

Conditional Operator in C

Overview

The conditional operator is the one and only ternary operator in the C programming language. It can be used as an alternative for if-else condition if the 'if else' has only one statement each. The conditional operator takes an expression and executes the first statement if the expression evaluates to be true, and the second statement if the expression evaluates to be false.

Scope

- In this article we will learn about the conditional operator in the C programming language, its syntax, its working.
- We will also see when to use the conditional operator over if-else condition.
- Then finally article will answer some of the most frequently asked questions about the conditional operator.

The conditional operator in the C programming language

I believe you would have found yourself in this situation one or more times, where you write an if-else code just to execute a single statement. That is kind of a fuss, right? The conditional operator was designed especially for this problem. The conditional operator can help you make decision-making statements in just a single line, where an if-else would take more than one line.

The conditional operator takes three operands, so it is a ternary operator. As a matter of fact, the conditional operator is the only ternary operator available in the C programming language, so the names ternary operator and conditional operator are used alternatively to mean the conditional operator. The operands can be expressions, statements, constants, or variables. Since they always start with a condition as the first operand it is suitably named as conditional operator.

Syntax of conditional operator in C

```
expression ? statement1 : statement2
```

The expression will be treated as a logical condition, and any non 0 value will be considered as true, and 0 as false. The statement1 and statement2 can be a statement, expression, variable, or a constant. One of the statements will get executed based on the result obtained from the evaluation of the given expression.

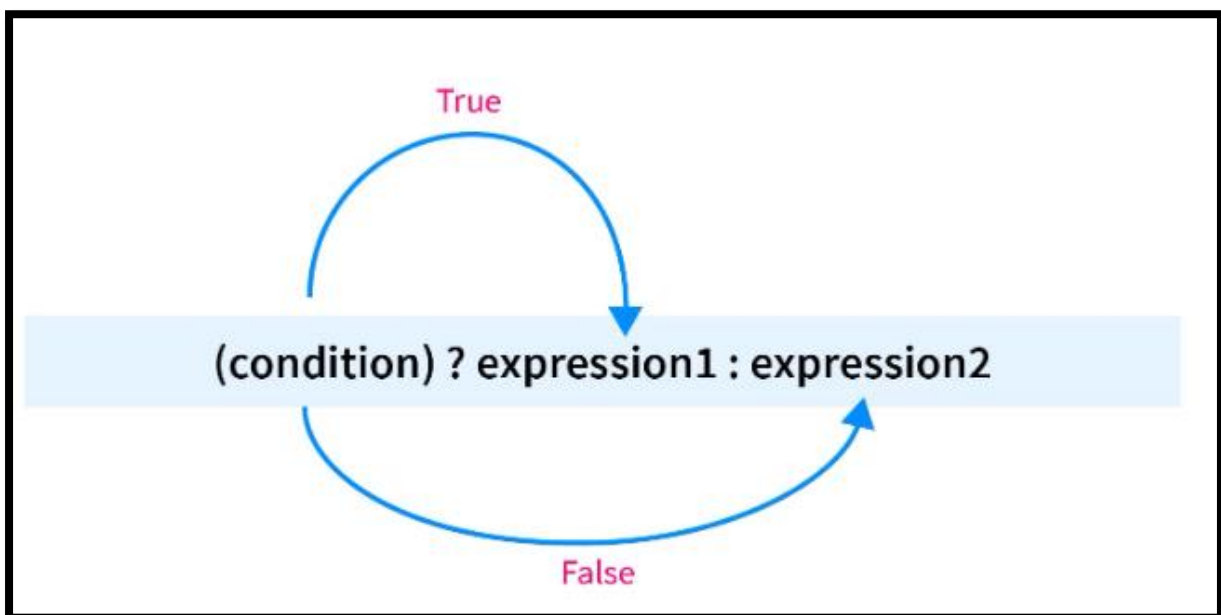
Working of conditional operator in C

The conditional operator of the C programming language works as follows:

- The condition is evaluated first and the result of the condition is implicitly converted to bool.
- If the condition evaluates to be true the first statement -- the statement after the question mark will get executed.
- If the condition evaluates to be false then the second statement -- the statement after the colon gets executed.

The result of the conditional operator is the result of one of the two expressions present in the conditional operator. Only one of the last two operands or expressions gets evaluated, the other expression just gets ignored.

The below diagram will help you remember the working of the Conditional Operator in C much more efficiently.



Another version of the conditional operator in C

The conditional operator is a shorter version of the if-else statement for the cases where the if statement would only have one statement. The conditional operator also has a shorter version of itself. This version of the conditional operator was designed to help with the process of checking a condition and assigning a value to a variable, one of the most commonly used use cases in a programming world.

