Entry Technical Test  **Junior Quant Analyst (C++ & Python) – Risk Systems**

Chenavari Investment Managers

London, England, United Kingdom

## Introduction

In this test, you are encouraged to use the AI model you are most comfortable with. Please let us know which AI model you are using and, whenever possible, share the specific questions you asked it to answer.

## Evaluation criteria

Accuracy | Code cleanliness, conciseness, learnability, efficacy (simpler is better) | Human insight | Critical thinking.

## Question 1: OCR

The provided images *behltreu.png* and *ahwm5.png* (see contain historical prices for the bond index **BEHLTREU** and the future **AHWM5**, respectively. AHWM5 is a future on BEHLTREU expiring on **20 June 2025**. We expect a correlation between the two sets of historical prices.

Write a simple Python script for Windows that:

* Uses tesseract.exe and the Pillow package (or a suitable alternative OCR tool that runs on Linux);
* Performs OCR on the two provided images behltreu.png and ahwm5.png. If the tool does not perfectly parse all dates, you may manually add missing values;
* Cleans the data manually or using Python to produce a final CSV with three columns:  
  date, bond index price, future price.

## Question 2: modeling stress scenarios

We are interested in the variation of the future's price relative to movements in the underlying bond index. Define a 3-business-day (3BD) stress as follows:

Using **Python or Excel (whichever you prefer)**:

* Model the 3BD variations in the future's price as a **linear function** of the 3BD variations in the bond index price; please provide script, plots, coefficients.
* When plotted, the stress values of the bond index (x-axis) versus the future (y-axis) should approximate a straight line.

**Bonus**:

* Model 7BD and 14BD variations in a similar way.
* Generalize the linear model to allow analysis for any time period that makes sense. Feel free to comment, compare, and visualize results.
* Implement and compare a more advanced model in Python, such as a **polynomial regression** or **machine learning approach**, and compare it with the linear model.

## Question 3: Recursive impact on funds exposure

Chenavari manages multiple funds, some of which hold equity stakes in other internal funds. The ownership structure is as follows:

|  |  |  |
| --- | --- | --- |
| **Fund A** | **Fund B** | **Percentage of Fund B owned by Fund A** |
| Alpha | Beta | 60% |
| Alpha | Epsilon | 20% |
| Beta | Delta | 10% |
| Zeta | Theta | 35% |
| Beta | Gamma | 15% |
| Alpha | Delta | 20% |

### Part A

Copy this table into a CSV. Write a **C++ program for Linux** that:

* Reads the CSV;
* Builds a **directed acyclic graph (DAG)** using smart pointers to represent fund ownership;
* Defines a recursive function *Ownership(X, Y)* that returns the percentage of Fund **Y** owned by Fund **X**. For example: *Ownership(Alpha, Delta) = 0.6 \* 0.1 + 0.2*

This solution should work with **any similarly structured CSV**.

### Part B

|  |  |
| --- | --- |
| **Fund** | **Present Value (today) of AHWM5 in Fund** |
| Alpha | £ 300,000 |
| Beta | £ 250,000 |
| Gamma | £ 100,000 |
| Delta | £ 1,000,000 |
| Theta | £ 500,000 |

On **May 14**, the prices for AHWM5 and BEHLTREU were **300.38** and **299.4425**, respectively. On the same day, most funds held a position in the AHWM5 future. Assume that each fund's present value (PV) in AHWM5 is **proportional to the price**.

Copy this table into a CSV. Extend your C++ program to:

* Use the previously built ownership graph to compute the **Total PV of AHWM5** in each fund;
* This should be calculated recursively using the Ownership function.  
  For example: *TotalPV(Alpha) = 300,000 + Ownership(Alpha, Beta) \* 250,000 + Ownership(Alpha, Gamma) \* 100,000 + …*

**Bonus**:  
Using the May 14 data and assuming a **3BD stress move of +2%** in BEHLTREU, compute the **change in TotalPV** for each fund between **May 14** and the predicted value on **May 19**.