

Internal Assessment Test 1 – Sep 2016 - SCHEME OF EVALUATION

Sub:	Object Oriented Modeling and Design	Code:	10CS71
Date:	06/09/2016	Duration:	90 mins
Max Marks:	50	Sem:	VII
Branch:	ISE/CSE		

Note: : Answer any 5 questions

Total marks: 50

1. What is object orientation? Explain in brief the stages involved in OO methodology. (10M)

(Definition of object orientation)

2M

Software is organized as a collection of discrete objects that incorporate both State and behavior.

OO methodology Stages

The process for OO development and graphical notation for representing OO concepts consists of building a model of an application and then adding details to it during design. 0.5 M

(5 different stages with explanation each carries 1.5 M)

The methodology has the following stages:

5*1.5=7.5M

- **System conception** : Software development begins with business analysis or users conceiving an application and formulating tentative requirements
- **Analysis:** The analyst must work with the requestor to understand the problem, because problem statements are rarely complete or correct.
The analysis model is a precise abstraction of what the desired system must do, not how it will be done. It should not contain implementation decisions.
- **System design:** The development teams devise a high – level strategy – the system architecture for solving the application problem.
They also establish policies that will serve as a default for the subsequent, more detailed portions of design.
The system designer must decide what performance characteristics to optimize, choose a strategy of attacking the problem and make tentative resource allocations.
- **Class design** : The class designer adds details to the analysis model in accordance with the system design strategy.
The focus of class design is the data structures and algorithms needed to implement each class.
- **Implementation:** Implementers translate the classes and relationships developed during class design into particular programming language, database or hardware.
During implementation, it is important to follow good software engineering practice so that traceability to the design is apparent and so that the system remains flexible and extensible.

2. a) Elaborate on the major themes that are well supported in object oriented technology? (6M)

(six points with explanation each carries 1 M)

Abstraction: focus on essential aspects of an application while ignoring details. (1 M)

Encapsulation :(Information Hiding) separates the external aspects of an object , that are accessible to other objects from internal implementation details. (1 M)

Combining data and behaviour (1 M)

Sharing (reuse) (1 M)

Emphasis on the essence of an object: OO technology stresses what an object is, rather than how it is used. (1 M)

Synergy: Identity, classification, polymorphism, and inheritance characterize OO languages. Use all together. (1 M)

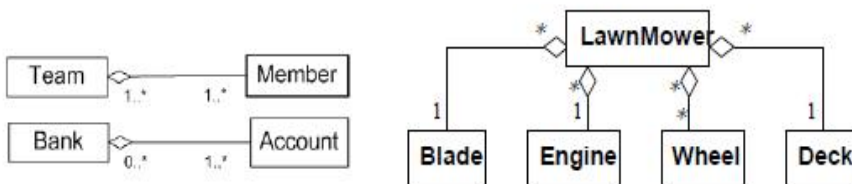
b) What is aggregation and composition? Give their respective UML notations (4M)

(Aggregation definition and Examples each carries 1 M)

2*1=2M

Aggregation

- ✓ is a strong form of association in which an aggregate object is formed using other objects as parts
- ✓ An aggregate object is treated as a unit in many operations

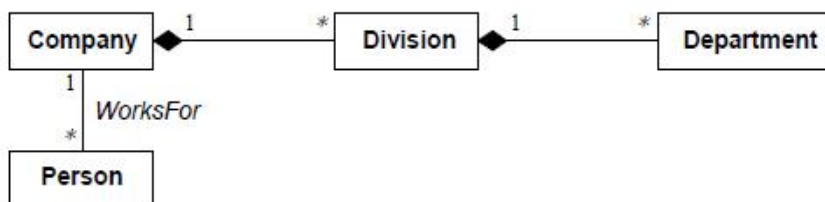


(composition definition and Examples each carries 1 M)

2*1=2M

Composition

- ✓ Is a form of aggregation with two additional constraints
- ✓ Deletion of an assembly objects triggers deletions of all constituent objects

