Solved Question Paper June 2017

1.a) What are the benefits of Object Oriented Methodology in real life applications ? Briefly explain each element of the state diagram with respect to dynamic modeling. 10mark

It's a approach through which modelling of the systems are done by visualizing the system based on the real world concepts.

There are several benefits of object oriented modelling. Reuse and emphasis on quality are the major highlights of Object Oriented Methodology. Object Oriented Methodology provides resistance to change, encapsulation and abstraction etc. Object-oriented modeling reduces the need for maintenance and increases both reliability and flexibility.

Major benefits of Object Oriented Modelling is that :

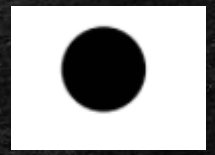
- Development of the system becomes fast
- Quality of the system increases
- Improves the communication between users, analysts etc.
- The ability to tackle more challenging problems
- It gives the freedom of use of existing code and design.

- Helps in the development of complex system with less risk due to basic properties like class, objects and inheritance.
- Helps in understanding problems, communicating with experts from distance, modelling enterprises and designing programs and databases.
- It reduces the need for maintenance and if needed maintenance is easier.
- It increases both reliability and flexibility.
- Modular Architecture : We can replace or add any one component (**module**) without affecting the rest of the system. This is because there is clear division between components.
- It gives the freedom of use of existing code and design : In this each object is a stand alone component that can be reused not only within a specific problem domain, but also in completely different problem domains, having the requirements of similar objects.
- It focuses on data relationships. We cannot develop a successful system when data relationships are not well understood.
- It provides all of the insight of an ER diagram and contains additional information related to the methods to be performed on the data.

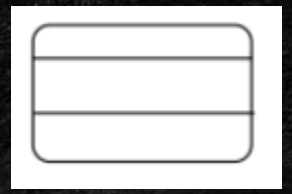
- The ability to tackle more challenging problem domains
- Improved communication among users, analysts, designers and programmers.

Elements of the state diagram with respect to dynamic modelling are :

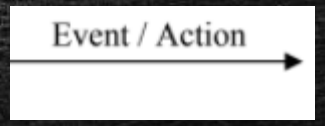
Initial State : This shows the starting point or first activity of the flow. Its denoted by a solid circle. This is also known as pseudo state, where the state has no variables describing its further and no activities, to be done.



State : Represents the state of an object at an instant of time. In a state diagram, there will be multiples of such symbols, one for each state of the object. Its denoted by a rectangle with rounded corners and compartments.

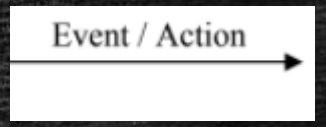


Transition : An arrow indicating the object to transition from one state to the other. The actual trigger even and action causing the transition are written beside the arrow, separated by a slash. Transitions that occur because the state has completed an activity are called “triggerless” transitions.

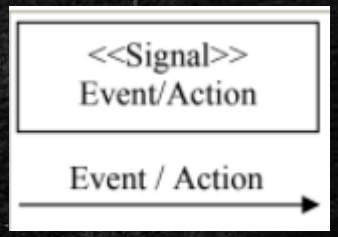


History States : A flow may require that the object go into the trance, or wait state, and on the occurrence of a certain event, go back to the state it was in when into a wait state-its last active state. This is shown in a state diagram with the help of a letter H enclosed within a circle.

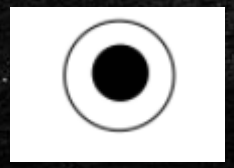
Event and Action : A trigger that causes a transition to occur is called as an event or action. Every transition need not occur due to the occurrence of an event or action directly related to the state that transitioned from one state to another. An event/action is written above a transition that it causes.



Signal : When an event causes a message/trigger to be sent to a state that causes the transition; then, that message sent by the event is called a signal.



Final state : The end of the state diagram is shown by a bull's eye symbol, also called a final state. A final state is another example of a pseudo state because it does not have any variable or action described.



1.b)What is concurrency? Explain how concurrency is identified in a system. How will you manage concurrency in an "Online Railway Reservation, System" ? Explain with the help of a diagram. (10marks)

Concurrency is an essential concept in computer programming. It is the idea of handling more than one task at a time.

