The Problem

Day traders (http://en.wikipedia.org/wiki/Day_trading) buy and sell different types of financial instruments, including stocks, possibly many times within the same trading day, but sell all their positions before the close of the market for that trading day. In this project, we simulate a day trader’s activities for a specific stock and in the process, collect a few useful statistics.

The Set Up

Kirby, our day trader, starts with a fixed amount of cash in his account as he starts his trading day and before the end of the trading day, he will sell all his positions and as a result, ends up with only cash in his account. Kirby’s starting cash amount is an input to the program that you will write to simulate one of his trading days for a particular stock. Before he starts his trading activities, Kirby decides how many transactions he wants to perform. A transaction is either a buy or a sell. In case of a buy, on your request, Kirby enters the number of shares to buy and the price per share. If he has sufficient funds for the purchase, you execute the transaction. For a sell, on your request, Kirby enters the number of shares to sell. If he is holding sufficient number of shares, on your request, he enters the sale price per share and you execute the transaction. After each such transaction, you should inform Kirby of the number of shares that he holds and his current balance. Once all transactions have been completed, if Kirby is still holding shares of the stock that he is trading, you ask him to provide a price per share so you sell his outstanding shares. Finally, you would print a number of statistics to summarize Kirby’s trading day transactions.

Details

As your program runs, the first order of business is to ask Kirby to enter his opening balance. Throughout this write up, your prompt is printed using this font and Kirby’s input has been underlined and italicized.

Starting cash amount? **12000**

Next, you need to ask Kirby the number of transactions (buy/sell) that he wants to perform during that trading day.

Number of trades for today? **5**

So, five (5) transactions. You would give Kirby a bit of information at this point.

You do not own any shares, but have $12000.0 in cash.
Knowing that at this point Kirby doesn’t hold any shares, you prompt him for the number of shares he wants to buy and the price per share. Again, his input is underlined and italicized.

Number of shares to buy? 120
Price per share? 3.45

You give Kirby some feedback.

120 shares for $3.45 per share cost $414.0.
Currently holding 120 shares and have $11586.0 in cash.

Ask Kirby for the type of the next transaction (trade number 2).

Trade number 2 (buy/sell)? buy
Number of shares to buy? 250
Price per share? 3.5

The second and third lines above are in response to Kirby’s transaction type, which is a buy. More feedback for Kirby.

250 shares for $3.5 per share cost $875.0.
Currently holding 370 shares and have $10711.0 in cash.

Prompt Kirby again for his next trade and once processed, give him feedback.

Trade number 3 (buy/sell)? sell
Number of shares to sell? 50
Sale price? 4.125
50 shares for $4.125 per share is worth $206.25.
Currently holding 320 shares and have $10917.25 in cash.

Let’s see Kirby’s next two transactions.

Trade number 4 (buy/sell)? buy
Number of shares to buy? 380
Price per share? 5.2
380 shares for $5.2 per share cost $1976.0.
Currently holding 700 shares and have $8941.25 in cash.

Trade number 5 (buy/sell)? sell
Number of shares to sell? 250
Sale price? 4.8
250 shares for $4.8 per share worth $1200.0.
Currently holding 450 shares and have $10141.25 in cash.

At this point, we have carried out all five transactions that Kirby requested. So, we are done. However, Kirby is a day trader and has to sell all the shares (450) that he is currently holding. We need to know price per share so we can complete our last trade for the day.
Trading day is almost over and you are left with 450 shares and $10141.25 in cash. Enter the price to sell these shares: 5.125

We are finished with the interactions. let’s give him the final statistics.

Total shares bought for the day is 750 with an average of $4.3533333333333335 per share. Minimum and maximum price per share bought are $3.45 and $5.2, respectively.
Total shares sold for the day is 750 with an average of $4.95 per share. Minimum and maximum sale price per share are $4.125 and $5.125, respectively. End of day balance is 12447.5, a gain of $447.5.

Input validation
You should assume that all the input that Kirby provides are valid. That is, he will enter an integer to represent the number of transactions to perform. For each transaction type, he enters either buy or sell. He also enter an integer for the number of shares to buy or to sell. Finally, he enters a floating-point numbers for price-per-share. Therefore, you need not perform any error checking.

buy Rules
The very first transaction that Kirby perform is a buy and as such, you don’t ask him for the transaction type, rather, you ask him for the number of shares and price-per-share. More generally, if Kirby does not hold any shares, you do not ask him for the transaction type. That is, for example, if for his first transaction, Kirby buys 100 shares and for the second transaction, he sells them all, then for the third transaction, you shouldn’t ask him for the transaction type as it has to be a buy; he doesn’t have any shares to sell.