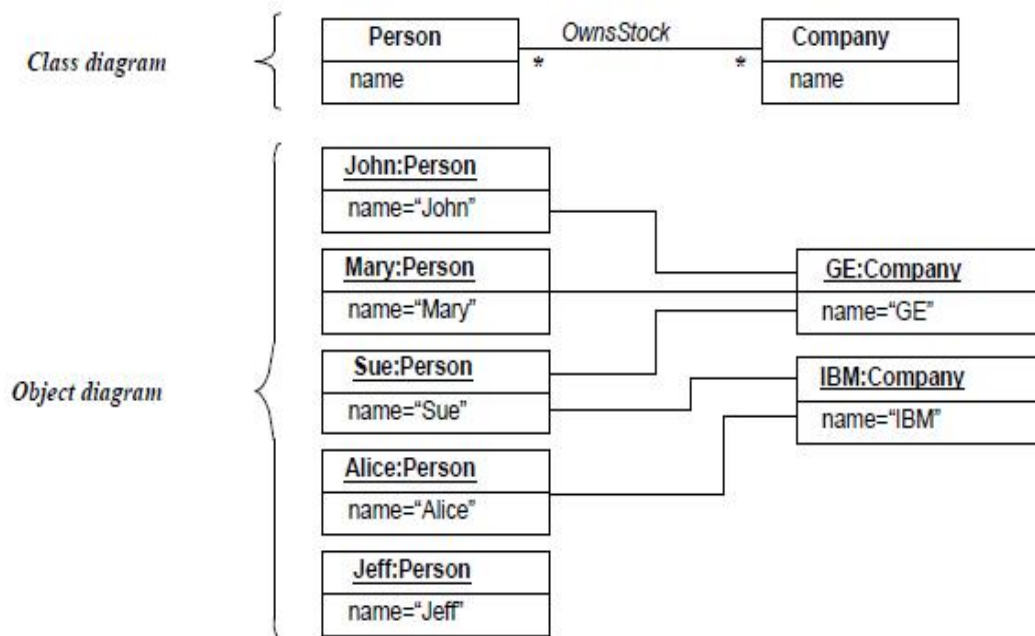


3. Prepare a meta model that supports the following object modeling concepts: a association, association ends, multiplicity, class name, qualified association. (10M)
(definition and Examples each carries 2 M)

5*2=10M

i) association: (2 M)

An **association** is a description of links (association instances) with common semantics and structure



ii) **Association end name** Associations have ends. They are called ‘Association Ends’. They may have names - Rolenames (which often appear in problem descriptions). (2M)

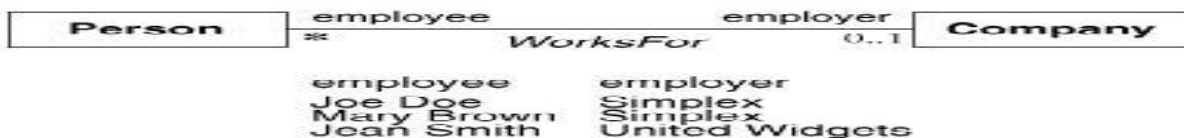


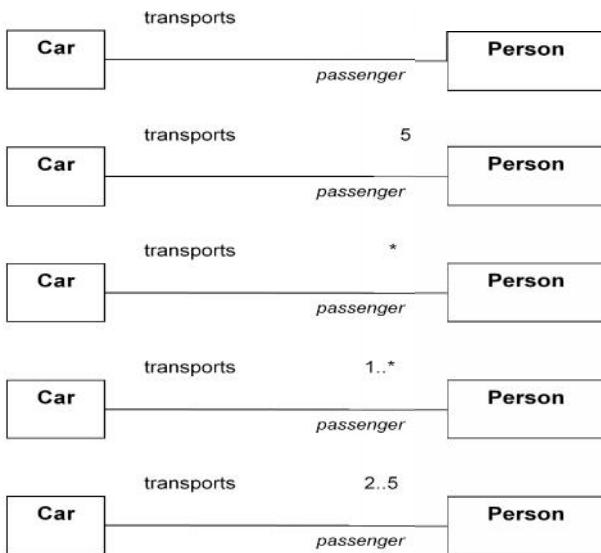
Figure 3.12 Association end names. Each end of an association can have a name.

Object-Oriented Modeling and Design with UML, Second Edition by Michael Blaha and James Rumbaugh, ISBN 0-13-1-015020-4, © 2005 Pearson Education, Inc., Upper Saddle River, NJ. All rights reserved.

iii) **Multiplicity (use any 2 indicators)** Multiplicity defines the number of objects associated with an instance of the association. UML diagrams explicitly list multiplicity at the end of association lines.

