**CSCI 1411 – Lab 02 – C++ Variables and Input/Output**

**Goals:**

* Use of variables of type int, double, float, char, string
* Use of operators +, -, /, \*
* Understand integer division
* Understand Modulus %

**Development Environment:** csegrid (Centos), g++ (all students must use csegrid)

**Skills:** declare variables, use operators

Reading: Chap 2 and Chap 3

**Deliverables:** 1) This lab with 3 screen shots, 2) lastnameFirstLab02.cpp

**Part I – Skills Practice (5 points)**

* Use Putty to log in to the csegrid.ucdenver.pvt. Note: If off-campus you would need to first click into Cisco Any Connect. Make a directory called **lab02** (remember it is case sensitive). Change directory into lab02. Create a file called **lastnameFirstLab02.cpp** using nano (use your last name and your first initial)

**mkdir lab02  
cd lab02  
nano augustinetLab02.cpp**

* Place the following text in the file (Do not cut and paste. You will learn more by typing it in. If you do cut and paste all of the single and double quotes will be incorrect)

**#include <iostream> //needed for cin and cout**

**#include <string> //needed to use string**

**using namespace std; //needed for string and cin and cout**

**int main( )**

**{**

**double celsius, fahren;**

**int count;**

**string firstName, lastName; //use descriptive variable names**

**cout << "What is your first name?" << endl;**

**cin >> firstName;**

**cout << "What is your last name?" << endl;**

**cin >> lastName;**

**cout << "What is the fahrenheit temperature?\n";**

**cin >> fahren;**

**celsius = (fahren - 32) \* (5.0 / 9.0);**

**//make sure to use parenthesis**

**/\*remember and int divided by int is an int**

**so you have to make it into a double with 5.0\*/**

**cout << firstName << " " <<lastName**

**<<", the Celsius temperature is "**

**<< celsius << endl;**

**return 0; //that is a zero not an Oh**

**}**

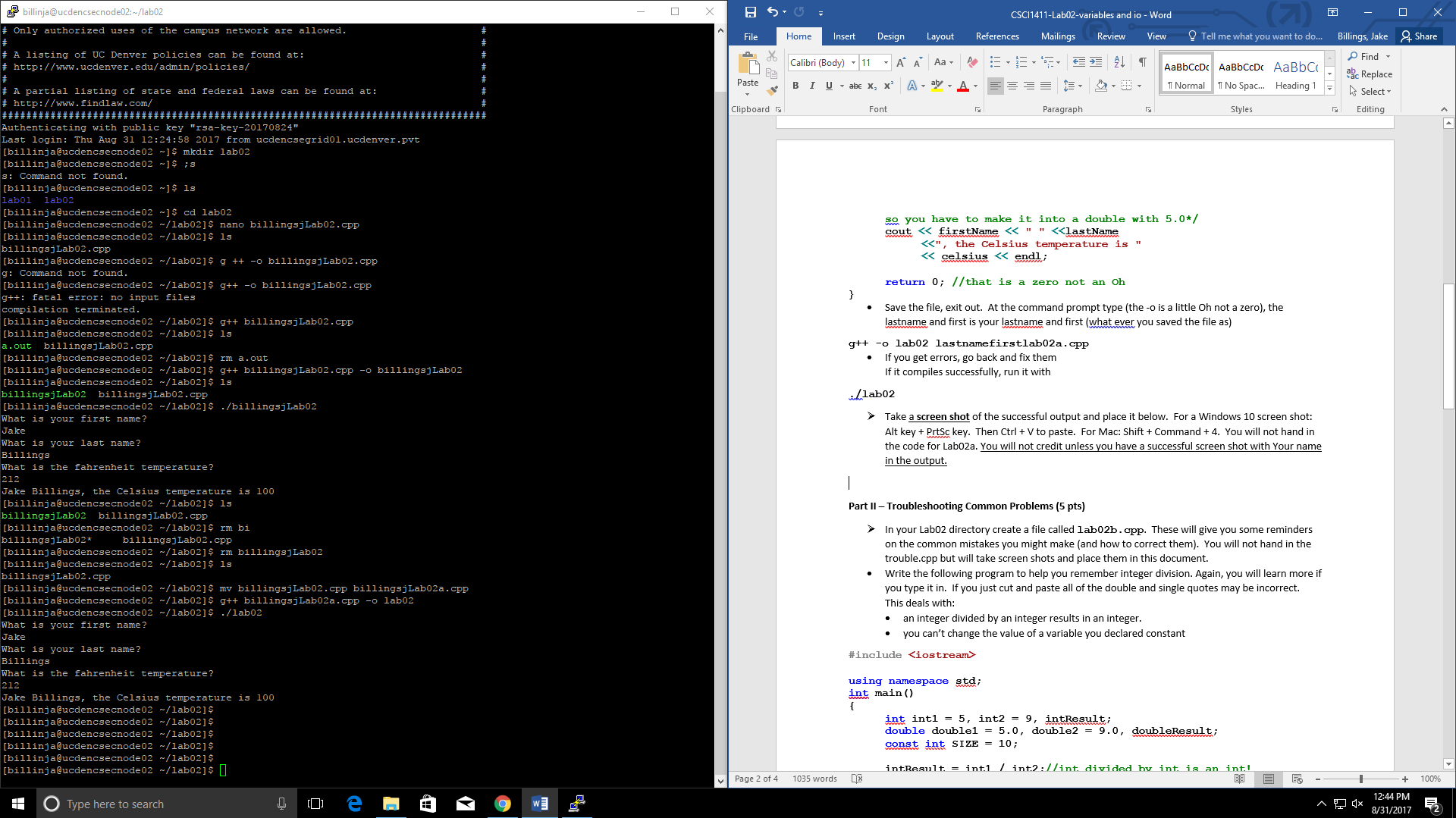
* Save the file, exit out. At the command prompt type (the -o is a little Oh not a zero), the lastname and first is your lastname and first (what ever you saved the file as)