If the number of shares that Kirby has requested times price-per-share exceeds Kirby’s balance, you should inform him of that fact, ignore the buy transaction, and move on to the next transaction, if any. The cost of purchases are subtracted from Kirby’s balance.

sell Rules
You should prompt Kirby for the transaction type only if he has at least one share to sell. If the number of shares that Kirby wishes to sell is larger than the number of shares that he holds, you should inform him of that fact, ignore the sell transaction, and move on to the next transaction, if any. The proceeds from sales are added to Kirby’s balance.

sell all Rules
Once you perform all transactions that Kirby has requested, if there are any shares outstanding, you should ask Kirby for a sale price and sell them all at that price. Therefore, if Kirby asks for n transactions, you may have to perform n + 1 transactions, the last one being the sale of his outstanding shares. You should treat this transaction as any other sell transaction.

Statistics
As you perform transactions, you should keep track of the cumulative total number of shares bought and the cumulative total number of shares sold. In addition, you should keep track of the cumulative purchase amount and cumulative sale amount for the entire day so that you can
calculate the average purchase price and the average sale price for each share purchased and sold. The following table shows the types of information that you need to keep as you interact with Kirby.

<table>
<thead>
<tr>
<th>Transaction</th>
<th>Transaction Type</th>
<th>Shares for Transaction</th>
<th>Price Per Share</th>
<th>Transaction Amount</th>
<th>Shares Owned after Trans</th>
<th>Cumulative Shares Bought</th>
<th>Cumulative Purchase Amount</th>
<th>Cumulative Shares Sold</th>
<th>Cumulative Sale Amount</th>
<th>Cash Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>buy</td>
<td>120</td>
<td>$3.45</td>
<td>$414.0</td>
<td>120</td>
<td>120</td>
<td>$414.0</td>
<td>0</td>
<td>$0.0</td>
<td>$12000.0</td>
</tr>
<tr>
<td>2</td>
<td>buy</td>
<td>250</td>
<td>$3.5</td>
<td>$875.0</td>
<td>370</td>
<td>370</td>
<td>$1289.0</td>
<td>0</td>
<td>$0.0</td>
<td>$10711.0</td>
</tr>
<tr>
<td>3</td>
<td>sell</td>
<td>50</td>
<td>$4.125</td>
<td>$206.25</td>
<td>320</td>
<td>370</td>
<td>$1289.0</td>
<td>50</td>
<td>$206.25</td>
<td>$10917.25</td>
</tr>
<tr>
<td>4</td>
<td>buy</td>
<td>380</td>
<td>$5.2</td>
<td>$1976.0</td>
<td>700</td>
<td>750</td>
<td>$3265.0</td>
<td>50</td>
<td>$206.25</td>
<td>$8941.25</td>
</tr>
<tr>
<td>5</td>
<td>sell</td>
<td>250</td>
<td>$4.8</td>
<td>$1200.0</td>
<td>450</td>
<td>750</td>
<td>$3265.0</td>
<td>300</td>
<td>$1406.25</td>
<td>$10141.25</td>
</tr>
<tr>
<td>sell</td>
<td></td>
<td>450</td>
<td>$5.125</td>
<td>$2306.25</td>
<td>0</td>
<td>750</td>
<td>$3265.0</td>
<td>750</td>
<td>$3712.50</td>
<td>$12447.5</td>
</tr>
</tbody>
</table>

The first row of this table represents the state of your program after having received the starting balance from the user, but before having performed the first trade. In this example, Kirby has requested five (5) transactions. The second through the sixth row each represent one transaction (one iteration of a for-loop.) The numbers reflect the state of the program after the transaction has been completed. For example, the second transaction is a buy for 250 shares for the price of $3.50 per share. The total cost of that transaction is $250 \times 3.5 = 875.0$. He currently holds 370 shares (he bought 120 shares in the first trade) and has spent a total of $1289.0 ($414.0 for the first purchase and $875.0 for this one) and is left with $10711.0 (12000.0 - 414.0 - 875.0).

The last row is a forced sell. That is, after having performed the five transaction, Kirby holds 450 shares and we we know that he has to sell them all. Once that step is executed, we could for example, find the average purchase price by dividing the cumulative purchase amount ($3265.0) by the cumulative number of shares bought (750) to get $4.353, the average purchase price per share. Likewise, we can divide the cumulative sale amount ($3712.5) by the number of shares sold (750) to get $4.95, the average sale price per share.

The minimum and maximum purchase prices and sale prices are calculated/updated as each transaction takes place. For example, the price-per-share of the first buy can be used to set the minimum and the maximum purchase prices. For each subsequent buy transaction, the price per share is compared with the minimum purchase price that has been seen so far to determine if it needs to be updated. The same logic can be used to find the maximum purchase price.

