

Bài 1:

Viết chương trình cho phép nhập vào từ bàn phím một số nguyên n và hiển thị ra màn hình các các hình có hình dạng như dưới đây với kích thước tùy theo giá trị nhập vào.

Input: Số nguyên dương n

Output: Ví dụ sau

For example:

Input	Output
5	* *** ***** ***** ***** ***** ***** ***** ***** ***** * ** **** ***** ***** ***** ***** ***** *** *

```
# Vòng lặp for chạy từ 1 -> n
n = int(input())
# In tam giác đều
# vd : n = 3 i chạy từ :0 -> 2
# dòng i=0 : in (n-0) dấu cách và (2*0+1) dấu sao
# Tương tự đến dòng thứ i = n-1(dòng thứ 2): in (n-2) dấu cách và (2*2+1)dấu sao
for i in range(n):
    print(' '*(n-i) + '*'*(2*i+1))
# In dấu cách dòng
print()

# In hình chữ nhật
# i chạy từ 0 -> n-1
# dòng i = 0: in 5 dấu sao
# Tương tự đến dòng i = n-1 và in ra HCN
for i in range(n) :
    print('*'*5)
print()

# In hình tam giác đều ngang
# i chạy từ 0 -> n-1
# In ra tam giác vuông xuôi :
# dòng i = 0: 2*0 + 1 : in ra 1 dấu sao
# Tương tự đến dòng i = n-1 : in ra 2*2+1 dấu sao
for i in range(n) :
    print('*'*(2*i + 1))
# In tam giác vuông ngược bằng cách đổi range i từ n-2 -> 0
for i in range(n-2,-1,-1):
    print('*'*(2*i+1))
```

Bài 2:

Số hoàn chỉnh là số bằng tổng tất cả các ước nhỏ hơn nó, ví dụ $6 = 1 + 2 + 3$ là một số hoàn chỉnh. Viết chương trình cho phép thực hiện các công việc sau:

- Nhập một số nguyên dương n từ bàn phím
- Hiển thị các từ "Co" hoặc "Khong" tương ứng với n có phải là số hoàn chỉnh hay không?

Input:

- Dòng 1: Số nguyên dương n

Output:

- Dòng 2: Các giá trị "Co" hoặc "Khong" tương ứng.

Ví dụ:

Input	Output
6	Co
3	Khong
11	Khong

```
n=int(input()) # Nhập số cần kiểm tra
so=0 # khai báo 1 biến là tổng các ước
for i in range (1,n): # dùng 1 vòng lặp để kiểm tra từ 1 đến n-1 số nào là
    ước của n
    if n%i==0: # kiểm tra xem n có chia hết cho i không
        so+=i # nếu chia hết thì cộng vào biến tổng các ước
# sau khi kết thúc vòng Lặp kiểm tra tổng các ước có bằng n không và in kết
quả
if so==n:
    print("Co")
else:
    print("Khong")
```

Bài 3:

Câu hỏi 1

Not complete

Đạt điểm 1,00

Viết chương trình cho phép thực hiện các công việc sau:

- Nhập một số nguyên n từ bàn phím
- Hiển thị các từ "Co" hoặc "Khong" tương ứng với n có phải là số nguyên tố hay không?

Input:

- Dòng 1: Số nguyên n

Output:

- Dòng 2: Các giá trị "Co" hoặc "Khong" tương ứng.

Ví dụ:

Input	Output
1	Khong
3	Co
14	Khong

```
# Nhập vào số nguyên n
n = int(input())

check = True
# Nếu số đó bé hơn 2, số đó không là số nguyên tố
if n < 2:
    check = False

# Kiểm tra từ 2 đến n-1 xem n có ước nào không (có chia hết số nào ko)
else:
    for i in range (2,n,1):
        # Nếu tìm thấy một ước i nào đó, n không phải là snt dừng vòng lặp
        if n % i == 0:
            check = False
            break

# In ra kết quả tương ứng
if check == True:
    print("Co")
else:
    print("Khong")
```

Bài 4:

Viết chương trình cho phép thực hiện các công việc sau:

- Nhập hai số nguyên dương x, y từ bàn phím ($x < y$)
- Tính tổng bình phương các số từ x đến y

Input:

- Dòng 1: Số nguyên dương x, y

Output:

- Dòng 2: Tổng bình phương các số từ x đến y

Ví dụ:

