



## Bài 5: Lệnh rẽ nhánh và vòng lặp

---

TS. Trịnh Tuấn Đạt  
Viện CNTT-TT, ĐHBK Hà Nội



# Nội dung

---

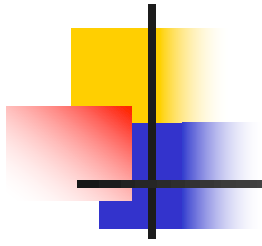
1. Lệnh rẽ nhánh
2. Vòng lặp
3. Lệnh break, continue, pass
4. Lệnh else với vòng lặp



# Nội dung

---

1. **Lệnh rẽ nhánh**
2. Vòng lặp
3. Lệnh break, continue, pass
4. Lệnh else với vòng lặp



# 1. Lệnh rẽ nhánh

---

1.1. Lệnh if

1.2. Lệnh if... else

1.3. Lệnh if...elif...else

1.4. Lệnh if lồng nhau

# 1.1. Lệnh if

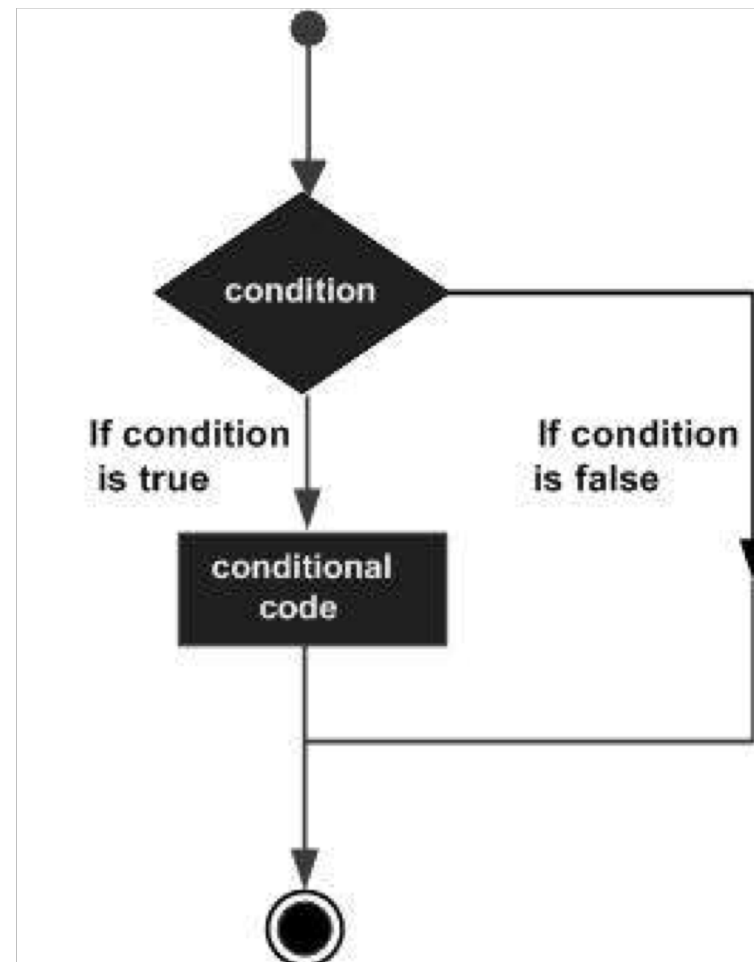
## ■ Cú pháp (Lưu ý dấu : và sự thụt dòng)

```
if bieu_thuc:  
    cac_lenh
```

## ■ Ví dụ

```
tuoi = 10;  
if tuoi <= 18:  
    print("Chao em")  
    print("Tam biet")
```

Chao em  
Tam biet





## 1.1. Lệnh if

---

- Trong Python, các giá trị khác null và khác 0 là True. Các giá trị là null hoặc 0 là false

```
var1 = 100
if var1:
    print("Var 1")
    print(var1)
```

```
var2 = 0
if var2:
    print("Var 1")
    print(var2)
print("Good bye!")
```

Var 1  
100  
Good bye!