**g++ -o lab02 lastnamefirstlab02a.cpp**

* If you get errors, go back and fix them  
  If it compiles successfully, run it with

**./lab02**

* Take a **screen shot** of the successful output and place it below. For a Windows 10 screen shot: Alt key + PrtSc key. Then Ctrl + V to paste. For Mac: Shift + Command + 4. You will not hand in the code for Lab02a. You will not credit unless you have a successful screen shot with Your name in the output.



**Part II – Troubleshooting Common Problems (5 pts)**

* In your Lab02 directory create a file called **lab02b.cpp**. These will give you some reminders on the common mistakes you might make (and how to correct them). You will not hand in the trouble.cpp but will take screen shots and place them in this document.
* Write the following program to help you remember integer division. Again, you will learn more if you type it in. If you just cut and paste all of the double and single quotes may be incorrect. This deals with:
* an integer divided by an integer results in an integer.
* you can’t change the value of a variable you declared constant

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**int int1 = 5, int2 = 9, intResult;**

**double double1 = 5.0, double2 = 9.0, doubleResult;**

**const int SIZE = 10;**

**intResult = int1 / int2;//int divided by int is an int!**

**cout << int1 << "/" << int2 << "=" << intResult;**

**doubleResult = double1 / double2;//double divided by a double is a double**

**cout << double1 << "/" << double2 << "=" << doubleResult;**

**cout << "Division with one double and one int is a double\n";**

**cout << "5/9.0 =" << 5 / 9.0 << endl;**

**cout << "5.0/9 =" << 5.0 / 9 << endl;**

**cout << "Convert an int to a double" << double(int1);**

**cout << "You can increment an int or double with ++:" << endl;**

**int1++;**

**cout << int1++ << endl;**

**cout << "But you can not change the value of a const";**

**//take a screen shot of the error below then remove the statement**

**SIZE++;**

**return 0;**

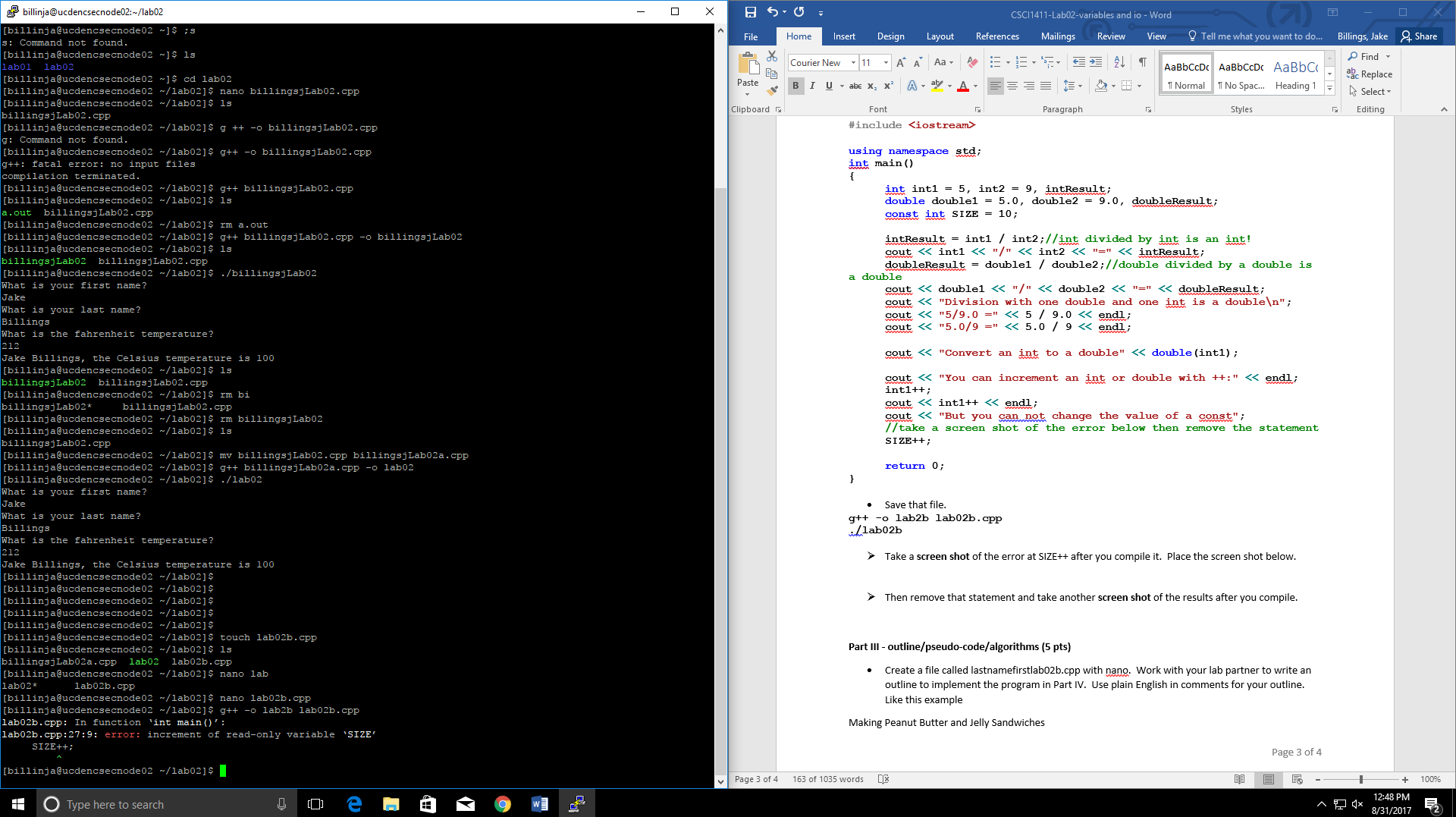
**}**

* Save that file.

