Component-based Software Development

Date: Monday 12th January 2015
Time: 09:45 - 11:45

Please answer THREE Questions:

Answer the Question in Section A and any TWO Questions from Section B

Use a SEPARATE answerbook for each SECTION.

This is a CLOSED book examination

The use of electronic calculators is NOT permitted
Section A

1. a) In an OO framework, what is a component? Briefly discuss reusability in such a framework. (2 marks)

b) What is a component model? Why is it important? Illustrate your answer with an example. (4 marks)

c) Briefly outline the phases of the idealised component life cycle. (2 marks)

d) What is the role of a repository in the idealised component life cycle? Why is it important? (2 marks)

e) Briefly outline the W model for the development of component-based systems, and show how the idealised component life cycle fits in with the W model. (4 marks)

f) Consider the lab exercises for this course, in which you constructed an ATM (automated teller machine) system in different ways, using different component models. Based on your results for the lab exercises, discuss how well each of the following categories of component models supports the W model:

   i) component models with objects as components; (2 marks)
   ii) component models with architectural units as components; (2 marks)
   iii) component models with encapsulated components. (2 marks)
Section B

2. a) Explain how encapsulated components are different from objects and architectural units, and the merits and demerits of encapsulated components. Illustrate your answer with suitable examples. (4 marks)

b) Briefly explain the X-MAN component model in terms of:

   i) its components and composition operators; (2 marks)
   ii) its development life cycle. (2 marks)

c) In X-MAN what entities contribute to:

   i) behaviour; (1 mark)
   ii) data passing. (1 mark)

d) Consider a Cruise Control System (with collision detection) used in a car. When activated, the system can achieve a cruising speed set by the driver, by controlling the brake and the accelerator of the car.

   To activate the system, the driver selects a desired cruising speed (on a touchpad) and then presses the ‘activate’ button. Once activated, the system continuously detects other cars ahead of it and its own current speed, and issues outputs to apply the brake or the accelerator in order to achieve and maintain the cruising speed. The outputs have the range of 0 to 2, with 0 meaning no depression on the pedal and 2 meaning the pedal is fully depressed.

   At any time, the driver can also temporarily override the Cruise Control System by manually depressing the brake to slow down the car. The Cruise Control System must therefore always detect the manual brake position and bypass its brake controller when necessary. The system contains a radar, a brake position detector, a speed detector and a controller for the brake and accelerator.
i) Using the X-MAN component model, identify suitable components required for the system. For each component, list the services it provides. (2 marks)

ii) Compose the identified components using suitable connectors to implement the system. (2 marks)

iii) Identify the interface of the system. (2 marks)

iv) Specify all data channels between all services in the system. (2 marks)

v) Find one reasonably reusable composite component in your design. Specify the interface of such a composite component. (2 marks)
3. a) Explain how objects can be used as components, and the merits and demerits of objects as components. Illustrate your answer with suitable examples.  
   (4 marks)

b) Briefly explain the EJB (Enterprise JavaBeans) component model in terms of:
   i) its components and their composition;  
      (2 marks)
   ii) its development life cycle;  
      (1 mark)
   iii) the steps involved when a remote client calls an EJB component residing in an EJB server.  
      (1 mark)

c) Compare and contrast the EJB and JavaBeans component models. Focus on component definition, composition mechanism, and development life cycle.  
   (4 marks)

d) Consider the Cruise Control System in Question 2.
   i) Identify suitable components to implement the Cruise Control System in EJB. For each component, list the services (signatures) and briefly describe what they do.  
      (4 marks)
   ii) Construct the Cruise Control System in EJB using the components you identified in i).  
      (2 marks)
   iii) Use a suitable notation to express the control flow in your system.  
      (2 marks)
4. a) Explain how architectural units are composed, and the merits and demerits of this composition mechanism. Illustrate your answer with a suitable architecture description language. (4 marks)

b) Briefly explain the UML 2.0 component model in terms of
   i) the components it defines and their composition; (2 marks)
   ii) the associated development life cycle. (2 marks)

c) Compare and contrast UML 2.0 with any other ADL. (4 marks)

d) Consider the Cruise Control System in Question 2.
   i) Using the UML 2.0 component model, identify suitable components for the system. For each component, list the services it provides and requires, and summarise what it does. (2 marks)
   ii) Compose the identified components to implement the system and specify the system’s interface. (4 marks)
   iii) Describe in detail how the control logic of the system is realised. You can use e.g. a UML sequence diagram or pseudocode. (2 marks)