Programming Assignment 3
Divide and Conquer

March 26, 2024

As the US stock market has entered a Bear market, Dr. Chen decided to apply his machine learning knowledge to build a stock price prediction model. He plans to manage his retirement funds using this model. His dream of early retirement relies critically on a computer program that is able to identify when to purchase and when to sell a stock to maximize profits. This program will take as an input time series of predicted stock price and find two time points, purchasing time A and selling time B, so that the profit is defined as the stock price at B minus the stock price at A, is maximized. As Dr. Chen witnessed the major financial consequences of a short squeeze of the stock of the American video game retailer GameStop (NYSE: GME), he decided that this program should not allow for short selling. In this model, the stock must be purchased before it can be sold, i.e., $A < B$. Figure 1 illustrates the optimal purchasing and selling points of the AMZN stock during the given time frame.

Figure 1: AMZN stock price. For the given time window, purchasing at A and selling at B generates the maximum profit.

In the project, you will help Dr. Chen to implement this critical piece of code, i.e., to quickly

identify purchasing and selling points to maximize the profit in a given time frame. Your program (TradeStock) should run in the command line. It takes two arguments: name of a binary file that contains stock prices in chronological order and the algorithm being used. It prints out your name, the name of the input file, the name of the algorithm, the optimal purchasing time, the optimal selling time, and the estimated profit. Each input file consists of the following in binary format:

- n, the length of the stock price series. It has type int (4 bytes in big-endian).
- n stock prices. Each price has type float (4 bytes in big-endian).

For example, the input file mystock0.bin contains 10 prices in chronological order:
10 92.4224 34.3862 59.6091 34.2989 36.5064 54.0163 21.1790 64.0800 16.7446 34.2500
The following command finds the optimal purchasing and selling point for mystock0.bin:
java -jar TradeStock.jar mystock0.bin 2
and returns the following:
Yixin Chen
mystock0.bin
Theta(n) Divide and Conquer
6, 7, 42.9010
Note that the series of stock prices uses 0-based index. In this example, the 6-th and 7-th prices are 21.1790 and 64.0800, respectively. The program finds the maximum profit of 42.901.

Specific requirements of the project include:

1. Use divide and conquer strategy to design your program. (12 pts + 4 pts)
   Explain in English the design and present pseudo code. If your program passes all the test cases and has a time complexity of $\Theta(n \log n)$, you will receive 12 pts. Partial credit (8 pts) will be given if your program passes all test cases and has a time complexity of $\Theta(n^2)$ (i.e., a brute force algorithm). Extra credit (2 pts) will be awarded if you implement a linear time ($\Theta(n)$) divide and conquer algorithm. Additional extra credit (2 pts) will be awarded if you implement a $\Theta(n)$ decrease and conquer algorithm. Note that you will only receive partial credit (8 pts) if you transform this project into a problem that has a well-known divide-and-conquer solution.
2. Provide time complexity analysis in the report. (8 pts)
   Please include the recurrence equation and solve it. If your program does not use divide and conquer and the analysis is correct, partial credit (6 pts) will be awarded.

Please submit a report, JAVA source code and .jar file to Blackboard before Wednesday, April 10th, 2024.