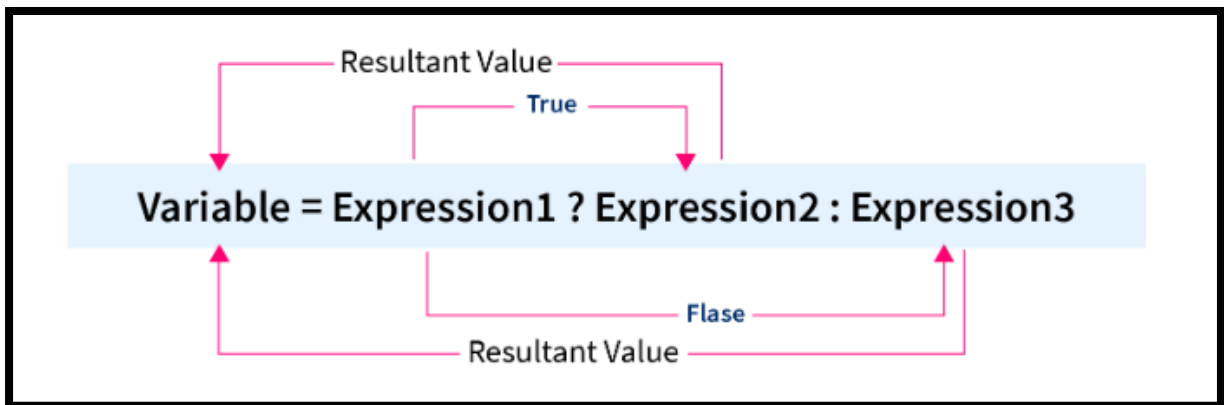
Syntax of the conditional operator in C:

```
variable = condition ? value1 : value2
```

Working of the conditional operator in C:

The working of this version of the conditional operator is more similar to that of the original conditional operator. The compiler will first evaluate the condition and then store value1 to the variable if the condition is evaluated to be true, and the value2 will be assigned to the variable in case if the condition evaluates to be false.

The below diagram will help you in remembering the working of the Conditional Operator in C much more efficiently



Difference between conditional operator in C and if-else statement in C

The if-else statement and the conditional operator of the C programming language are more similar to each other. They perform almost the same operation -- they check a condition and execute a statement based on the result obtained from the evaluation of the condition. The conditional operator was designed so that the user could write code more efficiently. The conditional operator can be used in the case of an if-else statement if the if-else statement only has one statement to execute. The conditional operator reduces the number of lines of code in a program. Some noteworthy differences between the conditional operator and the if-else statement of the C programming language are listed in the below tabular column:

Conditional operator in C

The conditional operator is a single programming statement and can only perform one operation.

The conditional operator can return a value and so can be used for performing assignment operations.

if-else statement in C

The if-else statement is a block statement, you can group multiple statements using a parenthesis.

The if else statement does not return any value and cannot be used for assignment purposes.

Conditional operator in C

The nested ternary operator is complex and hard to debug.

if-else statement in C

The nested if-else statement is easy to read and maintain.

Examples of the Conditional operator in C

All these theories can be exhausting, so let's jump into some coding and see some situations where the use of the conditional operator can be extremely helpful.

print whether the given number is odd or even:

```
#include <stdio.h>

int main() {
    int num;
    scanf("%d", &num);
    (num % 2 == 0)? printf("The given number is even") : printf("The given number is odd");

    return 0;
}
```

Input:

13

Output:

The given number is odd

Input:

6

Output:

The given number is even

Working:

- You will already be familiar with the first three lines of code. They initialize a variable num and read an integer input.
- The conditional operator will first check for the given condition, in the first input i.e., 13 the condition $13 \% 2 == 0$ evaluates to false so the second part of the statement gets executed. So the string "The given number is odd" gets printed as the output.

In the second input i.e., 6 the condition $6 \% 2 == 0$ evaluates to true, so the first statement gets executed and the string "The given number is even" gets printed as the output.

Frequently Asked Questions (FAQ) about the Conditional operator in C

Let's now answer some of the most Frequently asked questions about the conditional operator.

- **Is the conditional operator in C and the ternary operator of the C programming language the same?**

yes, the conditional operator is sometimes referred to as the ternary operator.

- **Why is it called a ternary operator?**

In the C programming language the operators are classified as unary, binary, and ternary based on the number of operands they require. In C programming the conditional operator is the only ternary operator, and hence it is also referred to as the ternary operator.

- **Why is it called a conditional operator in C?**

The first operand for the conditional operator is always a condition, so it is called the conditional operator.

- **Can I use conditional operator in C instead of if statement?**

The conditional operator cannot be replaced in the position of the if statement, cause the conditional operator requires three operands so you have no option but to state the else part to execute.