## 1.1. Lệnh if

---

### ■ Lệnh if trên 1 dòng

```
number = 20;  
if number%2==0 and number%5==0 :print("Chia het cho 2");print("Chia het cho 5");  
print("Tam biet")
```

```
Chia het cho 2  
Chia het cho 5  
Tam biet
```

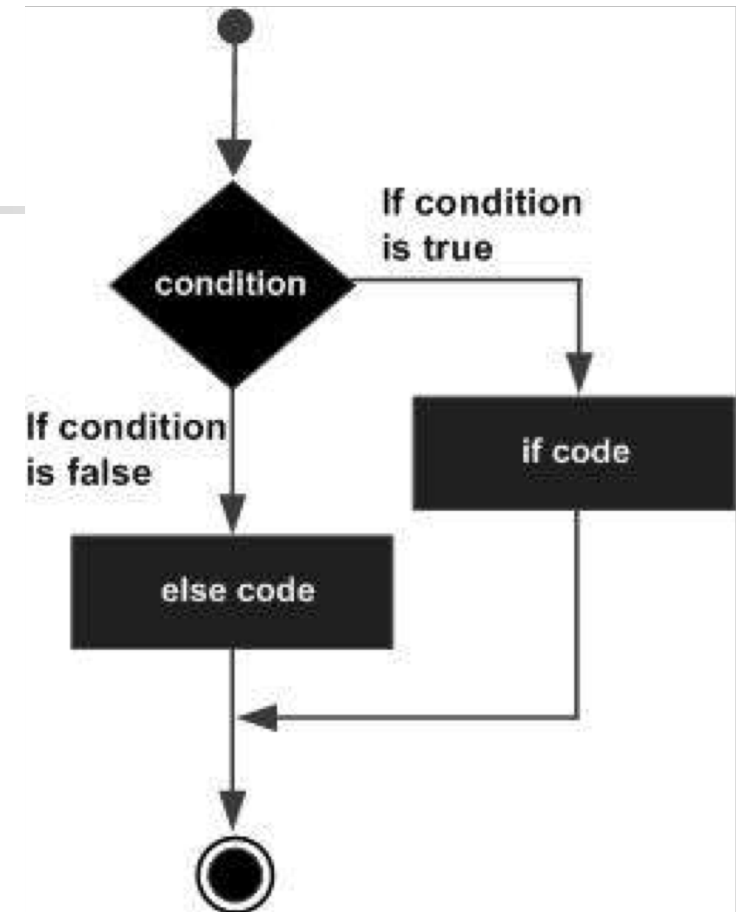
```
number = 19;  
if number%2==0 and number%5==0 :print("Chia het cho 2");print("Chia het cho 5");  
print("Tam biet")
```

```
Tam biet
```

## 1.2. Lệnh if...else

### ■ Cú pháp

```
if bieu_thuc:  
    cac_lenh  
else:  
    cac_lenh
```



- Một lệnh **else** có thể được kết hợp với lệnh **if**. Lệnh else chứa khối code được thực thi nếu biểu thức điều kiện trong lệnh if là False.
- Lệnh else là tùy chọn, và nếu có, chỉ có **duy nhất** một lệnh else sau lệnh if.





## 1.2. Lệnh if...else

---

### ■ Ví dụ

```
var1 = 100
if var1:
    print("1-True")
    print(var1)
else:
    print("1-False")
    print(var1)
```

```
1-True
100
2-False
0
Good bye!
```

```
var2 = 0
if var2:
    print("2-True")
    print(var2)
else:
    print("2-False")
    print(var2)

print("Good bye!")
```



## 1.3. Lệnh if...elif...else

---

- Cú pháp

```
if bieu_thuc1:  
    cac_lenh  
elif bieu_thuc2:  
    cac_lenh  
elif bieu_thuc3:  
    cac_lenh  
else:  
    cac_lenh
```

