**413 Group Written Report**

**Project 2**

(i) Describe in words how you would use a Monte Carlo method to approximate the initial arbitrage free price of this European derivative.

First, simulate Brownian motion matrix that are i.i.d with size M, N. Then based on the Black Sholes model , we can get a matrix of stock price with size M, N+1. After filling in this matrix, compute the payoff based on the stock price for each row where . Next, compute the unique arbitrage free initial price:, price development of the asset is simulated, and value of the claim is computed in terms of an expected value in this way.

(ii) Create a Jupyter notebook to implement the scheme you described in part (i).

See on the uploaded Jupyter notebook.

(iii) Describe in words how you would use the binomial model approximation to the Black-Scholes model, to approximate the initial arbitrage free price for the European derivative. (Hint: you will need to use a NON-recombining tree and will need to use a systematic way of labeling the nodes in the tree.)

Using the binomial model for approximation to the Black-Scholes model

Time period:

r is free interest rate

Price goes up:

Price goes down:

j = 0, 1, 2, ……，N-1

As usual, the risk neutral probability is .

Supposed that the stock price is , payoff value is , the price for European derivative is and the initial arbitrage free price is . Therefore, the non-recombining tree is that.

Then, by in the question, we have:

At maturity,

In addition,

Therefore, by dynamic moving backward to get the previous values until obtaining the initial arbitrage free price

(iv) Add a section to your Jupyter notebook to implement the scheme you described in part (iii).

See on the uploaded Jupyter notebook.

(v) To test your code, use a simple example first where T = 2, S0 = 5, r = 0.1 and σ = 0.05 with different time discretizations. Then try your code for T = 10, S0 = 5, r = 0.02 and σ = 0.1. In both examples, compare the prices obtained. How close are they? How about the computation time for the two methods? Can you draw any conclusions?

In this part, we are testing the speed of convergence of the initial arbitrage-free price of the European Contingent Claim of both the Monte Carlo method and the Non-Recombining Binomial Tree method using two different sets of parameters. Also, we want to examine the run time of the two methods to test their computational performance and efficiency.

Here, we fix all other variables and only alter the time discretization (). However, as the time discretization becomes finer, the computation time must increase for the Monte Carlo and the non-recombining binomial tree methods. In fact, the number of terminal nodes for a **non-recombining tree** with N time steps is , and the number of terminal nodes for a **recombining tree** with N time steps is N+1 as we learned from the lecture.

As for the computer science niche, we may probably say the run-time complexity and space complexity for the Monte Carlo algorithm is O(MN), and the run-time complexity and space complexity for the Non-Recombining Binomial Tree algorithm is O (). So, in the end, there are instances when each algorithm will perform better in terms of run time, depending on the relative value of M and N. We can verify this in the result below.

Through a multiple trial-and-error on the code, we find that we cannot pick to be less than 0.08 from a laptop computer with 16GB RAM memory. is the most “refined” value we can choose as for the time discretization.

Based on this observation, we run different test cases on the two methods to compare their arbitrage free prices and their corresponding run time:

* **First set of parameters:**

Fix

Set.

Try for different

Table

Description automatically generated

Chart, line chart

Description automatically generated

Table

Description automatically generated

A picture containing shape

Description automatically generated

* **Second set of parameters:**

Fix

Set.

Try for different

Table

Description automatically generated

Chart, line chart

Description automatically generated

Table

Description automatically generated

A picture containing text

Description automatically generated

**Conclusion:**

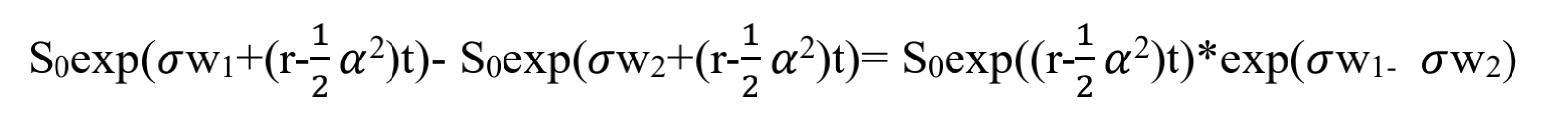
The binomial model is a straightforward but powerful technique to solve complex pricing problems. On the other hand, the Monte Carlo method is very flexible in handling high-dimensional financial issues.

