DISCLAIMER: We are using actual stock market data for academic purposes only. I am NOT in any way suggesting that you invest or trade in the stock market and other financial markets. Stock market investing is RISKY, and you may lose money!

CCPROG2 MP SPECIFICATION PART 4 *サルバドル・フロランテ*
AY2024 Term 2

- Part 4 covers the application of text file processing in combination with previous ideas learned.
- Please be reminded again that you are NOT allowed to use functions which were not discussed in class (unless specified otherwise).
- Question? Press this link and post your question in the Canvas discussion thread: Q&A on the Machine Problem (MP).

IMPORTANT NOTES:

- **use double data type, not float, for real numbers**
- **do NOT use I/O redirection for this MP part**

PRELIMINARY NOTE
Since you're already reading this PDF document, it means that you have already extracted **`MP4-SPECS.zip`** file. Examine the contents the folder named as **`SHD`** – and verify that it 30 contains TXT files, one for each of the 30 PSEi companies. Note also that all dates in all TXT files follow the "MM/DD/YYYY" format.

Challenge #9: Read From an Input Text File, Write to an Output Text File [10 pts].
Read and follow the instructions in the accompanying skeleton file **`GROUPNUMBER-C9.c`** to produce a C program that will accomplish the following:

1. Read the contents of an input text file containing the stock historical data for the company corresponding to the symbol specified as actual parameter in the **main()** function. For example, if the actual parameter is "AC", then your program should read ALL the contents of the file "AC.TXT" stored in the **`SHD`** folder.
2. Write the contents of the output text file which contains basically the same set of values as the input text file but with TWO differences:
    a. the date values are formatted as **"YYYY-AAA-DD" (**for example, "2021-OCT-27") where "AAA" represents the first 3 characters of the month in word (in capital letters), i.e., "JAN", "FEB", …, "NOV", "DEC".
    b. the rows of data are listed in CHRONOLOGICAL order starting from the oldest date to the most recent date.

The name of the output text file should be **`"GROUPNUMBER-SYMBOL.txt"`** where SYMBOL is the stock's symbol. For example, if your group number is 1, and the input text file is **"AC.txt"**, then the output text file should be named as "**01-AC.txt**".

Keep in mind that:

- The input text file should be read from the SHD folder (if the company symbol exists).
- The output text file should be in the same folder as your C source file.

Non-compliance in the output file format will be considered as a logical error. Your program should produce the same format and contents as **`"C9-AC-EXPECTED.txt"`** using **"AC.txt"** as input file.

IMPORTANT RESTRICTIONS:

- You should **NOT** use input/output redirection to produce the output file for this challenge. Instead, you should call the four basic text file processing commands **`fopen()`**, **`fscanf()`**, **`fprintf()`** and **`fclose()`** to achieve the requirements.
- The SHD folder and its contents should NOT be modified in any way. This means that you should NOT create any new file inside the SHD folder, and that you should not change the contents of any of the 30 text files inside the SHD folder.

DELIVERABLES: Submit four files via Canvas before the indicated submission deadline

1. C header file named as **`GROUPNUMBER-C6.h`** (ONLY IF YOU USED your C6 header file in your solution)
2. C source file named as **`GROUPNUMBER-C9.c`**.
3. Your program's actual output file named as **`GROUPNUMBER-AC.txt`** using **`AC.txt`** as input.
4. Redirected output text file of the bbtest comparison result named as **`GROUPNUMBER-C9-BBTEST.txt`**.

Challenge #10: Stock Portfolio [10 pts + 10 bonus pts]. **The 10 bonus points will be awarded if and only if (a) the solution is logically correct, AND (b) the specifications were all complied with properly. No bonus point will be awarded otherwise.**

Investors/traders buy and sell stocks[1]. The collection of stocks currently held by an investor/trader is referred to as a **portfolio**.

Refer to the accompanying example **`PORTFOLIO.txt`** file. Note that other filenames can be used for the portfolio name (such as **myportfolio.txt**, or something more generic such as **abc.txt**). The portfolio text file follows the format below:

```
<stock symbol> <number of shares> <buy date>
<stock symbol> <number of shares> <buy date>
      :              :                :
      :              :                :
<stock symbol> <number of shares> <buy date>
```

For example, the first line of **`PORTFOLIO.txt`** contains:

```
AC      80      01/08/2015
```

This means that the investor/trader bought 80 shares of AC stock on January 8, 2015.

