Your grade: 96.77%

Your latest: **96.77%** • Your highest: **96.77%** • We keep your latest score. Review your overall course grades here.

Next item

Instructions

1. Please upload a PDF file containing your report, wherein you should describe how you carried out each task.

OOP%20MIDTERM.pdf

Grading Rubric

2. Please upload a zip file containing your code, ideally so it can be easily built and run

5 / 5 points

oop-midterm.zip

Grading Rubric

Rate the code style concerning indentation, descriptive comments
Score **5/5**

3. Please upload a video demonstrating your code running. It should clearly show how your code completes the various tasks.

43 / 45 points

2023-07-17%2012-53-21.mp4

Grading Rubric

Please refer to the report, code and video.

Rate TASK 1: Compute candlestick data CODE

0: hard fail: no attempt at implementing

2: soft fail: attempted but clearly does not work at all

4: pass: attempted and nearly works or very basic

6: solid pass: working if a bit basic implementation

8: very good: technically impressive implementation

10: outstanding: very impressive implemetation with professional level code

Score **8/10**

Rate TASK 1: Compute candlestick data DESCRIPTION

0: obvious fail

1: basic write-up with obvious issues

2: decent write-up but limited detail

3: outstanding write-up, very clear writing, lots of detail

Score **3/3**

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

0: hard fail: no attempt at implementing

2: soft fail: attempted but clearly does not work at all

4: pass: attempted and nearly works or very basic

6: solid pass: working if a bit basic implementation

8: very good: technically impressive implementation

10: outstanding: very impressive implemetation with professional level code

Score **10/10**

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

0: obvious fail

1: basic write-up with obvious issues

2: decent write-up but limited detail

3: outstanding write-up, very clear writing, lots of detail

Score **3/3**

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

0: hard fail: no attempt at implementing

2: soft fail: attempted but clearly does not work at all

4: pass: attempted and nearly works or very basic

6: solid pass: working if a bit basic implementation

8: very good: technically impressive implementation

10: outstanding: very impressive implemetation with professional level code

Score **10/10**

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

0: obvious fail

1: basic write-up with obvious issues

2: decent write-up but limited detail

3: outstanding write-up, very clear writing, lots of detail

Score **3/3**

Did they submit the correct items?

2 points for Code in a zip file

2 points for report in a PDF file

2 points for video
Score **6/6**

4. Please upload your code in text format (not screenshots) in a PDF file here. Please clearly label with start and end comments exactly which sections of code you **2 / 2 points**

personally wrote without assistance. 5% of the marks for this coursework are reserved for this part.

CODE%20PRINT.pdf

Grading Rubric

In the PDF, is all student code included in a text format (not screenshots) and is it clearly labelled which code the student personally wrote?

- ○ **0 points** Not clearly labelled and/ or code is not in text format
- ◉ **2 points** Clearly labelled and in text format

5. We would like to reward students who follow their own ideas and make an effort to go beyond the basic requirements. Please highlight which aspects of your work are original/creative/exceptional in up to 100 words. You can focus on one aspect, e.g. originality, or you can cover several, depending on what you were asked to do for this coursework. 20% of the marks for this coursework are reserved for this part. **10 / 10 points**

I use ASCII characters all through for the plot and graph. To beautify the text graph visualisation, I print the asterisks in green and red colour. I also applied good OOP concepts by passing output of function directly to another function as argument. Syntax and inbuilt classes not taught in class: typedef and struct were used for the robustness of the code base. Most challenging part was task 2 and making the code efficient for the larger dataset, I solve this by using the vector of vector as grid system make changes to it and only print it at once

Grading Rubric

Read the statement made by the student about their work. This should help you to find original/creative/exceptional elements in their work. How original/creative/exceptional do you think their work is based on what you see in the submitted work, guided by their statement? Note that students do not need to be creative, exceptional and original to score well here. They can focus on one aspect, e.g. originality, or they can cover several, depending on what they were asked to do for this coursework.
Score **10/10**

Please provide feedback on this work.

What went well?

What did not go so well?

What should the student focus on to improve their work?
Excellent.