Intro to Programming

Team Project Specification – Semester 1 2022

This project is worth **30%** of your total marks for this module.

Teams will consist of three students.

All members of each team are expected to make an equal contribution to the project and participate in all aspects of it, including problem development and analysis, coding and documentation. Teams should work together when possible and class time is an ideal opportunity to do this. Individual marks will be awarded where it is apparent that team members have not made an equal contribution to the project.

Note: Where it is obvious that a team member has made little or no contribution to the work effort of the project deliverables; this will result in a mark of zero.

The project solution and design document must be uploaded to Moodle by Friday **09/12/2022** no later than 5pm.

The project is based on how Arrays are handled in C++. You will be given a skeleton outline of the code during class and you need to build the code around this to complete the project. You can produce the code to meet the minimum requirements of the project, but extra marks are reserved for teams that provide additional functionality, such as error trapping and handling, etc.

The design document must include a reflective section from each student, in which you discuss how you approached this project, what your contribution was, what you learned about all aspects of the project: team work, problem analysis, coding, what worked and could have been done better, etc.

The project itself requires you to develop a menu-driven command line application to manipulate the contents of an array of 12 integer numbers, as follows:

1. **DISPLAY:** Displays all of the values in the collection to the computer screen.

2. **TOTAL:** Calculates the total of all the values in the collection.

3. **AVERAGE:** Calculates the average of all the values in the collection.

4. **LARGEST:** Outputs the largest value of all the values in the collection.

5. **SMALLEST:** Outputs the smallest value of all the values in the collection.

6. **OCCURRENCES OF VALUE:** Outputs the number of occurrences of a particular value in the collection.

7. **SCALE UP:** Multiplies each value in the collection by the scale factor entered.

8. **REVERSE:** Rearranges the contents of the collection so that they are in reverse order. (NOTE: This is not simply displaying the collection in reverse order on the screen).

9. **ZERO BASE:** Adjust all of the values in the collection so that the smallest value will be zero.

10. **EXIT:** Quits the application.

Final Deliverables:

Fri., 9th Dec.

Upload fully compiled, working version of project solution to moodle together with design document.

Note: One upload per team only