

NATIONAL UNIVERSITY OF SINGAPORE  
SCHOOL OF COMPUTING

SEMESTER I (AY2005-06)  
EXAMINATION FOR

**CS2103: SOFTWARE ENGINEERING**

November 2005

Time Allowed: 2 Hours

**INSTRUCTIONS TO CANDIDATES**

1. This examination paper consists of **TEN(10)** questions and comprises **twenty-two (22)** printed pages. Answer **ALL** questions.
2. Write your answers in the **blank spaces** in this answer book only.
3. This is an **OPEN BOOK** examination.
4. Please fill in your **Matriculation Number** below. Also write your matriculation number on the top right hand corner of every page.

**Matriculation Number:**

--	--	--	--	--	--	--	--	--	--

**For Official Use Only :**

	Marks
Question 1 (i) to (x) (max 10)	
Question 2 (max 10)	
Question 3 (max 10)	
Question 4 (max 10)	
Question 5 (max 15)	
Question 6 (max 15)	
Question 7 (max 12)	
Question 8 (max 5)	
Question 9 (max 5)	
Question 10 (max 8)	
TOTAL: (max 100)	

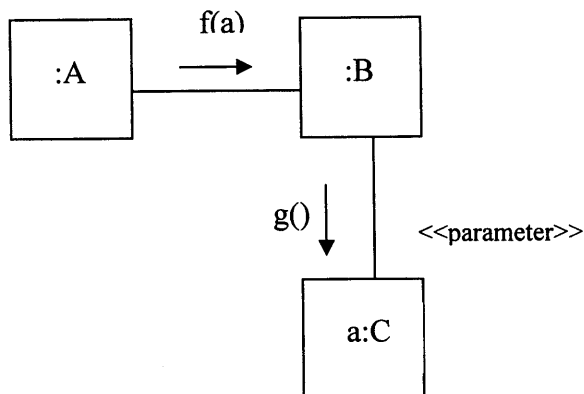
**Question 1 : It has 10 Multiple Choice Questions. Each MCQ is evaluated with 1 mark. Fill answers for these MCQ questions in the following table.**

Question	(i)	(ii)	(iii)	(iv)	(v)	(vi)	(vii)	(viii)	(ix)	(x)
Answer										

**Question (i)** When is the cost of a change , in a software development , the smallest

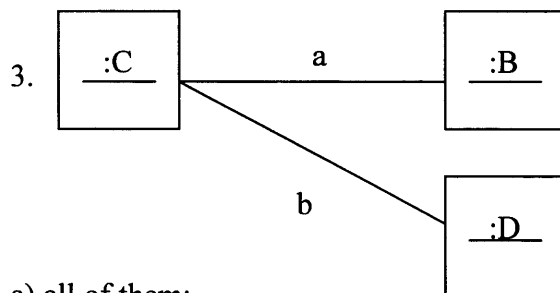
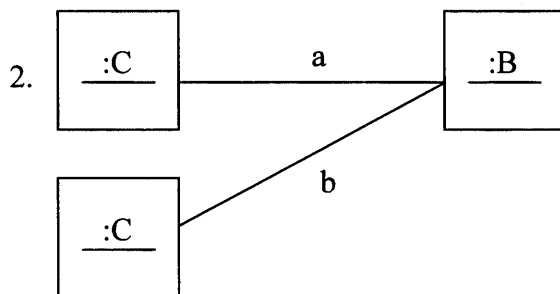
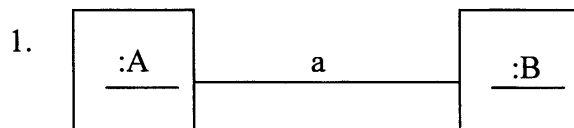
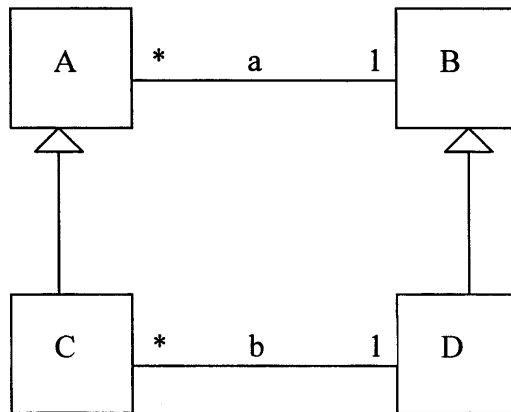
- a) During the planning phase
- b) During the design phase
- c) During the implementation phase
- d) During the testing phase
- e) None of the above.

