



Trường ĐH Khoa Học Tự Nhiên Tp. Hồ Chí Minh

TRUNG TÂM TIN HỌC



Lập trình Python cơ bản

Bài 5: Cấu trúc lặp



Phòng Lập Trình - Mạng

2023

Nội dung

1. Giới thiệu
2. Cấu trúc lặp
3. Dùng else với cấu trúc lặp
4. Sử dụng break, continue và pass



1. Giới thiệu

- ❑ Vòng lặp là thuật ngữ dùng để chỉ hành động lặp đi lặp lại nhiều lần. Trong chương trình, đôi lúc chúng ta muốn chạy đi chạy lại một phần code nào đó thì đó là lúc chúng ta cần đến vòng lặp.
- ❑ Có 2 loại vòng lặp chính trong Python:
 - Vòng lặp **while**
 - Vòng lặp **for**

Nội dung

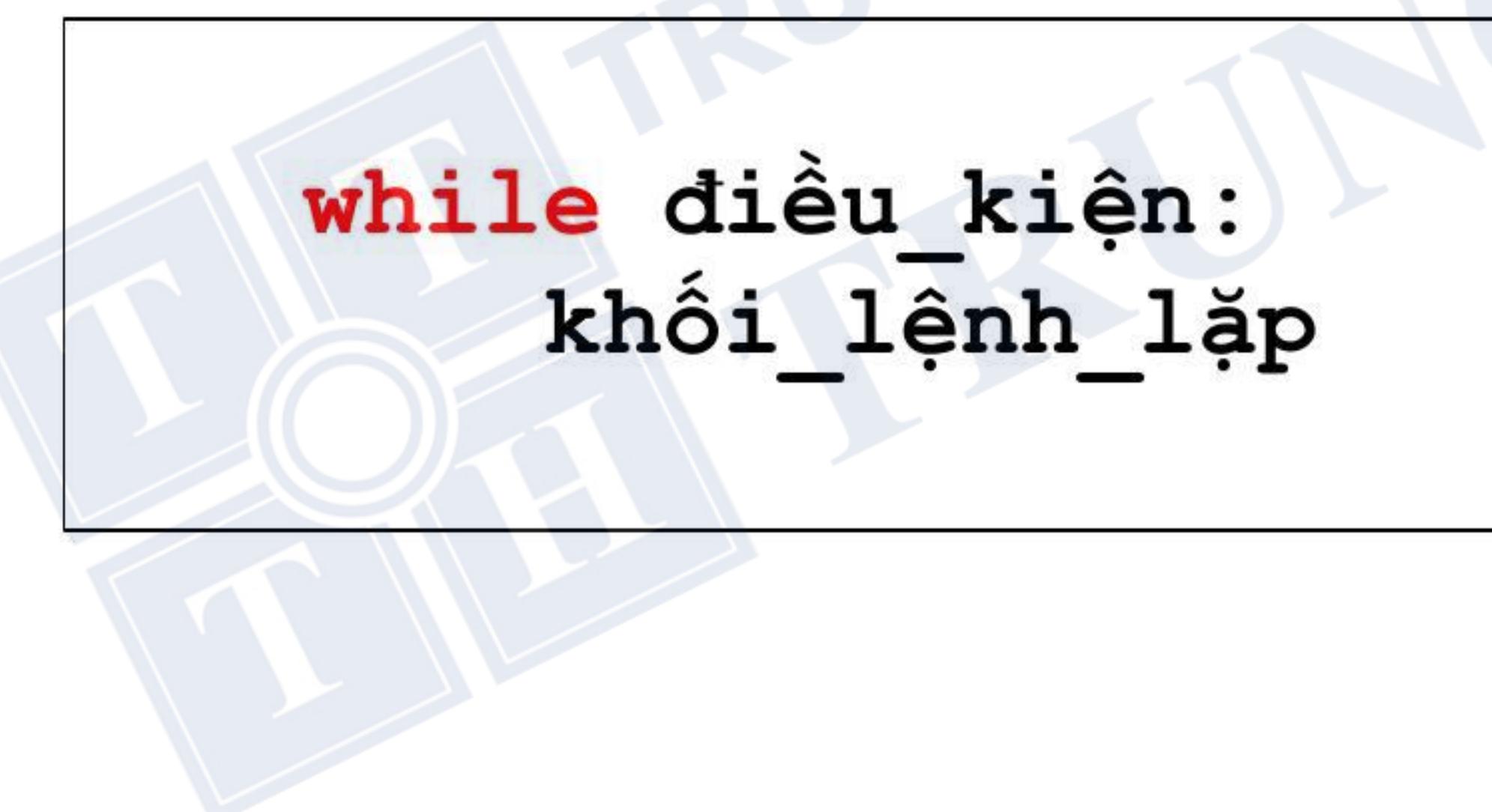
1. Giới thiệu
2. Cấu trúc lặp
3. Dùng else với cấu trúc lặp
4. Sử dụng break, continue và pass



2. Cấu trúc lặp

❑ Vòng lặp while:

- Dùng để thi hành một lệnh hay khối lệnh trong khi biểu thức điều kiện còn đúng.
- Điều kiện được kiểm tra trước khi các lệnh được thi hành.



