảng sau khi đổi chỗ\n",doiCho(pos1 - 1, pos2 - 1))

## h) Đảo ngược trật tự các phần tử của danh sách

"""

Phương thức reverse() : đảo ngược list ban đầu trong python

Cắt (slice) list : Đảo ngược list và tạo ra một list mới

"""

print("Mảng lúc đầu: \n",array)

array.reverse()

print("\nMảng sau khi đảo ngược: \n",array)

## i) Xuất tất cả các số lớn thứ nhì của danh sách

#arr2 = []

def Max2List():

    arr2 = sorted(array)

    print("Số lớn thứ 2 trong danh sách là:",arr2[-2])

print(array)

Max2List()

## j) Tính tổng các chữ số của tất cả các số trong danh sách

def TinhTong(*array*):

    sum = 0

    for i in *array*:

        sum += i

    return sum

print(array)

print("Tổng tất cả các số trong danh sách là:",TinhTong(array))

## k) Đếm số lần xuất hiện của một số trong danh sách

print("Danh sách mảng:", array)

x = int(input("Nhập số cần đếm: "))

print("Số lần xuất hiện của %d là:" %x ,array.count(x))

## l) Xuất các số xuất hiện n lần trong danh sách

def SoLanXuatHien():

    n = int(input("Nhập số lần xuất hiện: "))

    for i in array:

        if n == array.count(i):

            print("Số xuất hiện %d lần là: %d" %(n,i))

            break

SoLanXuatHien()

## m) Xuất các số xuất hiện nhiều lần nhất trong danh sách

def SoXuatHienNhieuNhat():

    count = 0

    num = array[0]

    for i in array:

        arr3 = array.count(i)

        if arr3 > count:

            count = arr3

            num = i

    return num

print("Số xuất hiện nhiều nhất là:",SoXuatHienNhieuNhat())