Intervals are used to express multiplicity: (2M)

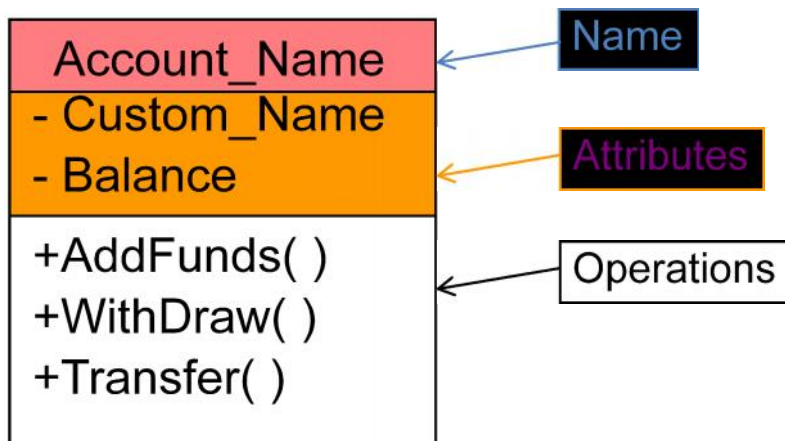
Indicator	Meaning
0..1	Zero or one
1	One only
0..*	Zero or more
1..*	One or more
n	Only n (where n > 1)
0..n	Zero to n (where n > 1)
1..n	One to n (where n > 1)



iv) **class name** : The name of the class is the only required tag in the graphical representation of a class. It always appears in the top-most compartment (2M)



Example : Account Class



v) **qualified association: Qualified associations** is an association in which an attribute called Qualifier the objects for a 'many' association' end. A qualifier selects among the target objects, reducing the effective multiplicity from 'many' to 'one'. Both below models are acceptable but the qualified model adds information. (2M)



Figure 3.22 Qualified association. Qualification increases the precision of a model.

Object-Oriented Modeling and Design with UML, Second Edition by Michael Blaha and James Rumbaugh. ISBN 0-13-1-015920-4. © 2005 Pearson Education, Inc., Upper Saddle River, NJ. All rights reserved.

4. a) What is UML? What is the importance of UML? (4 M)

An industry-standard graphical language for specifying, visualizing, constructing and documenting the software systems, as well as for business modeling.

The UML uses mostly graphical notations to express the OO modeling and design of software projects.

Simplifies the complex process of software design (2M)

Importance of UML (2M)

- Use graphical notation: more clearly than natural language and code.
- Help acquire an overall view of a system.
- UML is *not* dependent on any one language or technology.

(Mention any 2 points each carries 1 mark)

4. b) Draw class model using N-ary associations. Explain the drawbacks of N-ary associations (6M)

- An n-ary association is like a common (binary) association, except that more than two association roles involved in it.
- The UML symbol for n-ary associations is a diamond with lines connecting to related classes. If the association has a name, it is written in italics next to the diamond.

