



GNG1106 - Lab 8

Objectives

1. Practise the use of structure-typed variables and arrays, pointers to structure, and pass-by-value and pass-by-reference functions involving structure.
2. Master the fundamentals of sorting and the BubbleSort algorithm
3. Write program modules (functions) satisfying specifications. This is a skill required in collaborative software development.

Introduction

A class of students are evaluated by a set of assignments, a set of in-class exercises (ICEs), a set of labs, the midterm exam, and the final exam. A student's marks in these graded items as well as his/her average assignment mark, average ICE mark, average lab mark and the overall mark are encapsulated into a structure called "STUDENT_REC" (standing for "student record") together with the student's ID. In this lab, you will complete a partially written program so that

- it completes the students' record by computing their assignment average, ICE average, lab averages and overall grades,
- it allows the user to sort the class records using BubbleSort, where the sorting can be based on any single graded item, any average mark (of assignments, ICEs, or labs), the overall mark, or the student ID. It should also allow both ascending sort and descending sort, and
- it allows the user to print the class records.

A code template is given in the attached file where **adequate details and additional instructions are given**. You must implement all functions according to the instructions. **Feel free to design additional functions if you need.**

Instructions

Pre-Lab Submission (20%): Ensure you have submitted your pre-lab before attending the lab session.

Deliverable 1 (10%): Implement function

```
void printRec(STUDENT_REC);
```

Deliverable 2 (10%): Implement function

```
void printClassRec(STUDENT_REC *, int);
```

Deliverable 3 (15%): Implement function

```
void completeClassRecords(STUDENT_REC *, int);
```

Deliverable 4 (20%): Implement function

```
void bubbleSortClassRecords(STUDENT_REC *, int, int, int, int);
```

Deliverable 5 (25%): Implement function

```
int shouldAPrecedeB(STUDENT_REC *, STUDENT_REC *, int, int, int);
```

Submit only the completed code (one file only, including all deliverables). No need to submit screenshots of execution results.

Check out and Submission

You must check out with your TA before submitting the deliverables. During the check out, your TA may inspect your work and to-be-submitted deliverables, and ask you questions to further check your understanding. At the end of the check out, your TA will give you an initial mark for the in-lab component of this lab and let you know. You must then submit the deliverables before the due time of this lab. While this initial mark is likely to be the final, your TA reserves the right to reduce this initial mark after checking more carefully the deliverables you submit

Grading Criterion

- **Correctness (80%):** Correct syntax, logic and execution.
- **Style (20%):** Descriptive naming, appropriate indentation, ease of reading.