- Lệnh elif cho phép kiểm tra **lần lượt** nhiều điều kiện và thực thi khối code khi gặp biểu thức True đầu tiên (bỏ qua toàn bộ các khối khác)
- Lệnh elif là tùy chọn. Được dùng nhiều lệnh elif
- Theo sau elif cuối, có thể dùng else hoặc không



## 1.3. Lệnh if...elif...else

---

### ■ Ví dụ

```
diem = 7

if diem<5:
    print("Truot")
elif diem<6.5:
    print("Trung binh")
elif diem<8:
    print("Kha")
elif diem<9:
    print("Gioi")
else:
    print("Xuat sac")
```



## 1.4. Lồng các lệnh if

---

- Đôi khi cần kiểm tra thêm một điều kiện khác sau khi một điều kiện đã thỏa mãn (True). Khi đó, dùng các lệnh if lồng nhau
- Trong cấu trúc các lệnh if lồng nhau, có thể có cấu trúc if...elif...else bên trong cấu trúc if...elif...else khác.
- Cú pháp

```
if bieu_thuc1:
    cac_lenh
    if bieu_thuc2:
        cac_lenh
    elif bieu_thuc3:
        cac_lenh
    else:
        cac_lenh
elif bieu_thuc4:
    cac_lenh
else:
    cac_lenh
```



## 1.4. Lồng các lệnh if

---

### ■ Ví dụ

```
num = int(input("Nhập vào một số "))
if num%2 == 0:
    if num%3 == 0:
        print ("Chia hết cho 3 và 2")
    else:
        print ("Chia hết cho 2, không chia hết cho 3")
else:
    if num%3 == 0:
        print ("Chia hết cho 3, không chia hết cho 2")
    else:
        print ("Không chia hết cho 2, không chia hết cho 3")
```

Nhập vào một số 7  
Không chia hết cho 2, không  
chia hết cho 3



# Nội dung

---

1. Lệnh rẽ nhánh
2. **Vòng lặp**
3. Lệnh break, continue, pass
4. Lệnh else với vòng lặp



## 2. Vòng lặp

---

- Các lệnh được thực thi lần lượt: lệnh 1, lệnh 2, lệnh 3, ...
- Khi muốn thực thi một khối lệnh nhiều lần: dùng vòng lặp
- Các loại vòng lặp trong Python
  - Vòng lặp while
  - Vòng lặp for
  - Lồng vòng lặp lồng nhau



## 2.1. Vòng lặp while

---

- Vòng lặp while: lặp đi lặp lại các lệnh trong phần thân của vòng lặp miễn là điều kiện đã cho là True. Khi điều kiện là False, thoát ra khỏi vòng lặp.
- Cú pháp (chú ý dấu : và sự thụt dòng)

```
while bieu_thuc:  
    cac_lenh
```

