# CSE 400 Final

Olcay Taner YILDIZ

## I. Question (12 points)

Explain the following architectural styles by giving examples:

- Repository
- Model/View/Controller
- Peer-to-peer

## II. Question (8 points)

What is the difference between open and closed architectures?

## III. Question (15 points)

Consider the following design goals. For each of them, indicate the candidate patterns you would consider to satisfy each goal:

- Given a legacy banking application, encapsulate the existing business logic component
- Given a chess program, enable future developers to substitute the planning algorithm that decides on the next move with a better one
- Given a chess program, enable a monitoring component to switch planning algorithms at runtime, based on the opposing player's style and responce time

## IV. Question (15 points)

In a class hierarcy, A is the parent class of B, and B is the parent class of C. B and C can not see the variable a1 of A, but they can see the variable a2 of A. A and C can see the variable b1 of B, but only C can see the variable b2 of B. A and B can see the variable c1 of C but not the variable c2 of C. What are the visibilities of the variables a1, a2, b1, b2, c1 and c2?

## V. Question (16 points)

Write Java code to realize bidirectional many-to-many association between two classes Course and Student.

## VI. Question (16 points)

Design a relational database schema for the following object model: A *League* consists of multiple *tournament*s, where a *tournament* contains multiple *round*s. A *player* can play in multiple *tournament*s and of course a *tournament* is possible only with multiple *player*s. Assume League, Tournament, Player and Round have a name attribute and a unique identifier. Additionally, Tournament and Round have start and end date attributes.

## VII. Question (16 points)

Consider the List interface in the java.util package for ordered collections of objects. Write preconditions and postconditions in OCL for the following operations:

- int size() returns the number of elements in the list
- void add(Object e) adds an object at the end of the list
- Object remove() removes and returns an object from the end of the list
- Object get(int idx) returns the object located at index idx, 0 being the index of the first object in the list.