**Question (ii).** Given the below UML diagram, which of the following statements are true?



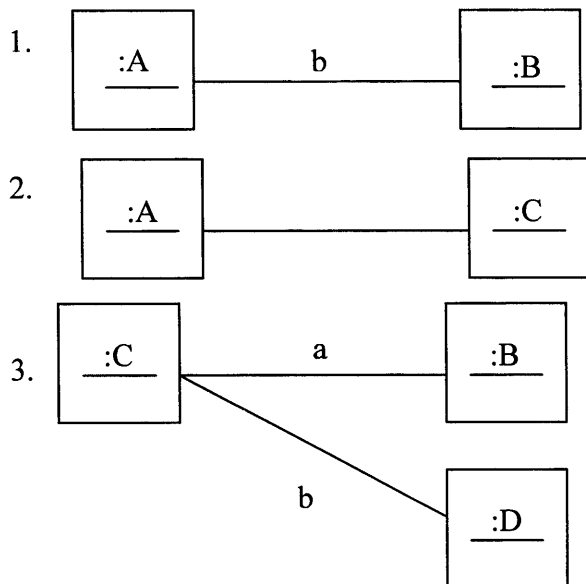
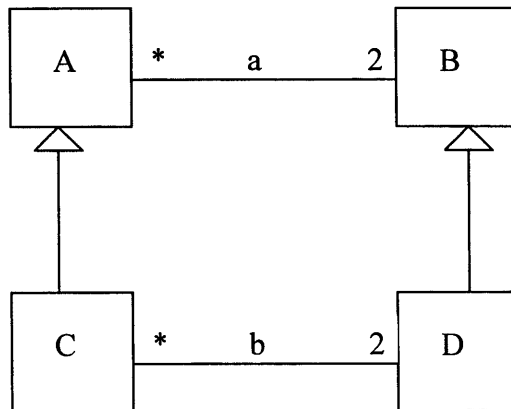
- a) The stereotype <<parameter>> is incorrectly defined in the object diagram;
- b) The message passing 'f(a)' is incorrectly defined in the class diagram;
- c) The generalization relationship between B and C is incorrectly specified;
- d) Object 'a' is passed as a parameter of a message;
- e) None of the above.

**Question (iii)** Given the below UML class diagram, which of the following objects diagrams are valid?



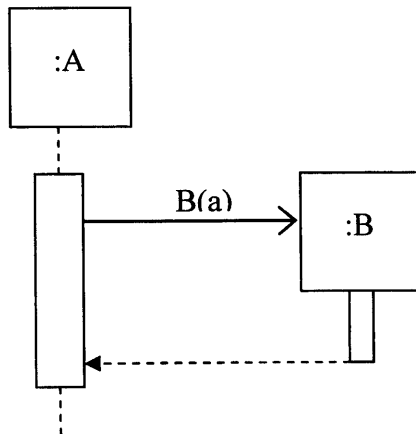
- a) all of them;
- b) only 1 and 2;
- c) only 3;
- d) only 1 and 3;
- e) None of the above.

**Question (iv).** Given the below UML class diagram, which of the following objects diagrams are valid?



- a) all of them;
- b) only 1 and 2;
- c) only 3;
- d) only 1 and 3;
- e) None of the above.

**Question (v).** Given the below UML diagram, which one of the following statements is true?



- a) The above diagram is wrong because the classifier role 'B' has no lifeline clearly drawn;
- b) The above collaboration diagram cannot show the message passing;
- c) The above diagram shows that an object of class 'B' has been created and activated;
- d) The above diagram clearly specifies that 'B(a)' is an operation of class 'A';
- e) None of the above.

**Question (vi).** What is the difference between the interaction diagrams and statecharts?

- a) Interactions diagrams specify the behaviour of more objects than statecharts can do;
- b) Interactions diagrams are the same thing as class diagrams, whereas statecharts do not specify all the possible behaviours of objects;
- c) Interactions diagrams show behaviour in particular interactions, whereas statecharts summarize the overall behaviour of objects;
- d) They are basically the same thing;
- e) None of the above.