2. Cấu trúc lặp

❑ Vòng lặp while:

- Ví dụ 1: In ra các số từ 1 đến 10

```
i = 1  
while i <= 10:  
    print("i =", i)  
    i += 1
```

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

2. Cấu trúc lặp

❑ Vòng lặp while:

- Ví dụ 2: In bảng cửu chương 3

```
print("CT in bảng cửu chương 3")
i = 1
while i <= 10:
    print("3 x", i, "=", 3*i)
    i += 1
```

CT in bảng cửu chương 3
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30

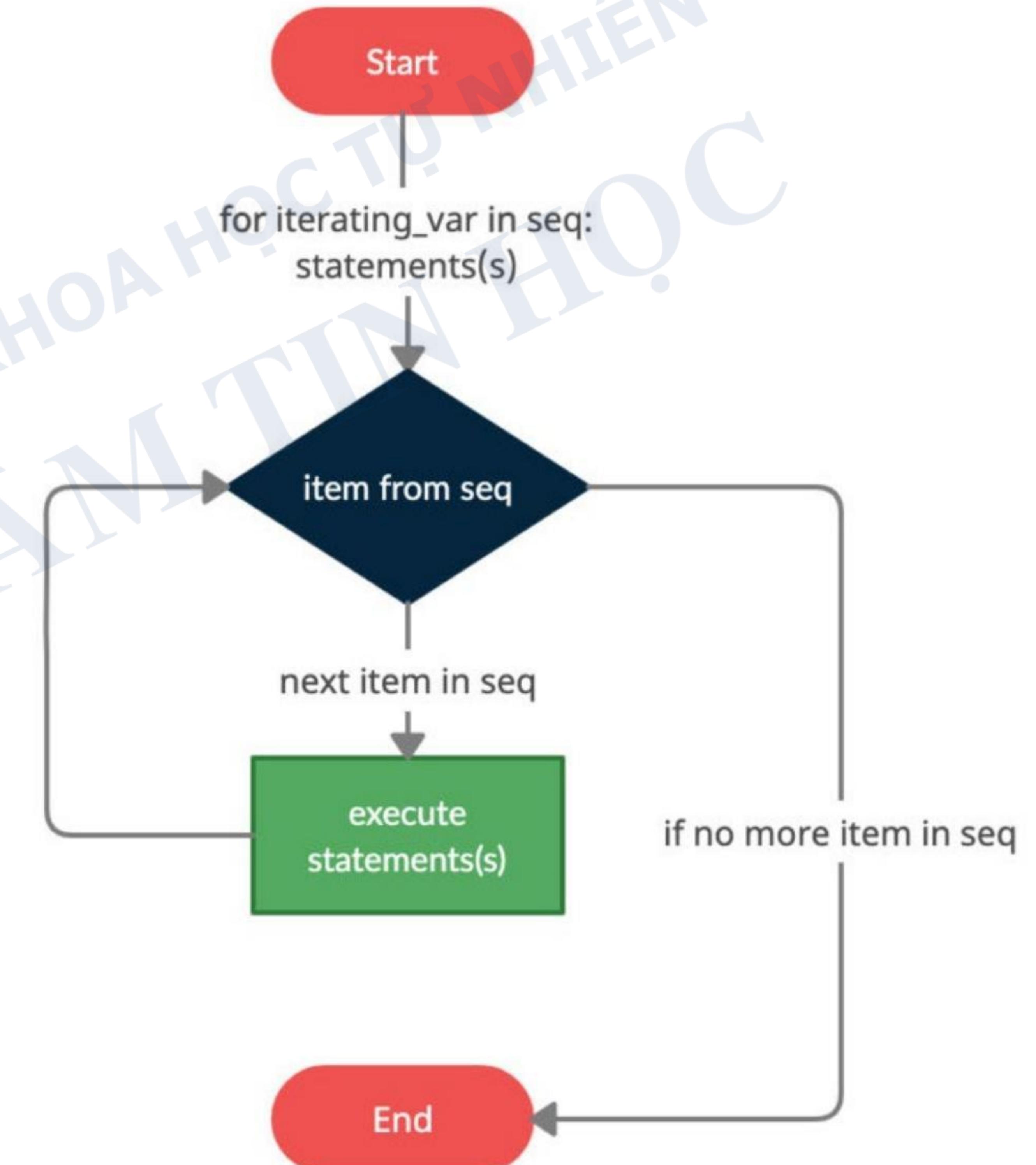
2. Cấu trúc lặp

❑ Vòng lặp for:

