School Of Mechanical & Manufacturing Engineering, NUST

Department of Mechanical Engineering

CS-114 - Fundamental of Programing

Lab Manual # 02

Course Instructor: Khawaja Fahad Iqbal

Lab Instructor: Muhammad Affan

Student Name: _____

CMS ID: _____

DATE:

_____

Lab Manual # 02

Relational and Logical Operators

Objectives:

The Objective of this LAB is to understand the use of Relational and Logical Operators in C++ Language. In addition, use these operators and make useful conditions.

Description:

Relational Operators:

There is often a need to compare two values in a program such as whether the value in one variable is greater than the value in the other variable. Depending upon the situation we can perform different operations for different cases.

Relational Operators (also known as Comparison Operators) are used to compare two values.

The value can either be true or false. In C and C++, 1 represents true and 0 represents false. However in C++, the data type has been introduced, which holds only two values either true or false. The value true corresponds to 1 and 0 corresponds to false.

Table 1: List of Relational Operators

Operator	Operator's Name	Example	Result
<	less than	5<10	1(true)
<=	less than or equal to	5<=10	1(true)
>	greater than	5>10	0(false)
>=	greater than or equal to	5>=10	0(false)
==	equal to	5==10	0(false)
!=	not equal to	5!=2	1(true)

Logical Operators

The logical operators are used to combine relational expressions or relational conditions. The expression containing logical operators is called logical expression or logical condition. It is also

called compound condition or compound expression. The output of a logical expression is also in logical form its value is either true or false. In C++, following logical operators are used:

Table 2: List of Logical Operators

AND (&&)	Returns true only if both operand are true.
OR (| |)	Returns true if one of the operand is true.
NOT (!)	Converts false to true and true to false.

1. **The && Operator**

The && operator is known as the logical AND operator. It takes two expressions as operands and creates an expression that is true only when both sub-expressions are true.

Here is an example of an if statement that uses the && operator:

```
if (temperature < 20 && minutes > 12);
cout << "The temperature is in the danger zone.";
```

In the statement above the two relational expressions are combined into a single expression. The cout statement will only be executed if temperature is less than 20 AND minutes is greater than 12. If either relational test is false, the entire expression is false and the cout statement is not executed.

2. **The || Operator**

The || operator is known as the logical OR operator. It takes two expressions as operands and creates an expression that is true when either of the sub-expressions are true.

Here is an example of an if statement that uses the || operator:

```
if (temperature < 20 || temperature > 100);
cout << "The temperature is in the danger zone.";
```

The cout statement will be executed if temperature is less than 20 OR temperature is greater than 100. If either relational test is true, the entire expression is true and the cout statement is executed.

3. **The ! Operator**

The ! operator performs a logical NOT operation. It takes an operand and reverses its truth or falsehood. In other words, if the expression is true, the ! operator returns false, and if the expression is false, it returns true. Here is an if statement using the ! operator:

```
if (!(temperature > 100));
cout<<"You are below the maximum temperature.\n";
```

First, the expression (temperature > 100) is tested to be true or false. Then the ! operator is applied to that value. If the expression (temperature > 100) is true, the ! operator returns false. If it is false, the ! operator returns true.

Table 3: Examples of Logical Operators

Operator	Operator's Name	Example	Result
&&	AND	3>2 && 3>1	1(true)
&&	AND	3>2 && 3<1	0(false)
||	OR	3>2 || 3>1	1(true)
||	OR	3<2 || 3<1	0(false)
!	NOT	!(3==2)	1(true)

Lab Task:

1. Write a program that determines if a person is eligible to vote based on their age (e.g., 18 years or older) using logical operators.
2. Write a program that takes an integer as input and checks if it falls within the range [10, 50] using logical operators.
3. Write a C++ program to compare two integers and find the maximum value.
4. Write a C++ program to calculate the average of three exam scores and determine if it's above a passing grade (e.g., average >= 60).

Home Task:

1. Create a program that takes a student's score as input and assigns a grade based on predefined criteria using logical operators (e.g., A, B, C, D, F).
   A-Grade: 90-100 Marks
   B-Grade: 75-90 Marks
   C-Grade: 60-75 Marks
   D-Grade: 45-60 Marks
   F-Grade: 0-45 Marks
2. Write a program that takes an integer as input and determines if it is both even and divisible by 5.
3. Create a C++ program that checks if a user-provided year is a leap year.
4. Create a C++ program that determines if a student is eligible for a scholarship based on their GPA (must have GPA >= 3.5) and attendance (must have attended at least 80% of classes).
5. Write a program that checks if a given character is a vowel (a, e, i, o, u) or a consonant using logical operators.