- Vòng lặp này có thể không chạy lần nào, khi bieu\_thuc có giá trị False

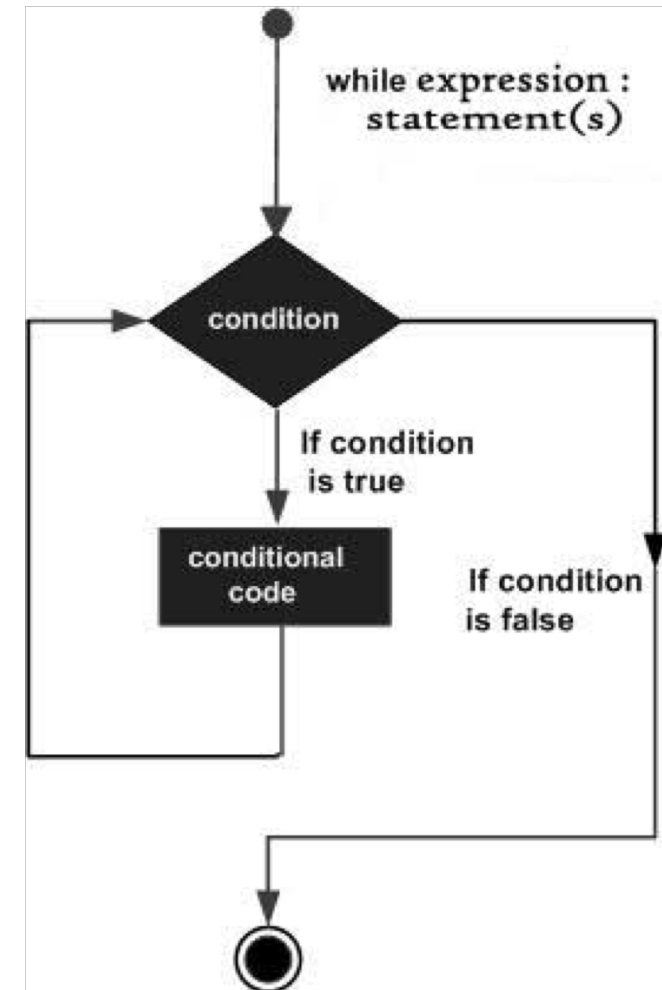


## 2.1. Vòng lặp while

```
count = 0
while (count < 9):
    print('So thu tu cua ban la:', count)
    count = count + 1

print("Good bye!")
```

```
So thu tu cua ban la: 0
So thu tu cua ban la: 1
So thu tu cua ban la: 2
So thu tu cua ban la: 3
So thu tu cua ban la: 4
So thu tu cua ban la: 5
So thu tu cua ban la: 6
So thu tu cua ban la: 7
So thu tu cua ban la: 8
Good bye!
```





# Vòng lặp vô hạn

---

- Vòng lặp vô hạn: vòng lặp mà điều kiện của nó luôn là True. Do vậy, vòng lặp không bao giờ kết thúc.
- Một vòng lặp vô hạn có thể dùng trong lập trình client/server, khi server cần chạy liên tục để các chương trình client có thể giao tiếp với nó khi cần



# Vòng lặp vô hạn

---

- Ví dụ sau là một vòng lặp vô hạn, để thoát khỏi vòng lặp này, nhấn phím CTRL+C.

```
while 1 : # Lenh nay tao mot vong lap vo han
    num = input("Hay nhap mot so: ")
    print("So da nhap la: ", num)

print("Good bye!")
```

```
Hay nhap mot so: 20
So da nhap la: 20
Hay nhap mot so: 29
So da nhap la: 29
Hay nhap mot so:
```



# Lệnh while trên một dòng

---

- Tương tự lệnh if, lệnh while có thể viết trên 1 dòng

```
count = 1  
while count < 11: print(count); count = count + 1
```

```
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```



## 2.2. Vòng lặp for

---

- Vòng lặp for dùng để duyệt các phần tử trong một dãy (List, Tuple, String)
- Cú pháp

```
for bien_vong_lap in day_sequence:  
    cac_lenh
```

- Các item trong dãy **day\_sequence** lần lượt được gán cho biến **bien\_vong\_lap**. Mỗi lần gán xong, khối lệnh **cac\_lenh** được thực thi



## 2.2. Vòng lặp for

```
for letter in 'Python':      # Ví dụ đầu tiên
    print('Chu cái hiện tại :', letter)

qua = ['chuoi', 'tao', 'xoai']
for qua in qua:              # Ví dụ thứ hai
    print('Bạn có thích an :', qua)

print("Good bye!")
```

```
Chu cái hiện tại : P
Chu cái hiện tại : y
Chu cái hiện tại : t
Chu cái hiện tại : h
Chu cái hiện tại : o
Chu cái hiện tại : n
Bạn có thích an : chuoi
Bạn có thích an : táo
Bạn có thích an : xoài
Good bye!
```



# Vòng lặp for và hàm range()

---