- Được sử dụng để lặp một biến dữ liệu qua một dãy (List, Tuple hoặc String) theo thứ tự mà chúng xuất hiện.

```
for iterating_var in sequence:
    khởi_lệnh_lặp
```

- iterating_var:** biến tạm dùng để chứa các giá trị trong sequence ở từng lần lặp.
- sequence:** danh sách (lists, strings, tuples, dictionary, set, range)

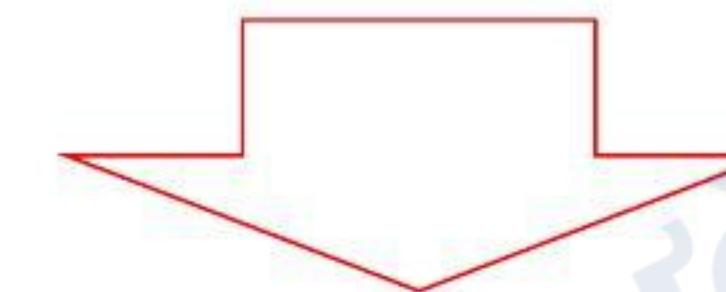


2. Cấu trúc lặp

❑ Vòng lặp for:

- Ví dụ 1: In các giá trị trong list:

```
danh_sach_hang_xe = ['BMW', 'Audi', 'Toyota', 'Ford', 'Honda']
for hang_xe in danh_sach_hang_xe:
    print('Hãng xe:', hang_xe)
```



Hãng xe: BMW
Hãng xe: Audi
Hãng xe: Toyota
Hãng xe: Ford
Hãng xe: Honda

2. Cấu trúc lặp

❑ Vòng lặp for:

- Ví dụ 2: Tính tổng các giá trị trong list:

```
danh_sach_so = [1, 3, 5, 7, 9]
tong = 0
for so in danh_sach_so:
    tong = tong + so
print('Tổng các giá trị trong list:', tong)
```

Tổng các giá trị trong list: 25

2. Cấu trúc lặp

❑ Vòng lặp for:

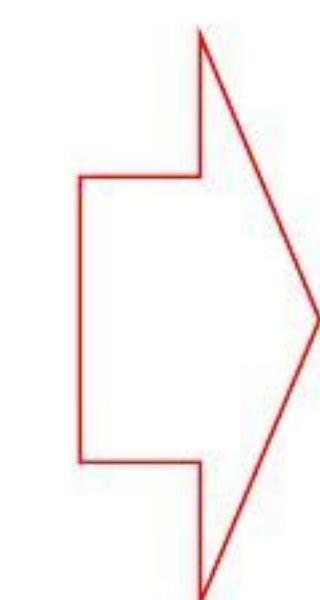
- Hàm `range()`

```
range(start, stop[, step])
```

- **start**: giá trị bắt đầu của vòng lặp.
- **stop**: giá trị để vòng lặp kết thúc (`stop-1`).
- **step**: bước nhảy (mặc định là 1).

- Ví dụ:

```
print('\nDạng 1:', list(range(10)))
print('\nDạng 2:', list(range(1, 10)))
print('\nDạng 3:', list(range(0, 10, 2)))
```



Dạng 1: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
Dạng 2: [1, 2, 3, 4, 5, 6, 7, 8, 9]
Dạng 3: [0, 2, 4, 6, 8]

2. Cấu trúc lặp

❑ Vòng lặp for:

- Ví dụ 3: In ra các giá trị số trong khoảng từ 1 đến 10:

```
for so in range(1, 11):  
    print('Số:', so)
```

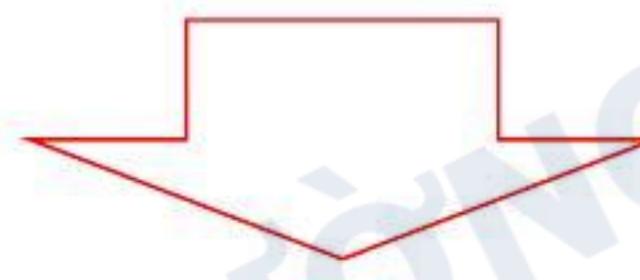
Số: 1
Số: 2
Số: 3
Số: 4
Số: 5
Số: 6
Số: 7
Số: 8
Số: 9
Số: 10

2. Cấu trúc lặp

❑ Vòng lặp for:

