Optimal Trading Strategies

The goal of this project was to design and implement a simple trading algorithm that could be used for trading strategies in investment firms. Investment firm X are responsible for trading a stock for company Y. Company X can predict the stocks for the company Y for the n days with a reasonable amount of accuracy. The predicted prices are Pn, where n is the day number. The goal of this strategy is to buy and sell stocks of company Y in order to max profits.

The strategy is that Y's stock would be bought on day b1. Then Y's stock will be sold on s1. The next day will repeat. If r = 1 the optimal trading strategy is buy at day 1 and sell at day 4. If r = 3 then the optimal trading strategy is to buy at day 1 and sell at day 4. Then buy on day 5 and sell at six.

Each group member played an important part to this contributed fairly. Tony Forsythe and Eric song helped with the create the main logic of the algorithm needed to created the trading strategy. Mark and Yao helped add their input and test the logic. All group members helped code certain parts of the projects and helped other group members if help was needed. Tony prepared the makefile and other components needed to run the program. All the documents including the project report and the README is written by Mark Madden and checked over by the other group members.

The algorithm is simple and works for every r value. What the algorithm does when r = 1 is create a 2D array. It will then sets all values in the 2D array to zero. The project will then keep track of a current variable. A loop will then run for as long as the counter is less than the size.In that loop it will run through that 2D array to loop. It will run a function that would look for the max in the parameters. The parameters are, the last element in the 2D array and the element from the prices file plus the current variable. The max number will be the value of the current element in the 2D array. The next line is to assign a new value to the curr variable. Curr is assigned the Max between either the current curr or the value of the previous element i the 2D array minus the current value in the prices file. The day to buy is determined right before the curr value changes. If the curr does not change the loop runs again. We track the change of the 2D array and the cur value to return the date of buy and sell.

The complexity for the algorithm when r = 1, 2 and 3 is O(n) because the for loop only runs through the array once.