**ITCS 1212L**

**Lab 8**

**Pre-lab Activity:** For these labs, you need to work on the algorithm for the main. You need to declare the variables including the arrays and the counters (e.g. for counting the number of stocks) and other variables. Think about some typical inputs for the program and their expected output as well as the functions are needed for this program as well as the logic of each function and the main(). Write down the typical inputs and what kind of output you expect.

**Lab 8a: Get Weather condition for a zipcode**

**Objective: Function Call as input, simple data processing**

Write a main program to get the weather forcast of a zipcode by calling the proper web services.

cout<<”Enter the zipcode: “;

cin>>zipCode;

### forecast = getInfo(zipCode);

Try it for different zipcodes and have you TA check your answers.

**TA Check\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Lab 8b: Get a Stock Value by passing a proper stock index.**

**Objective: Function Call as input, simple data processing**

Write a main program to retrieve at least three different stocks such as INTL, GOOG, and IBM (you can choose your own stocks) by using the proper web services.

cout<<”Enter the stock Sybmol: “;

getline(cin, stockSym);

stockPrice = getQuote(stockSym);

Make sure to declare the appropriate variables and write the algorithm/pseudocode first. Have you TA check your answers.

**TA Check\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Lab 8C: Retrieve silver, Copper and Stock prices**

**Objective: Function Call as input, simple data processing**

Assume that you need to retrieve the price of Silver, Copper, and a specific Stock. Write a main program to call the webservices functions such as getSilver(), getCopper() and getQuote() and show the prices for different dates(for silver and copper) and stock symbols.

Make sure to declare the appropriate variables and write the algorithm/pseudocode first.

**TA Check\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Lab 8d: My Portfolio**

Write a program that determines the stock price of four companies (INTL, GOOG, IBM, and MSFT) and with number of stocks for each one from the user, it calculates the total value of portfolio. It should include the following two functions, which are called inside the main() function:

* int getNoStocks() is passed the stock symbol (string) and returns the price for a specific stock symbol. You need to ask the user to enter number of stocks per stock symbol inside the function.
* double calculateTotalPortfolioValue() determines the total value of the portfolio.

Figure out the logic on paper first by drawing flowchart and/or the pseudocode.

Create a console project lab8d.cbj in codeblocks and type your menu-driven program in it.

**Lab 8e:**

For this lab you need to develop a program to help an investor with different investment options. The program includes a menu that gives the investor an option to check the value of the stock or silver, etc. and investor can make decision about it. Write the code for menu() function. Develop pseudo-code or flowchart for the main function.

Check the value of the following items:

1. Silver
2. Stock
3. Exit

When option 2 is selected, another message shows up like this:

Enter the stock symbol:

**TA Check: \_\_\_\_\_\_\_**

**Lab8f: Stock Market Portfolio**

Create a project named lab8f on CodeBlocks. This project is the modification to project 8d which in this case you will prompt the user to enter her/his stock portfolio and can calculate the value of the portfolio. A stock portfolio could include as many stocks as expected. You need to use a proper loop to prompt the user to enter the new stock symbol and a sentinel value for termination of the data entry. When the user enters the stock symbol, you need to ask the user also to enter the percentage of that stock in the portfolio. The stock values as well as the percentage of that particular stock in the entire portfolio will be stored in two arrays. You need a variable to count the number of stocks that is entered. You need to use web services to get the stock values.

**TA Check: \_\_\_\_\_\_\_\_\_\_**