LAB NR. 6
C PROGRAMMING

November 15, 2023

We will be playing today on a stock market. We will simulated the changes in the stock price, and look at the group of the players, who are using the random strategies.

Problem 1:

Simulated the price changes of the FORD Motor Company stock, over some period. Use the preprocessor `#define` directive to create the `TRDAYS` constant that has the value 500 which corresponds to the number of the days in the price history (around two years). Declare the array, which will hold the history of the stock price. The initial price is \$13.9. Each day it can change in the range from $90 - 110\%$ of the price from the previous day. Use the `rand` function from the `<stdlib.h>` header to generate pseudo-random numbers from $90 - 110$ to generate history. Find and print the minimal, and the maximal price in the stock history.

Problem 2:

Now you have a group of players. Use the preprocessor `#define` directive to create the `PLAYERNR` constant that has the value of the number of the players in your group. Each player has initially 100\$ at his account. Each day, the player is buying, or selling a random number of stocks. He cannot obviously sell more stocks than he has, neither he can buy more, than what the money in his account allows for. Simulate the trading history for each player, and calculate his total wealth at the end (the total money plus the stocks value, based on the last price)

Problem 3:

Sort the values of the players' wealth using the bubble sort algorithm (see `https://en.wikipedia.org/wiki/Bubble_sort` for the description. Print the unsorted, and the sorted values, the maximum and the minimum wealth in the group.