National University of computer and emerging sciences

LAB MANUAL 4

CL118-PROGRAMMING FUNDAMENTALS

BSCS-SPRING-2021

SCHOOL OF COMPUTING FAST-NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES, ISLAMABAD CAMPUS

National University of computer and emerging sciences

LAB 04 CONTROL STRUCTURES (IF ELSE and Ternary Operator)

1. If-else statement

The if keyword is used to execute a statement or block, if, and only if, a condition is fulfilled. Its syntax is:

```
if (boolean_expression)
{
    // Executes this block if
    // condition is true
}
else
{
    // Executes this block if
    // condition is false
}
```

Here, the condition (boolean_expression) is an expression that is being evaluated. If this condition is true, statement is executed. If it is false, the statement in else part is executed.
For example, the following code fragment prints the message (x is 100), only if the value stored in the x variable is indeed 100:

```
1 if (x == 100)
2   cout << "x is 100";
```

If x is not exactly 100, this statement is ignored, and nothing is printed.
If you want to include more than a single statement to be executed when the condition is fulfilled,

these statements shall be enclosed in braces {}, forming a block:

```
1 if (x == 100)
2 {
3    cout << "x is ";
4    cout << x;
5 }
```

2. If-else if Statements

In C/C++ **if-else-if ladder** helps user decide from among multiple options. The C/C++ *if* statements are executed from the top down. As soon as one of the conditions controlling the *if* is true, the statement associated with that *if* is executed, and the rest of the *else-if* ladder is bypassed. If none of the conditions is true, then the final *else* statement will be executed. On the other hand if a sequence of *if*s is used instead of *elseif*, the control goes to every *if* part even if some *if*-part has already been executed.

Syntax

```
if (condition)
    statement 1;
else if (condition)
    statement 2;
.
.
else
    statement;
```

3. Conditional or Ternary Operator (?:) in C/C++

The conditional operator is kind of similar to the if-else statement as it does follow the same algorithm as of if-else statement but the conditional operator takes less space and helps to write the if-else statements in the shortest way possible.

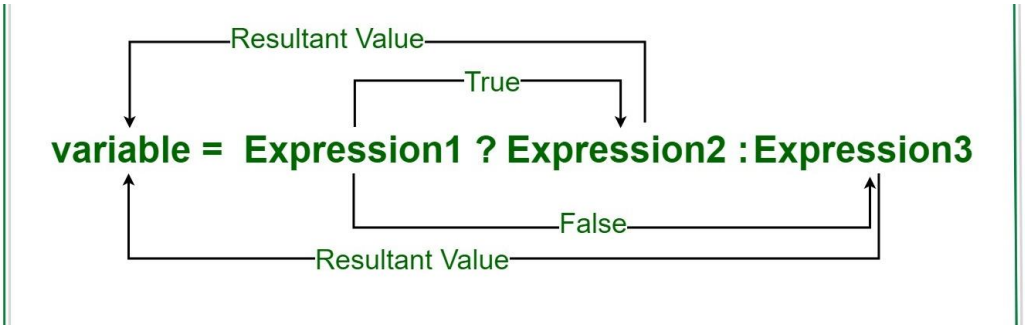

Figure 1-ternary operator in c++ -from geeksforgeeks

Syntax

```
variable = Condition ? Expression2 : Expression3
OR
Condition ? Expression2 : Expression3
```

Visualization into if-else statement

```
if(Expression1)
{
    variable = Expression2;
}
else
{
    variable = Expression3;
}
```

Program 1

```
int n= 20;
char isEven;
isEven= n%2==0 ? ‘y’ : ‘n’;
cout<< “isEven= ”<<isEven<<endl;
```

output:

```
isEven= y
```

Program 2

```
int x= 20;
n%2==0 ? cout<< “n is an even number”: cout<< “n is an odd
number”;
```

output:

```
n is an even number
```

TASKS

Problem 1:
Write a program that reads two integers from user and outputs the largest one using if else.

Problem 2:
Re-write the above program using ternary operator.

Problem 3:
A shop gives discount of 10% of the total purchase to its customers if a customer purchases for more than Rs. 2000.

Ask customer for total purchase value
Calculate and print total cost for user without discount and with discount.

Problem 4:

A university does not allow its student to sit in the exam if his/her attendance is less than 75% in any course/lab.
Write a program to ask a student about
total number of classes scheduled by instructor during the whole semester and
total number of classes he/she attended during the whole semester and print
percentage of his/her attendance. Finally display the student if he/she is allowed to sit in the exam or not.

Problem 5:

Note: use if-elseif statements for this program

Write a program that reads the score of a student in a subject and displays his grades according to the following criteria:

Score	Grade
>= 90	A+
80 – 89	A
70 – 79	B
60 – 69	C
50 – 59	D
< 50	F

Submission Instructions:

1. Save all .cpp files with your roll no and task number e.g. i20XXXX_Task01.cpp
2. Now create a new folder with name ROLLNO_LAB04 e.g. i20XXXX_LAB04
3. Move all of your .cpp files to this newly created directory and compress it into .zip file.
4. Now you have to submit this zipped file on Google Classroom.