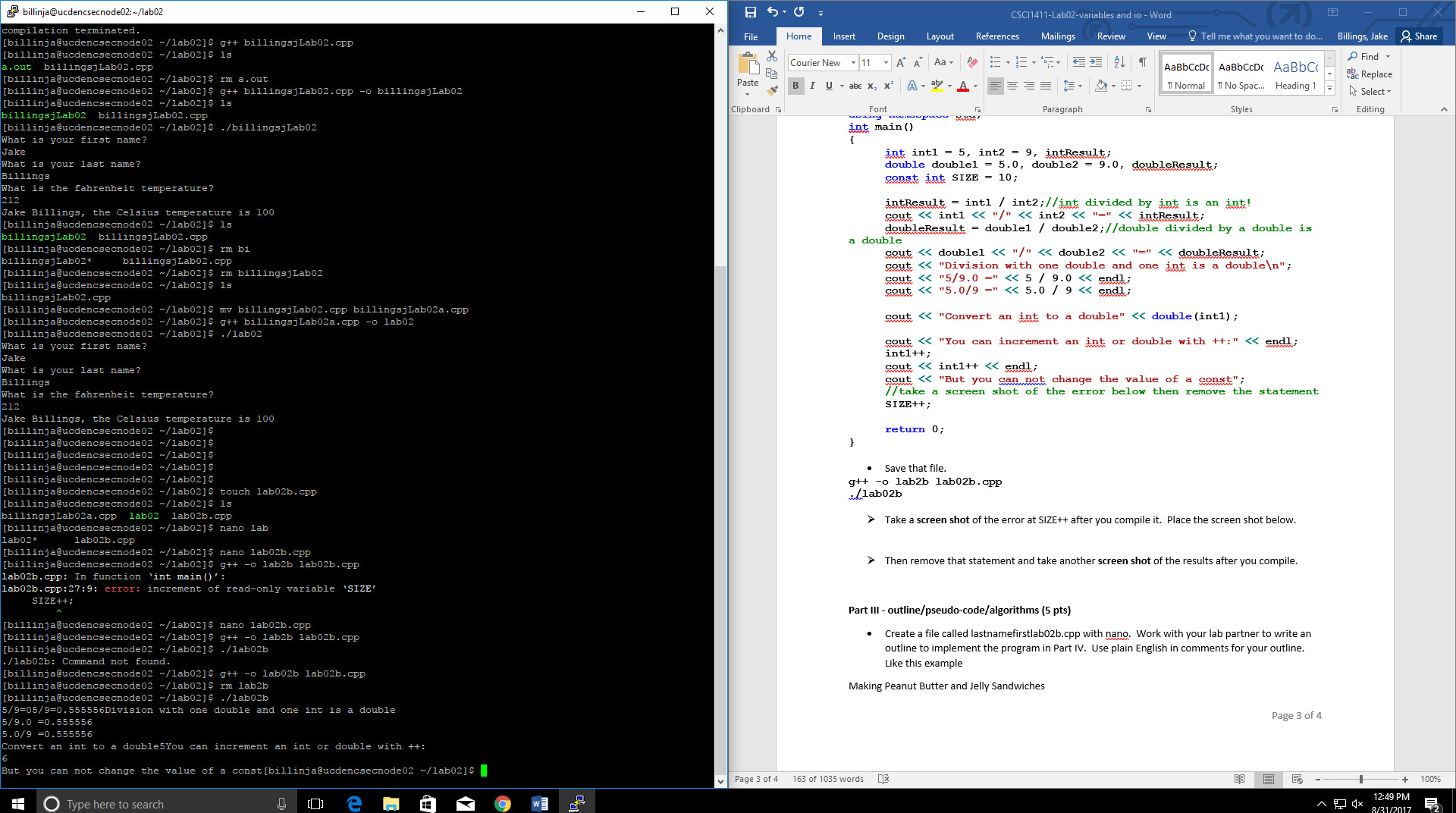
**g++ -o lab2b lab02b.cpp**

**./lab02b**

* Take a **screen shot** of the error at SIZE++ after you compile it. Place the screen shot below.



* Then remove that statement and take another **screen shot** of the results after you compile.

****

**Part III - outline/pseudo-code/algorithms (5 pts)**

* Create a file called lastnamefirstlab02b.cpp with nano. Work with your lab partner to write an outline to implement the program in Part IV. Use plain English in comments for your outline. Like this example

