

# Điều khiển lặp (2)

`huydq@soict.hust.edu.vn`

# Lệnh do...while

```
do {  
    <tập lệnh>  
} while ( <điều kiện> );
```

- Là vòng lặp mà lần lặp đầu tiên điều kiện không được kiểm tra
- Tập lệnh trong vòng lặp được thực hiện ít nhất một lần, ngay cả khi điều kiện lặp không bao giờ đúng

# Ví dụ

- Tính tổng của một dãy số nguyên với số cuối cùng là nhập là 0 (ver 3)

tổng = 0

do

{

input số

cộng số vào tổng

} while (số khác 0)

output tổng

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int tong=0, so;
```

```
    printf("Nhap cac so trong day\n");
```

```
    do
```

```
    {
```

```
        scanf("%d", &so);
```

```
        tong += so;
```

```
    }while (so != 0);
```

```
    printf("Tong la %d\n", tong);
```

```
    return 0;
```

```
}
```

# Kiểm tra giá trị bằng do...while

```
int n;

do {
    printf("Input a positive number: ");
    n = -1;
    fflush(stdin);
    if (scanf("%d", &n) != 1)
        printf("Input data is not a number\n");
}while (n<0);

printf("The number is %d\n", n);
```

# Kết quả chạy

- Input a positive number: abc
- Input data is not a number
- Input a positive number: -2
- Input a positive number: 5
- The number is 5

# Vòng lặp vô tận

- Trong vòng for các nhánh khởi tạo, điều kiện và cập nhật là tùy chọn.
- Có thể tạo vòng lặp vô tận bằng lệnh while hoặc for
  - Vòng lặp là vô tận khi điều kiện lặp là luôn luôn đúng

```
while (1)
{
    /* lặp vô tận */
}
```

```
for (;;)
{
    /* lặp vô tận */
}
```

# Lệnh **break**

- Lệnh trực tiếp cho phép thoát ngay khỏi vòng lặp mà không cần kiểm tra điều kiện lặp
- Thường được áp dụng trong các vòng lặp vô tận

```
for (;;)
{
    ...
    if (<điều kiện thoát OK>) break;
}
```

# Ví dụ

```
int n;
while (1) {
    printf("Input a positive number: ");
    fflush(stdin);

    if (scanf("%d", &n) != 1)
        printf("Bad number. Try again\n");
    else if (n<0)
        printf("Must be positive. Try again\n");
    else
        break;
}
printf("The number is %d\n", n);
```



# Kết quả chạy

Input a positive number: abc

Bad number. Try again

Input a positive number: -2

Must be positive. Try again

Input a positive number: 5

The number is 5

# Lệnh `continue`

- Lệnh yêu cầu chạy lại vòng lặp từ đầu
- Thường được áp dụng để bỏ qua lần lặp hiện tại khi một điều kiện nào đó xảy ra

```
for (...)  
{  
    ...  
    if (<cần bỏ qua lần lặp này>) continue;  
    ...  
}
```

# Ví dụ

```
int n;

do {
    printf("Input a positive number: ");
    fflush(stdin);
    if (scanf("%d", &n) != 1) {
        printf("Bad number. Try again\n");
        continue;
    }
    printf("The number is %d\n", n);
}while (n<0);
```

# Kết quả chạy

Input a positive number: abc

Bad number. Try again

Input a positive number: -2

The number is -2

Input a positive number: 5

The number is 5

# Bài tập

- Viết chương trình in ra tích của các thừa số nguyên tố của một số. Ví dụ,  $6 = 2 \times 3$ ,  $24 = 2 \times 2 \times 2 \times 3$ ,  $23 = 23$
- Viết chương trình in ra số nguyên tố đầu tiên lớn hơn một giá trị N.
- Viết chương trình để tính số  $\Pi$  theo công thức sau:  
$$\Pi/4 = 1 - 1/3 + 1/5 - 1/7 + 1/9 - 1/11 + \dots$$

# Lời giải (bài 1)

```
factor = 2;  
do  
{  
    if (n%factor==0) {  
        printf("%5d", factor);  
        n = n / factor;  
    } else {  
        factor++;  
    }  
} while (n>1)
```

# Lời giải (bài 2)

```
first = n+1;
while(1) {
    isPrime = 1;
    for (factor = 2; factor<first; factor++)
        if (first%factor==0) {
            isPrime = 0;
            break;
        }
    if (isPrime) {
        printf("The first greater prime is %d", first);
        break;
    }
    first++;
}
```