00034

THE OPEN UNIVERSITY OF SRI LANKA
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

FINAL EXAMINATION 2011/2012
BACHELOR OF SOFTWARE ENGINEERING

ECI 6267 Software Architecture

Date: 3rd March 2012 **Time: 9.30 – 12.30 hrs**

Answer only four (4) questions:

1. The ISV(*Independent Software Vendor*) Infinion® wants to move their stock trading application to SAAS (*Software as a Service*) model and make a new market offering by second quarter of this year. They want to run the project in Agile mode. Engineering team analyzed their current code base and found that its heavily coupled and convoluted. Therefore they have decided to overhaul the current code base and thoroughly enforce SOLID (Single responsibility, Open closed, Liskov substitution, Interface segregation, Dependency injection) principles of Object Oriented Programming.

a.) Explain the concept(s) of "Single Responsibility" and "Interface Segregation" citing concrete example for each.

[Marks 10]

b.) Define the terms "Coupling" and "Cohesion" used in software design. Explain how these Two phenomenons are related to each other in their contexts.

[Marks 5]

c.) What are the basic software development life cycle methods and justify the use of "Agile" software development method in above engagement.

[Marks 10]

2. Systems are often redesigned not because they are functionally deficient – the replacements are often functionally identical - but because they are difficult to maintain, port or scale. "Design quality metrics" provide uniform/standard information to the Designer/Architect regarding the ability of their design to survive the change or to be re-used. There will be design trade-offs that involves losing one quality or aspect of the design in return for gaining another quality or aspect.

a.) Explain the importance of measuring the quality of a software design. Explain design quality attributes that may be used to assess a given design citing examples.

[Marks 15]

b.) What are the main design trade-offs related to "multi-threading" and "single threading"?

[Marks 10]

3. Consider the facts given below to answer the questions asked.

* Very well known adage in the industry, "The Mythical Man-Month" describes many çommonly occurring problems in large and mid-scale software development projects and breaks them down.

* There are two important principles:

1. The Mythical Man-Month : Adding manpower to a late software project makes it later.
2. No silver bullet : There is not a single strategy, technique, or trick that can exponentially raise the productivity of programmers.

* It is observed that over 90% of the costs of a typical software system arise in the maintenance phase, and any successful piece of software will be inevitability be maintained.

a.) What are the Lehman's laws of software evolution?

[Marks 8]

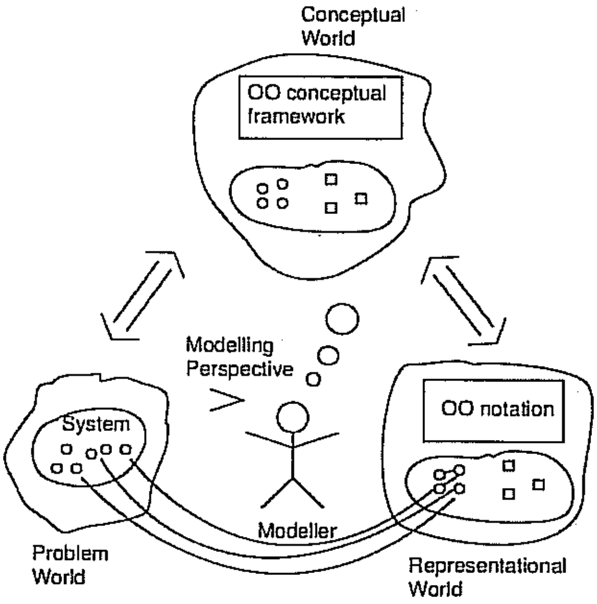
b.) What are the key elements of planning for software evolution?

[Marks 10]

c.) What are the different types of models used in software evolution? Briefly explain.

[Marks 7]

4. Modeling is very important in Software Architecture and Design. Shown below is how a Designer interacts with different worlds.



a.) What are the goals of software modeling ? Briefly explain.

[Marks 10]

b.) Briefly explain use of UML as a modeling language citing an example.

[Marks 15]

5. A software architecture defines both structure and behavior of a system. Software architecture is commonly organized in views, which are analogous to the different types of blueprints made in building architecture. A view is a representation of a set of system components and relationships among them.

a.) What are the types of architectural views that can be used to visualize a system ? Briefly explain.

[Marks 10]

b.) Explain event-driven architecture providing a real-life example.

[Marks 15]