

If else Statement in C

Sometimes we need to execute a block of statements only when a particular condition is met or not met. This is called **decision making**, as we are executing a certain code after making a decision in the program logic. For decision making in C, we have four types of control statements (or control structures), which are as follows:

- a) if statement
- b) nested if statement
- c) if-else statement
- d) if-else-if statement

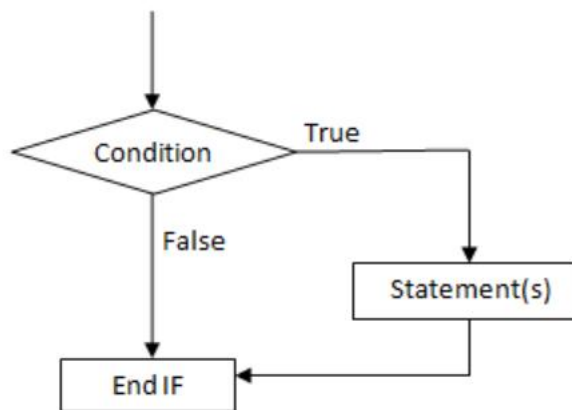
If statement in C

If statement consists a condition, followed by statement or a set of statements as shown below:

```
if(condition){  
    Statement(s);  
}
```

The statements inside **if** parenthesis (usually referred as if body) gets executed only when the given condition is true. If the condition is false then the statements inside if body are completely ignored.

Flow diagram of If statement



Example of if statement

```
#include <stdio.h>
int main(){
int num=70;
if( num < 100 ){
/* This cout statement will only execute, if the above condition is true */
printf(" number is less than 100");
}
printf("Outside if block");

if(num > 100){
/* This cout statement will only execute, if the above condition is true */
printf("number is greater than 100");
}
return 0;
}
```

Output:

number is less than 100

Nested if statement in C

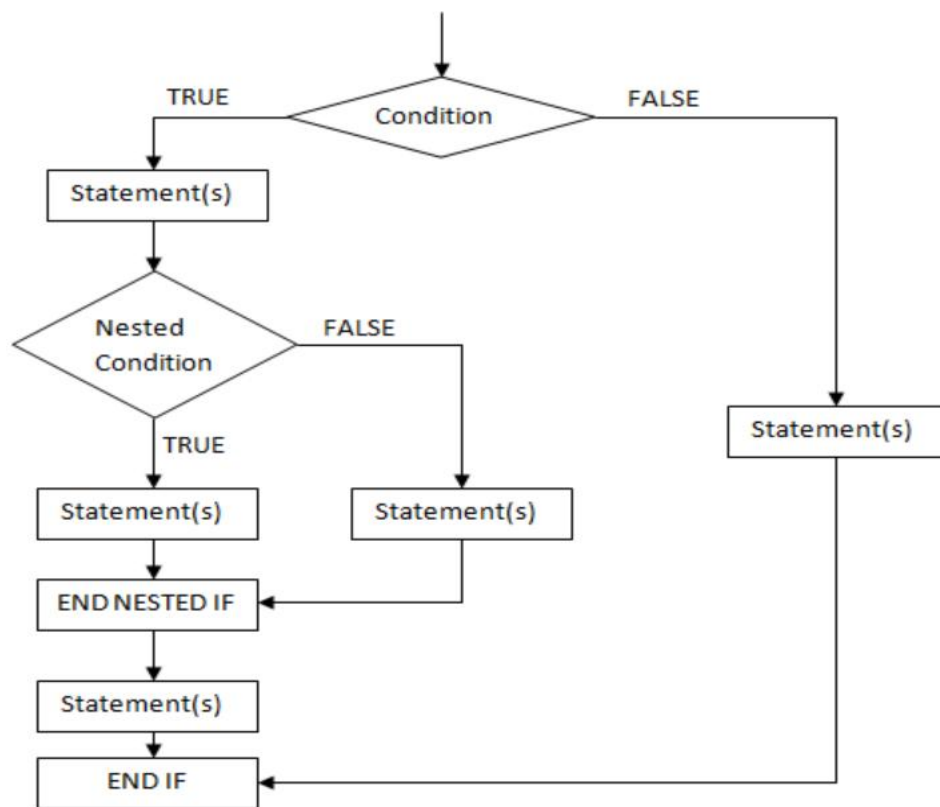
When there is an if statement inside another if statement then it is called the **nested if statement**.

The structure of nested if looks like this:

```
if(condition_1) {
Statement1(s);
if(condition_2) {
Statement2(s);
}}
```

Statement1 would execute if the condition_1 is true. Statement2 would only execute if both the conditions(condition_1 and condition_2) are true.

Flow chart of Nested If statement



Example of Nested if statement

```

#include <stdio.h>
int main(){
    int num=40;
    if( num < 100 ){ // true
        printf("number is less than 100");
        if(num > 50){
            printf("number is greater than 50");
        }
    }
    else{
        printf("number is smaller than 50");
    }
}
else{
    printf("number is greater than 100");
}
printf("After the if else block");
return 0;
}

```

Output:

```

number is less than 100
number is greater than 50

```

```

if(condition1)
{

```

```
// statement
}  
else  
{  
// statement  
}
```

If else Statement in C

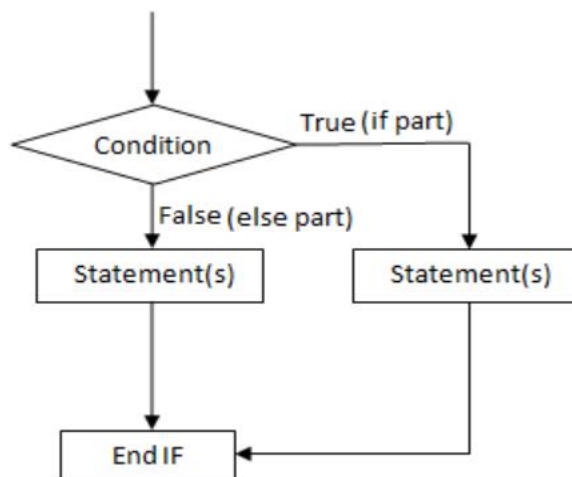
Sometimes you have a condition and you want to execute a block of code if condition is true and execute another piece of code if the same condition is false. This can be achieved in C using if-else statement.

