**Intro to the c compiler**

**LAB 2**

**SECTION A**

**SUBMITTED BY:**

**Brian Reber**

**SUBMISSION DATE:**

**9/18/2009**

**Lab Problem**

The purpose of this lab is to get acquainted with the programming development, as well as do some basic programming in C. We will be writing a program that accepts user input, and uses the input in a mathematical equation, and print out the result.

**Analysis**

In this program, we will be using the C language to input data. We will be using variables of type int to store our values, and then computing a final result using a given equation and using the user inputted values.

The formula we are to use is

a\*a/b + b\*b/a

**Inputs**:

a /\* first int value \*/

b /\* second int value \*/

**Outputs**:

c /\* the value of (a\*a/b + b\*b/a) \*/

**Design**

Algorithm:

1. Get first int value.
2. Get second int value.
3. Calculate the value of c using given formula.
4. Display the value of c.

**Testing**

We ran multiple tests for this program and got some interesting, yet expected results. Since we were using ints, the results were not accurate. During the division process, it would format it as an int, therefore discarding any number after the decimal point. This is not the best way to solve this problem, but it is what we were told to do. If we were interested in accuracy, we could use doubles for our values instead of ints. Other than the accuracy issue, we didn’t have any other problems.

**Comments**

In this lab, I learned about the value of using doubles versus ints and when each should be used. Although this lab may not have taught me many uses for each, I did learn the importance of analyzing which would be the best in this certain example.

Also, relating to the Sample Application:

This application ran a loop from 14 to 128. Each time, it would print the current number as a character, then its decimal value, and finally the hex value. This was interesting to see the actual representation of ASCII. I didn’t realize that it was possible to print out variables as a different type. For example, this program had a “short” variable, and it printed it out as a char, a decimal int and in hex.

**Implementation**

/\*

\* lab2-0.c

\*

\*

\* Programmer: Brian Reber Date: 9/11/09

\* Instructor: Daniels Class: CprE 185 Section: A

\*

\*/

// LAB2-0.c : Prints out a line on the console window.

#include "stdio.h"

int main(int argc, char\* argv[])

{

printf("Nathan Brinkman and Brian Reber's CprE 185 Lab programming project is almost done!\n");

return 0;

}

/\*

\* lab2-1.c

\*

\*

\* Programmer: Brian Reber Date: 9/11/09

\* Instructor: Daniels Class: CprE 185 Section: A

\*

\*/

// LAB2-1.c : Prompts the user for two int values, performs an equation and prints out the value.

#include "stdio.h"

int main(int argc, char\* argv[])

{

int a, b, c;

printf("Please enter an int for variable a: ");

scanf("%d", &a);

printf("Please enter an int for variable b: ");

scanf("%d", &b);

c = a\*a/b +b\*b/a;

printf("Nathan Brinkman and Brian Reber's CprE 185 Lab programming project is almost done!\n");

printf("C = %d\n", c);

return 0;

}

**Out.txt**

Please enter an int for variable a: Please enter an int for variable b: Nathan Brinkman and Brian Reber's CprE 185 Lab programming project is almost done!

The value for variable c is 11