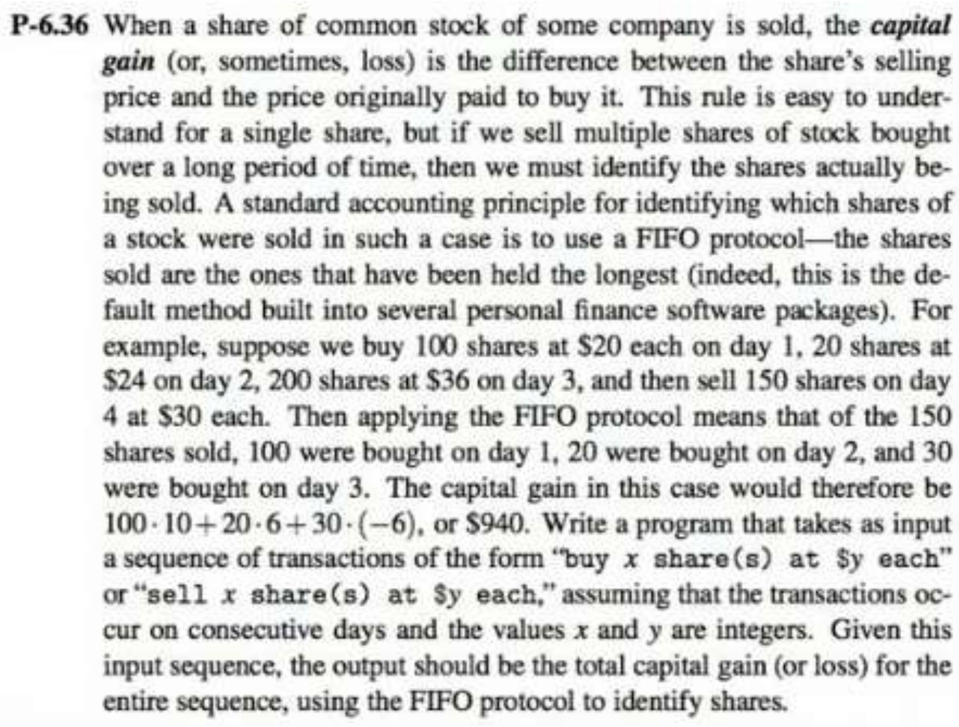
Queue Assignment

Write a Python program that uses the textbook's ArrayQueue class to implement a solution to the textbook's problem reproduced below:



P-6.36 When a share of common stock of some company is sold, the ***capital gain*** (or, sometimes, loss) is the difference between the share's selling price and the price originally paid to buy it. This rule is easy to understand for a single share, but if we sell multiple shares of stock bought over a long period of time, then we must identify the shares actually being sold. A standard accounting principle for identifying which shares of a stock were sold in such a case is to use a FIFO protocol—the shares sold are the ones that have been held the longest (indeed, this is the default method built into several personal finance software packages). For example, suppose we buy 100 shares at $20 each on day 1, 20 shares at $24 on day 2, 200 shares at $36 on day 3, and then sell 150 shares on day 4 at $30 each. Then applying the FIFO protocol means that of the 150 shares sold, 100 were bought on day 1, 20 were bought on day 2, and 30 were bought on day 3. The capital gain in this case would therefore be $100 \cdot 10 + 20 \cdot 6 + 30 \cdot (-6)$, or $940. Write a program that takes as input a sequence of transactions of the form "`buy` *x* `share(s) at $`*y* `each`" or "`sell` *x* `share(s) at $`*y* `each`," assuming that the transactions occur on consecutive days and the values *x* and *y* are integers. Given this input sequence, the output should be the total capital gain (or loss) for the entire sequence, using the FIFO protocol to identify shares.

Your program should read from a file named transactions.txt. I've supplied such a file with this assignment that can be used for testing your program, but by aware that I might test with other data as well.

Details/Hints

- Your program is only allowed to use the public methods of ArrayQueue: `enqueue()`, `dequeue()`, `is_empty()`, and `first()`. This implies that your code is *not* allowed to access `_data` or `_front`, which are supposed to be treated as private attributes. However, for debugging purposes, you might want to access one or both of these variables from the command line in the IDLE shell.
- You can assume that whenever a sell order is processed, a sufficient number of shares are available to sell so that the order can be satisfied.
- The `readlines()` method of Python file objects will create a list that contains one string element per line of the file. So, you can iterate over the lines of the input file by calling `readlines()` and then using a for statement to iterate over the elements in the list returned.
- Recall that the `split()` method can be used to split a string into its spaced-separated substrings. You can assume that each input line has exactly one space between words/numbers, so the number of shares sold will be the second element of a list producing by

calling `split()` on an input line and the value of the shares (preceded by a dollar sign) will be the fifth element of the list.

- Note that each buy order contains two pieces of information that need to be remembered for later processing of sell orders. I recommend (but do not require) storing each pair of data values as a list and storing this list in your program's queue. A key reason for using a list rather than, say, a tuple is that a list is mutable. So you can change an element of a list, and if that list is an element of the queue used by your program, the queue will "remember" this change. This can be very useful in this program because a sale can cause some but not all of the shares bought in a day to be sold and you need a way to record this change in the front element of the queue.
- As usual, you are not allowed to use features of Python that we have not covered. If in doubt, please ask.