Bài 1

#1

print('''

Twinkle, twinkle, little star,

    How I wonder what you are!

        Up above the world so high,

        Like a diamond in the sky.

Twinkle, twinkle, little star,

    How I wonder what you are!''')

#2

import platform

print(platform.python\_version())

#3

import datetime

current\_datetime = datetime.datetime.now()

print("Current date and time:", current\_datetime)

#4

from math import pi

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

print (f"The area of the circle with radius {str(r)} is: {str(pi \* r\*\*2)}")

#5

fname = input("Nhập tên : ")

lname = input("Nhập họ: ")

print (f"Hello {lname} {fname}")

#6

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

list = values.split(",")

tuple = tuple(list)

print('List : ',list)

print('Tuple : ',tuple)

#7

filename = input("Input the Filename: ")

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

print (f"The extension of the file is : {repr(f\_extns[-1])}")

#8

color\_list = ["Red","Green","White" ,"Black"]

print( f"{color\_list[0]} {color\_list[-1]}")

#9

exam\_st\_date = (11, 12, 2014)

formatted\_date = " / ".join(map(str, exam\_st\_date))

formatted\_text = f"The examination will start from : {formatted\_date}"

print(formatted\_text)

#10

number = int(input("Nhập số: "))

sum\_result = number + int(str(number) + str(number)) + int(str(number) + str(number) + str(number))

print("Tổng =", sum\_result)

Bài 2

1. Tính:

#1. Tính:

#a) (a + b)

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

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

sum = a+b

print("Tong hai so la", sum)

#b) a/b

div = a / b

print("Thuong hai so la: ", div)

#c) a^b

powfuc = pow(a, b)

print("a^b la: ", powfuc)

2. Tính diện tích hình chữ nhật

# cau 2: tinh dien tich hinh chu nhat

import math

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

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

def tinhDienTich():

    print(a\*b)

print(tinhDienTich())

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

# cau 3: xuat cac so nguyen to trong khoang cho truoc

import math

def kiem\_tra\_so\_nguyen\_to(n):

    if n == 1:

        return False

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

        if n % i == 0:

            return False

    return True

def liet\_ke\_so\_nguyen\_to(a, b):

    for i in range(a, b + 1):

        if kiem\_tra\_so\_nguyen\_to(i):

            print(i, end=' ')

try:

    a = int(input("Nhap so dau tien :"))

    b = int(input("Nhap so cuoi cung : "))

    if a < 0 or b < 0:

        print("Vui long nhap so tu nhien!")

    elif a > b:

        print("So thu nhat lon hon so thu hai!")

    else:

        liet\_ke\_so\_nguyen\_to(a, b)

except:

    print("Dinh dang dau vao khong hop le!")

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

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

N = int(input("Nhập vào số bạn muốn kiểm tra: "))

f3 = 0

f1 = 1

f2 = 1

if (N == 0 or N == 1):

    print("Đây là số trong dãy fibonacci")

else:

    while f3 < N:

        f3 = f1 + f2

        f2 = f1

        f1 = f3

    if f3 == N:

        print("Đây là số trong dãy fibonacci")

    else:

        print("Không phải số trong dãy fibonacci")

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

#không sử dụng đệ quy

def fibonacci\_iterative(n):

    if n <= 0:

        return 0

    elif n == 1:

        return 1

    fib\_prev = 0

    fib\_current = 1

    for \_ in range(2, n + 1):

        fib\_next = fib\_prev + fib\_current

        fib\_prev, fib\_current = fib\_current, fib\_next

    return fib\_current

n = int(input("Nhập n: "))

r = fibonacci\_iterative(n)

print(f"Số Fibonacci thứ {n} là {r}")

#đệ quy

def fibonacci\_recursive(n):

    if n <= 0:

        return 0

    elif n == 1:

        return 1

    else:

        return fibonacci\_recursive(n - 1) + fibonacci\_recursive(n - 2)

n = int(input("Nhập n: "))

r = fibonacci\_recursive(n)

print(f"Số Fibonacci thứ {n} là {r}")

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

# Đệ quy

def fibo(n):

    if n < 2:

        return 1

    else:

        res = fibo(n-1) + fibo(n-2)

        return res

n = int(input("Nhập n: "))

sum = 0

for i in range(0, n):

    r = fibo(i)

    sum += r

    print(r)

print("Tổng dãy Fibonacci", sum)

# không đệ quy

def fibo(n):

    f0 = 0

    f1 = 1

    fn = 1

    if (n < 0):

        return -1

    elif (n == 0 or n == 1):

        return n

    else:

        for i in range(2, n):

            f0 = f1

            f1 = fn

            fn = f0 + f1

        return fn

n = int(input("Nhập n: "))

sum = 0

for i in range(0, n):

    r = fibo(i)

    sum += r

    print(r)

print("Tổng dãy Fibonacci", sum)

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

import math

n = int(input("Nhập số n: "))

Tong = 0

if n < 0:

    print("Số không hợp lệ! Vui lòng nhập lại")

else:

    for i in range ( 1, n + 1):

        Tong += i

        kq = math.sqrt(Tong)

print(f"Căn bậc 2 của {n} số nguyên là: {kq}")

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

# 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: "))

delta = pow(b, 2) - 4\*a\*c

if delta < 0:

    print("Phương trình vô nghiệm!")

elif delta == 0:

    nghiemKep = -(b / (2 \* a))

    print("Phương trình có 2 nghiệm kép x1 = x2 = ", nghiemKep)

elif delta > 0:

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

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

    print(f"Phương trình có 2 nghiệm phân biệt: x1 = {x1}, x2 = {x2}")

9. Tính n!

n = int(input("Nhập số cần tính giai thừa: "))

def giaiThua(n):

    if n == 0:

        return 1

    return n \* giaiThua(n - 1)

print(giaiThua(n))

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

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.

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

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

b) Xuất tất cả các số Fibonacci

c) Tìm số nguyên tố lớn nhất

d) Tìm số Fibonacci bé nhất

e) Tính trung bình các số lẻ

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

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

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

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

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

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

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

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