```
for var in range(5):  
    print(var)
```

0  
1  
2  
3  
4

```
for var in range(5,10):  
    print(var)
```

5  
6  
7  
8  
9



# Lặp qua index của dãy

---

- Một cách khác để lặp qua các item là qua chỉ số (index) của dãy. Ví dụ

```
qua = ['chuoi', 'tao', 'xoai']
for index in range(len(qua)):
    print('Ban co thich an :', qua[index])

print("Good bye!")
```

```
Ban co thich an : chuoi
Ban co thich an : tao
Ban co thich an : xoai
Good bye!
```





## 2.3. Lồng vòng lặp

---

- Có thể sử dụng một vòng lặp bên trong một vòng lặp khác
- Cú pháp (vòng lặp for cũng có thể nằm trong vòng lặp while và ngược lại)

```
for bien_vong_lap in day_seq:  
    for bien_vong_lap in day_seq:  
        cac_lenh1  
        cac_lenh2
```

```
while bieu_thuc:  
    while bieu_thuc:  
        cac_lenh1  
        cac_lenh2
```



## 2.3. Lồng vòng lặp

### ■ Ví dụ

```
for i in range(1,11):  
    for j in range(1,11):  
        k = i*j  
        print (k, end=' ')  
    print()
```

|    |    |    |    |    |    |    |    |    |     |
|----|----|----|----|----|----|----|----|----|-----|
| 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10  |
| 2  | 4  | 6  | 8  | 10 | 12 | 14 | 16 | 18 | 20  |
| 3  | 6  | 9  | 12 | 15 | 18 | 21 | 24 | 27 | 30  |
| 4  | 8  | 12 | 16 | 20 | 24 | 28 | 32 | 36 | 40  |
| 5  | 10 | 15 | 20 | 25 | 30 | 35 | 40 | 45 | 50  |
| 6  | 12 | 18 | 24 | 30 | 36 | 42 | 48 | 54 | 60  |
| 7  | 14 | 21 | 28 | 35 | 42 | 49 | 56 | 63 | 70  |
| 8  | 16 | 24 | 32 | 40 | 48 | 56 | 64 | 72 | 80  |
| 9  | 18 | 27 | 36 | 45 | 54 | 63 | 72 | 81 | 90  |
| 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100 |

|                    |
|--------------------|
| 1, 2, 3, 4, 5,     |
| 2, 4, 6, 8, 10,    |
| 3, 6, 9, 12, 15,   |
| 4, 8, 12, 16, 20,  |
| 5, 10, 15, 20, 25, |



# Nội dung

---

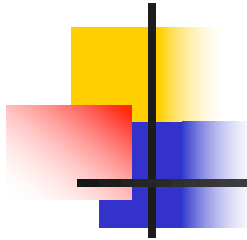
1. Lệnh rẽ nhánh
2. Vòng lặp
3. **Lệnh break, continue, pass**
4. Lệnh else với vòng lặp



## 3. Lệnh break, continue, pass

---

- Lệnh break: Thoát vòng lặp
- Lệnh continue: Bỏ qua các lệnh trong bước lặp hiện tại, chuyển sang bước lặp tới
- Lệnh pass: Lệnh rỗng



## 3.1. Lệnh break

---

- Lệnh break: kết thúc vòng lặp hiện tại và chuyển điều khiển tới cuối vòng lặp
- Lệnh break được dùng trong cả vòng lặp while và vòng lặp for
- Giả sử đang sử dụng lồng vòng lặp, lệnh break kết thúc sự thực thi của vòng lặp bên trong và bắt đầu thực thi dòng code tiếp theo của khối bên ngoài



## 3.1. Lệnh break

---

```
for letter in 'Python':  
    if letter == 'h':  
        break  
    print('Chu cai hien tai :', letter)
```

```
Chu cai hien tai : P  
Chu cai hien tai : y  
Chu cai hien tai : t
```



## 3.1. Lệnh break

---

```
var = 10
while var > 0:
    print('Gia tri bien hien tai la :', var)
    var = var - 1
    if var == 5:
        break

print("Good bye!")
```

```
Gia tri bien hien tai la : 10
Gia tri bien hien tai la : 9
Gia tri bien hien tai la : 8
Gia tri bien hien tai la : 7
Gia tri bien hien tai la : 6
Good bye!
```



## 3.1. Lệnh break

---

```
l = ["abc123", "ab1234", "a12345"]
for s in l:
    for i in s:
        if i=='1':
            break
        print(i, end = " ")
    print(s, "is processed")
print("Good bye!")
```

```
a b c abc123 is processed
a b ab1234 is processed
a a12345 is processed
Good bye!
```





## 3.2. Lệnh continue

---

- Lệnh continue bỏ qua toàn bộ các lệnh còn lại, chuyển điều khiển tới đầu vòng lặp
- Lệnh continue có thể được sử dụng trong cả vòng lặp while hoặc vòng lặp for.



## 3.2. Lệnh continue

---

```
for letter in 'Python':  
    if letter == 'h':  
        continue  
    print('Chu cai hien tai :', letter)
```

```
Chu cai hien tai : P  
Chu cai hien tai : y  
Chu cai hien tai : t  
Chu cai hien tai : o  
Chu cai hien tai : n
```



## 3.2. Lệnh continue

---

```
var = 10
while var > 0:
    var = var - 1
    if var == 5:
        continue
    print('Gia tri bien hien tai la :', var)
print("Good bye!")
```

```
Gia tri bien hien tai la : 9
Gia tri bien hien tai la : 8
Gia tri bien hien tai la : 7
Gia tri bien hien tai la : 6
Gia tri bien hien tai la : 4
Gia tri bien hien tai la : 3
Gia tri bien hien tai la : 2
Gia tri bien hien tai la : 1
Gia tri bien hien tai la : 0
Good bye!
```



## 3.3. Lệnh pass

---

- Là lệnh không làm gì cả
- Dùng khi cần phải có một lệnh gì đó, nhưng tạm thời chưa biết làm gì cả
- Ví dụ:

```
if 5:  
    print("Good bye")
```

**Error**

```
if 5:  
    pass  
    print("Good bye")
```

**OK**



# Nội dung

---

1. Lệnh rẽ nhánh
2. Vòng lặp
3. Lệnh break, continue, pass
4. **Lệnh else với vòng lặp**



## 4. Lệnh else với vòng lặp

---

- Python cho phép sử dụng một lệnh else kết hợp với một lệnh vòng lặp (**while** và **for**)
- Khi else được sử dụng với vòng lặp while, lệnh else được thực thi khi điều kiện là False.
- Khi else được sử dụng với vòng lặp for, lệnh else chỉ thực thi khi vòng lặp for chạy bình thường (không gặp lệnh break)



# Sử dụng lệnh else với vòng lặp while

- Khi else được sử dụng với vòng lặp while, lệnh else được thực thi khi điều kiện là False.
- Ví dụ: in các số nhỏ hơn 5, khi số không nhỏ hơn 5, thực thi lệnh else

```
count = 0
while count < 5:
    print(count, " là nhỏ hơn 5")
    count = count + 1
else:
    print(count, " là không nhỏ hơn 5")
```

```
0  là nhỏ hơn 5
1  là nhỏ hơn 5
2  là nhỏ hơn 5
3  là nhỏ hơn 5
4  là nhỏ hơn 5
5  là không nhỏ hơn 5
```



# Sử dụng lệnh else với vòng lặp for

- Lệnh else chỉ thực thi khi vòng lặp for chạy bình thường (không gặp lệnh break)

```
numbers = [11, 33, 55, 39, 55, 75, 37, 21, 23, 41, 13]
```

```
for num in numbers:
    if num%2 == 0:
        print ('Danh sach chua 1 so chan')
        break
else:
    print ('Danh sach khong chua so chan nao')
```

Danh sach khong chua so chan nao