Lab 2

Name: .....

Objectives:

- Understand memory architecture.
- Translate between binary and decimal number formats.
- Discuss the differences between, and some issues with, integer and floating-point formats.
- Create a Flow chart and use it to implement a C++ program.
- Begin exploration of basic input and output.

Procedures:

Binary Numbers and Memory Architecture

1. From the information in your textbook and classroom lectures describe how memory is structured in the computer, along with what is meant by *addressing*.

2. Translate the following decimal values into 8-bit binary numbers (DO NOT USE A CALCULATOR!):
   a) 11

   b) 27

   c) 56

   d) 38

3. What is the decimal representation of the following binary numbers (DO NOT USE A CALCULATOR!):
   a) 11011001

   b) 00110101

   c) 10101010

d) 11111111

Numeric Data Types – Integers and Floating Point

4. What types of data can be represented in C++ programs?

5. Explain how integers are represented in C++.

6. Visit the following site and read its contents:
http://www.cprogramming.com/tutorial/floating_point/understanding_floating_point.html
Explain what the author means by *Accuracy* and *Precision*. How do integers and floating-point numbers differ concerning *Accuracy* and *Precision*?

7. Visit the following site and read its contents:
http://www.cprogramming.com/tutorial/floating_point/understanding_floating_point_representation.html
Based on your reading there and in the textbook, along with classroom lectures, explain how floating-point numbers are represented in C++.

8. Examine the program to the right:
Look up the "isnan" definition for c++.

a) What is the function of isnan?

```
1 #include<iostream>
2 #include<math.h>
3 using namespace std;
4
5 int main()
6 {
7    float x = 1.0;
8    float y = 0.0;
9    float z = 0.0;
10    y = x + 0.000001;
11    z = x + 0.0000000000000001;
12    cout << "X > X   -> " << (x>x) << endl;
13    cout << "Y > X   -> " << (y>x) << endl;
14    cout << "Z > X   -> " << (z>x) << endl;
15
16    cout << "11.1/0.0 -> " << 11.1/0.0 << endl;
17    cout << "0.0/0.0  -> " << 0.0/0.0 <<endl;
18    cout << "log(0)   -> " << log(0)  << endl;
19    cout << "sqrt(-1) -> " << sqrt(-1) << endl;
20
21    float a = 2.0/10.0;
22    cout << "a == a   -> " << (a==a) << endl;
23    cout << "a -> " << a << "     2.0/10.0 -> " << 2.0/10.0 << endl;
24    cout << "a == 2.0/10.0 -> " << (a == 2.0/10.0) << endl;
25
26    cout << "ISNAN(sqrt(-1)) -> " <<  isnan(sqrt(-1))  << endl;
27    cout << "ISNAN(5) -> " <<  isnan(5)  << endl;
28
29    return 0;
30 }
```

b) What do think that the output of each of the cout statements in the program will be?

Line 12:

Line 13:

Line 14:

Line 16:

Line 17:

Line 18:

Line 19:

Line 22:

Line 23:

Line 24:

Line 26:

Line 27:

c) Type in the program and save it as prog2a.cpp. Compile prog2a.cpp using your compiler. What is the command that you issued to compile the program? If you used an IDE how did you do it?

d) Run the program you compiled above. Examine its output and compare it to the estimations you made in section

(b) of this question (above). Explain any discrepancies between your estimation and the actual results. Also, explain the output for Line 14.

Flow charts, program design, and implementation

10. Create a complete C++ program that lets a user calculate the perimeter and area of a rectangle given the length and width. Print the length and width, as well as the perimeter and area, to the screen.

a) Create a Flow chart to conceptualize the problem, listing the inputs, outputs, and processing steps that must occur in the program. Make sure to check your algorithm by hand tracing the output for a given set of inputs. Make sure to assign appropriate variable names to your variables. Draw the flow chart below:

b) Write a complete C++ program to implement the algorithm from your flow chart, declaring the variables you

identified in the previous step. Assign some reasonable values for the length and width. Attach a copy of this

program to your lab sheet.

c) Compile your program.

d) Run and check your program. Make sure the output is correct.

11. Examine the following program:

a) Type in the program. Compile the prog2b.cpp source file and run it.

```
1 #include<iostream>
2 using namespace std;
3
4 int main()
5 {
6    int x;
7    cout << "Type in an integer: ";
8    cin >> x;
9    cout << "You typed the number " << x << endl;
10
11    return 0;
12 }
```

b) Explain the operation of Line 8. What does *cin* do?

You can use Google to look up the answer if this was not covered in class yet.

c) Line 7 is often referred to as a prompt. Explain why we need a prompt for this program. (what would happen if we did not have one?)

d) Run the program again, and when prompted for an integer, enter a floating-point number. What is the output of the program? How/why does it differ from your input?

e) Alter the area/perimeter program you wrote in section 10 of this lab. Use *cin* to read the length and width from the keyboard. Attach a copy of your modified program to this lab sheet.