- Ví dụ 4: In câu chào:

```
danh_sach_ten = ['Minh', 'Hoa', 'Nam', 'Lan', 'Trinh']
for ten in range(len(danh_sach_ten)):
    print("Xin chào", danh_sach_ten[ten])
```



Xin chào Minh
Xin chào Hoa
Xin chào Nam
Xin chào Lan
Xin chào Trinh

Nội dung

1. Giới thiệu
2. Cấu trúc lặp
3. Dùng else với cấu trúc lặp
4. Sử dụng break, continue và pass



3. Dùng else với cấu trúc lặp

- Python hỗ trợ dùng **else** với cấu trúc lặp:

- Đối với vòng lặp **while**: khối lệnh **else** sẽ được thực hiện khi điều kiện lặp trở thành **False**.

```
while điều_kiện:  
    khối_lệnh_lặp  
else:  
    khối_lệnh_else
```

- Đối với vòng lặp **for**: khối lệnh **else** sẽ được thực hiện sau khi for **đã duyệt xong** danh sách.

```
for iterating_var in sequence:  
    khối_lệnh_lặp  
else:  
    khối_lệnh_else
```

3. Dùng else với cấu trúc lặp

- Python hỗ trợ dùng **else** với cấu trúc lặp:

- Ví dụ 1: Cho người dùng nhập vào 1 chuỗi. Đếm số ký tự trắng (space) có trong chuỗi:

```
chuoi = input('Nhập chuỗi: ')
i = 0
dem = 0
while i <= len(chuoi)-1:
    if chuoi[i] == ' ':
        dem += 1
    i += 1
else:
    print(f'Chuỗi "{chuoi}" có chứa {dem} ký tự trắng')
```

Nhập chuỗi: trungtamtinhoc
Chuỗi "trungtamtinhoc" có chứa 0 ký tự trắng

Nhập chuỗi: Python cơ bản
Chuỗi "Python cơ bản" có chứa 2 ký tự trắng

3. Dùng else với cấu trúc lặp

- Python hỗ trợ dùng **else** với cấu trúc lặp:

- Ví dụ 2: Cho người dùng nhập vào n giá trị số nguyên. Tính giá trị trung bình của các giá trị nhập:

```
n = int(input('Bạn muốn nhập bao nhiêu giá trị?\t'))
tong = 0
for i in range(1, n+1):
    gia_tri = int(input(f'Nhập số thứ {i}: '))
    tong += gia_tri
else:
    print(f'Giá trị trung bình: {tong/n}')
```

Bạn muốn nhập bao nhiêu giá trị?
Nhập số thứ 1: 6
Nhập số thứ 2: 4
Nhập số thứ 3: 9
Nhập số thứ 4: 2
Nhập số thứ 5: 5
Giá trị trung bình: 5.2

