

Các cấu trúc điều khiển

Cấu trúc if/else. Việc thực hiện được chuyển hướng tùy theo biểu thức điều kiện nhận giá trị đúng (1) hay sai (0).

Ví dụ:

```
if (size>MAXSIZE )
    size=MAXSIZE;
```

hay:

```
minval=a; maxval=b;
if (a>b) {
    maxval=a; minval=b;
}
```

hay:

```
if (a>b) {
    maxval=a; minval=b;
}
else {
    minval=a; maxval=b;
}
```

Cấu trúc switch case. Thực hiện theo nhiều nhánh rẽ.

<pre>switch (EXPR) { case VAL1: BLK1 [break;] case VAL2: BLK2 [break;] ... case VALm: BLKm [break;] default: BLK0 }</pre>	<pre>float Func(float a, float b, int op) { float result; switch(op) { case '+': result=a+b; break; case '-': result=a-b; break; case '*': result=a*b; break; case '/': result=a/b; break; case Sqrt: case EXP: printf("Not implemented"); break; default: printf("Exception: bug!"); } return result; }</pre>
--	--

Biểu thức nguyên EXPR được lượng giá và so sánh với VAL1, VAL2, ..., VALm; việc thực hiện chương trình được chuyển đến đầu đoạn chương trình BLKj mà có giá trị VALj bằng với EXPR; nếu EXPR không bằng với bất kỳ giá trị VALj nào thì đoạn BLK0 được thực hiện. Sau đó các lệnh được thực hiện tiếp, nếu gặp một lệnh **break** đầu tiên thì thoát ngay khỏi **switch case**, bỏ qua các lệnh còn lại trong cấu trúc này.

Cấu trúc lặp for.

```
for ([khởi động]; [điều kiện dừng]; [các lệnh trước khi lặp lần kế]) {  
    // Các lệnh trong vòng lặp  
}
```

Ví dụ. Tính tổng các số lẻ từ 1 đến 2000.

```
for (sum=0, i=1; i<2000; i=i+2)  
    sum = sum + i;
```

Ví dụ. Nhập và tính tổng 10 số nguyên:

```
for (sum=0, i=1; i<=10; i++) {  
    printf (" So %d = ", i) ;  
    scanf ("%d", & x) ;  
    sum = sum + x;  
}
```

Cấu trúc lặp while/do while.

Cấu trúc while () :

```
while ([Điều kiện dừng])  
{  
    // Các lệnh trong vòng lặp  
}
```

Cấu trúc do...while () :

```
do  
{  
    // Các lệnh trong vòng lặp  
} while ([Điều kiện dừng]);
```

Vòng lặp dừng khi biểu thức [Điều kiện dừng] nhận giá trị sai (0) hay gặp một chỉ thị **break** bên trong vòng lặp. Ngoài ra, khi gặp chỉ thị **continue** thì lần lặp kế tiếp sẽ thực hiện, bỏ qua các lệnh bên dưới.

Ví dụ: Tính tổng các số lẻ từ 1 đến 2000 được viết tương tự như trên nhờ dùng *while()* :

```
sum = 0; i=1;  
while (i<2000)  
{  
    sum = sum + i;  
    i = i+2;  
}
```

Nhập và tính tổng các số thực dương, dừng khi gặp số âm hay 0:

```
sum = 0;  
do {  
    printf("nhap so:"); scanf("%f", &x);  
    if (x > 0)  
        sum = sum + x;  
} while (x > 0);
```