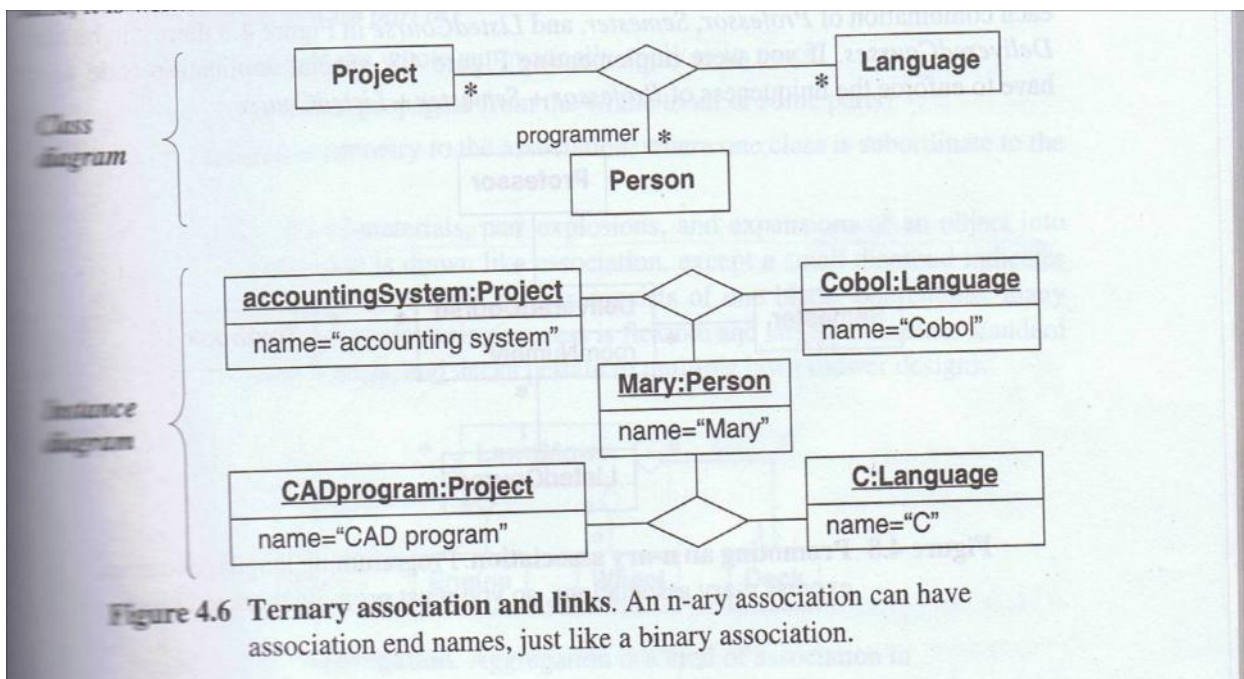
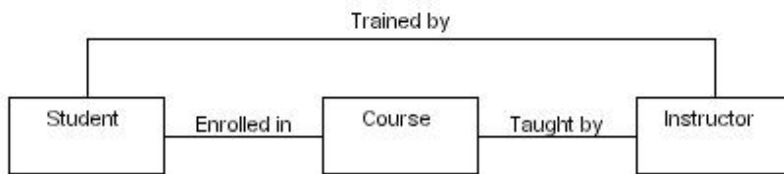
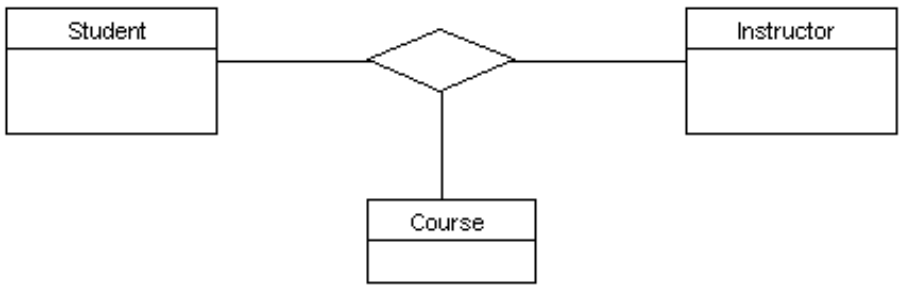


Figure 4.6 Ternary association and links. An n-ary association can have association end names, just like a binary association.

Definition + Example 4M



Drawback: Programming languages don't support N-ary association. (2M)

5. a) Explain the different events types with examples. (6M)

➤ Signal event (2 M)

Signal Event

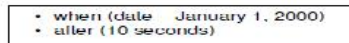


- ✓ A signal event is event of sending or receiving information.
- ✓ A signal is a one way message from one object to another.
- ✓ Most signal classes have attributes indicating values they convey.

➤ Time Event (2 M)

Time Event

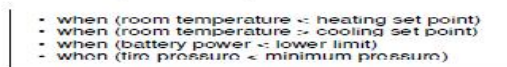
- ✓ A time event is an event caused by the elapse of a time interval or by matching an absolute time point.
 - Expression is continuously tested,
 - Whenever it changes from false to true, the event happens
- ✓ UML notation for a change event (**after** or **when** followed by expression that evaluates a condition)



➤ Change Event (2 M)

