Institute of Computer Sciences & Information Technology (ICS&IT)
Faculty of Management and Computer Science (FMCS)
The University of Agriculture, Peshawar

Lab No. 4: Introduction to I/O Statements, Operators and String in C++

Objective:

This lab is about familiarization with cin and cout and string in C++ and with different operators in C++.

C++ User Input

You have already learned that cout is used to output (print) values. Now we will use cin to get user input.

cin is a predefined variable that reads data from the keyboard with the extraction operator (>>).

In the following example, the user can input a number, which is stored in the variable x. Then we print the value of x:

Example

```
int x;
cout << "Type a number: ";  // Type a number and press enter
cin >> x;                   // Get user input from the keyboard
cout << "Your number is: " << x; // Display the input value
```

String Types

The string type is used to store a sequence of characters (text). This is not a built-in type, but it behaves like one in its most basic usage. String values must be surrounded by double quotes:

Example

```
string greeting = "Hello";
cout << greeting;
```

To use strings, you must include an additional header file in the source code, <string> library:

Example

```
// Include the string library
#include <string>

// Create a string variable
string greeting = "Hello";

// Output string value
cout << greeting;
```

Boolean Types

A boolean data type is declared with the bool keyword and can only take the values true or false.

When the value is returned, true = 1 and false = 0.

Example

```
bool isCodingFun = true;
bool isFishTasty = false;
cout << isCodingFun;  // Outputs 1 (true)
cout << isFishTasty;  // Outputs 0 (false)
```

Arithmetic Operators

Arithmetic operators are used to perform common mathematical operations.

Operator	Name	Description	Example
+	Addition	Adds two values	a + b
-	Subtraction	Subtracts one value from another	a - b
*	Multiplication	Multiplies two values	a * b
/	Division	Divides one value by another	a / b
%	Modulus	Returns the division remainder	a % b

Assignment Operators

Assignment operators are used to assign values to variables.

In the example below, we use the assignment operator (=) to assign the value **10** to a variable called **x**:

Example

```
int x = 10;
```

Comparison Operators

Comparison operators are used to compare two values.

Note: The return value of a comparison is either true (1) or false (0).

In the following example, we use the **greater than** operator (>) to find out if 5 is greater than 3:

Example

```
int x = 5;
int y = 3;
cout << (x > y); // returns 1 (true) because 5 is greater than 3
```

A list of all comparison operators:

Operator	Description	Example
=	Simple assignment operator, Assigns values from right side operands to left side operand	C = A + B will assign value of A + B into C
+=	Add AND assignment operator, It adds right operand to the left operand and assign the result to left operand	C += A is equivalent to C = C + A
-=	Subtract AND assignment operator, It subtracts right operand from the left operand and assign the result to left operand	C -= A is equivalent to C = C - A
*=	Multiply AND assignment operator, It multiplies right operand with the left operand and assign the result to left operand	C *= A is equivalent to C = C * A
/=	Divide AND assignment operator, It divides left operand with the right operand and assign the result to left operand	C /= A is equivalent to C = C / A
%=	Modulus AND assignment operator, It takes modulus using two operands and assign the result to left operand	C %= A is equivalent to C = C % A

Compound assignment (+=, -=, *=, /=, %=, >>=, <<=, &=, ^=, |=)

Compound assignment operators modify the current value of a variable by performing an

operation on it. They are equivalent to assigning the result of an operation to the first operand:

expression	equivalent to...
y += x;	y = y + x;
x -= 5;	x = x - 5;
x /= y;	x = x / y;
price *= units + 1;	price = price * (units+1);

and the same for all other compound assignment operators. For example:

```
// compound assignment operators
#include <iostream>
using namespace std;

int main ()
{
  int a, b=3;
  a = b;
  a+=2;                    // equivalent to a=a+2
  cout << a;
}
```

Expressions

Multiplication, mode and division have higher precedence than addition and subtraction.

Associativity: left to right

C++ String

Strings are used for storing text

A string variable contains a collection of characters surrounded by double quotes

Example: Create a variable of type string and assign it a value:

string greeting = "Hello";

String Concatenation: The + operator can be used between strings to add them together to make a new string. This is called concatenation

String Length: To get the length of a string, use the length() or size() function

Access Strings: You can access the characters in a string by referring to its index number inside square brackets [ ]

Note: String indexes start with 0: [0] is the first character. [1] is the second character, etc.