The binomial model with **a non-recombining tree** may take less time to run with larger time discretizations. However, as we reduce our time discretizations to smaller values, it is clear that the Monte Carlo method converges to the initial arbitrage-free price of the European contingent claim faster. Also, the results obtained from the Monte Carlo Method are more accurate than the Binomial Method with the **non-recombining tree**. The observation will be amplified if we have more periods (T) and significant volatility (σ).

(vi) Does using antithetic variables improve the accuracy or computation time for the Black-Scholes model?

In order to better our estimation we use the antithetic variables to reduce the variance. The goal is to find another unbiased estimator with smaller variance. Thus we introduce negative dependence between pairs of replications to reduce variance. The pairs are independent and identically distributed.

First, we can prove that S(t) is a monotone function.

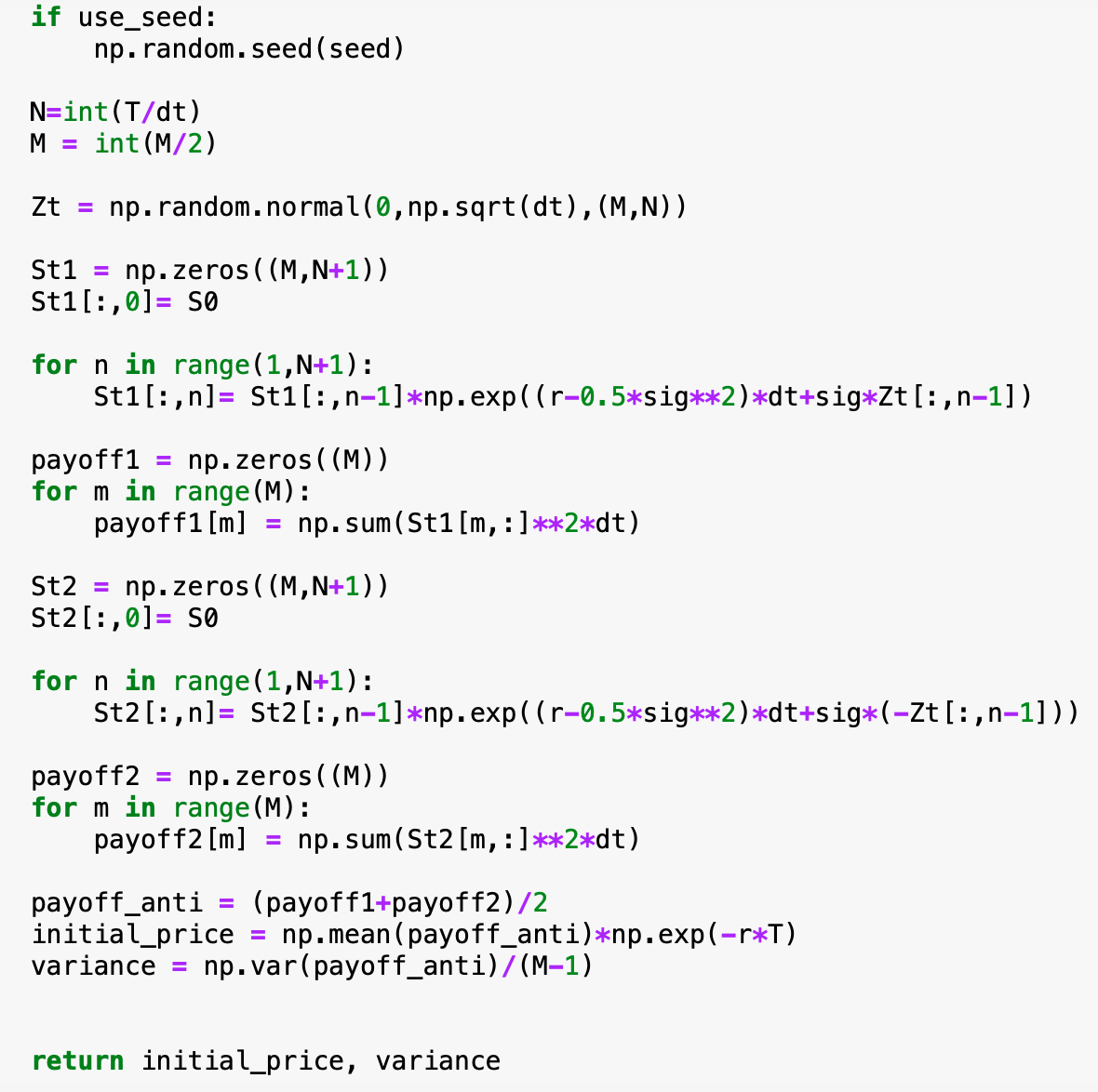


If W1>W2, the result would be positive. For ex is a monotone function, we can easily come to the conclusion that S(t) is also a monotone function which meets the condition where we could use the antithetic variables.

Thus we could get a new variance through the antithetic variable method. We also insert a time function into the calculation process to measure the computation time of this method.

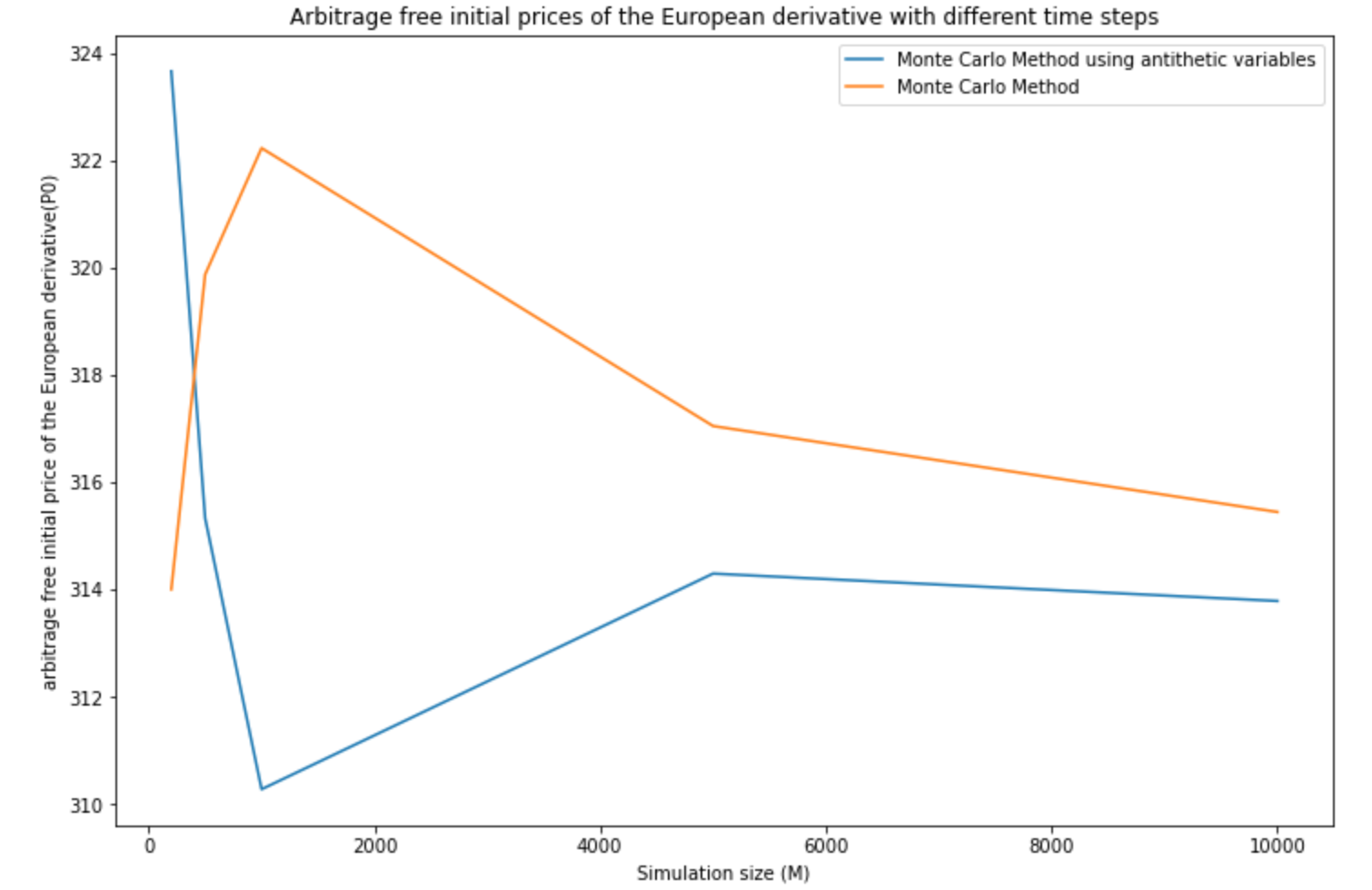
First we list all the parameters derived from inputs and outputs. And thus form a function of Monte Carlo Antithetic Method. We set up the matrixes after the setting of function.

