Introduction to Ternary Operator in C

- In the C language ternary operator is allowing for executing or running any code based on the given value to the condition, and the condition result value of the expression returned to the output.

- The important use of a ternary operator decreases the number of lines of code and increases the performance of the application.

- Most of the research articles claimed that the expression result is faster than a statement result (conventional if-else condition).

- Ternary operator contains 3 expressions; Expression1, Expression2, and Expression3.

- Expression1 holds the condition to check, Expression2 will hold true value evaluated by Expression1 condition, and Expression3 will hold false value evaluated by Expression1 condition.

Advantages:

- It reduces the code.
- Improves performance.
- Overcome conventional use of if and else condition always.

How does ternary operator work in C language?

C language ternary operator works based on the ternary operator(?), If the condition is evaluated true then it executes the true expression value at the left-hand side of the colon(:) symbol and if the condition is evaluated false then it executes false expression value at the right-hand side of the colon(:) symbol.

Syntax:

```
Expression1?Expression2:Expression3;
```

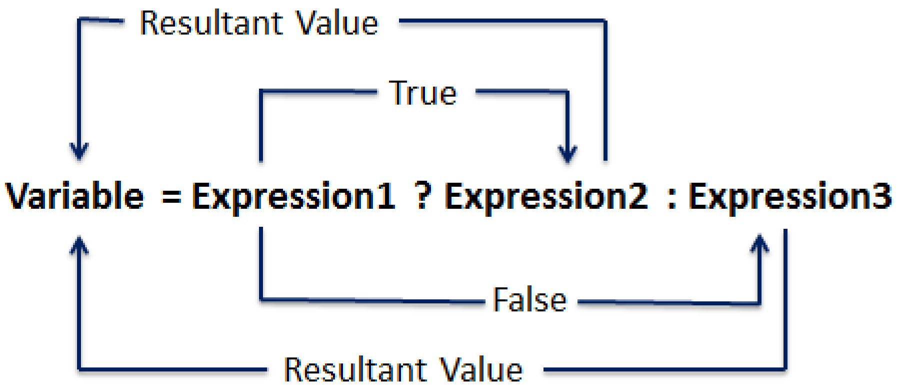
Or

```
Condition?true value:false value;
```

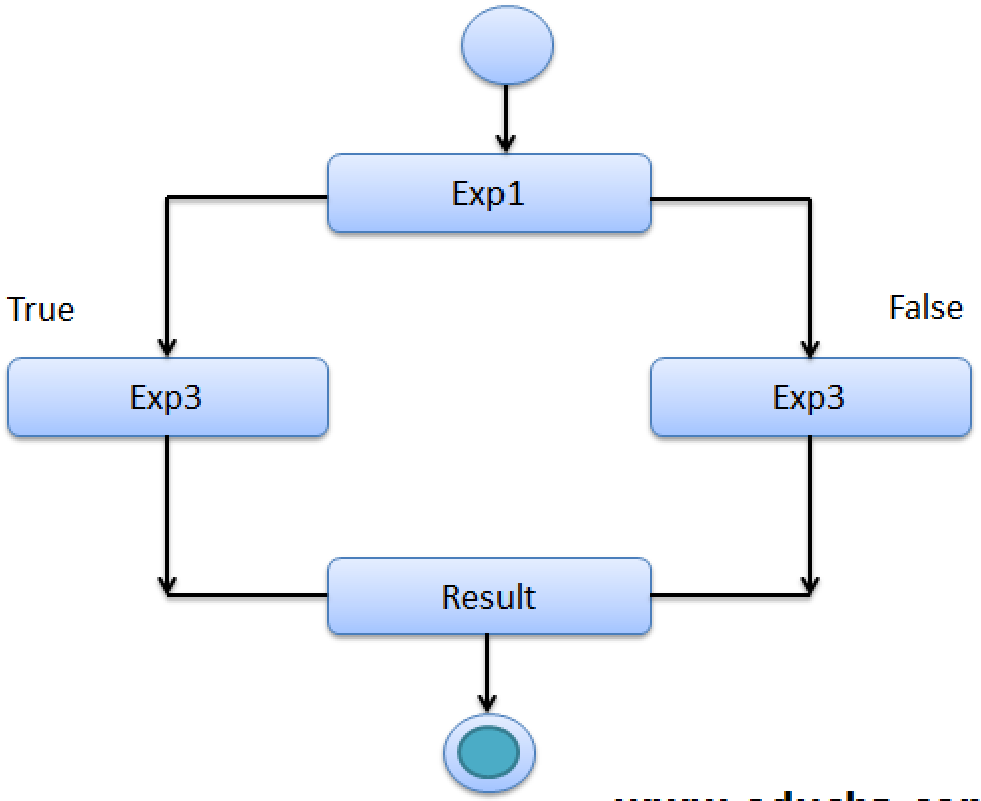
What is the return value of the ternary expression?

```
Boolean result= Condition?true value:false value; it returns the Boolean value(true/false).
```

