

Name: _____

Name: _____

1 Introduction

Welcome to CSI 201! In this lab you will first write and run a simple C++ program, and then explore what the different parts of it do.

- It's okay to be confused! This lab will ask you to make guesses about things you probably haven't seen before. Just give it your best shot, and feel free to ask me questions if you need to.
- **Work in pairs.** We'll do this a lot in this class. The two of you should share a laptop, and share your ideas too.
- If you have prior programming experience, try to work with someone who has also programmed before, so that you don't short-circuit your partner's chance to ask questions.
- There isn't anything to turn in for this lab. We will talk about each of the questions towards the end of class.

2 Your First Program

In this section, you will learn how to create a new C++ project, put code in it, and run it. You may need to refer back to these instructions for future exercises and assignments until it is second nature.

Creating a new Visual Studio project.

- Open Microsoft Visual Studio 2012 Express: `Start` → `All Programs` → `Microsoft Visual Studio Express 2012` → `VS Express for Desktop`
- Make a new project: `File` → `New` → `Project`
 - Select `Visual C++` → `Win32` → `Win32 Console Application`
 - Choose the project name. (Let's use `first_program` for today. Note where the project is saved.)
 - Click `OK`.
 - In the wizard, click on `Application Settings`. Check the box next to `Empty project`.
 - Click `Finish`.
- Add a new source code file:
 - Click on `Project` → `Add New Item`
 - Select `Visual C++` → `Code` → `C++ File (.cpp)`
 - Choose the name of the new file. For this lab, use `first_program.cpp`
 - Click `Add`.
- Type in the following very simple program. For future labs, use this program as a starting template.

```
#include <iostream>
using namespace std;

// Every program must have a main
void main() {
    cout << "Hello!" << endl;
}
```

- Be sure to save your code.
- Run the program: **Debug** → **Start Without Debugging** (or, press Ctrl+F5). If a “This program is out of date” dialog pops up, click **Yes** to rebuild, then run.

Exercise 1 *Follow the instructions: create a new project called `first_program`, type in the code above, and run it. If errors appear, figure out what you typed wrong and fix it.*

What happens when you run your code?

Answer:

Comment 1 *All of the laptops in the classroom dual-boot into both Mac OS X and Windows, but Visual Studio only runs in Windows. If your computer boots into OS X, switch operating systems by restarting the computer. Hold down the ‘Option’ key when you see the grey loading screen. Let go when you see the startup icons appear and choose Windows.*

3 Understanding the Program

In this section you’ll do experiments to figure out what the different parts of the program are. Feel free to make changes to your program to see what happens. Don’t worry about seeing errors appear—you won’t break your computer.

Exercise 2 *Some of the lines of our program end with a semicolon. This will be important in C++.*

Does removing semicolons change any?

Answer:

Does adding semicolons change anything? To the end of lines? In the middle of lines?

Answer:

Comment 2 *In general, never put a semicolon at the end of a line that starts with a '#'. We'll talk about why later.*

Exercise 3

One of the lines of our program is a comment. Comments never, ever affect what the program does.

Which line is the comment? Why do you think so? Try to add another comment. Did that work? Do comments have to be on their own line?

Answer:

Exercise 4 *One of the lines of code prints a word to console output. Which line is it? How can you change the program to also print some other words as well?*

Answer:

Comment 3 *In this class, 'print' doesn't mean to make a paper copy of a document. In computer science, 'print' means to write something to the console, which is the black screen with white letters that appears when you run your program.*

Exercise 5 *Every program must have one function called main. Let's try see what happens if we have two. Copy and paste so that your code looks like this:*

```
#include <iostream>
using namespace std;

// Every program must have a main
void main() {
    cout << "Hello!" << endl;
}
void main() {
    cout << "Hello!" << endl;
}
```

Does this cause a problem? What happens?

Answer:

Exercise 6 *Is the first line important? What happens if you remove it? Take a guess at what it does.*

Answer: