CPT109 C Programming and Software Engineering 1 – Group Project

Assessment Number	3 (group project)
Contribution to Overall Marks	15%
Issue Date	21st October 2024
Submission Deadline	15th December 2024, 23:59

Assessment Objective

This assessment aims to evaluate students' ability to develop a significant software solution to a real-world problem by working as a member of a team. Your team will be given a vague specification and is expected to deliver a software product using C programming language, which meets the specifications before the due date. This size and type of project is suitable for development in a modular format, and so teams are encouraged to devise program structures that allow various parts of the code to be developed independently by each team member. Being a team player means you are expected not only to apply the knowledge gained during the lectures, laboratory classes, and assignments to specify, design, implement, test, and document your own code, but also to cooperate with your teammates so that the whole project will be delivered on time with good quality.

Grouping

There are 404 students enrolled in this module, and you will be divided into groups consisting of 5 students. Groups will be formed in two stages as follows: Firstly, students will be given the option to choose their own group members. Students failing to form a group will then be randomly assigned to a group. Randomly formed groups will contain students with a range of abilities based on their performance in previous assignments. Each group will then develop a Planet Catalogue System following the attached specification using C programming language (C90).

Final Deliverables

Each group should submit the following (only 1 submission per group):

1. A report (PDF file), based on the Software Development Process applied throughout this semester:

    a) Problem Statement (Specification: formulate the problem generally) (10%).

    b) Analysis: interpret the vague software requirements provided by your design brief and determine a very clear specification for your software design. (10%)

    c) Design: explain how your program is structured in terms of functional blocks and describe what each block does. A flow diagram may be useful here showing how functions interconnect and what data is passed between them. (10%)

    d) Testing: explain how the program has been tested and verified. Ensure any usernames or passwords required to test the program are listed in a table. (10%)

2. All C source code (.c and .h files) and the final executable demonstration file (.exe). The source

code must be appropriately commented. Include any data files needed for execution. (50%)

3. **Contribution form** (The contribution form is part of the Assessment 3 download package). The group should agree on the percentage of contribution of each member to each section and submit one copy of the form **signed by all** members.

NOTE: Different individual contributions to the project may lead to different marks for different members of the same group. The module leader may call group members for a short oral test if necessary.

Submission Procedure

All the above-mentioned files/deliverables (report, source code (.c and .h) files, data files, executable, manual, contribution form, must be compressed into **a single file (.rar or .zip)**. **The group leader** (identified on the group formation form) must submit this single compressed file on Learning Mall using his/her account.

NOTE: Each group should only submit **ONE** copy on Learning Mall by the group leader. Make sure your report has a title page and ensure **ALL** group members' names and ID numbers are on it.

Marking Scheme

The general marking scheme is shown as follows:

Documentation	(50%)
Overall Report Quality	10%
Problem Specification	10%
Analysis	10%
Algorithm Design	10%
Testing	10%
Coding	**(50%)**
Implementation/coding style	40%
Robustness	10%

Important notes:

University late submission policy will be applied.

University Academic Integrity policy will be applied rigorously.

You can find information relating to both of these in ebridge.

General Guidelines

The project descriptions are deliberately given in the form of simple customer specifications, which (as in the real world) are incomplete and often ambiguous, rather than a set of exact functional specifications. The group members should work methodically together (as the developers in a real world software project would) to:

1. Analyse and formalise the customer specifications (at this stage, the various design choices and the software features can be subject to the group's creativity).

2. Design and decompose the functional and programmatic aspects of the problem and allocate constituent tasks to each group member. You are expected to use a top-down design which can then be modularised so that the tasks for each member can be clearly determined.

3. Implement the product with frequent meetings to report progress and decisions to each other and re-evaluate the agreed courses of action.

4. Implement test procedures, debug and correct the program. Each program module should be independently testable. Testing of each module and the program as a whole should be performed.

5. Finalise the deliverables.

The specifications are only basic, and most of the design choices should be made during your group meetings. The disambiguation of the customer specifications can be based on the student's logic and real life experience.

Assessment will be based on whether the product/program offers reasonable functionality and features (for the group size, allocated time, and project difficulty), its design quality, flexibility, robustness, software bugs, and other stated deliverables.

If the group cannot implement all of the system features mentioned, it is better to have a few features fully working without run-time crashes than none of the required features working properly due to bugs or disrupting ripple effects between modules in the project. However, the corresponding marks deduction will be applied depending on the missing features.

If any group issues arise during the project, it is important to raise these with the module leader as soon as they do so that resolutions can be found quickly. It is important to work with your group, and to share the work accordingly. All group members should be responsible for some of the coding. Individuals may not produce your own work for submission independent of the group.

Planet Catalogue System
Project Specification

Overall Description:
Your team has been hired by a group of researchers to develop a software system to catalogue planets within the universe. The main requirements from the researchers include:

Customer Specifications:
The implemented software should have the following capabilities:

- **Manage Planet Information**: Handle different attributes of planets such as name, type, size, distance from their star, and any other relevant data.
- **Manage Researcher Accounts**: Each researcher should have a unique account number.
- **Search and Retrieval**: Allow researchers to search for planets by name, type, or other attributes. Enable researchers to view detailed information about each planet.
- **Statistics and Reporting**: Provide statistics such as the number of planets catalogued, the number of discovered planets, and the number of planets observed.

System Users: The system should cater to the following user functionalities:
- Admin:
    - **Manage Planet Information**:
        - Add, edit, or remove planets from the catalogue.
        - Set or modify planet attributes and their respective details.
    - **View Statistics**:
        - Number of planets catalogued.
        - Number of discovered planets.
        - Number of planets observed.
    - **Manage Researcher Accounts**:
        - Add, edit, or remove researcher accounts.
        - View researcher activity and contributions.

- Researchers:
    - **Planet Search and Retrieval**:
        - Search for planets by name, type, or other attributes.
        - View detailed information about each planet.
    - **Account Management**:
        - Register for a new account.
        - Edit personal account details.
        - View past contributions and activities.

CPT109 Group Project
Contribution Form

Name	ID Number	Contribution (%)	Signature
1.			
2.			
3.			
4.			
5.			
Total		**100%**	