Name: Minh Anh Bui

**CS 37 - Exam #1**

**20 Points Possible**

(2) 1a. >> is the stream extraction operator.

b. << is the stream insertion operator.

(1) 2. When is the ignore method required?

The ignore method is required whenever we have to enter a string or character data after having entered a numerical value.

(1) 3. (True/False) Once the setw manipulator is used, it does not need to repeated for the printing out of additional values. False

(1) 4. (True/False) The default justification for C++ is left-justification. False

(1) 5. Write the cout statement needed to print out your first name using a field size of 15 followed by your last name using a field size of 15 where both your first name and last name are left-justified. Please assume that the variable f contains your first name and the variable l contains your last name. That is, your cout statement should use the variables f and l.

cout << left << setw(15) << f << setw(15) << l << endl;

(1) 6. Suppose that the variable total already has the value 345.6789. Write the cout statement to create the following line of output: **Your total amount is $345.68** (total is the variable to use in your cout statement.)

cout << setprecision(2) << showpoint << fixed << “Your total amount is $” << total << endl;

(1) 7. Assume that the variable sum has already been assigned a value. Write the necessary C++ statement(s) to print the variable sum in octal and hexadecimal.

Octal:

cout << setbase(8) << sum << endl;

Hexadecimal:

cout << setbase(16) << sum << endl;

(2) 8a. C++ supports value and reference parameters. How does the programmer decide which type of parameter to use when creating a function?

The type of parameter to use when creating a function depends on what will be passed through it. Numerical values are pass by reference and pass by value. However, strings, arrays, or structs are pass by reference only.

b. (True/False) Strings can be passed by using value or reference parameters. False

(10) 9. You are to write two functions for a program that keeps track of information for one stock along with a line of code to call each function and its prototype. The data to be entered by the user is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Stock Name | Shares | Buy Price | Current Price | Stock Type |
| Joe’s Technology | 150 | 69.34 | 77.55 | Large Cap |

a. Write a function to allow the user to enter in the above data for the stock **in the order given above**.

Please place your function here:

void input (char stockname[20], int &shares, float &buyprice, float &currentprice, char stocktype[20])

{

cout << "Enter the stock name: " << endl;

cin.getline (stockname, 20);

cout << "Enter the number of shares: " << endl;

cin >> shares;

cout << "Enter the buy price: " << endl;

cin >> buyprice;

cout << "Enter the current price: "<< endl;

cin >> currentprice;

cin.ignore(20, '\n');

cout << "Enter the stock type: " <<endl;

cin.getline (stocktype, 20);

(

Specify one line of code which could call your function: input (stockname, shares, buyprice, currentprice, stocktype);

Place the function prototype here: void input (char [20], int &, float &, float &, char [20]);

b. Write a function that will calculate and return the total cost (shares \* buy price), the current value (shares \* current price), and the profit (current value – total cost) for the stock.

Please place your function here:

void calculate (float &totalcost, int &shares, float &buyprice, float &currentvalue, float &currentprice, float &profit)

{

totalcost = ((float)(shares))\*buyprice;

currentvalue = ((float)(shares))\*currentprice;

profit = currentvalue – totalcost;

}

Specify one line of code which could call your function: calculate(totalcost, shares, buyprice, currentvalue, currentprice, profit);

Place the function prototype here: void calculate (float &, int &, float &, float &, float &, float &);