Xinyu Tang
CS 5200

(10pts) List nouns that are candidate classes or attributes:

Classes: Users(faculty, student), courses(session ,teaching course, my course), Module, Lesson, Widgets, exam, assignment and question

Attributes: Username, password, first name, last name, email, phone address, course Name, seat capacity, section number, semester, faculty name, module name, lesson name, type

(10pts) List verbs as candidate relations between classes
Faculty and student are inherited from the user class.
Session is inherited from the course class; teaching course and my course are inherited from session.
Each faculty can has(is composed of) zero or many teaching courses.
Each teaching course is composed of one or many modules. Each module is composed of one or many lessons and each lesson is composed one or many widgets(enumeration).
Each students has zero or many courses. Each course is composed of one or many exams and assignments. Each assignment or exam is composed of one or many questions.
User is associated with course.

(10pts) Generalization/specialization (inheritance, if applicable, explain) - show parts of your diagram that specifically illustrates the use of inheritance

Faculty and student are inherited from the user class.
Session is inherited from the course class; teaching course and my course are inherited from session.
All above inherited instances have same attributes and methods as their parent class and they also have their own attributes and methods.

(10pts) Associations, aggregation and/or composition, e.g., empty or filled in diamonds (1 to * or 1 to 1..*, if applicable, explain) - capture any lifecycle dependencies between classes using aggregation or composition.

User class is associated with courses class
Each faculty can has(is composed of) zero or many teaching courses.
Each teaching course is composed of one or many modules. Each module is composed of one or many lessons and each lesson is composed one or many widgets(enumeration).

Each students has zero or many courses. Each course is composed of one or many exams and assignments. Each assignment or exam is composed of one or many questions.

(20pts) Classes vs. attributes Or vice versa, where some attributes might actually be better modeled as classes. Document how the class diagram evolved from a naive first approximation to the final result. Explain your decisions and support them with relevant portions of the class diagram. Show your final class diagram as one single diagram.

At the beginning, I didn’t realize that faculty’s teaching course and student’ course can both inherit from the course class.

(5pts) Correct data types, e.g., Date, String, Integer, List, Array, Enumeration, etc.
Most of the datatypes are strings and integers. Widgets is enumeration.

(10pts) Cardinality - for every single association, show the number of instances participating in a relation

User class is associated with courses class, the diagram shows that a student user attends at least one course; and each course is taught by one teacher.

(10pts) Remove any inadequate or redundant relationships, entities or attributes (if applicable, explain) - if you identify redundant associations, entities, or attributes, explain how/why you removed it. For instance, the problem statement might have irrelevant information that you might need to ignore. Also, the text might describe contradictory or ambiguous descriptions. Finally the statement might use different terms to refer to the same thing. Make sure you make a compelling argument for your decisions to ignore a particular noun or verb as irrelevant or redundant or an overloaded term.

I include as much useful information I could in the diagram. The only things I ignore are the exam types (a simple essay assignment, a submission assignment, or an exam) and evaluation types(as essay questions, multiple choice questions, fill in the blank questions, and many more types of questions). What I can do is to put them as enumeration data type.

I also separated teaching course from student course. It’s possible to combine them together, but the relationship would be not that clear than what I shown in my UML.