**Question 3**

**EXECUTIVE SUMMARY**

What your report is about,  
What you did,     
why you did it i.e why is it interesting   
how you did it   
Observations and results  
Conclusion and recommendations

**INTRODUCTION**Introduce vba and computer programming in general   
Show how computer programming and  
Introduce VBA

**VBA MACROs**

Introduce the macro scenarios in real life and   
modeling strategy/Introduction to the model   i.e  what it does and how it works

Special features like user forms, msgboxes, loops, if else statements  etc,  
How to test the model,  
limitations or weaknesses of the model takes too long time(with many lines), need to draw a chart every time  
uses of the model – who uses the model and how frequent   
*Plus any other analysis that would make the user understand the advantages of your VBA model better. This is why you are allowed to make any assumptions that would make your modelling more interesting***.**

Assume that you invest in equity. You will wonder how much You'll wonder how much profit you can make, and how much loss we'll lose if things get bad.

The purpose of this model is to visualize the price path of a stock using the Monte Carlo style simulation. Monte Carlo simulation refers to a methodology that solves a problem by generating random situations with random numbers. The price of the equity used in this model is based on Brownian motion.

(1)

S(T) is the price after T period, in this case a year after which is 250 business in this case. S(0) is the current price, r is expected return, σ is the volatility, and Z is the standard normal random variable with mean 0 and variance 1.

With this model, the investors can presume the future stock price based on given current price, expected return, volatility and the simulation step size. It will show the price path from now to a year later. Users can easily get the current price of the stock. By Using historical data such as ROE, Users can calculate the expected return, and can get a volatility as well from the notion of the standard deviation. Users can get other variables from the market data, but they have to choose themselves how many times they will simulate the model.

If the user input the simulation step size 10, it will operate the simulation 10 times. I used For/Next loop to build the 10 times simulation. I used simulation step size variable t and outer loop in the code to take a simulation step size that the users write.

Each simulation runs 250 times calculation in inner loop. In first iteration, the price S(T) would be the current price so I used If/Then function to use current price in the first iteration in the inner loop. From the second iteration, the loop calculates two factors. First, it calculates Z-value.

[1] rdraw = Application.NormSInv(rnd)

Using above code, it is able to draw a random number from the inverse of cumulative standardized normal distribution. Within inner loop, this statement inputs the random number between 0 and 1 in the rdraw variable which can be used as Z-value. To show the expected price in each day, I used the below code.

[2] Cells(28 + i, 2 + m) = Cells(27 + i, 2 + m) \* Exp((r - 0.5 \* (v ^ 2)) \* (1 / 250) + v \*

\_ Sqr(1 / 250) \* rdraw)

I converted the Brownian motion statement to VBA code. This code estimates the price of day ‘t’ using the price of the day ‘t-1’. The time increment of the code is one day, so I used 1/250 of period T in above Brownian motion (1). The old price which is the price of day ‘t-1’ is saved in the (27 + i, 2 + m) cell. The new price will be saved in the (28 + i, 2 + m) cell which is one cell below the previous price. After operating the inner loop for 250 times, it will input new simulation results in the next column, if the user’s step simulation variable is larger than 1.

To make a model, I set default value for the variables that I mentioned above but let users can change the variables on the spreadsheet. When the user click the ‘simulation’ button, the model will operate simulations according to the simulation step size that the users enter and it will show the each simulation results. The user can see the figures are changing according to the dates. It is hard to presume how the price will change only with the numbers, so I draw the line chart.

[3] Worksheets(“Question3”).Shapes.AddChart.Select

[4] ActiveChart.SetSourceData Source:=Range("B28").CurrentRegion

In code [3] Addchart function is used to draw the chart in the “Question3” sheet. (John GreenStephen, 2011) Drawing the chart, it is important to choose the source. In code [4], It choose the region that the cell(B28) belong to. I choose the cell(B28) because it is the cell which the price of first day of first simulation saved, so it can be recognized as the region regardless of what simulation step size the user have chosen, if it is not zero.

The model will generate the chart and data table every time the user clicks the ‘simulation’ button. I could insert the chart in the ‘insert’ tab in the Excel spreadsheet, but I decided to generate chart with VBA because the model can cope with various situation when I use VBA. For example, if I used an already-made-chart with the maximum band 20 and minimum band -10, the chart cannot show the price path properly, when the user input the current price larger than 20. In addition, if I used already-made-chart which has already-selected region, the chart cannot select the region properly after the simulation step size has changed. This method has the disadvantage of taking more time operating the model, but it is user friendly that it gives more choice to users and they don’t need to select the region every time they operate the model.

As the model, generate the chart and data table every time the user operating the model, it is needed to delete the exiting chart and data table. I used below codes.

[6] Rows(27 & ":" & 278).Select

[7] Selection.Clear

[8] Worksheets("Question3").ChartObjects(1).Delete

In code[6], it selects the rows from 27 to 278 where the data table belong, and delete the data in the selected rows. In code[8], it deletes the existing chart in the spread sheet named “Question3”.

Users can use this model before buying the equities, presuming the price after a year and the best or the worst scenario that users can encounter during the investment.

**CONCLUSION & RECOMMENDATION**Summarize key points and  
Recommendations i.e action the customer should take

# References

Glasserman, P. (2013). Monte Carlo Methods in Financial Engineering. In P. Glasserman, *Monte Carlo Methods in Financial Engineering* (p. 596).

John Green, S. B. (2011). Excel 2007 VBA Programmer's Reference. In S. B. John Green, *Excel 2007 VBA Programmer's Reference* (p. 1176).