**What is due and when**

In order to manage complexity and to experiment with different features of a software system, it is always a good idea to develop it in stages. To that end, you will first develop a partial solution for this project and then progressively add more features to it until you implement all the required features. This will be done in three stages.

**Phase 1 — to be completed by Monday, 15 September at 7:00pm — 10 points:** for this phase, your solution should behave in the following manner. You should use cWolf to submit your solution. *No late submissions will be accepted.*
Number of trades for today? 5
Trade number 1 (buy/sell)? buy
Number of shares to buy? 120
Price per share? 3.45

120 shares for $3.45 per share cost $414.0.

Trade number 2 (buy/sell)? buy
Number of shares to buy? 250
Price per share? 3.5

250 shares for $3.5 per share cost $875.0.

Trade number 3 (buy/sell)? sell
Number of shares to sell? 50
Price per share? 4.125

50 shares for $4.125 per share worth $206.25.

Trade number 4 (buy/sell)? buy
Number of shares to buy? 380
Price per share? 5.2

380 shares for $5.2 per share cost $1976.0.

Trade number 5 (buy/sell)? sell
Number of shares to sell? 250
Price per share? 4.8

250 shares for $4.8 per share worth $1200.0.

Bought 750 shares and sold 300 shares.

Here is pseudo-code (http://en.wikipedia.org/wiki/Pseudocode) for this phase.

Prompt the user for the number of trades to perform and store it in a variable called num_trades. Using num_trades, set up a for-loop and for each trade do:

  Prompt the user for the type of trade that he/she wants to perform and store it in action. Prompt the user for the number of shares to trade and store it in num_shares. Prompt the user for the price-per-share and store it in price_per_share. Print the number of shares, price-per-share, and the total cost of the trade. If the trade is a buy:
    Add the number of shares to the cumulative number of shares bought. Otherwise:
    Add the number of shares to the cumulative number of shares sold. Print the cumulative number of shares bought and sold.
Phase 2 — Demo due by September, 19 at 5:00pm — 20 points: For this phase, your program interacts with the user and at each iteration, prints the state of the program. The focus here is on updating the value of the variables at each iteration of the loop based on the user input. Here is how your program should interact with the user and the values that it should produce.

Starting cash amount? **12000**
Number of trades for today? **5**
Currently holding 0 shares and have $12000.0 in cash.
Shares owned 0.
Cumulative shares purchased 0.
Cumulative purchase amount 0.0.
Cumulative shares sold 0.
Cumulative sales amount 0.0.

Number of shares to buy? **120**
Price per share? **3.45**

120 shares for $3.45 per share cost $414.0.
Transaction 1
Transaction Type is buy
Currently holding 120 shares and have $11586.0 in cash.
Cumulative shares purchased 120.
Cumulative purchase amount 414.0.
Cumulative shares sold 0.
Cumulative sales amount 0.0.

Trade number 2 (buy/sell)? **buy**
Number of shares to buy? **250**
Price per share? **3.5**

250 shares for $3.5 per share cost $875.0.
Transaction 2
Transaction Type is buy
Currently holding 370 shares and have $10711.0 in cash.
Cumulative shares purchased 370.
Cumulative purchase amount 1289.0.
Cumulative shares sold 0.
Cumulative sales amount 0.0.

Trade number 3 (buy/sell)? **sell**
Number of shares to sell? **50**
Sale price? **4.125**

50 shares for $4.125 per share worth $206.25.
Transaction 3
Transaction Type is sell
Currently holding 320 shares and have $10917.25 in cash.
Cumulative shares purchased 370.
Cumulative purchase amount 1289.0.
Cumulative shares sold 50.
Cumulative sales amount 206.25.

Trade number 4 (buy/sell)? buy
Number of shares to buy? 380
Price per share? 5.2

380 shares for $5.2 per share cost $1976.0.
Transaction 4
Transaction Type is buy
Currently holding 700 shares and have $8941.25 in cash.
Cumulative shares purchased 750.
Cumulative purchase amount 3265.0.
Cumulative shares sold 50.
Cumulative sales amount 206.25.

Trade number 5 (buy/sell)? sell
Number of shares to sell? 250
Sale price? 4.8

250 shares for $4.8 per share worth $1200.0.
Transaction 5
Transaction Type is sell
Currently holding 450 shares and have $10141.25 in cash.
Cumulative shares purchased 750.
Cumulative purchase amount 3265.0.
Cumulative shares sold 300.
Cumulative sales amount 1406.25.

Here is pseudo-code for this phase.