Change Event

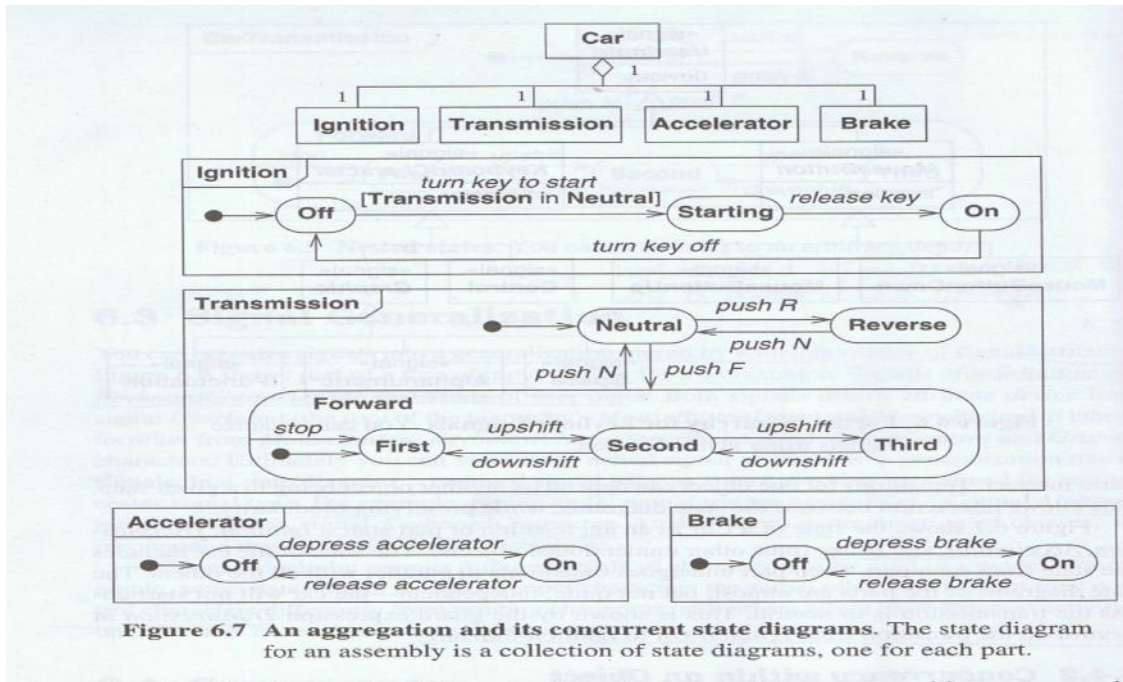
- ✓ A change event is an event that is caused by satisfaction of a boolean expression.
 - The condition must be checked often enough so that it seems continuous from the application perspective
- ✓ UML notation for a change event (**when** followed by boolean expression)



b) Explain aggregation concurrency with a suitable example. (4 M)