For simplicity, we assume that:

- the encoded values are valid, i.e.,
  - the stock symbol exists – this means that there is always a corresponding text file in the **`SHD`** folder
  - the number of shares is a whole number (integer) greater than zero
  - there are stock historical data on the buy dates
- the shares of stocks were bought at the Close price
- the contents are not necessarily in any alphabetical order (by stock name) or chronological order (by buy date)
- there may be several lines that contains the same stock symbol (i.e., an investor/trader may buy the same stock multiple times on the same or different dates, and possibly with different number of shares)

Read and follow the instructions in the accompanying skeleton file **`GROUPNUMBER-C10.c`** to produce a C program that will accomplish the following:

1. Read using **`fscanf()`** the contents of the named portfolio text file. For each row of portfolio data, compute the corresponding P/L and %P/L with reference to a date that will be supplied as a function parameter. This requires that the text file for a specified symbol be opened as well, and the that stock historical data be read also using **`fscanf()`**. **Your program should read the stock historical data from the appropriate text file in the `SHD` folder. You are also required to implement the binary search algorithm and use it to search for the Close price using `<buy date>` as search key.**
2. Write using **`fprint()`** to an output text file which will contain results formatted as follows:

```
PORTFOLIO PERFORMANCE AS OF <reference date >

<stock symbol> <number of shares> <buy date> <buy price> <reference price> <P/L> <%P/L>
<stock symbol> <number of shares> <buy date> <buy price> <reference price> <P/L> <%P/L>
      :                              :                                          :
      :           :                  :                                          :
<stock symbol> <number of shares> <buy date> <buy price> <reference price> <P/L> <%P/L>
```

   a. Assume that the reference date supplied is valid, and all the buy dates in the portfolio text file are less than the reference date.
   b. The reference price is the Close price on the reference date.
   c. The buy price, reference price, P/L and %P/L values should be written with 6 digits after the decimal point.

[1] Note that we'll only consider LONG position, i.e., buy the stock and then later sell the stock. There is another kind of position called SHORT position where the trader sells the stock and then later buys it. Read more about long vs. short position here.

The name of the output text file should be the same as the portfolio text file preceded with your group number and a dash. For example, if your group number is 1, and the input text file is **"PORTFOLIO.txt"**, then the output text file should be named as "**01-PORTFOLIO.txt**".

Keep in mind that:

- The stock historical data should be read from text files located in the SHD folder
- The output text file should be in the same folder as your C source file.

Non-compliance in the output format will be considered as a logical error. Your program should produce the same format and contents as **`C10-EXPECTED.txt`** using the sample **PORTFOLIO.txt** as input file.

You are also required to create your own portfolio file for testing purposes. Edit the accompanying file **`MYTEST.txt`** by adding at least 10 rows of data after the existing 1st line and use it in Test #2 as specified in the **main()** function.

IMPORTANT RESTRICTIONS:

- You should **NOT** use input/output redirection to produce the output file for this challenge. Instead, you should call the four basic text file processing commands **`fopen()`**, **`fscanf()`**, **`fprintf()`** and **`fclose()`** to achieve the requirements.
- The SHD folder and its contents should NOT be modified in any way. This means that you should NOT create any new file inside the SHD folder, and that you should not change the contents of any of the 30 text files inside the SHD folder.

DELIVERABLES: Submit SIX files via Canvas before the indicated submission deadline

1. C header file named as **`GROUPNUMBER-C6.h`**
2. C source code named as **`GROUPNUMBER-C10.c`**
3. Portfolio text file named as **`MYTEST.txt`** that you created and used in Test #2 as specified in the **main()** function
4. Your program's actual output file named as **`GROUPNUMBER-PORTFOLIO.txt`**
5. Your program's actual output file named as **`GROUPNUMBER-MYTEST.txt`**
6. Redirected output text file of the bbtest comparison result named as **`GROUPNUMBER-C10-BBTEST.txt`**.

--The End --