Making Peanut Butter and Jelly Sandwiches

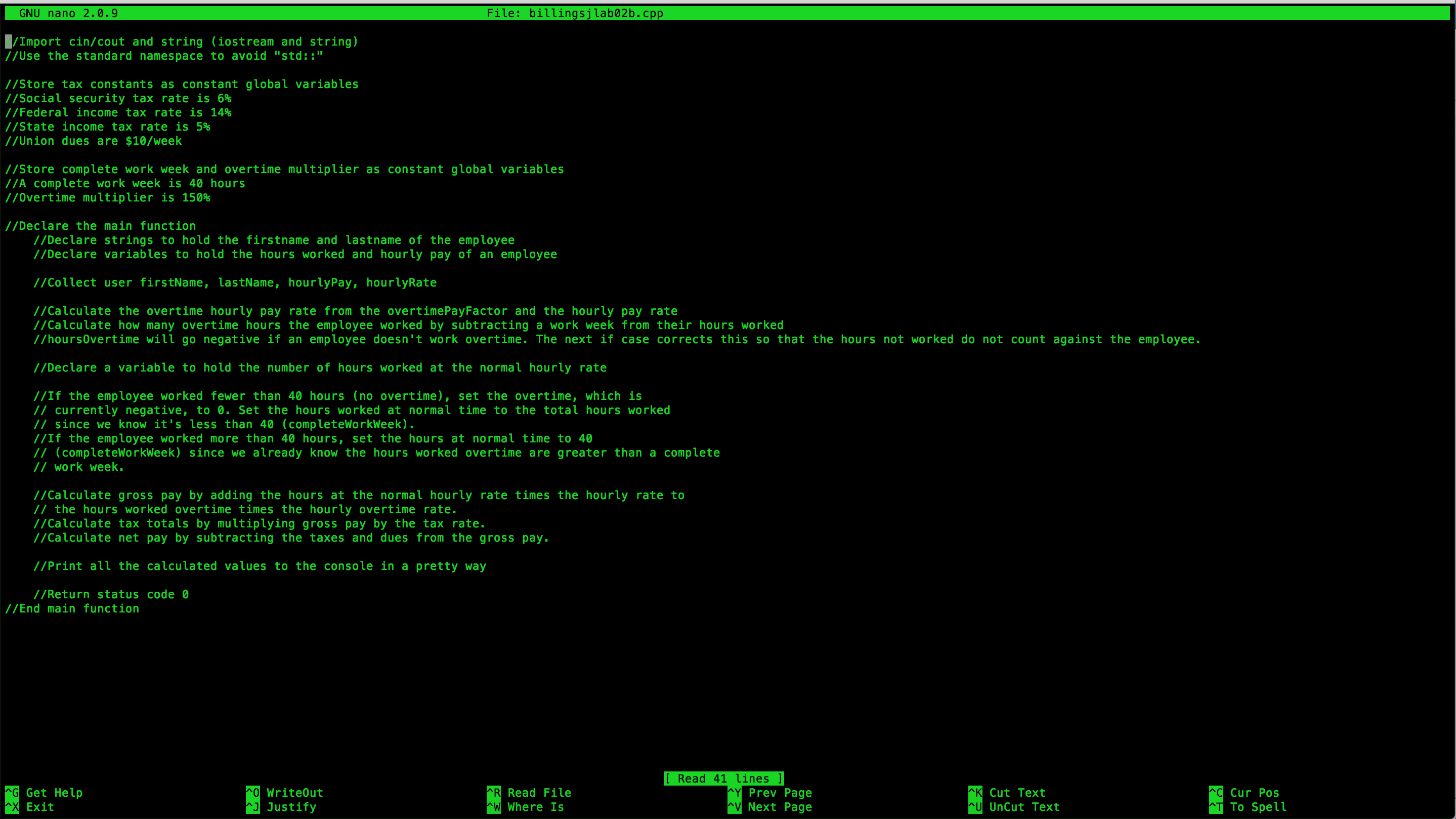
**//Begin Main**

**//Ask Customer name**

**//Print “Spread peanut butter on bread”**

**//Print “Spread jelly on bread”**

**//End Main**

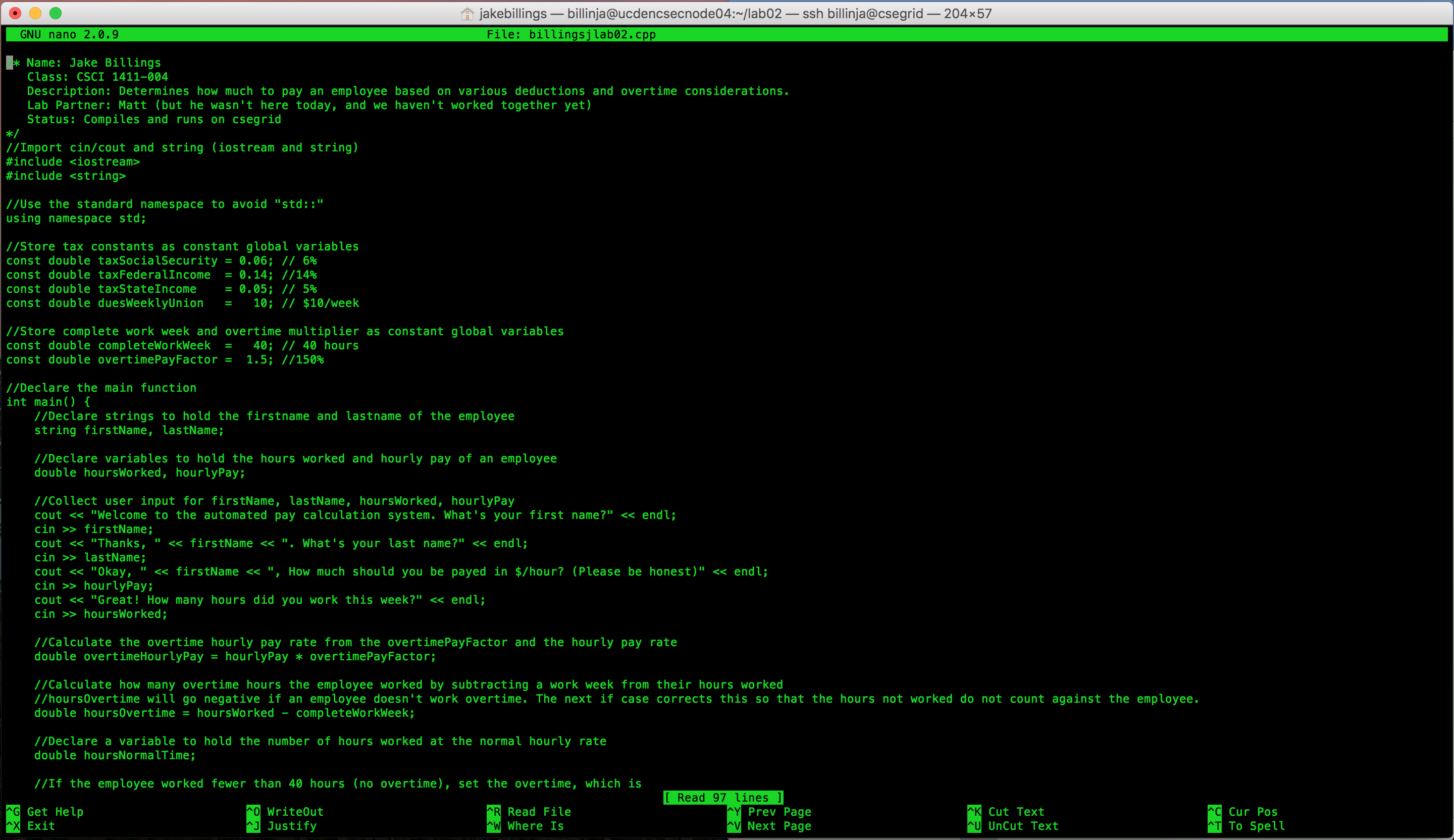


**Part IV -Employee pay rate implementation. (10 pts)**

* Write a program to determine how much to pay an employee. Ask the employees First Name, Last Name and hourly pay rate, and number of hours worked in week one, and week two. If the employee works more than 40 hours per week, they earn “time and a half” (\* 1.5) for the additional hours. From the gross pay, 6% is withheld for Social Security tax, 14% is withheld for Federal Income tax, 5% is withheld for State income tax and $10 per week for union dues. The program should ask the appropriate questions, then provide the gross pay (without deductions) and the net-take-home pay.
* Call the program **yourlastnameFirstinitialLab02.cpp** (e.g. augustinetLab02.cpp). Every program you write should have the following block at the top in comments. Make sure to fill in the Name, Class, Description and Lab Partner. Ensure your status is accurate.

/\* Name:  
Class: CSCI 1411-00X  
Description: [fill in description]

Lab Partner:  
Status: successfully compiled and run on csegrid [if it doesn’t run or meet all of the requirements, list the actual status!/\*



(See file for full code and next page for screenshot of application running)

* You should work with your lab partner on this, but each of you must write your own code.
* Implement the program in part II, placing the actual code under each appropriate line of the outline.
* Ensure Part IV runs and compiles, complete with comments from Part III. Hand in yourlastnameLab02.cpp, and this lab with the screenshots.