Then we take the negative Zt to get a negatively correlated function and obtain the new variance and initial price through the calculation of payoff1 and payoff2.

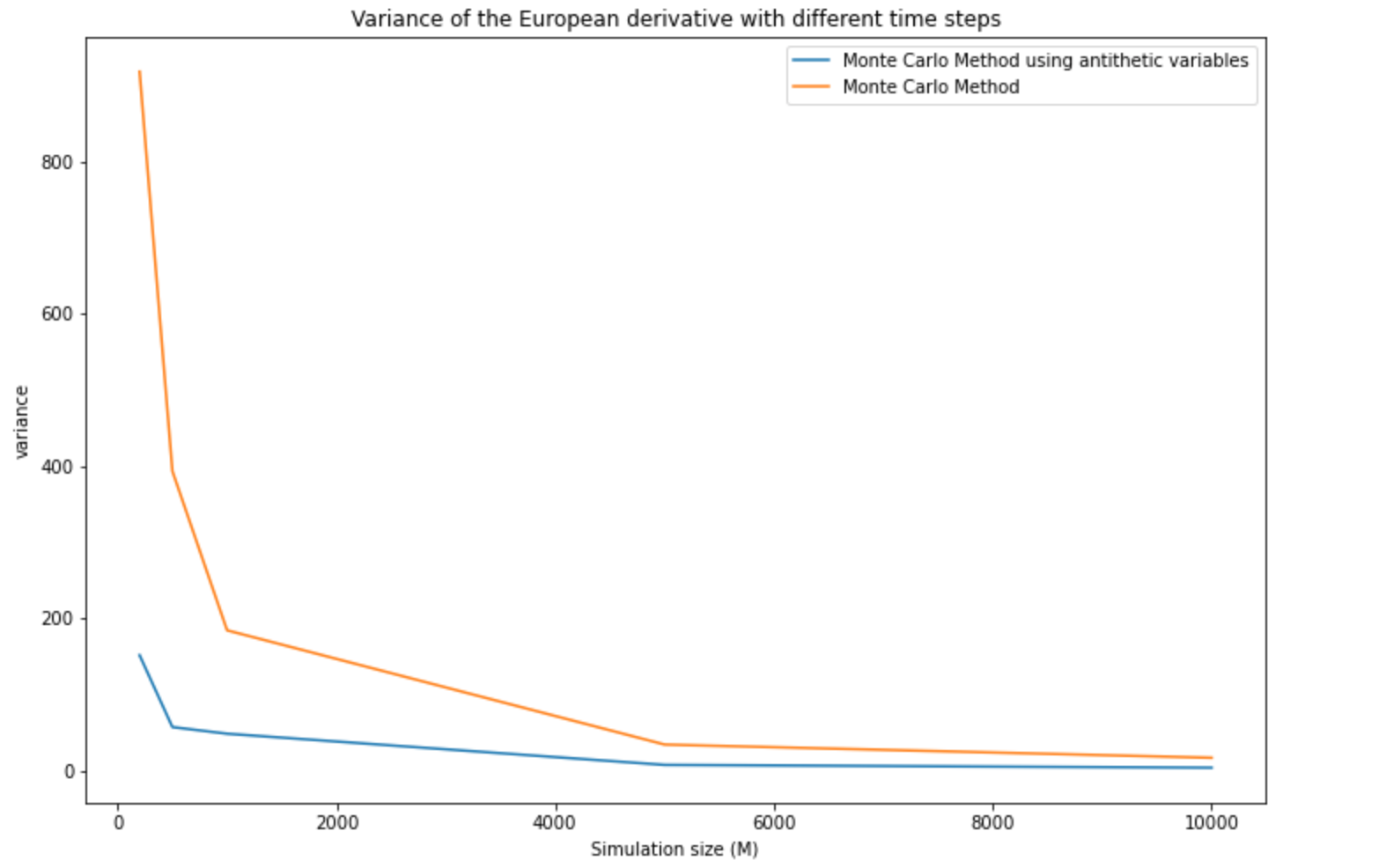


We have three charts to display the results that we have obtained.

