MPPFinal Exam 11/22/13

Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ StudentId:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

There are a total of 60 points possible. Note that there are questions on the back of this page. You must be done at 12:00 Noon, so budget your time wisely to complete all the questions in 2 hours.

1. (12pts). You have just beenhired as an object oriented design expert. Your boss wants you to evaluate their current software to determine if it is a good OO Design. She asks you to give her the criteria you will use and a one sentence description of what each criteria means. List your top 4 criteria for determining a good OO design along with your one sentence descriptionof each criteria.
   * + **Any 4 of the following:**
   1. **Information hiding – enforce program to public methods that are constant while allowing the internal class implementation to change. Keep internal methods and attributes private.**
   2. **Encapsulation = strong cohesion within classes for data and methods**
   3. **Open for extension, closed for change – add new classes for additional requirements, don’t change existing code.**
   4. **P2I - Program to interface instead of program to code – makes code less fragile, and more flexible and reusable. Use abstract classes and interfaces to enforce this.**
   5. **Inheritance and polymorphism – the “is-a” relationship can be implemented with inheritance and we can call the same method on different sub-classes to get the behavior we need.**
   6. **Delegation and propagation – classes delegate methods to their associates and the solution propagates as needed for the use case.**
2. (6pts). Dynamic Binding:
   1. In Java what is the purpose of dynamic binding?
   2. Describe when and how dynamic binding occurs.
   3. Will a static method use dynamic binding? Explain your answer.
   4. Will a private method use dynamic binding? Explain your answer.

**Used with polymorphism to select the correct instance of a method from the class inheritance hierarchy.**

**Done by the JVM at run time.**

**Static methods and private methods are not inherited so they use early binding not dynamic binding.**

1. (4pts) Describe two differences between abstract classes and interfaces.
   1. **An interface has no method implementations, just method signatures. Abstract Classes may have method implementations.**
   2. **Abstract classes may have concrete attributes and instance variables defined. Interfaces can only have constant (final static) fields.**
2. (4pts) Describe two ways that abstract classes and interfaces are the same.
   1. **Neither one can be instantiated**
   2. **Classes can inherit from both. Classes “implement” zero to many interfaces.Classes “extend” at most one Abstract Class.**
   3. **Both offer a way to program to interface by abstracting out common API methods that can be used by clients.**
3. (4pts) What is the purpose of a static factory method? Give one advantage to using that option and one disadvantage.

**Classes can provide Static factory methods instead of a public constructor. It allows clients to create a new instance of the target class based on parameters passed in to the static factory method.**

**Advantages are:**

* + **Can have a meaningful name – e.g. to create a class instance with special properties**
  + **Do not have to create an object every time**
  + **Can return objects of a subtype**
  + **Can maintain associations**

**Disadvantage is you cannot create a subclass from a class without a public or protected constructor.**

We create an online MUM Java Course which is a series of Java tutorials. We allow staff, faculty, and students to become members of our Java Course. A member could be both on staff and a student, or change from staff to faculty or from a student to staff, etc.Students will pay $50 to become a member, staff will pay $25, and faculty can become a member for free. In the case of a member being a combination of types, they get the lowest price available from their combination. For example, a member who is both student and staff will pay $25.

Members must complete the Java tutorials for our course in order. They can only take a tutorial if they have completed the preceding tutorials in the course. They pay their membership fee when signing up for our course.

1. (12 pts) Create a UML class diagram for the problem description. Show the major class attributes and methods, and label your associations. Make a note of any design assumptions you make.
2. (10 pts) Create a sequence diagram for the use case of a member signing up for our Java Course.
3. (3 pts)Write pseudo-code snippets that show Member’s associations.
4. (5 pts) Write pseudo-code for the method that determines member fees to sign up for our course. Show the method signature and the method implementation.

public class Member {

//here are Member's associations

private List <MemberRole> myRoles;

private JavaCourse myJavaCourse;

private List <JavaTutorial> completedJavaTutorials;

private JavaTutorial myCurrentTutorial;

private List <TutorialTranscriptEntry> myTutorialTranscript;

class JavaCourse {

private List <Member> myMembers;

abstract class MemberRole {

private Member myMember;

class JavaTutorial {

private List <Member> myMembers;

// here is using polymorphism to find the smallest fee based

//on a Member's roles

private double getMyMemberFee() {

double fee = Double.MAX\_VALUE;

for (MemberRole role : myRoles) {

if (role.getMemberFee() < fee)

fee = role.getMemberFee();

}

return fee;

}