Programming Assignment 4, Spring 2018

OBJECTIVE

The objective of this assignment is to give you more practice with creating and using ADTs, multi-file program format, and file reading and writing ("text view").

ASSIGNMENT SUBMISSION

To get credit for this assignment, you must

- ✓ write a multi-file program in C
- ✓ submit your project as a tar file through Canvas exactly as instructed (naming, compatibility, etc.)
- ✓ submit your assignment on time

PROBLEM STATEMENT

For this assignment, you need to implement a program that helps a financial company keep track of its clients' stocks. All the necessary data is provided in *txt* and *csv* files and it is your job to put this information together and produce one output file.

Input Files

The first input file is called *clients.txt* and contains the following information about each client:

- client id
- name
- phone number
- e-mail

Sample file is provided so that you could see the file format. In order to store this information, you need to create client data type that is capable of storing this information, with each bullet above constituting one field within the data type. You should provide structure pointer and prototypes (create, destroy, getters, setters) in .h file and structure definition and functions implementations in .c file. If you choose to provide more than one create, then be aware that there is no function overloading in C and each create has to have a slightly different name.

The second input file is called *stocks.csv* and contains the following information about stocks:

- stock symbol
- its current price

Sample file is provided so that you could see the file format. In order to store this information, you need to create stocks data type that is capable of storing this information, with each bullet above constituting one field within the data type. You should provide structure definition and operations in .h file and functions implementations in .c file. For this ADT, there is no need to hide structure definition details from the end-user.

The third input file is called *stock_client.txt* and contains the following information about each client:

- client id
- number of different stocks
- for each stock the client owns, the stock symbol and the number of shares

Sample file is provided so that you could see the file format. You do not need to create a data type for this information.

Output Files

Your program is to produce one output file called *summary.csv* that combines the information from all the input files and for each client lists all the client information, namely:

- client id
- client name

- client phone number
- client e-mail
- for each stock the client owns
    - the stock symbol
    - the number of shares
    - price per share
- total value of a client (various shares x their stock values)

Sample file is provided so that you could see the file format and the contents to be included.

Program Specs

- The program needs to follow object-oriented principles and use structures to define data types and their operations
    - one ADT for client objects that is to follow ADT principles of information hiding
    - one ADT for stock objects that does NOT need to follow ADT principles of information hiding
    - generic array ADT implementation (use the one provided in class and add the operations needed by your program); this ADT is to follow ADT principles of information hiding
- The generic list should be used to store a list of clients and a separate list of stocks
- All ADTs need to abide by cohesion principle, which means that no ADT should rely on another ADT (e.g. not list functions inside a client ADT)
- The minimum number of files for your program is 7 files: 2 for generic list + 2 for client data type + 2 for stocks + 1 for a driver
- The driver should make the use of functions
- Your executable file should be called *pr4.out*
- All program files must reside in one folder, called pr4, and the contents of the entire folder must be compressed using tar utility (check lab 1 directions to make sure you are compressing an entire folder and not individual files)
- Your tar file should be called *pr4.tar*
- Your program has to follow basic stylistic features, such as proper indentation (use whitespaces, not tabs), meaningful variable names, etc.
- You need to comment your code
- You are not allowed to use global variables
- **Your program must compile in gcc gnu 90 – programs that do not compile will receive a grade of 0. Also, please test your code on the CSS gate server.**

Extra Credit (15%)

Extra credit includes the following possibilities (as always, comment at the top of your code):

- Make stock ADT a proper ADT with information hiding and create a list of clients that contains all information about the client (including their stocks) that is to be printed to the output file
- Add information to the stocks list and write it out, along with the statistics, to an output file, i.e. for each stock keep track of the overall number of shares owned by the clients and write it out along with the information like the most and least popular stock, other stats

If you have some other idea, run it by the instructor first.

Program Submission

On or before the due date, use the link posted in *Canvas* next to Programming Assignment 4 to submit your tar file. Make sure you know how to do that before the due date since late assignments will not be accepted. Few things to keep in mind:

- Please stick with the input/output format provided in the assignment.

- If you're working in a group, only one member of the group should submit the assignment. The group member who is submitting the assignment should mention the partner's name in the comment box when submitting the assignment.
- If one student claims that he/she works with another student in a group but both submitted different codes, both students will lose some points.