Class Project: Back tester

During the last 10 homework, we created different pieces of a back-tester. To complete the steps to finalize this project, you will gather all the prior parts to build a whole back tester.

You will first find a strategy that you would like to implement.

https://learn.alphadroid.com/blog/algorithmic-trading-strategies/common-types-of-algorithmic-trading-strategies/

Please try not to be too complicated, I would like to advise you a momentum strategy, a pair correlation one or even an arbitrage one, if you really don't want to look for more.

You will find some data to play with on this website:

https://quant.stackexchange.com/questions/141/what-data-sources-are-available-online

If you got some data from another class, please feel free to reuse it.

Part I: Analysis

You will draw a model of this strategy with R, Python or Excel to present when you will want to place orders. This model doesn't need to be complicated. Your goal is to create a reference model.

Part II: Create the back tester

Once, you got your model working, you are going to implement the backtester by using all the prior homework. You will also consider removing all the pointers and replace them by smart pointers. You will also try to use more C++11 and C+14.

The market simulator should be realistic enough but doesn't need to be perfect. Using the one we created during this course should be enough. Your backtester should be able to give the position of your trading strategy at any time and also know the P&L (Profit-Loss).

Part III: Create the unit testing for your strategy

You will use google test for the whole project and test the critical part of your projects.

Grading:

1- We will focus on checking the match of the results of the results from your analysis part I and your back tester part II. The results should be the same. If not, you should determine why there is a gap.
2- We will check the presentation of your files, of your classes, of your cmake files
3- Coding style: keep the same coding style for the whole project
4- The unit tests will be key in the grading, we need to know how your check your code

5- We will not judge your trading strategy
6- We will not give you more points if your strategy is very profitable, if the sharp ratio is better or if you don’t take any risk