University of Glasgow

Friday 29 April 2016
9.30 am – 11.00 am
(Duration: 1 hour 30 minutes)

DEGREES of MSc in Information Technology, MSc in Software Development

Software Engineering (M)

This examination paper is wo s

The use of a calculator is not permitted in this examination.

INSTRUCTIONS TO INVIGILATORS

Please collect all exam question papers and exam answer scripts and retain for school to collect. Candidates must not remove exam question papers.

3. This question is about design principles and system architecture.

a) Systems should always be designed to increase cohesion. Name and provide a description for three different types of cohesion.

[6]

b) What are two techniques you might use to anticipate obsolescence during design? What is a potential drawback to using these techniques?

[6]

c) Model-View-Controller (MVC) is an architectural pattern.

i) Please provide a graphical representation of the MVC architecture.

[4]

ii) Described how MVC adheres to the design principles to **design for testability** and **reduce coupling.**

[4]

1. This question is about object oriented modelling with UML.

Scenario: A music school is developing a new software system to manage their courses, tutors, and students. The system should keep track of the name, address, and phone numbers for tutors and students. The system should also store the qualifications of the tutors.

The music school offers recurring courses on a regular schedule. Each recurring course is led by a specific music tutor. A specific course can accommodate up to ten students, and students can enrol in any number of courses. Each student is assigned a tutor as their advisor when they join the music school.

a) What design pattern should be applied to represent **Courses** in this scenario? Please provide the name of the pattern and a description of the pattern's implementation.

[6]

b) Give a UML class diagram for the proposed system. Show the relationships described and include all attributes, associations, and multiplicities for each class.

[10]

c) Please provide the Java implementation for the Student class's attributes and constru

[4]

2. This question is about *design patterns*.

a) What problems does the Player-Role pattern solve?

[4]

b) Provide an example scenario where the Player-Role pattern should be used.

[2]

c) Describe the implementation of the Player-Role pattern and provide the pattern's graphical representation.

[6]

d) The Abstraction-Occurrence pattern is related to the Player-Role pattern. What is the key difference between the semantics of these two patterns?

[4]

e) Describe an anti-pattern for the Player-Role pattern and describe why this is an anti-pattern.

[4]