Input	Output
3 15	1235
4 6	77

```
x=int(input()) # Nhập số x
y=int(input()) # nhập số y

tong=0
for i in range(x, y+1, 1):
    tong += i*i
print(tong)
```

Bài 5:

Viết chương trình cho phép thực hiện các công việc sau:

- Nhập số thực x và số nguyên y từ bàn phím
- Tính và hiển thị giá trị biểu thức x^y (kết quả lấy 3 chữ số sau dấu chấm thập phân)

Input:

- Dòng 1: Số thực x
- Dòng 2: Số nguyên y

Output:

- Dòng 2: Tính và hiển thị giá trị biểu thức x^y

For example:

Input	Output
2	8.000
3	
5.5	30.250
2	

```
# Nhập số thực x và số nguyên y
x = float(input())
y = int(input())

# Dùng biến p để lưu trữ giá trị x mũ y
p = x**y
# Sử dụng f-string để làm tròn kết quả
print(f"{p:.3f}")
```

Bài 6:

Viết chương trình nhập một số nguyên n từ bàn phím, tìm bội số của số đó với các số từ 1 đến 10, sau đó in kết quả ra màn hình.

Input: Số nguyên n

Output: Bội số của n với các số từ 1 đến 10

For example:

Input	Output
3	3
	6
	9
	12
	15
	18
	21
	24
	27
	30
-5	-5
	-10
	-15
	-20
	-25
	-30
	-35
	-40
	-45
	-50

```
# Nhập số x
x=int(input())

# vòng lặp từ 1 - 10 để tính bội
for i in range (1,11):
    print(x*i) # mỗi lần lặp in ra tích của x với i
```

Bài 7:

Viết chương trình cho phép, nhập một số nguyên dương n từ bàn phím và tính các biểu thức sau (kết quả lấy 3 chữ số sau dấu chấm thập phân):

$$sa = 1^2 + 2^2 + 3^2 + \dots + (n-1)^2 + n^2$$

$$sb = 1 + \frac{1}{1.2} + \frac{1}{2.3} + \frac{1}{3.4} + \dots + \frac{1}{(n-1)n} + \frac{1}{n(n+1)}$$

Input: Số nguyên dương n

Output:

- Dòng 1: Giá trị biểu thức **sa**
- Dòng 2: Giá trị biểu thức **sb**

For example:

Input	Output
5	55.000 1.833
3	14.000 1.750

```
# Nhập số nguyên n
n = int(input())
# Khai báo sa và sb
sa = 0
sb = 1

# sử dụng vòng lặp for để tính tổng từ 1 đến n và cộng dồn vào sa và sb
for i in range (1,n+1):
    sa+= i**2

for i in range (1,n+1):
    sb += 1 / (i*(i+1))

# Sử dụng f-string để làm tròn kết quả
print(f"{sa:.3f}")
print(f"{sb:.3f}")
```

Bài 8:

Viết chương trình cho phép, nhập một số nguyên dương n từ bàn phím và tính các biểu thức sau (kết quả lấy 3 chữ số sau dấu chấm thập phân):

$$sc = \sqrt{3 + \sqrt{3 + \sqrt{3 + \cdots \sqrt{3}}}} \text{ (có n dấu căn)}$$

$$sd = 1 + 1.2 + 1.2.3 + 1.2.3.4 + \cdots + 1.2.3.4 \dots n$$

Input: Số nguyên dương n

Output:

- Dòng 1: Giá trị biểu thức **sc**
- Dòng 2: Giá trị biểu thức **sd**

For example:

Input	Output
6	2.302 873.000
2	2.175 3.000

```
from math import sqrt # khai báo sử dụng hàm từ thư viện toán học
n=int(input()) # Nhập số n

sc=0 #khai báo biến tổng.
for i in range (1,n): # vòng lặp từ 1 - 10
    sc=sqrt(sc+3)
print(f"{sc:.3f}") #một cú pháp Lấy 3 chữ số sau dấu phẩy

sd=0 # khai báo biến tổng sd
tich=1
'''khai báo biến tích dùng để Lưu giá trị các tích sau mỗi vòng lặp
ví dụ: Lần Lặp 1 tích bằng 1
Lần Lặp 2 tích bằng 1*2 bằng 2
tiếp tục cứ thế'''
for i in range(1,n+1):
    tich*=i # như giải thích bên trên
    sd+=tich # cộng dồn tổng các tích
print(f"{sd:.3f}")
```


Bài 9:

Viết chương trình thực hiện các công việc sau

- Nhập vào số nguyên dương n
- Hiển thị số n viết theo thứ tự ngược lại.