This is how an if-else statement looks:

```
if(condition) {  
    Statement(s);  
} else {  
    Statement(s);  
}
```

The statements inside “if” would execute if the condition is true, and the statements inside “else” would execute if the condition is false.

Flow chart of if else statement



Example of if-else statement

```
#include <stdio.h>
int main(){
    int num=36;
    if( num < 50 ){
        //This would run if above condition is true
        printf("num is less than 50");
    }
    else {
        //This would run if above condition is false
        printf("num is greater than or equal 50");
    }
    return 0;
}
```

Output:

num is greater than or equal 50

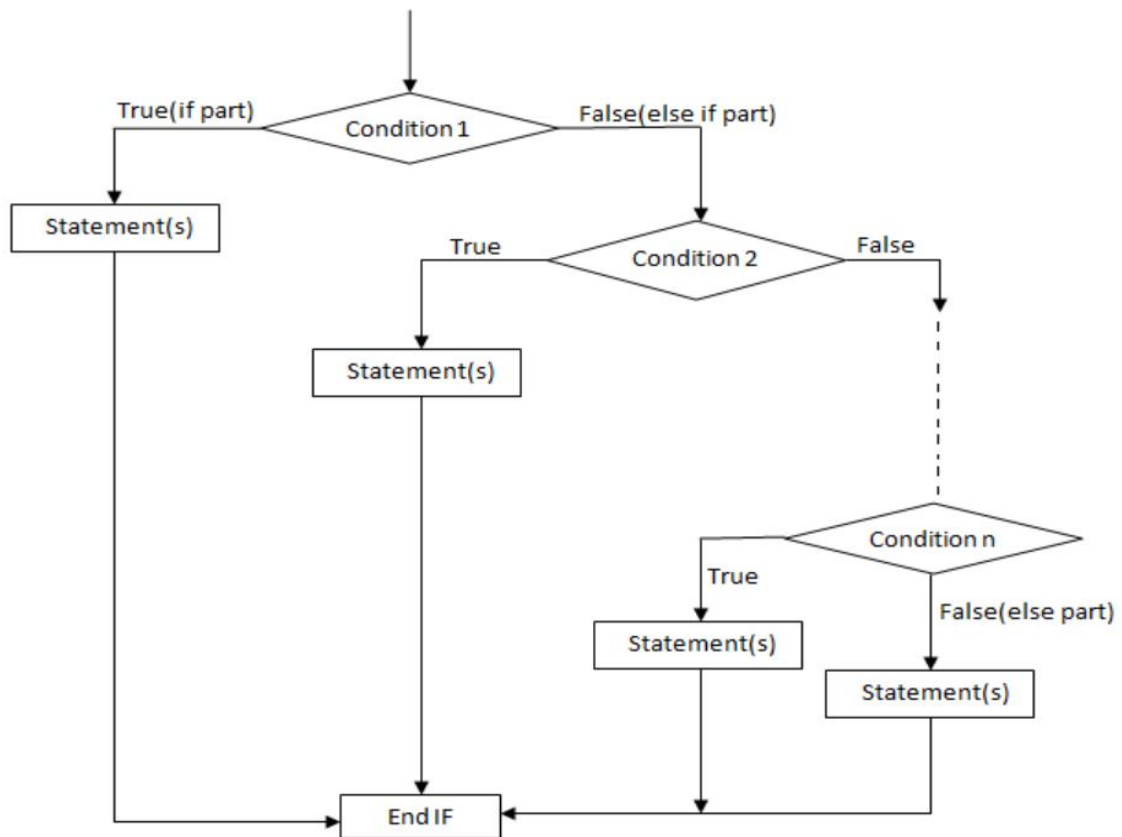
if-else-if Statement in C

if-else-if statement is used when we need to check multiple conditions. In this control structure we have only one “if” and one “else”, however we can have multiple “else if” blocks. This is how it looks:

```
if(condition_1) {
    /*if condition_1 is true execute this*/
    statement(s);}
else if(condition_2) {
    /* execute this if condition_1 is not met and condition_2 is met */
    statement(s);}
else if(condition_3) {
    /* execute this if condition_1 & condition_2 are not met and condition_3 is met */
    statement(s);}...
else {
    /* if none of the condition is true then these statements gets executed */
    statement(s);}
```

Note: The most important point to note here is that in if-else-if, as soon as the condition is met, the corresponding set of statements get executed, rest gets ignored. If none of the condition is met then the statements inside “else” gets executed. **Example of if-else-if**

Flow chart of If else if else statement



Example of if else-if else statement

```

#include <stdio.h>
int main(){
    int num;
    printf("Enter an integer number between 1 & 99999: ");
    scanf("%d", &num); // 1234
    if(num <100 && num>=1) {
        printf("Its a two digit number");
    }
    else if(num <1000 && num>=100) {
        printf("Its a three digit number");
    }
    else if(num <10000 && num>=1000) {
        printf("Its a four digit number");
    }
    else if(num <100000 && num>=10000) {
        printf("Its a five digit number");
    }
    else {
        printf("number is not between 1 & 99999");
    }
    return 0;}
  
```

Output:

Enter an integer number between 1 & 99999: 8976 Its a four digit number