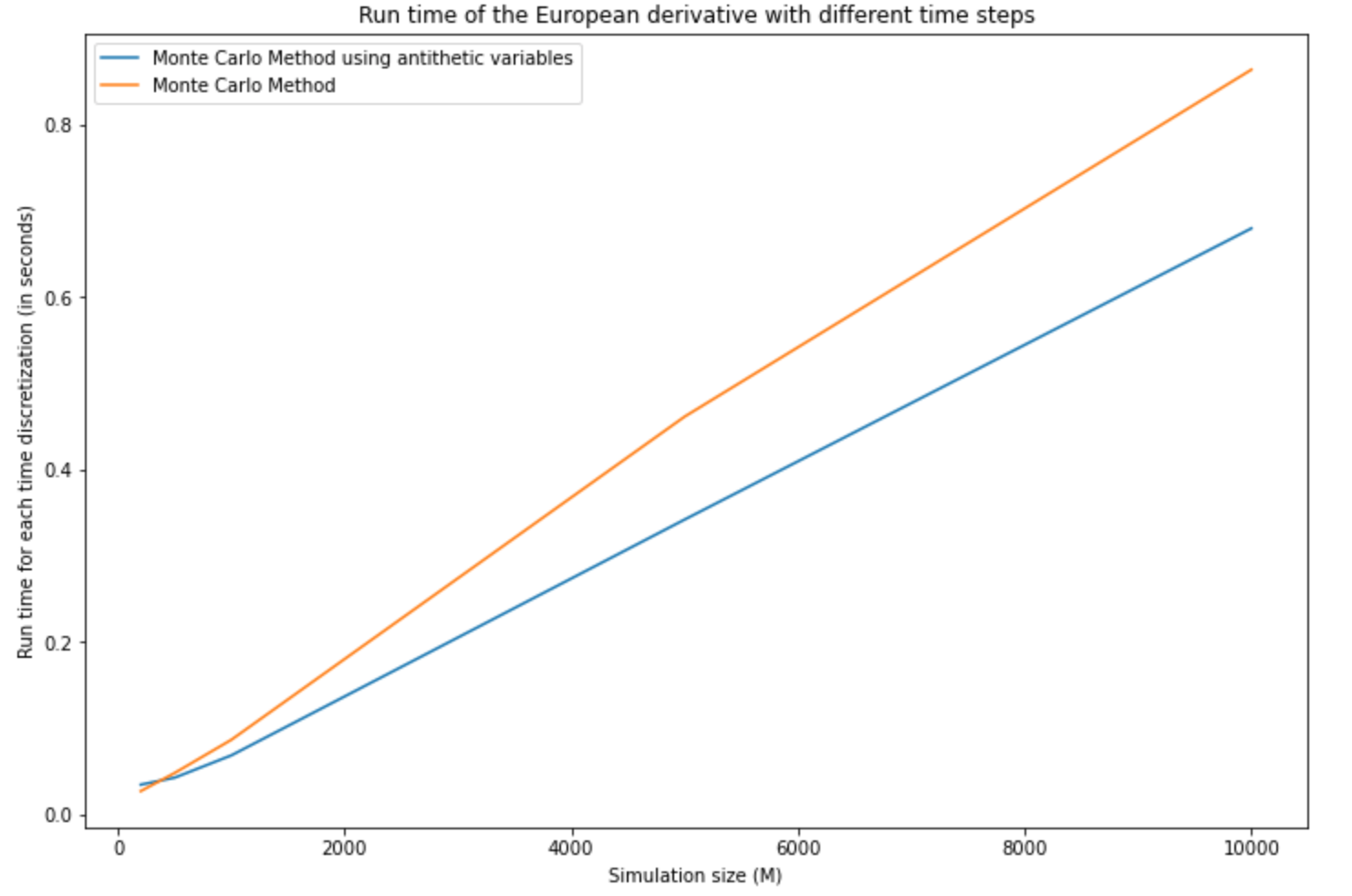
The first one shows the arbitrage free initial prices of the European derivative with different methods. We can see from the chart that as the simulation size increases the the gap between the prices that both methods obtain gets narrower.



