



NHẬP MÔN TƯ DUY TÍNH TOÁN

Bài 3: Vòng lặp và hàm trong python


Nội dung chính



Phần 1
Vòng lặp “while”

TRƯỜNG XUÂN NAM

3



Phần 2
Vòng lặp “for”

TRƯỜNG XUÂN NAM

8



Phần 3
Hàm

TRƯỜNG XUÂN NAM

12



Phần 4
Bài tập

TRƯỜNG XUÂN NAM

15



Phần 1

Vòng lặp “while”

Vòng lặp while



```
while expression:
```

```
# while-block
```

```
while expression:
```

```
#while-block-1
```

```
continue
```

```
#while-block-2
```

```
while expression:
```

```
# while-block
```

```
else:
```

```
# else-block
```

- Vòng **while** thực hiện lặp lại khối lệnh chừng nào biểu thức điều kiện còn đúng
- Phát biểu **continue** trong khối lệnh sẽ ngắt khối lệnh hiện tại và bắt đầu một vòng lặp mới
- Phát biểu **break** sẽ kết thúc vòng lặp ngay lập tức
- Khối **else** sẽ được thực hiện sau khi toàn bộ vòng lặp đã chạy xong, không bắt buộc phải có khối này
 - Khối này sẽ không chạy nếu vòng lặp bị “break”

Vòng lặp while

```
# In ra các số tự nhiên chia hết cho 7 nhỏ hơn 1000
```

```
n = 0
```

```
while n < 1000:
```

```
    if (n % 7) == 0:
```

```
        print(n)
```

```
    n += 1
```

```
# Tính tổng các số nhỏ hơn 1000 và không chia hết cho 3
```

```
t = 0
```

```
n = 0
```

```
while n < 1000:
```

```
    if (n % 3) != 0:
```

```
        t = t + n
```

```
    n += 1
```

```
print(t)
```

Vòng lặp while



```
# Với 10 triệu, gửi ngân hàng với lãi suất 5,1% hàng năm  
# tính xem sau bao nhiêu năm thì có ít nhất 50 triệu
```

```
so_tien = 10000000
```

```
lai_suat = 5.1/100
```

```
so_nam = 0
```

```
while so_tien < 50000000:
```

```
    so_nam += 1
```

```
    so_tien = so_tien * (1 + lai_suat)
```

```
    print("Số tiền sau", so_nam, "năm:", so_tien)
```

```
print("Sau", so_nam, "bạn sẽ có ít nhất 50 triệu.")
```

Vòng lặp while

```
# Ví dụ về lặp while có dùng break và else
# Nhập số n và kiểm tra xem nó có phải số nguyên tố hay không
n = int(input("Nhập số N: "))
x = 2
while x < n:
    if (n % x) == 0:
        print("N không phải số nguyên tố")
        break;
    x = x + 1
else:
    print("N là số nguyên tố")
```



Phần 2

Vòng lặp “for”

Vòng lặp for



- Cú pháp:

```
for <biến> in <danh-sách>:
```

```
    # khối for
```

```
else
```

```
    # khối else
```

- Vòng **for** cho phép sử dụng một <biến> lần lượt duyệt các giá trị trong <danh-sách>
- Tương tự như **while**, có thể sử dụng **break** và **continue**
- Khối **else** thực hiện sau khi toàn bộ vòng lặp đã chạy xong
 - Khối này sẽ không chạy nếu vòng lặp bị “break”
 - Không bắt buộc phải có khối này
 - Cách làm việc tương tự như ở vòng lặp **while**

Vòng lặp for



```
X = ['chó', 'mèo', 'lợn', 'gà']
```

```
# In ra các loài vật trong danh sách
```

```
for w in X:  
    print(w)
```

```
# In ra các loại vật, ngoại trừ loài 'mèo'
```

```
for x in X:  
    if x == 'mèo':  
        continue  
    print(x)
```

```
# In ra các loại vật, nếu gặp loài 'mèo' thì dừng luôn
```

```
for z in X:  
    if z == 'mèo':  
        break  
    print(z)
```

Vòng lặp for



Trường hợp một khoảng số khá lớn, không thể liệt kê được

Ta sử dụng hàm range để tạo ra khoảng số

In các số từ 10 đến 19: khoảng 10 đến 20, bước nhảy 1

```
for d in range(10, 20):
```

```
    print(d)
```

In các số từ 20 đến 11: khoảng 20 đến 10, bước nhảy -1

```
for d in range(20, 10, -1):
```

```
    print(d)
```

In các số lẻ từ 1 đến 100: khoảng 1 đến 100, bước nhảy 2

```
for d in range(1, 101, 2):
```

```
    print(d)
```



Phần 3

Hàm

- Cú pháp khai báo hàm rất đơn giản

```
def <tên-hàm>(danh-sách-tham-số):  
    <lệnh 1>  
    ...  
    <lệnh n>
```

- Ví dụ: hàm tính tích 2 số

```
def tich(a, b):  
    return a*b
```

- Hàm trả về kết quả bằng lệnh **return**, nếu không trả về thì coi như trả về **None**

Hàm với tham số mặc định

- Hàm có thể chỉ ra giá trị mặc định của tham số

nếu không nói gì thì mặc định b=1

```
def tich(a, b = 1):  
    return a*b
```

- Như vậy với hàm trên ta có thể gọi thực hiện nó:

```
print(tich(10, 20))           # 200
```

```
print(tich(10))               # 10
```

```
print(tich(a=5))              # 5
```

```
print(tich(b=6, a=5))         # 30
```

- Chú ý: các tham số có giá trị mặc định phải đứng cuối danh sách tham số



Phần 4

Bài tập

1. Viết chương trình cho phép người dùng nhập vào một dãy số tự nhiên (không biết trước độ dài), việc nhập dãy sẽ kết thúc khi người dùng nhập một số âm nào đó.
2. Viết hàm `isPrime` kiểm tra xem N có phải là số nguyên tố hay không?
3. Viết chương trình nhập hai số A và B , in ra tất cả các số nguyên tố nằm trong khoảng $[A, B]$.
4. Nhập 2 số A và B , tính và in ra màn hình ước số chung lớn nhất và bội số chung nhỏ nhất của hai số đó.
5. Nhập tọa độ 3 điểm A , B và C trên mặt phẳng 2 chiều. Hãy kiểm tra và chỉ ra hình dạng của tam giác ABC (đều, vuông, cân, vuông cân, tù, nhọn,...)