

# Final Report

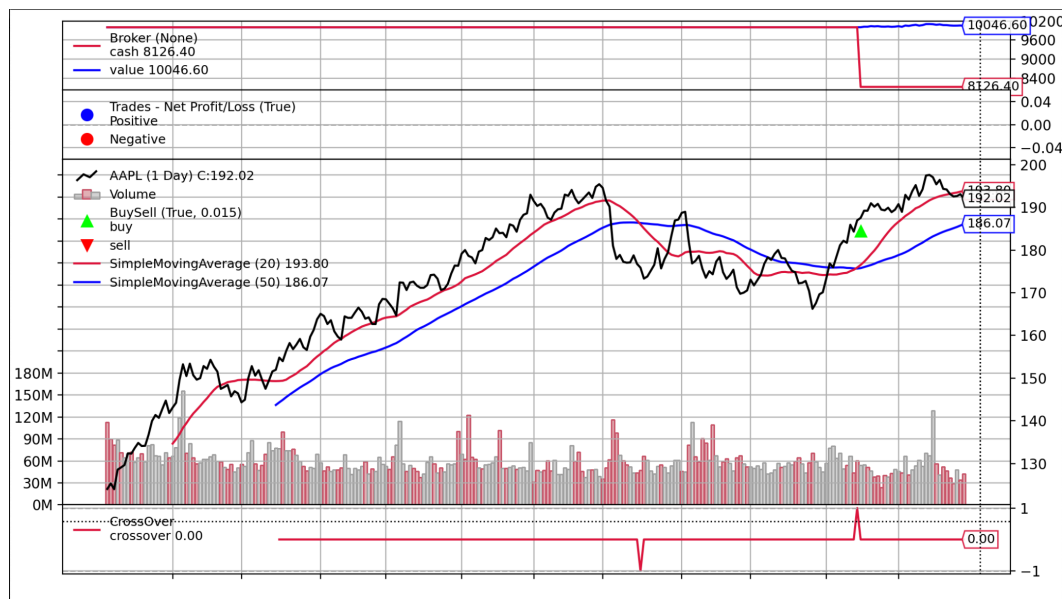
## Introduction

This project aimed to bridge the gap between backtesting and live trading by integrating a daily trading strategy developed using Backtrader with Interactive Brokers' (IB) paper trading account.

## Historical Data Backtest Strategy

In the script 'HistoricalDataStrategy.py', I implemented a simple moving average crossover strategy. The system will generate a buy signal when the short-term moving average crosses above the long-term moving average, and generate a sell signal when the short-term moving average crosses below the long-term moving average.

The strategy was backtested by the historical data from Yahoo Finance. After running the backtest, the program generated plots illustrating the moving averages and the corresponding buy and sell signals.



To improve the strategy's robustness, additional parameters can be introduced. For instance, adjusting the periods of the moving averages could allow for better alignment with different market conditions.

### **Real-time Backtest Strategy**

In the script 'RealTimeStrategy.py', I established a connection between Backtrader and the IB paper trading account. This setup allowed us to simulate live trading conditions and observe how the strategy performs in real-time.

The connection between Backtrader and the IB paper trading account was established using the `ib_insync` library. The program continuously fetched real-time market data from the IB API, enabling the strategy to respond to market changes promptly. Based on the signals generated by the moving average crossover strategy, trades were placed automatically in the IB paper trading account.

The real-time testing could generate actionable signals in a live market environment. However, the strategy's effectiveness varied with market conditions. Thus, the adaptive mechanisms are very crucial.