

NOTES

Bachelor of Science (First Year)
CSC-101-Programming fundamentals using C
Title of the Unit : V – Overview of C
Module Name : Control constructs: selection
Module Number : 21

Decision Making within a Program

Decision making is the thoughtful consideration and selection of a course of action from among available alternatives in order to produce a desired result. Most control statements in any computer language, including C, rely upon a conditional test that determines a course of action. The conditional test either evaluates to a true or a false. The concept of evaluating and obtaining a result is referred to as decision making in a programming language. "True" is considered the same as "yes," which is also considered the same as 1. "False" is considered the same as "no," which is considered the same as 0. Decision is a word which is normally taken in a moment where one is in a position to select one option from the available options which are obviously more than one. Decision making structures require that the programmer specify one or more conditions to be evaluated or tested by the program, along with a statement or statements to be executed if the condition is determined to be true, and optionally, other statements to be executed if the condition is determined to be false.

C programming language assumes any nonzero and non null values as true, and if it is either zero or null, then it is assumed as false value. In C language there are various methods which can be used to select an appropriate set of statements depending upon the user's input. A program control statement modifies the order of statement execution. Program control statements can cause other program statements to execute multiple times or not to execute at all, depending on the circumstances. .

FOUR different ways to take decisions which are as follows : **if statement, ifelse statement, the conditional operators and the switch statement.**

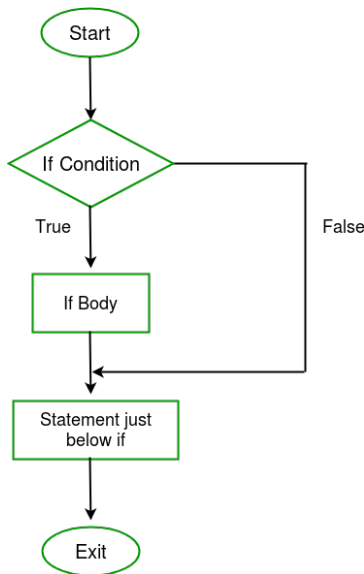
Conditions in C

Conditions may be a combination of **relational and logical operators**. The relational operators are used to compare expressions, asking questions such as, "Is y greater than 100?" or "Is z equal to 0?" An expression containing a relational operator evaluates to either true (1) or false (0). "True" is considered the same as "yes," which is also considered the same as 1. "False" is considered the same as "no," which is considered the same as 0. Logical operators combine two or more relational expressions into a single expression that evaluates to either "True" or "False".

Operator	Meaning
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equal to
!=	Not equal to

Operator	Meaning
&&	Logical AND
	Logical OR
!	Logical NOT

The 'if' statement evaluates an expression and directs program execution depending on the result of the evaluation. If expression evaluates to true, statement is executed. If statement evaluates to false, statement is not executed. In either case, execution then passes to whatever code follows the 'if' statement. The execution of statement depends on the result of expression. The line if (expression) and the line statement; are considered to comprise the complete if statement; they are not separate statements. An 'if' statement can control the execution of multiple statements through the use of a compound statement, or block. A block is a group of two or more statements enclosed in braces. A block can be used anywhere a single statement can be used.



If condition is true, statements in the block are executed followed by next statements after the block. If condition is not true, statements in the block are ignored/skipped and the next statement after the block is executed.

Else statement

An 'if' statement can optionally include an 'else' clause. The else clause is included as shown below:

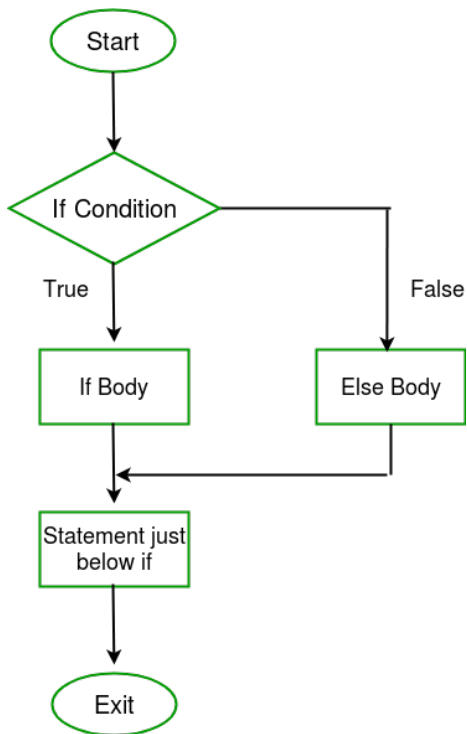
if expression evaluates to true, statements in if block are executed.

If expression evaluates to false, statements in else block are executed.

The combination of the 'if' and else clause is called the 'ifelse' statement. We can have an if statement without an else, but we cannot have an else statement without an if statement.

If else statement

The combination of the 'if' and else clause is called the 'ifelse' statement. This is the most common form of the 'if' statement. If expression is true, 'body of if' is executed; otherwise, 'body of else' is executed. The program then continues to execute the statements just below the if body.



Nested if else statement

We may sometimes need to nest if-else statements, which means we can use one if or else if statement inside another if or else if statement(s).

A nested if in C is an if statement that is the target of another if statement.

