CM2005 Object Oriented Programming mid term assignment

Introduction

The goal for this assignment is to develop a command-line technical analysis toolkit for the exchange platform.

TASK 1: Compute candlestick data

Candlesticks are a common visualisation used to show the trends in a trading system. The candlesticks show what happens in a series of time windows, in terms of the starting price, the end price and the highest and lowest prices seen. The following image shows a graphical plot of candlestick data:

Figure 1: example of a graphical candlestick plot. Each box represents the trading activity in a time period. The top and bottom of a box (the candle) represent the opening and closing value of a product. The top and bottom of the lines (the candlesticks) represent the highest and lowest value seen in that time frame. The red box on May 3rd indicates the value was lower at close than open, so value went down. You do not need to make a graphical plot – you will be making a text based plot.

In the first task, you need to compute candlestick data from the exchange data for a particular product. Here is an example of the fields needed for candlestick data:

Date	Open	High	Low	Close
2023-05-01	100	120	80	110
2023-05-02	110	130	100	120
2023-05-03	90	100	70	80
2023-05-04	95	110	70	100
2023-05-05	80	120	75	115

Note that you will need to do this separately for asks and bids, and separately for different products. So the table above might be the candlestick data for 'asks on ETH/UDST', for example.

Here is how you can compute each field:

Open = the average price per unit in the previous time frame. So say we want to compute the value for ETH/USDT asks from the list of items from the order book, which are all asks for ETH/USDT:

Amount	Price	Value (price X amount)
10	1	10
5	2	10

TOTAL	15		3	20
	MEAN price (total value / total price):			6.67

You can see that the mean price is 6.67.

Close: the average price per unit in this time frame (same as Open, but for the current time frame)

High: highest price seen this time frame

Low: lowest price seen this time frame

You should implement the candlestick data computation using a function which returns the following data type:

```
std::vector<Candlestick>
```

Note that it returns a vector of Candlestick objects. You will need to define your own Candlestick class that is suitable for representing candlestick data. One Candlestick object represents one candlestick.

TASK 2: Create a text-based plot of the candlestick data

The second task involves creating a text-based plot of the candlestick data. The figure above shows an example of a graphical plot. You should try and create something like that using text output. We recommend that you approach this by first manually typing out what kind of characters you might use in your plot to represent different things. Use your text editor to do this. E.g. you might use a '-' to represent the top of a box and a '|' to represent the stalk. Once you know what you are aiming for, you should write C++ function that takes a vector of Candlestick objects as its input and generates a plot from that data. You will need to cope with quite a few problems here – for example, how can you round the data so it is an appropriate range for your text-based plotter? How can you deal with horizontal and vertical positioning? You should attempt to use object-oriented techniques to make your code as neat and well organised as possible.

TASK 3: Plot a text graph of some other trading data

Task 3 involves plotting a text-based graph of some other trading data. It is up to you to decide what that data is, but it should be somehow derived from the crypto-currency data. Again, you need to plot it using text characters, and you should probably start by manually creating the text graph in your text editor so you know what you are aiming for, as we suggested for the previous task.

What to submit:

A PDF file containing your report, wherein you should describe how you carried out each task.

A zip file containing your code, ideally so it can be easily built and run

A PDF file containing all code. You can concatenate all cpp and h files into a single text file then save as a PDF.

A statement typed into the prompt on the Coursera platform which identifies aspects of your work that were challenging/ original/ creative/ exceptional.

A video demonstrating your code running

Marking criteria

Item	
Code style: indentation, descriptive comments	5
TASK 1: Compute candlestick data CODE	10
TASK 1: Compute candlestick data DESCRIPTION	3
TASK 2: Create a text-based plot of the candlestick data CODE	10
TASK 2: Create a text-based plot of the candlestick data DESCRIPTION	3
TASK 3: Plot a text graph of some other trading data CODE	10
TASK 3: Plot a text graph of some other trading data DESCRIPTION	3
Submit correct items: code as text in PDF, report as PDF, zip file of code and video	6
Originality and challenge of implementation	10
Clearly label all sections of the code that you personally wrote without assistance	2
TOTAL	**62**

Note that we will use all the submitted items to evaluate your work.