Many programming problems require that the program be able to :

- Stop what it's doing
- Currently deal with some other problem, and
- Return to the main process. There is a large class of problems in which you have to partition the problem into separately running pieces so that the whole program can be more responsive. Within a program, these separately running pieces are called threads, and the general concept called multithreading.

If we have more than one thread running that is expecting to access the same resource then there is a problem. To avoid this problem, a thread locks a resource, completes its task, and then releases the lock so that someone else can use it at a time. It is very important to handle threads properly.

In terms of object, concurrency in objects can be identified by the way they change the state. Concurrent objects are those that can change state independently. Aggregation implies concurrency.

Concurrency in OOAD is described and studied in dynamic modelling.

One important issue in system design is to find the concurrency in objects. If we identify non-concurrent objects, we can assign all the objects together in one thread of control, or process.

If the objects are concurrent in nature we have to assign them to different thread of control.

A thread of control is a path through a set of state diagrams on which a single object is active at a time.

Objects are shared among threads, that is, several methods of the same object can be active at the same time.

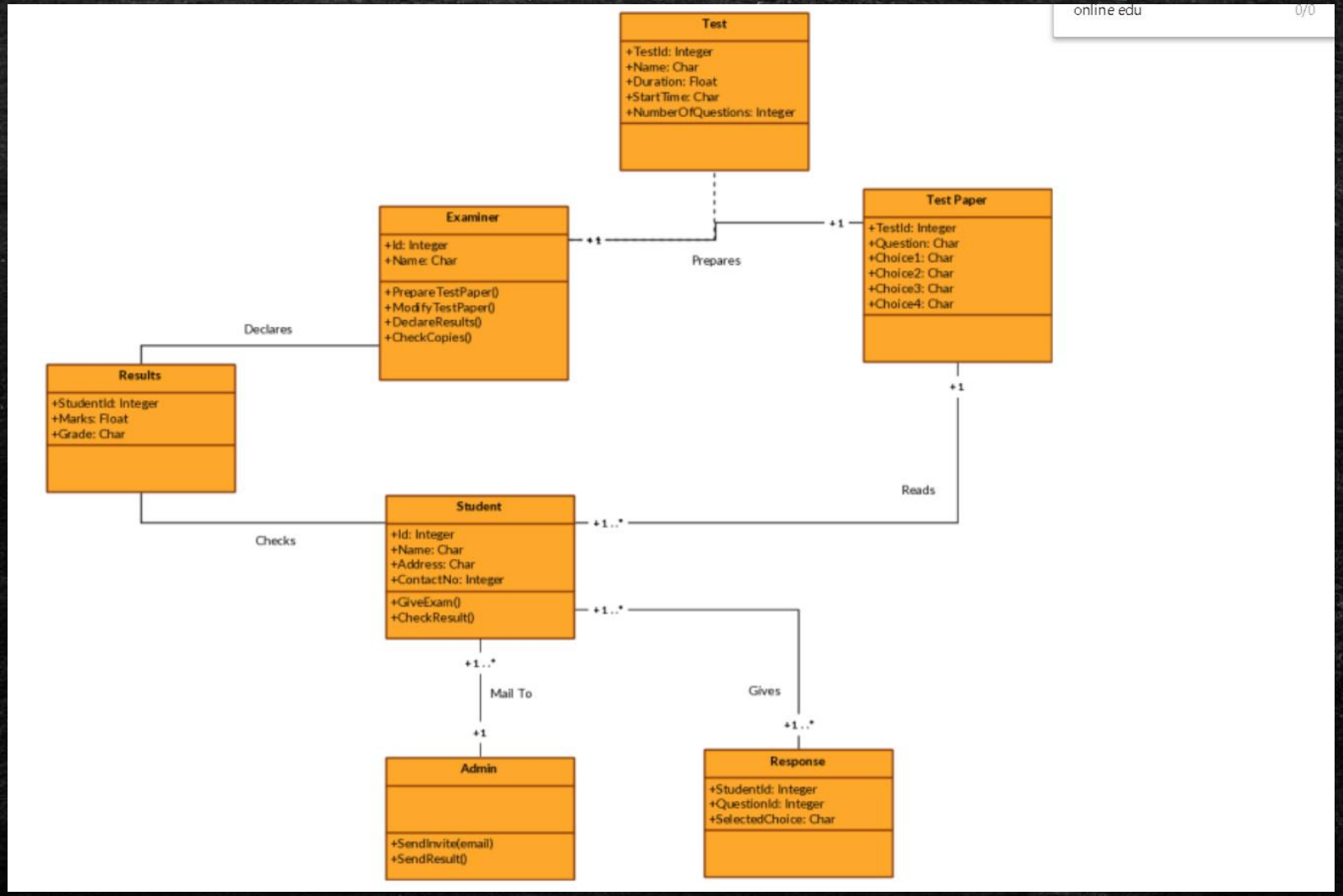
Thread splitting : Object sends a message but does not wait for the completion of the method.

