

Introduction to C Programming

— Basics of Programming (4) : Loops (1) —

Waseda University

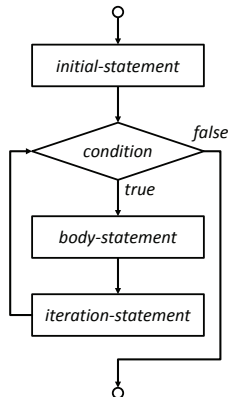
Today's Topics

- Usage of a Looping statements.
 - **for** statement

for statement

The general form of the **for** statement

```
for ( initial-statement ; condition ; iteration-statement ) {  
    body-statement  
}
```



Remark :

- **for** statement have Two semicolons and Three statements in the curly braces.
- *initial-statement* sets up the starting condition for the loop.

Example 1

The sum of 1-100.

```
#include <stdio.h>
int main(void){
    int Sum = 0, i;
    printf("Sum=%d\n",Sum);          /*Output for Sum*/
    for (i=1; i<=100; i=i+1){
        Sum = Sum+i;                 /*Add i to Sum*/
        printf("+[%d] =%d \n",i,Sum); /*Checking the value of Sum*/
    }
    printf("Answer is %d.\n", Sum);
    return 0;
}
```

- Type this program and run it.
- Let filename be "sum.c".

Remarks of Example 1

- The declaration and the assignment of a variable **Sum** can be:
int Sum = 0;
- The problem repeatedly execute the statement inside the **for** from **i=1** to **100**.
- **i++** is equivalent to **i=i+1**.
- **Sum+=i** is equivalent to **Sum=Sum+i**.
- If *body-statement* has only one statement, **{ }** can be omitted.

Example 2

The sum of 1-100.

```
#include <stdio.h>
int main(void){
    int Sum = 0, i;
    printf("Sum=%d\n",Sum);
    for (i=100; i>=1; i--){
        Sum = Sum + i;          /*Add i to Sum*/
        printf("+[%d]  =%d  \n",i,Sum);
    }
    printf("Answer is  %d.\n", Sum);
    return 0;
}
```

- Let's copy "sum.c" and filename be "sum2.c". (cp sum.c sum2.c)
- Type this program to "sum2.c" and run it

Remarks of Example 2

- The problem repeatedly execute the statement inside the **for** from **i=100** to **1**.
- $i--$ is equivalent to $i=i-1$.
- If *body-statement* has only one statement, $\{ \}$ can be omitted.

※ Check the value of **i** after the **for** loop in Example 2?

Operators for Performing Shortcuts

- Compound assignment operators : It combine the assignment operator with another basic arithmetic operation.
 - $a+=b$ is equivalent to $a=a+b$.
 - $a-=b$ is equivalent to $a=a-b$.
- The incrementation operator and the decrementation operator
 - $i++$ is equivalent to $i=i+1$ (increment).
 - $i--$ is equivalent to $i=i-1$ (decrement).

Example: The incrementation operator

```
for(i=1; i<=100; i++)  
    printf("%d\n",i);
```


Exercise

Exercise: Calculate the sum of an odd number

Calculate the sum S_n of an odd number from 1 to $2N - 1$ using **for** statement:

$$S_N = 1 + 3 + 5 + \cdots + 2N - 1,$$

where N is an input number. Let filename be "sum3.c".

Example:

Input number N: 100 [Enter]

Answer is 10000.

Remark:

- If you run an incorrect looping statement, the program doesn't exit in some cases. (**Infinite loop**)
- The program terminate an infinite loop by pressing **[Ctrl] + [c]** keys in the case of infinite loop.

Summary

The general form of the **for** statement

```
for ( initial-statement ; condition ; iteration-statement ) {  
    body-statement  
}
```

