INFO5100 Midterm Exam

Business Background

Imagine you are a engineer in University of California Information Systems team, and you are tasked to develop a register system to manage students' course registration, transfer, etc. The system needs to have the following functionalities:

Roles

- Student can browse Course, signup Course, drop Course, view Course schedule for each Semester, and view the Transcript for each Semester.
- Student can browse Submission of Assignment, create Submission of Assignment, delete Submission of Assignment, and update Submission of Assignment.
- Teacher can browse Assignment, create Assignment, delete Assignment, and grade Submission from Student.
- Teacher can create Evaluation rule for a Course based on Assignment.
- Course Manager can add Course, delete Course, delete Course, update Course, and browse Course.
- Finance Manager can add Rate for each Course credit, update Rate for each Course credit.
- Finance Manager can add Scholarship, delete Scholarship, update Scholarship, and browse Scholarship.
- Finance Manager can attach Scholarships to Student, delete Scholarships from Students, and browse Scholarship Assignments.

Course

- A Course will have fixed capacity when the signup count of Student reached capacity, student can no longer sign up to that Course.
- A Course will have number of credit. Students must sign up between 16 to 20 credits for each semester.
- A Course can be either online or offline, for offline Course, there must be a Location associated with it, for online Course, there must be a URL associated with it.
- A Course will have three grades: A, B, and C.
- A Course will have an Evaluation rules to calculate the final grade for Student. There are two types of Evaluation rules:
    - Points based: the Student's final grade for the Course will be based on the sum of all the points that Student get for each Assignment.

    - Rank based: the Student's final grade for the Course will be based on the rank of total points for each Assignment, then ranked against other Students.

Assignment

- An Assignment will have number of points.
- There are two types of Assignment:
    - Essay: which means the Submission must be submitted in the form of a texts.
    - Multiple Choice: which means the Submission must be submitted in the form of array of choices among: A, B, C, D.

Scholarship

- Scholarship will give Student finance benefit for sign up Course. The amount deducted can be either in the form of fixed amount or a rate. For example, $2000 or 10% of overall rate.
- There are two types of Scholarship:
    - Status based: The Scholarship can only be applied to Student with certain attribute. For example, from Canada, etc.
    - Merit based: The Scholarship can only be applied to Student with certain Grade.

Engineering Requirement

Based on the business background listed above, perform the following tasks:

- Design an application model that can support the listed cases.
- Implement the application model in Java programming language that implements the structure, relationships of the models. The classes implemented should also have corresponding functions to achieve the described tasks. For example,
- Feel free to make reasonable assumptions, but make sure document that in code in the form of Java comments.
- No database is required, if you want to persist data, use a file.
- Need to use at least one Java collection class.
- Need to use Interface at least once.
- Need to use Inheritance at least once.
- Need to use function overriding at least once.

Submission Requirement

You need to submit three items for this test:

- Model Diagram (can came from diagramming tool or as simple as a photo of your hand drawn diagram)

- Code (compressed in zip format)
- A document that answers the following question:
    - Describe the relationships of your models and functionality of each class.
    - Describe where you used Java collection class, and why you choose to use that Java collection class.
    - Describe where you used Interface, and why you choose to use Interface.
    - Describe where you used Inheritance, and why you used it.
    - Give an example of variable declaration (specify which class and which line).
    - Give an example of conditional control flow (specify which class and which line).
    - Give an example of loop (specify which class and which line).
    - Give an example of a Primitive type variable and a Object type variable (specify which class and which variable).
    - Give an example of Inheritance (specify the classes).
    - Give an example of function overriding.

Submission Method

Upload the required items to Blackboard before the deadline.

Tips

- The scope of the project is quite big, and besides the basic features, there are a lot can be implemented to make this a full fledged system. But try to focus on the MVP (minimum viable product) first, then try to expand and add features if you have time.
- Some of the entities in the model can be easily figured out, but there are many more abstract entities.

Grading Specification

The maximum points available is 100. You will be graded based on:

- Quality of the modeling
- Coding style and code quality
- The amount Java feature you used in code (such as Collection, Inheritance, Concurrency, Lambda, etc)
- Quality of work comparing to other students

Late submission will cause 10% loss of the overall points. For example, if your submission should get you 90 points, but you submitted after deadline, you will get 81 in total.