

CSE 400 Final

Olcay Taner YILDIZ

I. QUESTION (18 POINTS)

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

II. QUESTION (14 POINTS)

Consider the following implementation of the magic() method.

```
public void magic(int N){
    if (N < 0){
        return;
    }
    while (N > 0){
        if (N % 2 == 0)
            System.out.print("0");
        else
            System.out.print("1");
        N = N / 2;
    }
}
```

By exercising all possible paths generate example test cases.

III. QUESTION (12 POINTS)

Assign architectural style names to the following definitions that are given below:

- Subsystems access and modify a single data structure.
- Subsystems are classified into three different types. First group maintain domain knowledge, second group display it to the user, and third group manage the sequence of interactions with the user.
- A subsystem provides services to instances of other subsystems.
- Each subsystem can request and provide services.

IV. 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.

V. QUESTION (20 POINTS)

Consider the following design goals. For each of them, indicate the candidate patterns (draw UML diagram for each design pattern) 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 response time
- Given a chess program, enable undoing and redoing moves in a game.

VI. QUESTION (20 POINTS)

Apply the appropriate transformations (map associations to collections) to the following object model: A *School* consists of multiple *grades* (6th grade, 1st grade, etc), where a *grade* contains multiple *classes* (6-B, 1-C etc). An *teacher* can teach in multiple *classes* the same *course* and of course many teachers teach for a class. A *student* is registered to only one *class* and can take many *courses*. Assume School, Grade, Class, Teacher, Course, and Student have a name attribute and a unique identifier.