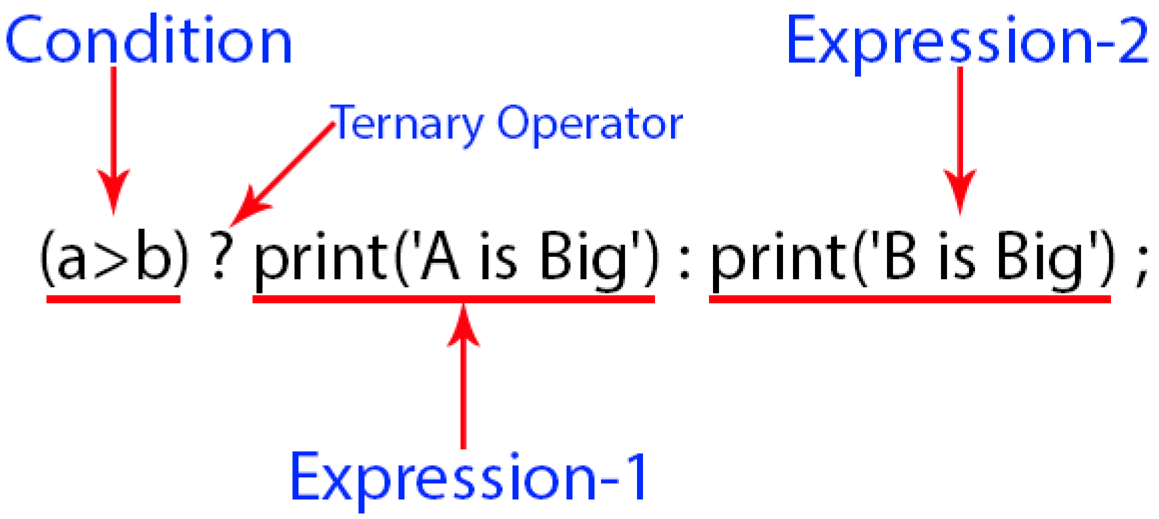
See the below images for better understanding of ternary operator:

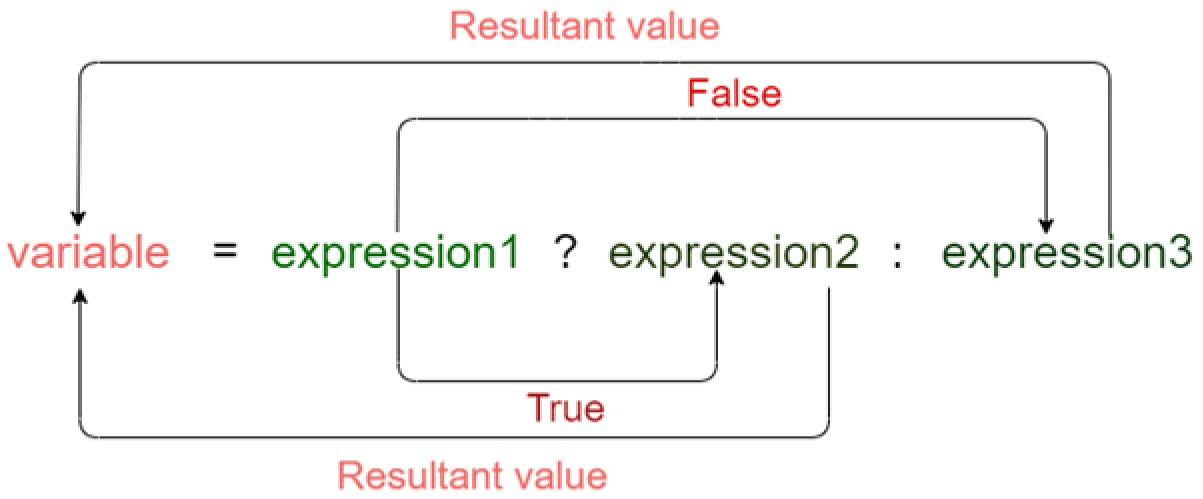


www.educba.com



www.educba.com


Condition
Expression-2
Ternary Operator
(a>b) ? print('A is Big') : print('B is Big') ;
Expression-1


Resultant value
False
variable = expression1 ? expression2 : expression3
True
Resultant value

Simple Ternary operator

a ? b : c => This ternary operator is similar to if-else statement. So it can be expressed in form of if-else statement.

