CS1342 Fall 2020

**Programming Project 4**

**Due date: For full credit by midnight Nov. 2, or no later than midnight Nov. 4 for -30 late points.**

**Assignment Overview**

The goal of this project is to gain understanding with object oriented programming by defining your own classes to leverage the power of object-oriented programming to analyze data.

**Background**

Data mining is the process of sorting through large amounts of data and picking out relevant information. It is usually used by business intelligence organizations, and financial analysts, but is increasingly being used in the sciences to extract information from the enormous data sets generated by modern experimental and observational methods. We have seen much of this being done during the pandemic.

These large data sets and the trend of analyzing them has come to be know as "Big Data" . You can read about Data Mining and Big Data online at <http://en.wikipedia.org/wiki/Datamining> and

<http://en.wikipedia.org/wiki/Big_data>

In this project, we want to do some preliminary data mining of the prices of any stocks of your choosing found at [https://finance.yahoo.com/quote/[stocksymbol]/history/](https://finance.yahoo.com/quote/%5bstocksymbol%5d/history/). For testing purposes, you can use a file of Apple's daily stock's prices at the link <https://s2.smu.edu/~etchison/cse1342/program4.csv> taken from <https://finance.yahoo.com/quote/AAPL/history/>. However, your solution should be modular enough such that you can download any of the stock price files located at Yahoo Finance and produce the statistical averages required for a particular period of time.

**Project Specifications**

1. You can download your own stock file by going to the link above. You insert your own stock symbol in the link. You can scroll down, set the dates and click on the download button which produces a comma delimited file. You can view the downloaded file with notepad or similar text editor. If you open it with Excel, it will show you the data as a spreadsheet.
2. You must implement a MyPortfolio class and an Stock class.

The Stock class contains the following attributes:

Stock Symbol a string

Date a string

Opening value a double

High value a double

Low value a double

Closing value a double

Volume an integer

and the following methods:

an appropriate constructor

accessors and mutators for all attributes

an overloaded << method that displays the Stock data in a nice format

feel free to add additional member functions.

The Portfolio class contains the following attributes:

A file object

An vector of stock objects (empty to begin with)

and the following methods:

a default constructor that does nothing…. Just { } with nothing in the {}

openFile()  
In this method, you are required to repeatedly prompt for the name of an input file until the user enters a filename for a file that can be opened for input.

get\_data\_list()

In this method, you are required to read a stock file you downloaded. You can assume it is valid. For each line read, a Stock object is created and added to the vector of stock objects.

average\_data()

In this method, you will receive a year from the keyboard and will average the data for each column for that particular year, and display the data in a nice format. Be sure to error check the year entered from the keyboard. If invalid, display an error message and ask for it again. You will also calculate and display the month within that requested year that had the highest opening value and the month that had the lowest closing value. If there is a “tie” you can display the earliest month of that year.

Code formatting:

See link <http://lyle.smu.edu/cse/1342/Supp.doc> for proper comments and code formatting.

How to submit:

A .zip file (lastname,firstinitial,p4) containing Stock.h, Stock.cpp, Portfolio.h, Portfolio.cpp , p4.cpp, program4.csv, your own stock file, and p4.out file MUST be uploaded to Canvas  
You should not wait until the last minute when servers could go down.

**Assignment Hints:**

Since there is so much data, do some testing with a smaller file (a subset of the data) and display the data in the Stock object to make sure that you have read the data correctly.

**Example Screen Output**

Open what file? fred

Bad file name, try again

Open what file? irving

Bad file name, try again

Open what file? Project4.txt

What year do you want averaged? A012

That is not a valid year, reenter 2012

Here are the averages for the Stock for year 2012

Opening Closing High Low Close Volume

<data> <data> <data> <data> <data> <data>

where <data> is your calculated averages

The month with the highest opening value was <month> with <amount>

The month with the lowest closing value was <month> with <amount>