Solved Question Paper June 2011

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

5.d)How do you implement constraints ?

Class constraints describe relationships that must be between the attribute values of an instance of the class; and preconditions and post-conditions specify what must be true before and after an operation is called. Once these are implemented by including code in the class which checks those conditions at the appropriate times, then the application becomes more reliable and robust.

All the preconditions that are specified for an operation should be explicitly checked in an implementation. Preconditions state properties of operation's parameters that must be satisfied if the operation is to be able to run to completion successfully. It is the responsibility of the caller of the operation to ensure that the precondition is satisfied when an operation is called.

If an operation does not check its parameter values, then, there is a possibility that wrong or meaningless values will go undetected, and this results in unpredictable run-time errors. A better strategy is to check its precondition for an operation and to raise an exception if a violation of the detection is detected.

Example :

```
Public class SavingsAccount

{
```

```
Public void withdraw(double amt)
{
If( amt >=balance)
{
// throw PreConditionUnsatisfied
}
balance=balance-amt;
}
Private double balance;
}
```

Any constraints can be checked at run-time by writing code that will validate the current state of the model. These checks increase overhead. That is why, except for the case of precondition checking, constraints are rarely implemented explicitly.

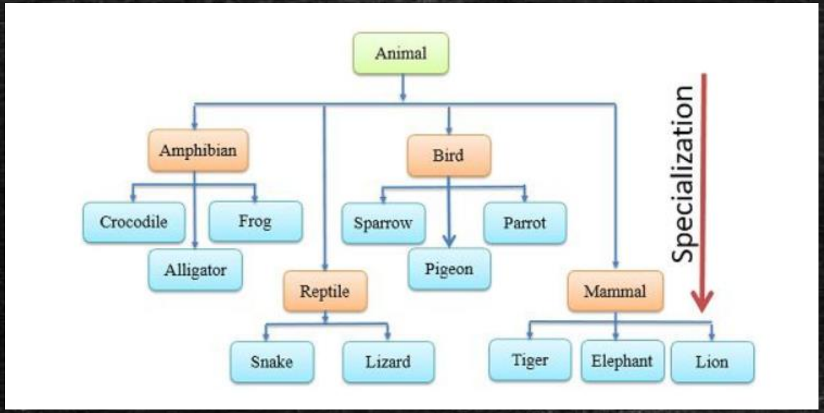
5.d)Define Serialization. Where it can be used and why ? (4marks)

Specialization :

Specialization is the reverse process of Generalization means creating new sub classes from an existing class. 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. It is a top-down design activity. In specialization, we split an entity to form multiple lower level entities. These newly formed lower level entities inherit some features of the higher level entities. It may happen that a higher level entity may not split further and hence, it may not have any lower level entity. Specialization is always applied on a single entity. It increases the size of a schema.

Specialization results in forming the multiple entity from a single entity. It creates new objects based on the difference between the existing ones and have some features of the parents.

Let us discuss an example of specialization. Let us consider entity **Animal**. The entity animal can further be spilt into **amphibian**, **reptiles**, **birds**, **mammals** etc.. Entity amphibian can be further split to **crocodile**, **alligator**, **frog**. The entity reptile splits to **snake, lizard**. The entity bird can be split to **sparrow**, **pigeon**, **parrot**. Mammals can be spilt to a **tiger**, **lion**, **elephant**.


Animal
Amphibian
Bird
Crocodile
Frog
Alligator
Sparrow
Parrot
Pigeon
Reptile
Mammal
Snake
Lizard
Tiger
Elephant
Lion
Specialization