➤ Suitable aggregation - 1 M

➤ State diagrams showing concurrency – 3 M



6. Explain state diagram and draw state model for telephone line, with activities. (10M)

States capturing – 2 M

Transitions – 2 M

Activities – 2 M

Diagram - 4M

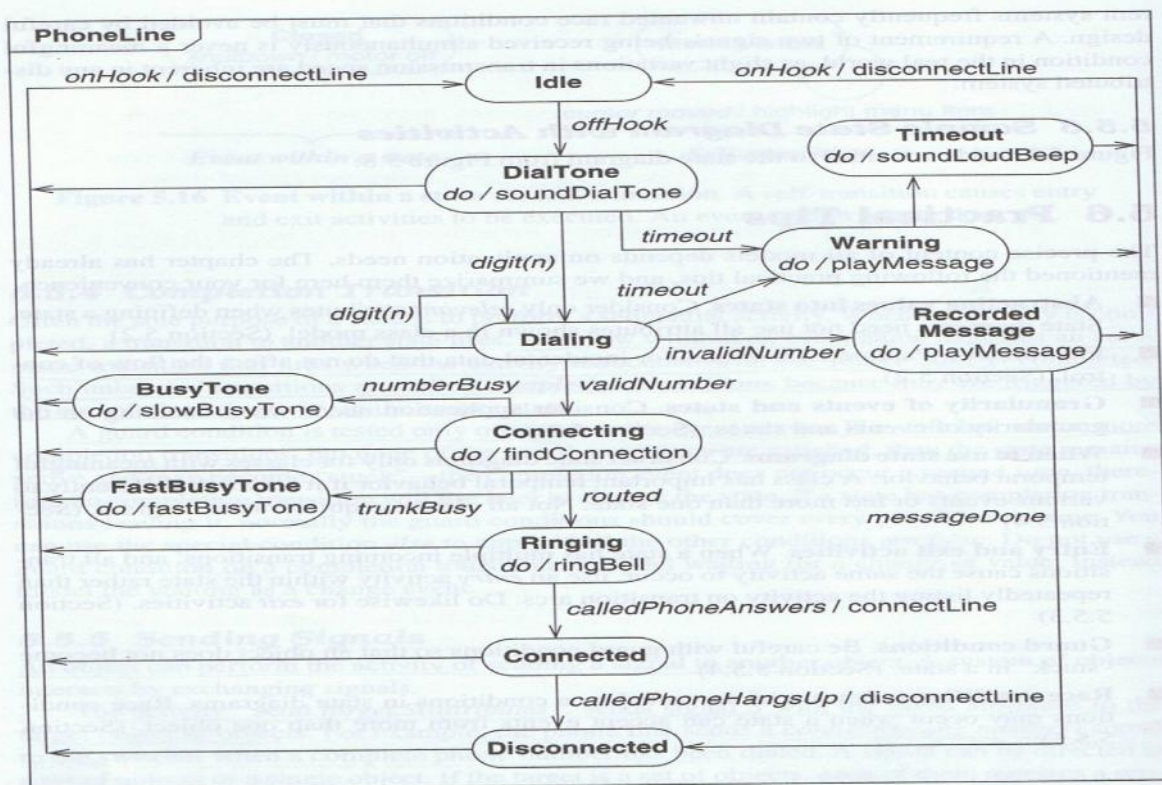
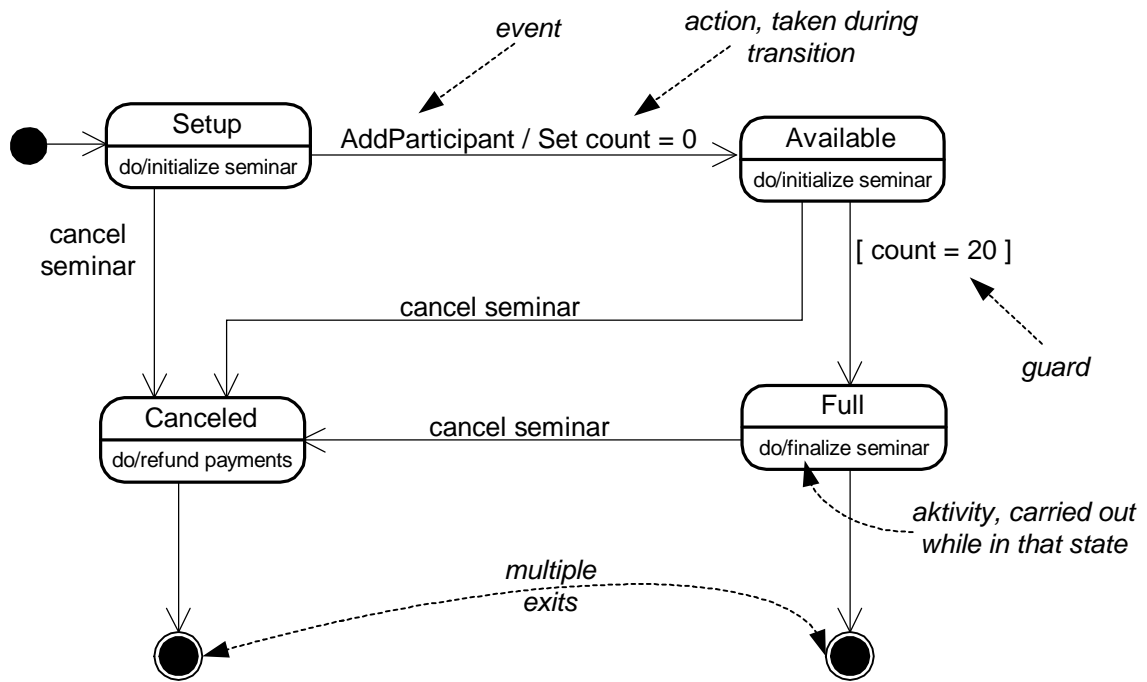


Figure 5.17 State diagram for phone line with activities. State diagrams let you express what objects do in response to events.

7. Prepare and explain state diagram for seminar registration. The seminar gets cancelled if there are less

than 2 registrations and registrations should not exceed a maximum of 50. Any participant can cancel his/her registration at any point of time. (10M)



States capturing – 2 M

Transitions – 2 M

Activities – 2 M

Guard Condition - 2M

Diagram -2M

