

Lecture Summary

Anatomy of your first C++ program

Recall the following simple program from first week lectures that demonstrates the basic structure of a C++ program, including libraries, defining the main function, and using the `cout` object for output.

```
#include <iostream>

int main()
{
    std::cout << "Hello, World!" << std::endl;
    return 0;
}
```

Now, let's break down the different parts of this program to make it understandable:

- **#include <iostream>**: This line is a preprocessor directive that tells the compiler to include the `iostream` library. The `iostream` library provides functionality for input and output operations, like printing text to the screen.
- **int main(){ ... }**: This is the main function of your program. Every C++ program must have a main function. The program's execution starts from here.
- **std::cout << "Hello, World!" << std::endl;**: This line uses the `cout` object from the `iostream` library to print text to the console. `std::cout` is used to output data to the console, and `<<` is the insertion operator that "inserts" the text or value on its right into the output stream. "Hello, World!" is the text you want to print. `std::endl` is used to insert a newline character and flush the output buffer.
- **return 0;**: This line indicates the end of the `main` function and the successful termination of the program. The value 0 is returned to the operating system, signifying that the program ran without errors.

When you compile and run this program, you'll see the output: `Hello, World!`

Declaring a Variable

In C++, a variable is a named location in memory that stores a value. Before using a variable, you need to declare it, which involves specifying the variable's data type and a name. Here's the general syntax:

```
data_type variable_name;
```

For example, to declare an integer variable named `age`:

```
int age;
```

Assigning a Value to a Variable

After declaring a variable, you can assign a value to it using the assignment operator `=`. The value you assign should match the variable's data type. Here's an example:

```
age = 25;
```

Output using cout

`cin` and `cout` are part of the C++ Standard Library and are used for input and output operations. `cin` is used for input (reading values from the user), and `cout` is used for output (displaying values to the user).

Output using cout: To display output to the user, you use `cout` along with the `<<` operator. You can output variables, constants, and text. Here's an example:

```
cout << "Your age is: " << age << endl;
```

Using Escape Character

Indicates "special" character output listed in the table below.

Escape Sequence	Description
<code>\n</code>	Newline. Position the screen cursor to the beginning of the next line.
<code>\t</code>	Horizontal tab. Move the screen cursor to the next tab stop.
<code>\r</code>	Carriage return. Position the screen cursor to the beginning of the current line; do not advance to the next line.
<code>\a</code>	Alert. Sound the system bell.
<code>\\</code>	Backslash. Used to print a backslash character.
<code>\"</code>	Double quote. Used to print a double quote character.

Carriage Return (\r)

```
#include <iostream>

using namespace std;

int main() {
    // Define the headers
    cout << "Hello World\r";
    cout << "Whole \n";
    return 0;
}
```

Lab Questions

1. Write a program that prints your household grocery list consisting of as many items as you need. Use `\n` to print each item of the list on a new line.

```
Household Grocery List:
```

```
1. Milk
2. Eggs
3. Bread
4. Rice
5. Apples
6. Chicken
7. Vegetables
8. Coffee
```

2. Write a program that creates and displays a table with four columns: Name, Age, Gender, and Height. The program should use `\t` for proper alignment and `\n` to move between rows. You need to display at least three entries in the table.

Name	Age	Gender	Height
John Doe	18	M	5.9
Jane Smith	20	F	5.6
Emily Davis	22	F	5.7

3. Write a program that creates and displays a table consisting of a list of students and their ITP scores in a tabular format. The program should use `\t` for proper alignment and `\n` to move between rows. You need to display at least three entries in the table. Note: Formatting of the code is all up to you.

Student Name	ITP Score
Alice Johnson	85
Bob Smith	90
Charlie Brown	78

4. Write a program that initializes an integer variable with a user's age and then prints a message that includes this age. The message should be of the format: "You are [Age] years old."

Requirements:

- Declare an int variable and initialize it with the user's age.
 - Use cout to print a message in the format: "You are [Age] years old."
5. Write a C++ program that initializes two integer variables representing the length and width of a rectangle. The program should then calculate the area of the rectangle and print a message in the format: "The area of the rectangle is [Area] square units."

6. Write a program that prints hello world and goodbye world. However, the word "Goodbye" should overwrite the word "Hello" when the output is displayed. You must use the carriage return (\r) escape character to achieve this.
7. Write a program that demonstrates the use of \r (carriage return) by first printing "Typing..." and then overwriting it with any statement (like "Welcome to IBA") on the same line.

Output Workflow:

- Print "Typing..."
 - Use \r to move the cursor to the beginning of the line
 - Print "Welcome to IBA" to overwrite "Typing..."
 - Your final output should be "Welcome to IBA"
8. Write a C++ program that displays a basic progress indicator using \r. The progress indicator should move from left to right and use \t or spaces to format the loading of the bar. The bar should gradually fill up using symbols (like #) as the progress increases.
 - The program should first print "Downloading..."
 - Use \r to move the cursor to the beginning of the line.
 - Then the following progress indicators should be printed

```
Progress [##      ] 20%
Progress [####    ] 40%
Progress [#####  ] 60%
Progress [#####  ] 80%
Progress [#####  ] 100%
Successfully downloaded
```