Input:

- Dòng 1: số nguyên dương n

Output:

- Dòng 2: số đảo ngược của n

Ví dụ:

Input	Output
132	231
111	111
1200	21

```
n = int(input()) #hàm Lấy ra độ dài n

m = 0
# Lấy dần các chữ số từ phải sang trái
# Ví dụ : 123 = 3 * 10^2 + 2*10 +1
while n > 0:
    # Lấy phần dư của n khi chia cho 10 và cộng vào biến m
    dv = n % 10 #Lấy chữ số hàng đơn vị
    n = n // 10 #Cắt chữ số hàng đơn vị
    m = m*10 + dv

print(m)
```

Bài 10:

Một số nguyên dương N có k chữ số, được gọi là số **amstrong** nếu số đó bằng tổng các lũy thừa bậc k các chữ số. Viết chương trình cho phép thực hiện các công việc sau:

- Nhập một số nguyên dương n từ bàn phím
- Hiển thị các từ "Co" hoặc "Khong" tương ứng với n có phải là số amstrong hay không?

Input:

- Dòng 1: Số nguyên dương n

Output:

- Dòng 2: Các giá trị "Co" hoặc "Khong" tương ứng.

Ví dụ:

Input	Output
153	Co
31	Khong
111	Khong

```
''' GIẢI THÍCH ĐỀ BÀI
VÍ DỤ SỐ 153 LÀ SỐ AMSTRONG VÌ 1^3 + 5^3 +3^3 =153
MŨ 3 VÌ SỐ 153 CÓ 3 CHỮ SỐ'''
n=int(input()) # nhập n
a=n # sử dụng biến a để đếm số chữ số
b=n # sử dụng biến b để tính tổng
cs=0 # biến cs để Lưu số chữ số
tong=0
'''vòng while dưới đây để đếm các chữ số
khi a còn khác 0 thì ta sẽ chia lấy nguyên a cho 10
đến khi a = 0 sẽ dừng vòng lặp và biến cs đã đếm đủ số chữ số của a
Ngoài ra các bạn có thể sử dụng hàm len() để đếm số ký tự thì sẽ nhanh hơn
khi sử dụng vòng while. Nhưng bản chất muốn đếm số chữ số thì nó sẽ như
vậy.'''
while a!=0:
    cs+=1
    a=a//10
''' vòng while thứ 2 dùng để tính tổng các chữ số Lũy thừa bậc K các chữ số'''
while b!=0:
    dv=b%10
    tong+=dv**cs
    b=b//10
'''Ở trên chúng ta phải dùng biến a và b gán bằng giá trị của n để sử dụng
với mục đích không làm mất đi giá trị của n ban đầu
và giờ chúng ta sẽ so sánh tổng với số n ban đầu.
nếu bằng nhau thì sẽ là số amstrong'''
if tong==n:
    print("Co")
else:
    print("Khong")
```

Bài 11:

Viết chương trình tìm số sinh viên tối thiểu của một lớp nếu biết số học sinh này xếp thành a hàng, b hàng, c hàng hoặc d hàng cũng đều vừa đủ với a, b, c, d là các số nguyên được nhập từ bàn phím. (Gợi ý: Tìm bội số chung nhỏ nhất.)

• Ví dụ: a = 2; b = 3; c = 4; d = 8 thì số sinh viên tối thiểu là 24.

Input:

- Dòng 1: số nguyên dương a
- Dòng 2: số nguyên dương b
- Dòng 3: số nguyên dương c
- Dòng 4: số nguyên dương d

Output: Số học sinh tối thiểu

For example:

Input	Output
2 3 4 8	24
2 3 4 5	60

```
a=int(input())
b=int(input())
c=int(input())
d=int(input())
bcnn=max(a,b,c,d) # Lấy bcnn là số Lớn nhất trong 4 số
"""Bắt đầu từ số Lớn nhất, tăng dần bcnn cho tới khi chia hết cho cả 4 số a,
b, c, d
Số cuối cùng có thể là a*b*c*d.
"""
while bcnn%a!=0 or bcnn%b!=0 or bcnn%c!=0 or bcnn%d!=0:
    bcnn+=1
print(bcnn)
```