Prompt the user for the starting balance and store it in a variable called balance.
Prompt the user for the number of trades to perform and store it in num_trades.
Using num_trades, set up a for-loop and for each trade do:
  if this is the first iteration of the loop:
    Prompt the user for the number of shares to buy and store it in num_shares.
  else:
    Prompt the user for the type of trade that he/she wants to perform and store it in action.
    Prompt the user for the number of shares to buy/sell and store it in num_shares.
    if the action is “buy”:
      Prompt the user for the price-per-share and store it in price_per_share.
      Calculate the transaction amount and store it in transaction_amount
      Print the number of shares, price-per-share, and the total cost of the trade.
Update the balance (subtract the transaction_amount from the balance)
Update cumulative shares owned
Update cumulative shares bought
Otherwise:
Prompt the user for the price-per-share and store it in price_per_share.
Calculate the transaction amount and store it in transaction_amount
Print the number of shares, price-per-share, and the total cost of the trade.
Update the balance
Update cumulative shares owned
Update cumulative shares sold
Print appropriate statistics

Complete Project due on or before September 24 at 11:00pm — 25 points: Under Details you saw an example of how a complete solution to this project should behave. Here is another example.

Starting cash amount? 15000
Number of trades for today? 8
You do not own any shares, but have $15000.0 in cash.

Number of shares to buy? 200
Price per share? 10.7
200 shares for $10.7 per share cost $2140.0.
Currently holding 200 shares and have $12860.0 in cash.

Trade number 2 (buy/sell)? buy
Number of shares to buy? 1000
Price per share? 11.0
1000 shares for $11.0 per share cost $11000.0.
Currently holding 1200 shares and have $1860.0 in cash.

Trade number 3 (buy/sell)? buy
Number of shares to buy? 200
Price per share? 10.5
You do not have sufficient funds to purchase 200 for $10.5 per share.
Your current balance is $1860.0 but 200 x $10.5 = $2100.0.
Currently holding 1200 shares and have $1860.0 in cash.

Trade number 4 (buy/sell)? sell
Number of shares to sell? 200
Sale price? 9.5
200 shares for $9.5 per share worth $1900.0.
Currently holding 1000 shares and have $3760.0 in cash.

Trade number 5 (buy/sell)? sell
Number of shares to sell? 500
Sale price? 10.5
500 shares for $10.5 per share worth $5250.0.
Currently holding 500 shares and have $9010.0 in cash.

Trade number 6 (buy/sell)? buy
Number of shares to buy? 150
Price per share? 11.6
150 shares for $11.6 per share cost $1740.0.
Currently holding 650 shares and have $7270.0 in cash.

Trade number 7 (buy/sell)? sell
Number of shares to sell? 800
You can’t sell 800 shares as you only have 650.
Currently holding 650 shares and have $7270.0 in cash.

Trade number 8 (buy/sell)? buy
Number of shares to buy? 100
Price per share? 12.5
100 shares for $12.5 per share cost $1250.0.

Trading day is almost over and you are left with 750 shares and $6020.0 cash.
Enter the price to sell these shares: 10.0
750 shares for $10.0 per share worth $7500.0.

Total shares bought for the day is 1450 with an average of $11.124137931034483 per share.
Minimum and maximum price per share bought are $10.7 and $12.5, respectively.
Total shares sold for the day is 1450 with an average of $10.10344827586207 per share.
Minimum and maximum sale price per share are $9.5 and $10.5, respectively.
End of day balance is 13520.0, a loss of $1480.0.

Late Submission/Demo Policy

No late submissions will be accepted. Emergencies will be handled on a case by case basis.

What to do for each due date

For phase 1 and the final version of the project, you should submit them by use of cWolf.
Submitting your project on cWolf is similar to the procedure that you have been using to submit your labs. Once you have copied your solution to cWolf, at the shell prompt, you will do:

~kooshesh/bin/submit 115

When prompted for the assignment, type

P1

When prompted for the file to submit, type in the name of your project and hit the return key.
You will have to demo the second phase of your project either during any lab, the workshop (Thursdays 2:00–4:00pm), or during the office hours.

**Grading**

The most important part of your grade is the correctness of your final program. Your program will be tested numerous times, using different inputs, to be sure that it meets the specifications. A correct program will receive 85% of the grade for this project. The remaining 15% is for programming style and design. This includes good design, use of spaces between operators and operands, comments, and the use of descriptive variable-names.

**Collaboration policy**

Programming projects must be your own work, and academic misconduct is taken very seriously. You may discuss ideas and approaches with other students and the course staff, but you should work out all details and write up all solutions on your own. **The following actions will be penalized as academic dishonesty:**

- Copying part or all of another student’s assignment.
- Copying old or published solutions.
- Looking at another student’s code or discussing it in great detail. You will be penalized if your program matches another student’s program too closely.
- Showing your code or describing your code in great detail to anyone other than the course staff or tutor.