User Input Strings: Use the extraction operator >> on cin to display a string entered by a user

TASKS:

1. Take two integers as input from user apply athematic assignment operations on them and print them on screen.

2. Write a code in C++ that takes values of a and b from the user and displays result of polynomial $a^2 + 2ab + b^2$.

3. Write a C++ program that takes two complex number from user and add them. Print the resultant complex number.

4. Write a C++ program to prompt the user to input 3 integer values and print these values in forward and reversed order.

5. Write a C++ program that take an integer value and character from user and display the integer and character on the console window.

6. Write a program that asks the user to enter a value for x and then displays value of following polynomial $2x^5 + 3x^4 - x^3 - 2x^2 + 7x - 6$. To calculate x^5 you will have to use pow (x, 5).

Note: include math.h library for pow.

7. Write a C++ program that take seconds from user and displays time in hours minutes and seconds format. For example, if user enters 3700, output should be 1 hour 1 minute and 40 seconds.

8. Write a C++ program to swap two variables values with and without using third variables.

9. Write a C++program to enter a string s1 and copy it to another string s2.

10. Write a C++ program to calculate the distance between the two points. Note: x_1, y_1, x_2, y_2 are all double values. Formula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

11. Complete the following table by writing the value of each expression in the Value column according to C++ language rules.

Expression	Value
28 / 4 – 2	
6 + 12 * 2 – 8	
4 + 8 * 2	
6 + 17 % 3 – 2	
2 + 22 * (9 - 7)	
(8 + 7) * 2	
(16 + 7) % 2 – 1	
12 / (10 - 6)	
(19 - 3) * (2 + 2) / 4	

12. Write a C++ program to find quotient and remainder of two integers, by taking input from the user.

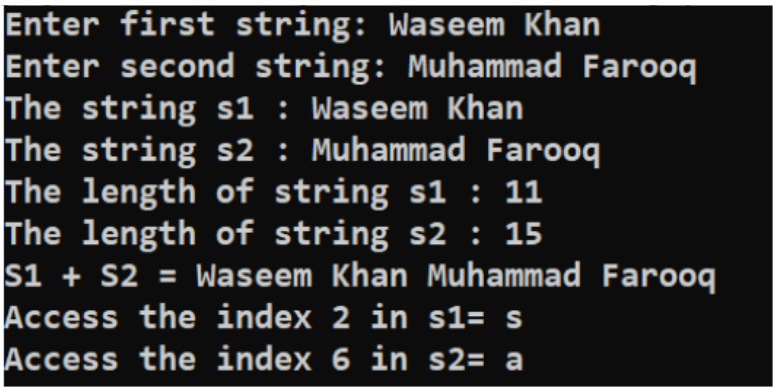
13. Write a C++ program which takes a character and display its previous character using decrement operator.

14. Write a program which asks for a number input from the keyboard. Add 34 to the number and then divide it by 2. Now, take modulus of that number with 4, then again add 4 to the number and then multiply the resultant value by 5. Display the result.

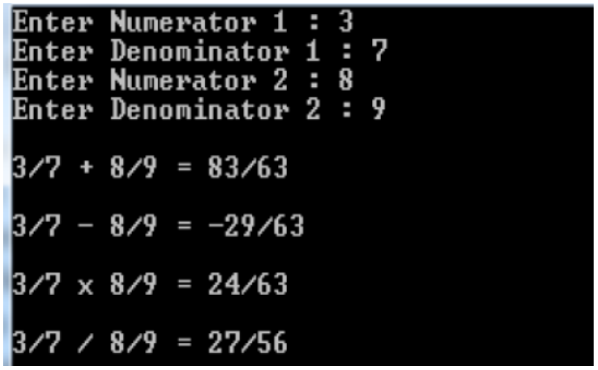
15. Write a program in C++ to compute the specified expressions and print the output.

(25 * 3.5 - 3 * 7.5) + 3 / (40.5 - 4.5)

16. Write a C++ program to take two strings s1 and s2 from user and display the following output.



17. Write a program that takes as input from the user the numerators and denominators of two fractions & gives the sum, difference, multiple and division of the two fractions. Assume that the numerators and denominators are integers.



18. Write a program that takes, as input from the user, the radius of a sphere, and outputs:

- The diameter of the sphere
- The surface area of the sphere
- The volume of the sphere