5

Nội dung

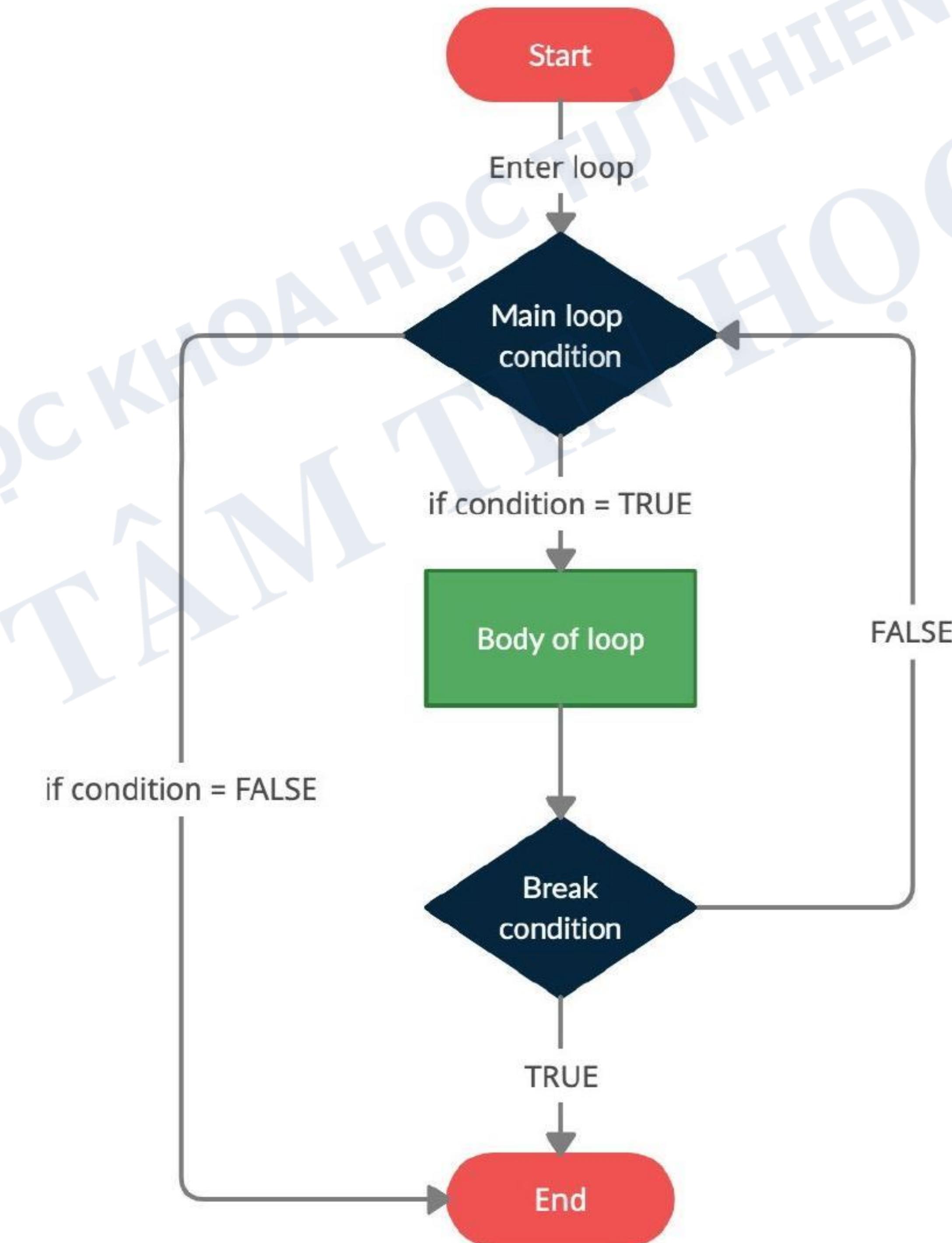
1. Giới thiệu
2. Cấu trúc lặp
3. Dùng else với cấu trúc lặp
4. Sử dụng break, continue và pass



4. Sử dụng break, continue và pass

❑ Sử dụng break:

- Công dụng: Cho phép thay đổi thứ tự thi hành của chương trình.
- **break**: Được dùng để kết thúc việc thực hiện của lệnh lặp.



4. Sử dụng break, continue và pass

□ Sử dụng break:

- Ví dụ 1: Cho người dùng nhập vào 1 số nguyên. Kiểm tra số nguyên đó có xuất hiện trong danh sách số hay không:

```
danh_sach_so = [1, 2, 3, 4, 5, 6, 7, 8]
gia_tri_tim = int(input('Giá trị cần tìm: '))
n = 0
while n <= len(danh_sach_so) - 1:
    if danh_sach_so[n] == gia_tri_tim:
        print(f'Tìm thấy giá trị {gia_tri_tim} tại vị trí index = {n}')
        break
    n += 1
else:
    print(f'Không tìm thấy giá trị {gia_tri_tim} trong danh sách')
```

Giá trị cần tìm: 7
Tìm thấy giá trị 7 tại vị trí index = 6

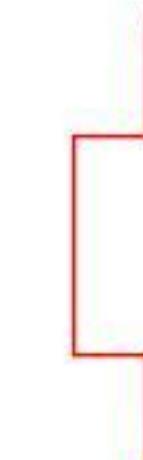
Giá trị cần tìm: 10
Không tìm thấy giá trị 10 trong danh sách

4. Sử dụng break, continue và pass

□ Sử dụng break:

- Ví dụ 2: Cho người dùng nhập vào 1 chuỗi. Kiểm tra chuỗi đó có chứa ký tự đặc biệt hay không:

```
ds_ky_tu_dac_biet = ['@', '!', '$', '#', '%', '&']
chuoi = input('Nhập chuỗi: ')
i = 0
while i <= len(chuoi)-1:
    if chuoi[i] in ds_ky_tu_dac_biet:
        print(f'Chuỗi "{chuoi}" có chứa ký tự đặc biệt')
        break
    i += 1
else:
    print(f'Chuỗi "{chuoi}" không chứa ký tự đặc biệt')
```



