A logo for college computing

Description automatically generated

|  |  |
| --- | --- |
| *Student Full Name* | Sara Arellano Arteaga |
| *Student Number* | 2024216 |
| *Module Title* | Algorithms & Constructs |
| *Assessment Title* | System Modelling & Build |
| *Lecturer/Supervisor* | Muhammad Iqbal |
| *Assessment Due Date* | 10th of May 2025 |
| *Date of Submission* | 10th of May 2025 |

**Declaration**

By submitting this assessment, I confirm that I have read the CCT policy on academic misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source.

I declare it to be my own work and that all material from third parties has been appropriately referenced.

I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

**Hospital Employee Management Program Report**

In the present report I will explain the choices for sorting and searching algorithms that I have used in my program. First of all, the institution that I have chosen is a “Hospital”, honestly because is the most familiar to me without the necessity of doing any research. The tasks were: sort, searching, add new objects and generating random data.

**Sorting algorithm:**

To sort the names, I used the **Recursive Merge Sort**, it is a popular sorting algorithm known for its efficiency and stability. It follows the divide-and-conquer approach. It works by recursively dividing the input array into two halves, recursively sorting the two halves and finally merging them back together to obtain the sorted array.

I have chosen Recursive Merge Sort method because is faster than simple algorithms like bubble, especially when the number of employees grows. Merge sort preserves the relative order of records with equal keys, which is useful when sorting employees with the same name or department.

**Searching algorithm:**

To search I have used the **Binary Search Algorithm**, in base to the class notes first the arrays must be sorted and the use of Binary is faster than Lineal. The binary search gets its name because the algorithm continually divides the list into two parts, always look at the center value and each time you get to discard half of the remaining list.

**Random employees generation:**

A utility method creates objects with randomly selected data. Each employee data is generated by combining randomly picked first names and surnames from predefined arrays, also from two different Enums take the Manager Type and the Department. We also had worked in class with it.

**Enums:**

The enums have been used to define fixed sets of constants. Like in Menu, Department and Manage. The main benefits of using it in my program are:

* Prevents invalid values
* Makes switch-case and selection logic easier to implement.
* Improves type safety and code clarity.

**Conclusion**

The combination of random employee generation, well-structured Enums, and efficient arrays allows the program to simulate a functional hospital employee management system using methods like the **Recursive Merge Sort** and **Binary Search**.