



## WSC – Quant Wing Induction Assignment

Hey!

Glad to see that you're interested to join the Quant Wing.

Before proceeding, please fill up the following form, we would like to use this for simple data collection.

<https://forms.gle/HedrVaD2UHm3mbHJ9>

Please attempt this assignment even if you are interested in **Competitive Coding, Machine Learning/Deep Learning or Finance** in general. All these fields are important for Quantitative Finance!

# Why should you attempt this assignment?

1. As a part of the Quant Wing, you will get to work on researching and developing trading strategies and automating them using python and cloud servers - **Algorithmic trading**. You will also be working on various **research projects** with the rest of the wing. Projects like these will add a lot of value to your resume.
2. **Micro challenges** throughout the year to help you polish your skills.
3. You will get to **network** with fellow like-minded people with similar interests and career paths which will help you grow yourself.

# Guidelines

1. You will probably find a lot of unfamiliar concepts in the questions. However, that should not stop you from attempting the questions. We will try to link as many resources in the questions as we can to help you out, however remember to **Google your queries**. Even if you have **0 knowledge**, please try and **google everything** and you will understand.
2. **No need to attempt and submit all 3 questions.** Even if you have completed 10% of the whole assignment, please do make a submission. This is how everyone starts out.
3. Try and not copy each other's code. It is fine if you use online resources and refer to code available online but make sure you understand each and every line you are writing.
4. There are 3 questions with different weightage each. Make sure to read all 3 and implement them in Python.
5. **Participation in the WorldQuant Alphathon is compulsory** for getting inducted into the wing, so make sure you at least start by registering for the competition at [platform.worldquantbrain.com](https://platform.worldquantbrain.com)
6. A few free APIs that you can use to download stock data are: Yahoo Finance API, Alpha Vantage, FYERS API

## Question 1

Code up a simulation of the Black Scholes Model in Python. In order to attempt, you will have to read about how it works.

Call Option price

$$C(S, K, \sigma, r, T, \delta) = Se^{-\delta T} N(d_1) - Ke^{-rT} N(d_2)$$

Put Option price

$$P(S, K, \sigma, r, T, \delta) = Ke^{-rT} N(-d_2) - Se^{-\delta T} N(-d_1)$$

where

$$d_1 = \frac{\ln(S / K) + (r - \delta + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}} \quad \text{and} \quad d_2 = d_1 - \sigma\sqrt{T}$$

S: Spot Price, K: Strike Price,  $\sigma$ : Volatility, r: risk-free rate,  $\delta$ : dividend (continuous rate)

(NOTE: You can ignore  $\delta$  for now by taking  $\delta=0$ )

Take hypothetical prices and rates and simulate the Black-Scholes model by pricing a call and a put option.

## Question 2

Come up with a simple trading strategy and back test it in python using technical indicators. Check out the following link for more info on technical indicators:

<https://www.investopedia.com/terms/t/technicalindicator.asp>

Feel free to choose the ticker and the timeframe that you would want to back test your strategy on.

Here is an example of a backtester:

<https://medium.com/codex/algorithmic-trading-with-relative-strength-index-in-python-d969cf22dd85>

Your strategy will be judged on innovation and scalability besides obviously the Profit and Loss Graph. (You are free to develop and back test Machine Learning/Deep Learning based strategies as well)

NOTE (optional): You can calculate and output the following metrics after your back test:

Win % - Profit making trades/ total trades

Loss% - Loss making trades/ total trades

Average annual returns

Maximum drawdown

Have a look at the following link for more details:

<https://medium.com/auquan/backtesting-basics-understanding-your-key-metrics-ed902c24c702>

### Question 3

This problem is the first step of a pairs trading strategy.

Choose any 10 stocks (of your choice) and form all possible pairs (90 pairs), analyse their historical returns and output those pairs that have stationary spreads.

Stepwise Process:

- 1) Download data for every stock.
- 2) For every pair, run linear regression (OLS) between the 2 stocks
- 3) Calculate the residuals (aka spreads) of the model.
- 4) Apply stationarity tests like AD Fuller test on these residuals/spreads.
- 5) Store and output those pairs that have stationary spreads (take a significance level of your choice)

**NOTE:**

- a) Check out the statsmodels module in python for the above.
- b) Terms that you will need to google are: stationarity, co-integration, AD Fuller test, OLS

# Submission Instructions

Create a GitHub repository called Quant-Wing-2022 and fill up the following form: <https://forms.gle/SMG4LBb9ZgMpwgA66>

Also send an email with the subject Quant-Wing-2022\_FULLNAME to [wsc.bitsgoa@gmail.com](mailto:wsc.bitsgoa@gmail.com) and attach your code files with it.

If you are not fluent with GitHub, just fill up the form and send an email.

In case you face any difficulties, feel free to contact

Omkar Kulkarni (Quant Head, WSC):  
<https://wa.me/918588041643>