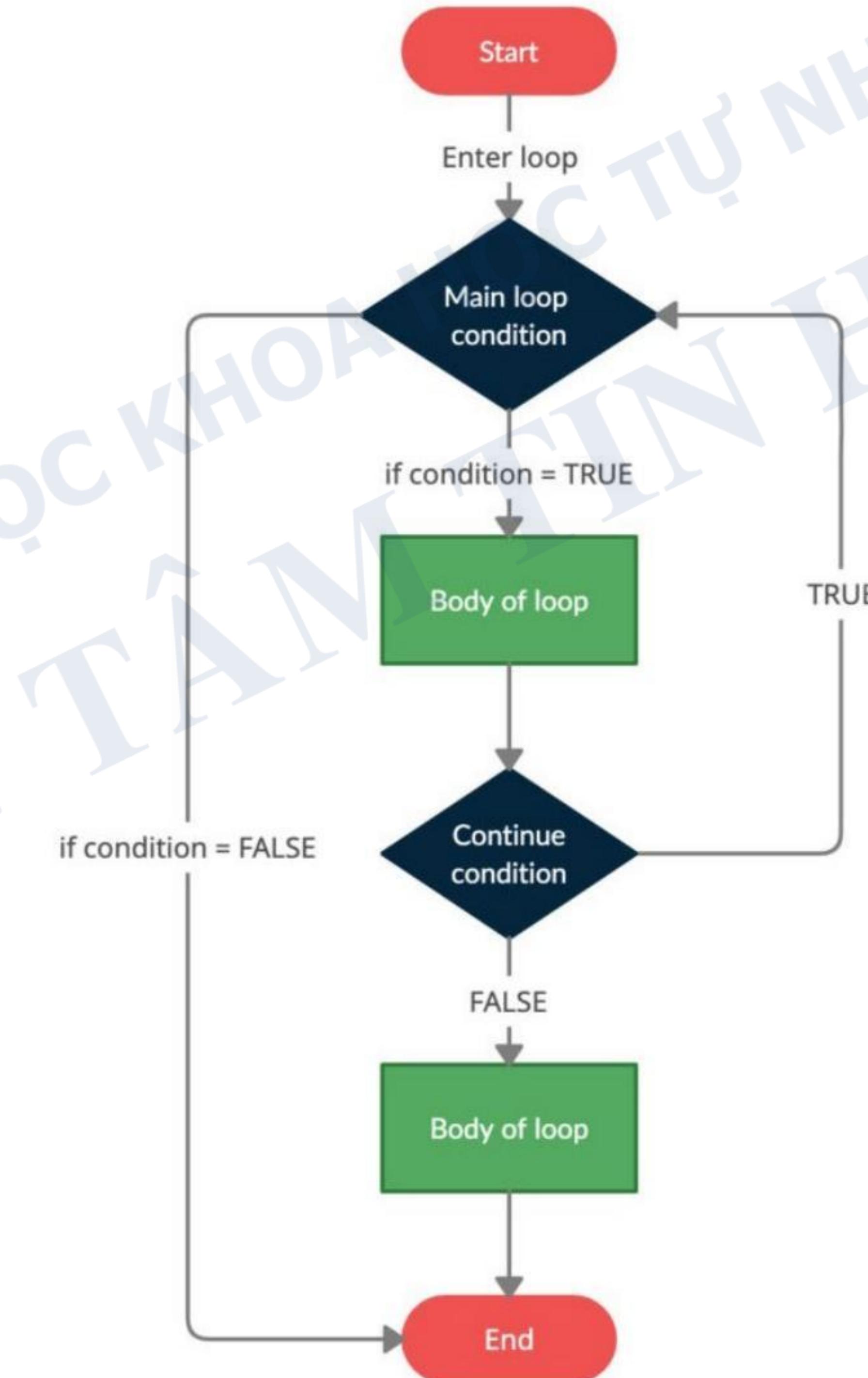
Nhập chuỗi: p@ssword
Chuỗi "p@ssword" có chứa ký tự đặc biệt

Nhập chuỗi: trungtamtinhoc
Chuỗi "trungtamtinhoc" không chứa ký tự đặc biệt

4. Sử dụng break, continue và pass

❑ Sử dụng continue:

- Bỏ qua lần lặp hiện hành và quay về đầu vòng lặp kiểm tra lại điều kiện, nếu thỏa thì tiếp tục lặp, nếu không thỏa thì thoát



4. Sử dụng break, continue và pass

□ Sử dụng continue:

- Ví dụ 1: Cho vòng lặp chạy từ 10 đến 1. In ra những giá trị là số lẻ:

```
n = 10
while n >= 1:
    if n % 2 == 0:
        n -= 1
        continue
    print('Số lẻ:', n)
    n -= 1
else:
    print('Kết thúc')
```

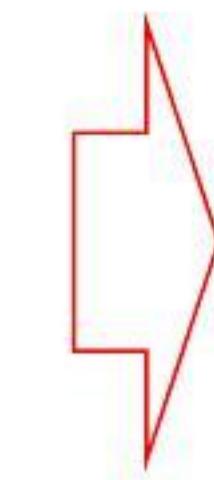
Số lẻ: 9
Số lẻ: 7
Số lẻ: 5
Số lẻ: 3
Số lẻ: 1
Kết thúc