The second chart shows the variance of the European derivative that the two different methods get. It’s quite obvious that the AV method reduces the variance and makes our estimation more accurate. And as the simulation size increases, the accuracy of the original method increases and comes close to the results of the AV method. So we can conclude that the AV method does improve the accuracy of the estimation. But this advantage would get weaker as simulation size increases.



The third chart shows that the AV method also has an advantage of computing time over the original MC method as the advantage would becomes more significant as the simulation size increases.



**Contribution Table**

|  |  |  |
| --- | --- | --- |
| **Member Name** | **PID** | **Contribution** |
| Haoning Qiu | A59011030 | Describe in words how you would use the binomial model approximation to the Black-Scholes model to organize the thought and mapping for two pictures for Binary tree for part(iii).  Improve the code of part(iv).  Organize the part(iii) into PowerPoint report. |
| Xiangji Kong | A59006307 | Write additional code in Jupyter notebook for testing the performance (run-time and convergence of the initial arbitrage-free price) of the Monte Carlo and non-recombining binomial algorithms for European pricing derivatives.  Summarize the part (v) insights in PowerPoint and the report. |
| Xuefeng Yang | A59005146 | Write the code for part(vi) to testify the improvement of accuracy of BS model after introducing antithetic variables and to compare the result with the result of scheme from part(ii) in the form of graphs. |
| Ying Zhang | A59000545 | Write the description for the antithetic method and explain the code and reason why we take the antithetic method. Analyze the results obtained and the improvements of AV method. |
| Zhenghua Ye | A59006278 | Write the code in jupyter notebook to implement the scheme described part (iii).  Modify the code in part (ii).  Create dataframes in the code of part(v) and part (vi) to improve the visibility of test results. |
| Zhuoer Jin | A59010948 | Describe in words how you would use a Monte Carlo method to approximate the initial arbitrage free price of this European derivative.  Write the code in jupyter notebook to implement the scheme described part (i). |