

CONFIDENTIAL

C Programming Introduction

Week 8: Loops

For HEDSPI Project

Cấu trúc vòng lặp While

- **Câu lệnh While**
 - Nếu biểu thức điều kiện đúng, các câu lệnh được thực hiện. Quá trình lặp lại đến khi biểu thức điều kiện sai.

```
while (expression) {  
    Statement1;  
    Statement2;  
    ...  
}
```

Cấu trúc vòng lặp While

- **Ví dụ**

```
#include <stdio.h>  
#define PERIOD '.'  
main() {  
    char C;  
    while ((C = getchar()) != PERIOD)  
        putchar(C);  
    printf("Good Bye.\n");  
}
```

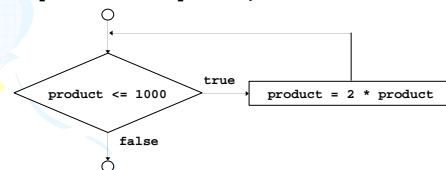


Result?

Cấu trúc vòng lặp While

- **Ví dụ:**

```
int product = 2;  
while ( product <= 1000 )  
    product = 2 * product;
```



Cấu trúc vòng lặp do-while

- **Câu lệnh Do-While**

- Vòng lặp do-while kiểm tra điều kiện sau khi thực hiện các câu lệnh. Nếu điều kiện đúng thì tiếp tục. Nếu điều kiện sai thì thoát khỏi vòng lặp. Vòng lặp này thực hiện ít nhất 1 lần.

```
do {  
    statement1;  
    statement2;  
    ...  
} while (expression);
```

Cấu trúc vòng lặp do-while

- **Ví dụ**

```
int i = 1, sum = 0;  
do {  
    sum += i;  
    i++;  
} while (i <= 50);  
printf("The sum of 1 to 50 is %d\n", sum);
```



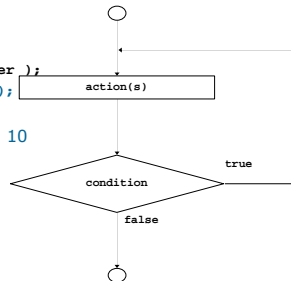
Result?

Cấu trúc vòng lặp do-while

- Ví dụ

```
counter = 1;
do {
    printf( "%d ", counter );
} while (++counter <= 10);
```

In các số nguyên từ 1 đến 10



Continue và Break

- **Câu lệnh Break và Continue**

- Câu lệnh **break** để thoát ngay lập tức khỏi vòng lặp

```
break;
```

- Câu lệnh **continue** bỏ qua các lệnh trong phần còn lại của vòng lặp và bắt đầu ngay vòng lặp tiếp theo.

```
continue;
```

Continue và Break

- Ví dụ

```
int c;
while ((c = getchar()) != -1) {
    if (C == '\n')
        break;
    else if (c >= '0' && c <= '9')
        continue;
    else putchar(c);
}
printf("*** Good Bye ***\n");
```

Bài 8.1

- Viết chương trình chép nội dung dữ liệu vào từ bàn phím ra màn hình, nhưng thay chuỗi các ký tự `` bằng 1 ký tự `.`.
- Có thể dùng `getchar()` và `putchar()` để thực hiện chương trình.

Bài 8.2

- Viết chương trình thay thế các ký tự `tab`, `\t`, `\b` bằng `\\` trong xâu vào. Sau đó in ra màn hình.
- Bạn có thể dùng `if` hoặc `switch`

Bài 8.3

- Viết và chạy chương trình tính lương cho 1 công ty sau. Quan sát cách thực hiện vòng lặp `while` trong chương trình.

Exercise8_3.c

```
#include <stdio.h>

int
main(void)
{
    double total_pay; /* company payroll */
    int count_emp; /* current employee */
    int number_emp; /* number of employees */
    double hours; /* hours worked */
    double rate; /* hourly rate */
    double pay; /* pay for this period */
    /* Get number of employees. */
    printf("Enter number of employees> ");
    scanf("%d", &number_emp);
}
```

```
/* Compute each employee's pay and add it to the payroll. */
total_pay = 0.0;
count_emp = 0;
while (count_emp < number_emp) {
    printf("Hours> ");
    scanf("%lf", &hours);
    printf("Rate > $");
    scanf("%lf", &rate);
    pay = hours * rate;
    printf("Pay is $%.2f\n", pay);
    total_pay = total_pay + pay;
    count_emp = count_emp + 1;
}
printf("All employees processed\n");
printf("Total payroll is $%.2f\n", total_pay);
return (0);
}
```

Bài 8.4

- Viết chương trình sử dụng vòng lặp *while* để phân tích kết quả thi: đếm số sinh viên đỗ và trượt
- Bạn hỏi người dùng nhập thông tin 1 sinh viên đỗ hay trượt bằng cách vào 1 nếu đỗ và 2 nếu trượt

Bài 8.6

- Sử dụng vòng lặp *do...while* để in ra các số nguyên nhỏ hơn 1 số cho trước.
- Chú ý, vòng lặp *do...while* luôn thực hiện ít nhất 1 lần.

Bài 8.7

- Viết chương trình tính điểm trung bình của 1 tập điểm.
- Chú ý:
 - Cần tính tổng và số điểm được nhập.
 - Đọc dữ liệu đến khi gặp 1 số âm.
 - In ra điểm trung bình

Bài 8.8

- Viết chương trình tính $n!$ sử dụng vòng lặp
- Có thể dùng:
 - Biến đếm i , chạy từ 1 đến n
 - Dùng f để lưu giá trị $i!$