4. Sử dụng break, continue và pass

❑ Sử dụng continue:

- Ví dụ 2: Cho danh sách tên. In ra những tên có số lượng ký tự khác 4:

```
danh_sach_ten = ['Mai', 'Cúc', 'Minh', 'Hoàng', 'Nam', 'Trân', 'Hồng']
for ten in danh_sach_ten:
    if len(ten) == 4:
        continue
    print('Xin chào', ten)
print('Kết thúc')
```

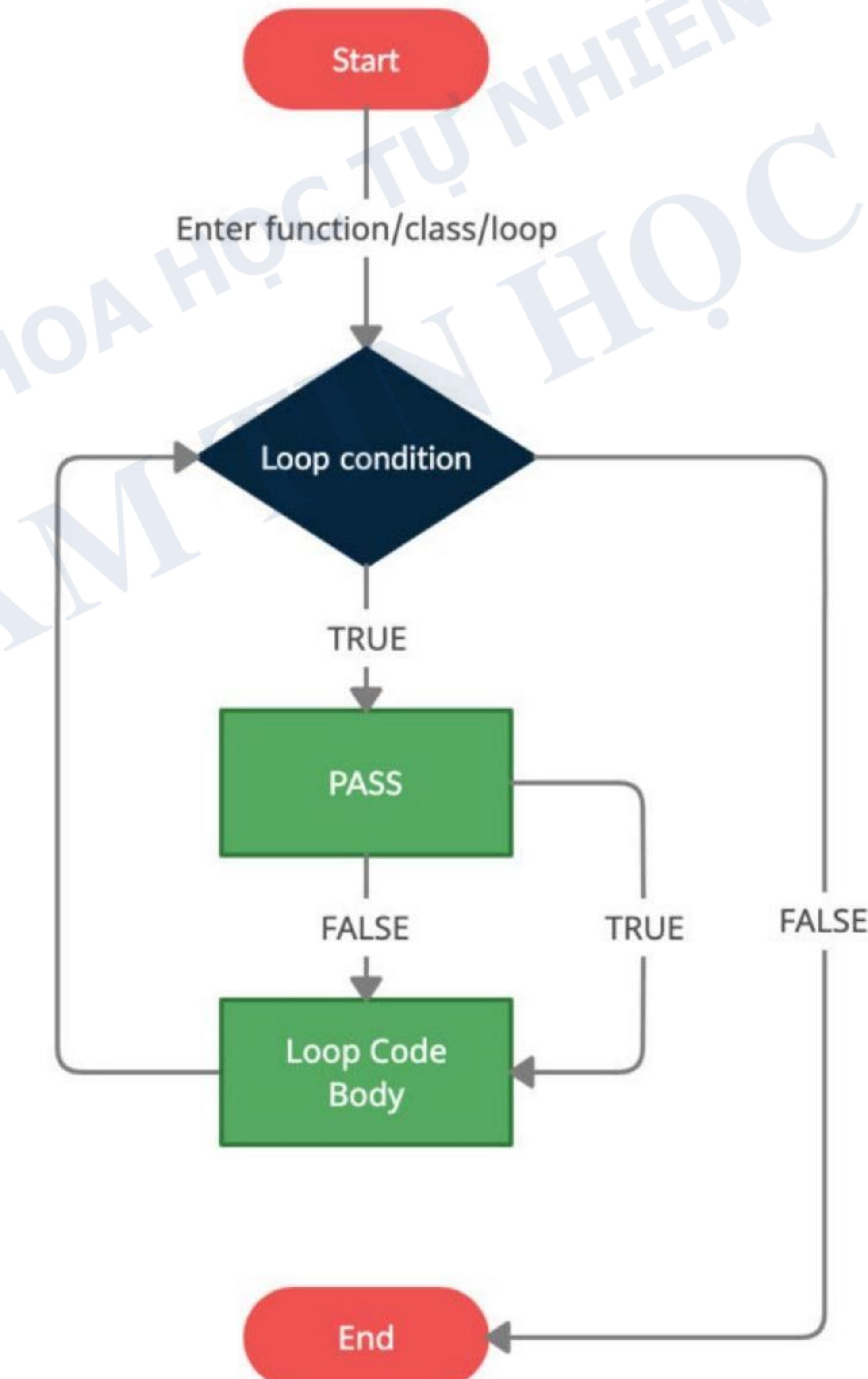


Xin chào Mai
Xin chào Cúc
Xin chào Hoàng
Xin chào Nam
Kết thúc

4. Sử dụng break, continue và pass

❑ Sử dụng pass:

- Được sử dụng khi một câu lệnh được yêu cầu nhưng ta không muốn bắt cứ lệnh hoặc code nào thực thi.
- Là một toán tử null, không có gì xảy ra khi nó thực thi.
- Sử dụng hữu ích ở những nơi code sẽ đi qua nhưng chưa được viết.



