

Vòng lặp (Loop)

- Khái niệm
- Vòng lặp (Loop) là một cấu trúc điều khiển trong lập trình, giúp lặp đi lặp lại một khối lệnh nào đó với số lần lặp BIẾT TRƯỚC hoặc KHÔNG BIẾT TRƯỚC
- Giúp giảm thiểu số dòng mã phải viết, tiết kiệm thời gian
- Giúp chương trình trở nên dễ đọc, dễ hiểu, dễ bảo trì hơn

Phân loại

Lặp với số lần lặp BIẾT TRƯỚC

For...loop

- Cú pháp (Syntax)
- Cấu trúc
- Luồng hoạt động
- Lưu đồ (Flowchart)

```
for(initialization, condition, updation) {  
    // body  
}
```

```
for(int i = 1; i <= 10; i = i + 1) {  
    printf("Hello world");  
}
```

Biến điều khiển vòng lặp i

Initialization - Khởi tạo giá trị cho biến điều khiển vòng lặp

Condition - Điều kiện kiểm tra xem vòng lặp có được tiếp tục thực thi hay không

Body - Đoạn mã được lặp đi lặp lại bởi vòng lặp for

Updation - Cập nhật lại giá trị (Tăng/Giảm) biến điều khiển vòng lặp

Vòng lặp for hoạt động dựa trên một biến điều khiển. Thông thường biến điều khiển này được gọi với các chữ cái i, j, k, ...

Bước 1: Initialization - Thiết lập giá trị khởi tạo cho biến điều khiển vòng lặp. Bước này chỉ xảy ra 1 lần duy nhất trong lần lặp đầu tiên của vòng lặp

Bước 2: Condition - Sau khi biến điều khiển vòng lặp (i) đã có giá trị, biến điều khiển này sẽ được kiểm tra với điều kiện condition.

Bước 3: Body - Sau khi biến điều khiển vòng lặp đã thỏa mãn điều kiện (condition) thì đoạn mã trong body của vòng lặp sẽ được thực thi

Bước 4: Updation - Cập nhật lại giá trị (Tăng/Giảm) biến điều khiển vòng lặp

Bước 5: Repetition - Sau khi biến điều khiển vòng lặp được cập nhật lại giá trị, vòng lặp quay lại Bước 2 và lặp đi lặp lại cho đến khi điều kiện condition không thỏa mãn (false) thì sẽ dừng lại

- Lặp lại tác vụ in ra chuỗi "Hello world" 10 lần
- Thông thường giá trị khởi tạo sẽ là các giá trị 0, 1 hoặc tùy thuộc vào tình huống của các bài toán trong thực tế
- Nếu thỏa mãn (true), đoạn mã trong phần body của vòng lặp mới được thực thi
- Nếu không thỏa mãn (false), vòng lặp ngay lập tức dừng lại

Lặp với số lần lặp KHÔNG BIẾT TRƯỚC

While...loop

- Cú pháp (Syntax)
- Cấu trúc
- Luồng hoạt động
- Lưu đồ (Flowchart)
- Lưu ý

```
while(condition) {  
    // body  
}
```

```
int result = 10;  
int answer;  
while(answer != 10) {  
    printf("Please input a random number");  
    scanf("%d", &answer);  
}  
printf("Congratulation !!!");
```

Condition - Điều kiện kiểm tra xem vòng lặp có được tiếp tục thực thi hay không

Body - Đoạn mã được lặp đi lặp lại bởi vòng lặp while

Vòng lặp while hoạt động dựa trên một điều kiện (condition) nhất định

Bước 1: Condition - Điều kiện khởi tạo để vòng lặp while có thể hoạt động được.

Bước 2: Body - Sau khi điều kiện điều khiển vòng lặp while (condition) được thỏa mãn (true), đoạn mã trong body của vòng lặp sẽ được thực thi

Bước 3: Repetition - Vòng lặp tiếp tục quay lại Bước 1 và lặp đi lặp lại cho đến khi điều kiện condition không thỏa mãn (false) thì sẽ dừng lại

- Nếu điều kiện thỏa mãn (true), đoạn mã trong phần body của vòng lặp mới được thực thi
- Nếu điều kiện không thỏa mãn (false), vòng lặp ngay lập tức dừng lại
- Vòng lặp vô hạn (Infinite loop) xảy ra khi điều kiện lặp (condition) mãi mãi luôn đúng.
- Cần chú ý trong việc thiết lập điều kiện lặp cho vòng lặp while
- Điều kiện lặp (condition) thường liên quan trực tiếp đến bài toán đang cần giải quyết trong thực tế
- Có thể tạo ra 1 biến dùng để kiểm soát vòng lặp while
- Biến kiểm soát vòng lặp sẽ được dùng để thiết lập điều kiện lặp
- Tùy thuộc vào các tình huống của bài toán trong thực tế, cập nhật lại giá trị của biến kiểm soát vòng lặp để condition không còn thỏa mãn -> Vòng lặp dừng lại
- ```
int result = 10;
int answer;
int loop = 1;
while(loop == 1) {
 printf("Please input a random number");
 scanf("%d", &answer);
 if (answer == result) {
 loop = 0;
 }
}
printf("Congratulation !!!");
```

break

Từ khóa dùng để ngay lập tức dừng vòng lặp và thoát ra khỏi vòng lặp tại bất kỳ thời điểm nào

```
for(int i = 1; i <= 10; i = i + 1) {
 printf("Hello world");
 if (i == 3) {
 break;
 }
}
```

Vòng lặp dừng lại tại thời điểm giá trị của biến kiểm soát vòng lặp i == 3

continue

Từ khóa dùng để dừng vòng lặp tại lần lặp hiện tại và chuyển đến lần lặp tiếp theo

Toàn bộ đoạn mã dưới từ khóa continue sẽ bị dừng lại (không thực thi) và ngay lập tức vòng lặp sẽ chuyển đến lần lặp tiếp theo

```
for(int i = 1; i <= 10; i = i + 1) {
 if (i == 3) {
 continue;
 }
 printf("%d, Hello world", i);
}
```

Vòng lặp sẽ bỏ qua toàn bộ logic trong body tại thời điểm i == 3

goto

Từ khóa dùng để chỉ định vòng lặp sẽ dừng lại và chương trình sẽ nhảy đến một nhãn (label) được khai báo trước

Không được khuyến khích sử dụng vì có thể gây khó hiểu và làm giảm tính rõ ràng của mã nguồn

```
for(int i = 1; i <= 10; i = i + 1) {
 if (i == 3) {
 goto exitLoop;
 }
 printf("%d, Hello world", i);
}
exitLoop:
printf("Exit loop when i is 3\n");
```

return

Từ khóa dùng để trả về một giá trị nào đó trong 1 hàm (Function)

Ngay lập tức dừng toàn bộ logic trong hàm -> Vòng lặp cũng sẽ bị dừng lại

Lệnh điều khiển vòng lặp