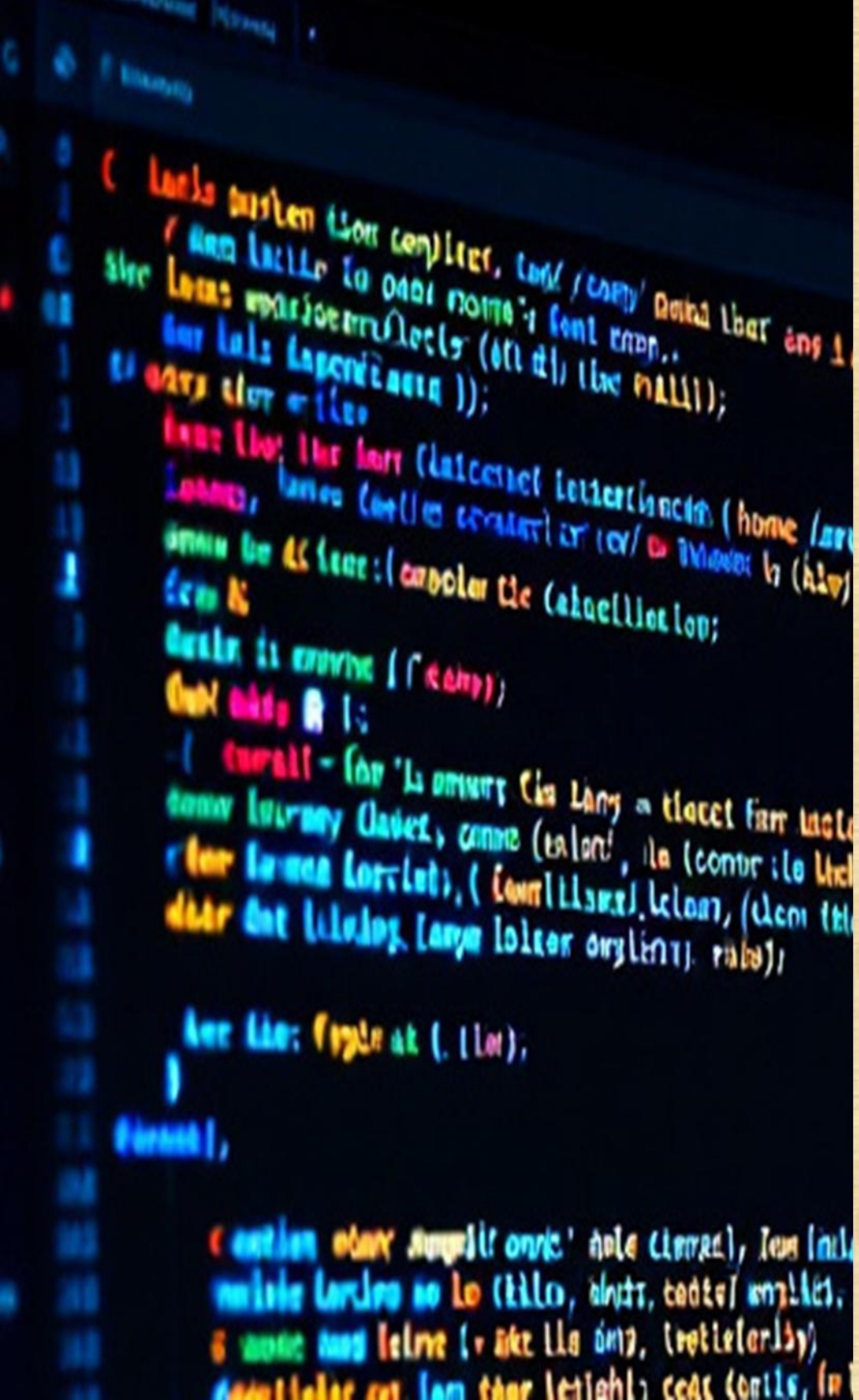
LIBRARY MANAGEMENT SYSTEM

- **Group No:** **07**
- **Members:**
  EG/2022/5003 : Dilhara K.K.V.R
  EG/2022/5113 : Jayawickrama J.A.N.D
  EG/2022/5290 : Rathnayake R.M.A.D
- **Objective:** To design **Library Management System** to maintain Book data and Member data for librarian by ***Adding, Removing, Searching and Sorting both Books and Members***. Furthermore, it allows ***issuing and returning books*** while ***updating number of copies*** and ***availability***.
- **Key Features:**
  - User-Friendly Console based Interface
  - Easy Sorting with different parameters
  - Easy Searching with different parameters
  - Auto Saving and Loading data sets through the text file permanently



Used Data structures and Algorithms

Data Structures

- **Dynamic Arrays** for store books data
- **Linked Lists** for store members data

Searching Algorithms

- **Linear search** for searching by Title and Author
- **Binary search** for searching by ISBN number

Sorting Algorithms

- **Quick sort** for sorting by Title and Author
- **Merge sort** for sorting by Year

- ❑ Quick sort and merge sort used for get faster performance than Bubble sort when using large number of data sets.
- ❑ Linear search doesn't need to sort the data, so we used linear search for Title and Author, because it is complex to sort these strings.
- ❑ Binary search is more faster than linear search in large data set. But it need sorted data sets. Therefore we used it for ISBN number.

Performance Comparison

Execution Time Analysis

We used Bubble Quick and Merge sort for check the execution time and noticed that bubble sort took significant delay to execute in large data sets.

Theoretical Analysis

According to the Big(O) Analysis it shows that both quick sort and merge sort take **n log(n)** time to execute while bubble sort get $\mathbf{n^2}$ time it is higher than other sorting methods.