BCS 370 Fall 2020

Capstone Project Description

Project Title: `College Student Homework Management System`

Business Requirements: The system manages and displays a college student's homework/assignments in the current semester. The objective of the application is to help the student efficiently manage his time in doing his course work.

(*To minimize your efforts, I removed the class scheduling part from the system.*)

Technical Specifications:

General functional requirements:
The homework management and lookup system should provide, at a minimum, "search", "add", "update" and "remove" functionalities. The "remove" function can be replaced with marking the assignment as "Completed" if you think it makes more sense.

Detailed function requirements:

1. **Data structure**: You may choose to use any data structure(s)/container(s) you see fit for this project. STL is encouraged to use. Or you may construct a custom data structure to your needs.

2. Must develop the project using Object Oriented Programming (OOP)

3. UML diagram

4. Must have the following functionalities:
   - 4.1. Add an assignment
   - 4.2. Update an assignment
   - 4.3. Search/lookup
     - 4.3.1. Search assignments which are due in # of days //0 means today; 1 means tomorrow; …
     - 4.3.2. Search by course
     - 4.3.3. Display all assignments that are in progress
   - 4.4. At least one method should feature tail recursion

5. Design necessary attributes for the homework/assignment object. Recommend including "estimated time to complete the assignment."

General guideline:
Data serialization: It's recommended that you save all homework data into a file and retrieve data from the file to make the application more useful. Since each assignment is an object, saving the collection of assignment objects into a binary file will be the most efficient solution.

Modularization: Classes/Objects, methods, functions should be well thought and designed before any implementation taken place. (Of course, the design is likely to be updated along with the development.) A UML diagram will be required as part of the project submission.

Submission Requirements:

You may create a compressed file of all the following files to submit.

1. Team writeup
    a. Team members
    b. UML diagram
    c. Key technical implementation descriptions
        i. data structure
        ii. tail recursion
        iii. complexity of searchByDueDate() and searchByCourse()
2. Source code (Preferable a **GitHub** URL)
3. Individual writeup

    Each member should write up a one-page document to describe his contribution to the project. (must include technical contribution)
    a. Design portion (which part in the UML, data structure) if applicable.
    b. Coding portion if applicable.
    c. Describe and explain your contribution.

Rubric: TBA