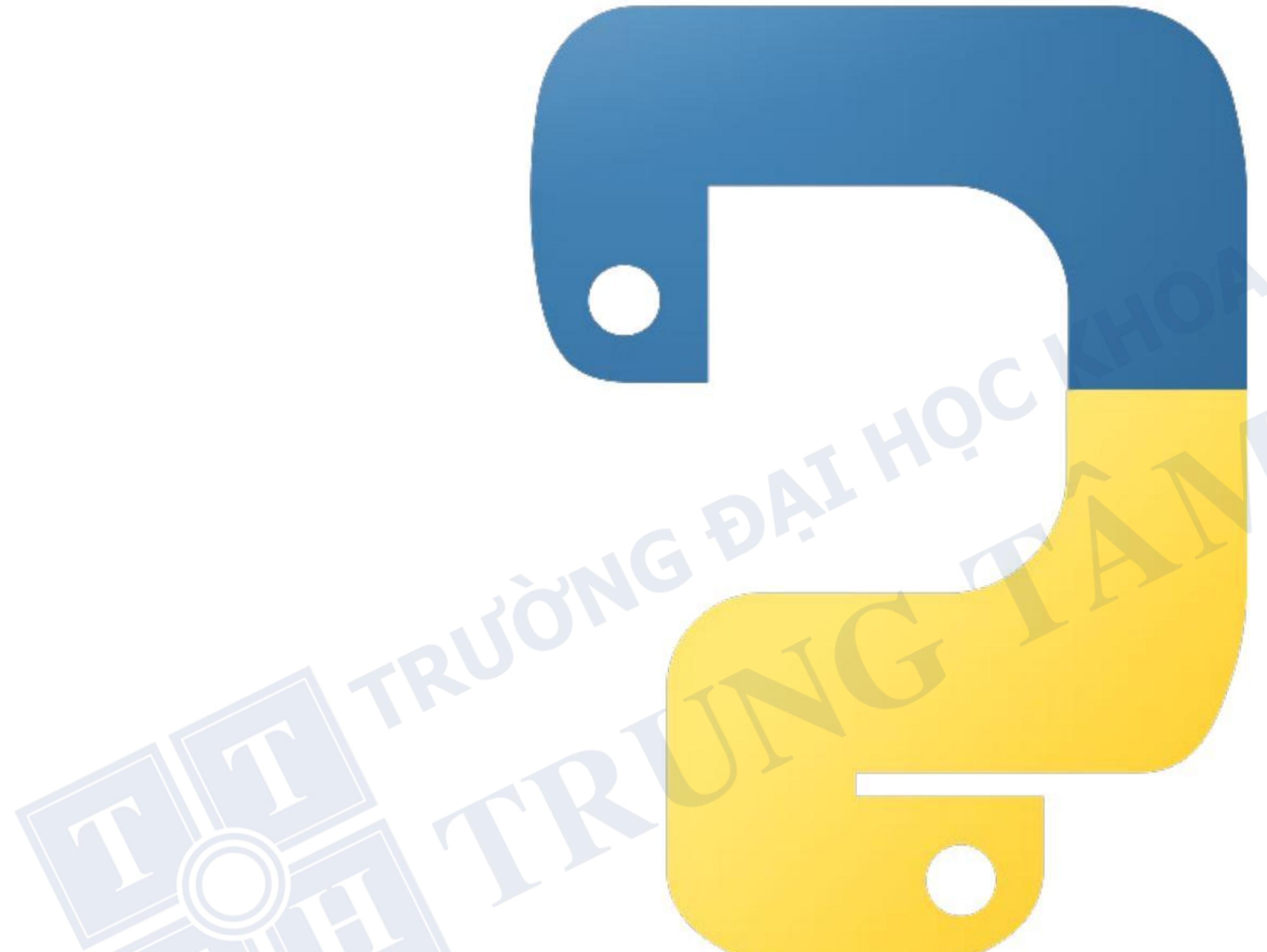
4. Sử dụng break, continue và pass

□ Sử dụng pass:

- Ví dụ: Cho danh sách số. In ra những giá trị là số chẵn:

```
danh_sach_so = [1, 3, 6, 33, 76, 29, 17, 60, 100, 47, 53, 88]
print('Danh sách số chẵn: ')
for so in danh_sach_so:
    if so % 2 != 0:
        pass
    else:
        print(so)
```

Danh sách số chẵn:
6
76
60
100
88



TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
TIT TRUNG TÂM TIN HỌC
TTH