Expression using Ternary operator:

```
a ? b : c
```

Expression using if else statement:

```
if ( a )

    then b execute

else

    c execute
```

Examples of Ternary Operator in C

Here are the following examples mention below

Example #1

```
/ C program to find largest among two
// numbers using ternary operator

#include <stdio.h>

int main()
{
   int m = 5, n = 4;

   (m > n) ? printf("m is greater than n that is %d > %d",
                    m, n)
           : printf("n is greater than m that is %d > %d",
                    n, m);

   return 0;

}
```

(OR)

Example #2

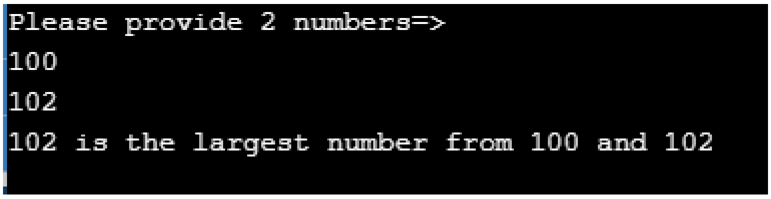
The largest number from 2 numbers with the ternary operator

Code:

```
#include<stdio.h>//line1
main()//line2
```

```
{
int first,second,largest;//line3
printf("Please provide 2 numbers=>\n");//lin4
scanf("%d %d", &first , &second);//line5
largest = (first > second) ? first : second;//line6
printf("%d", largest);//line7
printf(" is the largest number from %d and %d",first,second);
}
```

Output:



```
Please provide 2 numbers=>
100
102
102 is the largest number from 100 and 102
```

Explanation:

- **Line1** includes required library files to run the C language application
- **Line2** is the main method where the application starts from this main () method.
- **Line3** is an integer variable declaration for storing integer numbers (non-decimal numbers).
- **Line4** is the asking user to enter 2 numbers text.
- **Line5** is stored in the entered 2 integer numbers within the scanf method with %d operator.
- **Line6** is the ternary operator compares to 2 numbers which is largest.
- **Line7** printing the output on the console.

Nested Ternary Operator

a ? b: c ? d : e ? f : g ? h : i =>This Nested ternary operator can be broken into if, else and else-if statement. The expression can break into smaller piece in ternary operator and if else statement which are given below:

Expression using ternary operator:

```
a ? b

    : c ? d

    : e ? f

    : g ? h

    : i
```

Expression using if else statement:

```
 if a then b

    else if c then d

    else if e then f

    else if g then h

    else i
```

Convert the given source code using if else statement

```
int a = 2 > 3 ? 2 : 3 > 4 ? 3 : 4;
   printf(a);
```

Code using if else statement

```
   printf("Execute expression using if else statement: ");
    if ( 2 > 3 )
        printf(2);
    else if ( 3 > 4 )
         printf(3);
    else
        printf(4);
    return 0;
```

a ? b ? c : d : e => Below is the expansion of expression using ternary operator and if else statement.

Expression using ternary operator:

```
 a ?

      b ? c

    : d

: e
```

Expression using if else statement:

```
if ( a )

    if ( b )

        c execute

    else

        d execute

else

    e execute
```

C Code

```
int a = 4 > 3 ? 2 > 4 ? 2 : 4 : 3;
    printf(a);

   printf("Execute expression using if else statement: ");
    if ( 4 > 3 )
        if ( 2 > 4 )
            printf(2);
        else
            printf(4);
    else
        print(3)
```

Example #3

Largest numbers from 3 numbers with the ternary operator

Code:

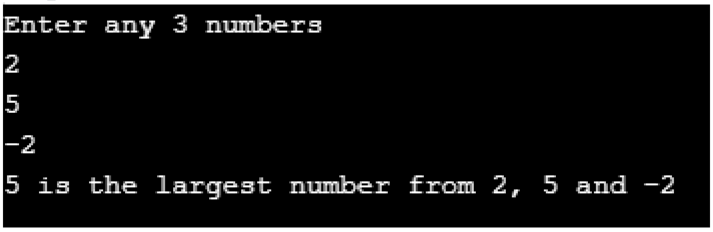
```
#include<stdio.h>//line1
int main()//line2
{
int firstNumber, secondNumber, thirdNumber, largest;//line3
printf("Enter any 3 numbers\n");//line4
scanf("%d %d %d", &firstNumber, &secondNumber, &thirdNumber);//line5

largest = (firstNumber > secondNumber) ? (firstNumber > secondNumber ?
firstNumber : thirdNumber) :  (secondNumber > thirdNumber ? secondNumber :
thirdNumber);//line5
printf ("%d", largest);//line6
printf (" is the largest number from %d, %d and
%d",firstNumber,secondNumber,thirdNumber);//line7
}
```

If condition

Then part

Else part

Output:



```
Enter any 3 numbers
2
5
-2
5 is the largest number from 2, 5 and -2
```

Replace the if else statement in the following code with nested ternary operator

```
int a = 1, b = 2, ans;
if (a == 1) {
    if (b == 2) {
        ans = 3;
    } else {
        ans = 5;
    }
} else {
    ans = 0;
}
printf ("%d\n", ans);
```

```
int a = 1, b = 2, ans;
ans = (a == 1 ? (b == 2 ? 3 : 5) : 0);
printf ("%d\n", ans);
```

ans is 3