**Question (vii).** How can we eliminate the non-deterministic transitions from statecharts?

- a) This is not possible;
- b) By adding guard conditions to transitions, all of them are evaluated to true;
- c) By adding guard conditions to transitions, at most one of them is evaluated to true;
- d) By adding to all transitions from a given state the same guard condition;
- e) None of the above.

**Question (viii).** What is true regarding transitions, entry and exit actions for statecharts?

- a) Transitions actions are performed when an object arrives at the state;
- b) Internal transitions can change the state, whereas entry and exit actions must be completed before the object can respond to any events.
- c) Entry actions are performed when an object arrives at the state, and exit actions are performed whenever an object leaves the state;
- d) All three kinds of actions are basically the same thing;
- e) None of the above.

**Question (ix).** What is true regarding internal and completion transitions for statecharts?

- a) Both internal and completion transitions can trigger when an event occurs;
- b) Both internal and completion transitions can change the active state;
- c) Internal transitions are executing all the entry and exit actions;
- d) When only one candidate event occurs, the completion transition is triggered;
- e) None of the above.

**Question (x).** What is true regarding composite states for statecharts?

- a) If a composite state is active, at least two of its sub-states must also be active;
- b) An event that is detected while an object is in a composite state cannot trigger outgoing transitions from the currently active sub-state;
- c) Unlike simple states, composite states cannot have activities, entry and exit actions;
- d) Composite states cannot have final states;
- e) None of the above.

**Question 2** (10 marks) From the following description, describe the use case, by identifying the actors, basic course of events and possible alternatives.

“In order to do the recruitment, some companies may pay (for simplicity, by credit card) for a job posting. Once published, their announcements may be viewed by job seekers.”

(Space for answer of question 2.)

(Space for answer of question 2)



**Question 3** (10 marks) From the following description, identify the main objects and their links. Draw an object diagram that shows these objects and links. Use the object diagram to draw the class diagram. Be sure to indicate the multiplicity, roles and name of each association. Justify your choices.

“Sichuan, Harbin, and Guandong are all provinces in China. Singapore, Malaysia, and China are all countries in Asia. Beijing is the capital of China. Kuala Lumpur is the capital of Malaysia.”

(Space for answer of question 3.)

(Space for answer of question 3.)

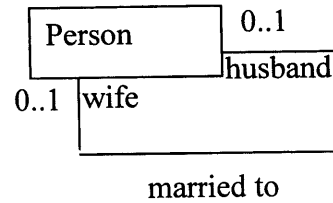
**Question 4** (10 marks) Draw a sequence diagram for the following scenario:

“The lecturer uploads the slides to the IVLE. The IVLE stores the slides files on a local folder and sends emails to all students of a class and SMSs to whose students who applied for this service. After the successful email broadcasting, IVLE acknowledges the lecturer that the students have received the email announcement. After getting the reply from the mobile phone company that the SMSs were successfully sent, IVLE acknowledges the lecturer that the students have received the SMSs announcement.”

(Space for answer of question 4.)

**Question 5** (15 marks)

Examine following class Person along with the association 'married to'. Provide full implementation of the class with constructors, accessors(get method), mutators(set methods), and any additional methods to effect the association.