Identification of concurrency :

Concurrency is identified in a dynamic model. Two objects are said to be concurrent (parallel) if they can receive events at the same time. Concurrent objects are required to be assigned to different threads of control.

If an object must perform two or more activities concurrently, then the events must be synchronized.

1.d)Draw the class diagram for an "Online Examination System". Make necessary assumptions.(10marks)



2.a)Differentiate between the following with examples :Specialization and Generalization(5marks)

Specialization	Generalization
It is a top-down activity.	It is a bottom-up activity.
Specialization involves the definition of a new class which inherits all the characteristics of a higher class and adds some new ones, in a subclass. OR Specialization is the reverse process of Generalization means creating new sub classes from an existing class.	Generalization extracts the common features from a collection of classes, and placing them higher in the inheritance hierarchy, in a super class. OR The process of extracting common characteristics from two or more classes and combining them into a generalized superclass, is called Generalization. Generalization is represented by a triangle followed by a line.
The higher level entity may not have lower level entities.	The higher level entity must have lower level entities.
Specialization increases the size of a schema.	Generalization reduces the size of a schema.
Specialization is applied on a single entity.	Generalization entities on group of entities.

Specialization results in forming the multiple entity from a single entity.	Generalization results in forming a single entity from multiple entities. Generalization clubs all the entities that share some common properties to form a new entity.
Creates new objects based on the difference between the existing ones and have some features of the parents.	Takes all the information that have universal nature within the entities and then forms a new entity.
Animal example	Furniture example

2.c) Difference between SAD and OOAD. (5mark)

Structured approach	Object Oriented Approach
Its focus is on processes and procedures.	Its based on the concept of objects.
Most direct way of implementing a desired goal.	Its an indirect way of system development because in this holistic view of application domain is considered, and objects are identified in the related problem domain.
Readjustment of some new changes in the system is very difficult.	It gives space for further enhancement of the system.
It is suitable for well-defined projects with stable user requirements.	It is suitable for risky large projects with changing user requirements.
follows **top-down approach**.	follows **bottom-up approach**.
Programs are divided into small self contained program segment known as **functions**.	Programs are divided into small entities called **objects**.
less secure as there is no way of **data hiding**.	more secure as having data hiding feature.

provides **less reusability,** more function dependency.	provides more reusability, less function **dependency**.
can solve **moderately** complex programs.	can solve any **complex** programs.
Less abstraction and less flexibility.	More abstraction and more flexibility.
Functions and data are not tide together.	Functions and data are tide together.
Modelling techniques used in this are DFD, flowcharts etc..	Modelling techniques used is UML which includes state diagram, class diagram, Use Case diagram, Sequence Diagram etc..
It is an old approach and is not preferred.	It is a common approach nowadays.

2.d) Difference between Relational Database and Object Oriented Database.(5marks)

Relational Database	Object oriented database
Database that stores data in tables that consist of rows and columns. Each row has a primary key and each column has a unique name.	Database that stores data in objects. An object is an item that contains data, as well as the actions that read or process the data.
relational database rely on relational model.	Object database rely on OOP
relational data base handle a single data.	Object database can handle different types of data.
It is based on mathematical principles called relational algebra.	It is based on objects.
Associations are not directly represented.	Associations are directly represented.
Faster for complex queries	Slower than relational databases for relational databases.
Supports multiple writers and readers.	Does not support multiple writers and readers.
RDBMS store only data.	OODBMS store data and methods.