GROUP MINI PROJECT

At the end of mini project, you should be able to:

- Construct a computer program by applying suitable data structures and algorithms.
- Develop good relation, interact with other people and work together in an effective way with your team mates to achieve the same objective.

Instructions

Each group must SUBMIT both parts; Programming and Report. The Programming SHOULD use C++ language ONLY. Plagiarism will NOT BE TOLERATED; if detected; the penalty may be severe and may lead to failure to this project. **The due date to PRESENT AND SUBMIT both parts is on WEEK 14 during your class/lab sessions.**

Programming

- The theme for the project is developing an information system. You may choose either a **booking** system, a **reservation** system, an **ordering** system, an **inventory** system, a **rental** system OR a **membership** system. Each group should decide a **unique PROJECT TITLE**. For example, Cinema Ticket Booking System is only for one group.
- The system should have following functions as a MENU list:
    1. Add item (new item into a list using queue concept)
    2. Process item in (1) (processed item into a new normal list)
    3. Display item (based on the new list in no. 2 in ascending order using simple sort technique)
    4. Exit

Sample MENU for Cinema Ticket Booking System.

```
*** WELCOME TO CINEMA TICKET BOOKING SYSTEM ***
_____________________________________________

1. Book a ticket
2. Process the booking
3. View all tickets
4. Exit

Enter your choice:

Exit code: 0 (normal program termination)
```

- Create a complete program where it **SHOULD APPLY** the concept of **class and objects** and **array of struct**.
- Make sure your program has **NO ERRORS** and able to **RUN** as per intended output.

Report

The report is actually the **COMMENTS** in your program:-

1. **Main program file:**
    a. At the top of the file, write your project title, the team members' names and matric numbers.
    b. Above the main function, describe the flow of your main program.
2. **Interface class file:**
    a. At the top of the file, write your project title, the team members' names and matric numbers.
    b. Above the class definition, describe the project class.
3. **Implementation class file:**
    a. At the top of the file, write your project title, the team members' names and matric numbers.
    b. Above each function header, describe the function process.

Rules & Regulation

Teams: **Work in a PAIR/TRIO**. It is expected that each team member contributes and shares balanced workloads to the success of the project. Any form of unfairness between team members, such as the intention to cover the other member's work, is considered a violation of the honesty policy and will cause ALL members to receive ZERO (0) mark.

Bonus: Highly encourage between each team to have UNIQUE program – NO SIMILARITY or ALIKE with others group. To get bonus points, you need to justify that the overall ALGORITHM ANALYSIS for the algorithm you have applied. Also, applying LINKED LIST concept in your program would be great. A fancy GUI may be another possibility, but the functionality is CRUCIAL. After all, we are seeking for better programming techniques in this class, not just better GUI.