Your First C++ Program
Today

- In today’s lecture we will
  - Write our first C++ program
  - Analyze the different components of C++ programs
Problem

• Programs are written to solve problems

• Imagine that you have been asked to solve the following problem

  o Your summer surveying job requires you to study some maps that give the distance in kilometers and some that use miles. You and your co-workers prefer to deal in metric measurements. Write a program that performs the necessary conversion
Your First C++ Program

```cpp
//***********************************************************
// File name: hello.cpp
// Author:    Shereen Khoja
// Date:      08/23/2006
// Purpose:   This program displays a welcome message to
//             the user
//***********************************************************

#include <iostream>
#include "stdafx.h"

using namespace std;

int main()
{
  string name;
  cout << "Type your name, then press enter" << endl;
  cin >> name;
  cout << "Hello " << name << "!" << endl;
  return 0;
}
```

Program Output
Type your name, then press enter
Chadd
Hello Chadd!
Building an Application

Source Code

```
#include <hdr>
int main()
{
    ...
}
```

Preprocessor

Modified Source Code

```
hdr
int foo()
{
    ...
}
int main()
{
    ...
}
```

Object Code

```
foo: 01010
main:10111
```

Compiler

```
cl, gcc, cc, g++
```

Linker

```
link, ld
```

Executable Code

```
app.exe
```

1010

1101
Language Elements

- Key Words
  - Have special meaning in C++
    - using, namespace, int

- Programmer-Defined Identifiers
  - Names made up by the programmer
    - name

- Operators
  - Perform operations
    - *, =

- Punctuation
  - Used to mark the beginning and end of the program
Syntax

- Rules that must be followed when constructing a program
- Controls the use of key words, programmer-defined identifiers, operators, and punctuation
Variables

- Names storage location in the computer's memory
- Holds data
- The data can change
Program Components

- The C++ program on the previous slide consists of the following elements:
  - Comments
  - Preprocessor directives
  - Standard namespace
  - `main` function
  - Declaration statements
  - Executable statements
Comments

• Comments are
  - How you explain in English what the different parts of your program do
  - Ignored by the compiler
  - Very important

• The editor in Visual Studio will colour code your comments. They will be green
Comments

- There are two ways to write comments
  - // I am a comment
    - Anything after // till the end of the line will be a comment
  - /* I am another comment */
    - You must start the comment with /* and end it with */ in this style of comment
Preprocessor directives

- `#include <iostream>`
- `#` signifies preprocessor directive
- Processed before program translation
- `#include` tells the preprocessor to look for libraries
- `<>` signifies part of standard C++ libraries
- We’ll see other examples of preprocessor directives later
Preprocessor directives

- `iostream` is the input/output stream library
- It is needed to output data to the screen and read in data from the keyboard
- `#include` takes the contents of the library file and places them in the current program
Namespace std

- using namespace std;

  Indicates that we will be using objects (cout & cin) that are named in a region called std.

  The statement ends in a semicolon.

  The statement appears in all our programs.
main Function

```c
int main()
{
    // program statements
    return 0;
}
```

- Every program must have a main function
- It is where the start of your program execution begins
- `return 0;` ends the main function and indicates that the program terminated successfully
- Everything within the double braces `{}` should be indented
Program Statements

- There are two types of statements that you can write inside the main (or any other) function
  - Declaration statements
    - Specify the data that is needed by the program
  - Executable statements
    - Perform operations

- All statements must end with a semicolon;
Program Statements

- Declaration statements
  - No declaration statements in our program

- Executable statements

```cpp
cout << "Type your name, then press enter" << endl;

cin >> name;

cout << "Hello " << name << "!" << endl;
```
Program Skeleton

- All programs in C++ should have the following skeleton

```cpp
/***********************************************************
// File name: filename.cpp
// Author:    Your Name
// Date:      09/01/2004
// Purpose:   Description about what the program does
/***********************************************************
#include <iostream>
using namespace std;

int main()
{
    // declaration statements

    // executable statements

    return 0;
}
```
Summary

• Today we
  o Wrote our first C++ program
  o Introduced the basic components of a C++ program

• To see the program in action you should test it in Visual Studio .NET

• We covered p. 12 - 33 from your textbook