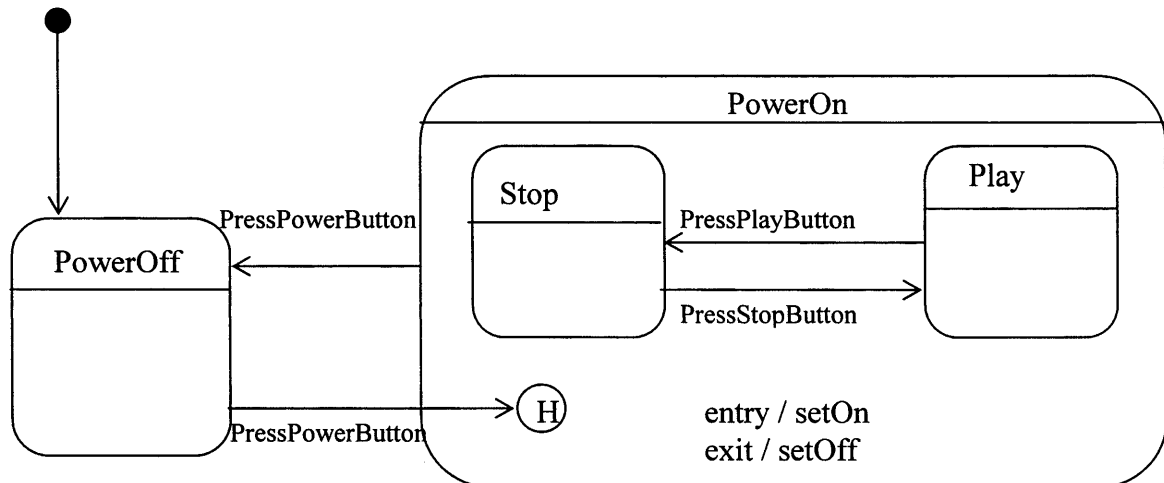


(Space for answer of question 5.)

(Space for answer of question 5.)

**Question 6** (15 marks)

- (i) Implement a cassette player class based on the information provided in the following state chart .



- (ii) Use State Pattern to re-model the behaviour of cassette player class. You have to provide a class diagram to illustrate your new model. In your diagram, include classes, attributes, operation signatures, and associations as per the information available in the state chart. You are not required to add any additional information.

(Space for answer of question 6.)

(Space for answer of question 6.)



**Question 7 (12 marks)**

Consider following information about a class structure :

Class UMember - attribute name , method getName() which returns a string value.  
Class UGraduate inherits from UMember ; and has an additional method getStipend();  
Class UEmployee inherits from UMember; and has an additional method getPaid();  
Methods getStipend() and getPaid() both return void.  
Class UTA inherits from UGraduate and UEmployee class.

Provide implementation for above in Java. Method body of above methods can be left empty in your code. Since Java does not have a direct mechanism for coding multiple inheritance, introduce an appropriate mechanism to realize it. Also write a test class to illustrate polymorphic invocation of getStipend() and getPaid() methods for UTA objects.

(Space for answer of question 7)

(Space for answer of question 7.)

**Question 8 (5 marks)**

For the following code fill up the table that follows the code. Write 'ok' if each of the statement in the main method compiles and executes fine or else write 'compile error' or 'run-time' error with reason as appropriate. Assume default access modifier to be public and that classes A , B, and C are correctly coded.

```

class A { //... }
class B extends A { //..... }
class C { //..... }

class Test {
    public static void main( String args[] ) {
        Object o;
        A a;
        B b = new B();
        C c;

        a = b;
        o = b;

        a = (A)o;
        b = (B)o;
        a = (A)(B)o;
        b = (B)a;
        c = (C)o;
    }
}

```

Statement	Answer 8
a = (A)o;	
b = (B)o;	
a = (A)(B)o;	
b = (B)a;	
c = (C)o;	

**Question 9 ( 5 marks)**

Read the following code fragment . Class A does not contain any methods that B requires, and is a completely unrelated class. But Class A must still be present for B to function correctly. Suggest an alternative implementation in Java to overcome this problem of dependency. Avoid suggesting another definition of the constant in B.

```
public class A {  
    public static String NAME = "MyForum";  
  
    public A() {  
        System.out.println("Subscribe to " + NAME);  
    }  
}  
  
public class B {  
    public B() {  
        System.out.println("Renew your subscription to " +  
A.NAME);  
    }  
}
```

(Space for answer of question 9)

**Question 10** (8 marks)

Read following code fragment for a function which attempts to purchase a property when landed on a specific square on a game board.

- (i) Draw a flowgraph .
- (ii) Based on the flowgraph, find the Cyclomatic Complexity and list all the linearly independent paths.

// methods isPropertyOwned(..) , haveEnoughMoney(..) , isPropertyOnWishList(..)  
 // return a boolean value.

```
void PurchaseProperty(P) {
    if ( isPropertyOwned(P)) {
        if (haveEnoughMoney(P.getRental())) {
            payRental(P.getRental());
        }
    }
    else {
        if (isPropertyOnWishList(P) ) {
            if (haveEnoughMoney(P.getValue()) {
                payValue(P.getValue()) ;
                addToTitleList(P.getTitle()) ;
            }
        }
    }
}
```

(Space for answer of question 10)

---END OF PAPER---