OBJECT ORIENTED TESTING

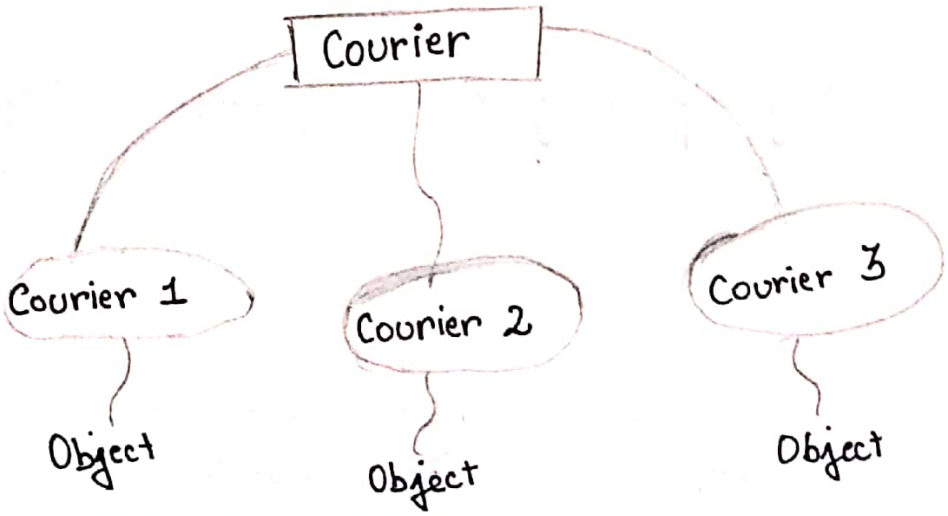
- RUPASA KARMAKAR 11610170 C03

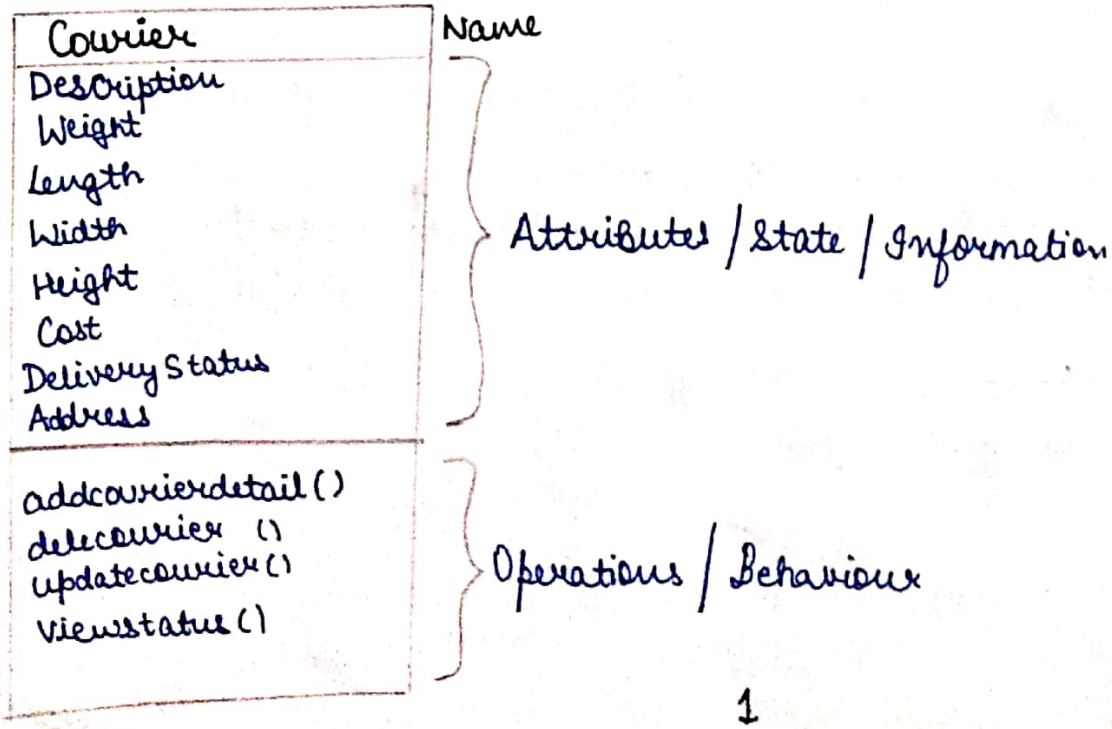
What is object orientation?

In object orientation, an action is initiated by sending a message to an agent who is responsible for that action. An agent acts as a receiver and if it accepts a message (request), it becomes its responsibility to initiate the desired action using some method to complete the task.

Classes and Objects

All objects are instances of class. The class describes the structure of instances which include behaviour and information





Inheritance

All classes inherit information from the upper class. Hence, information from a base class in common to all derived classes. However each derived class also have some additional information of its own. Each derive class inherits the attributes of its base class and this process is known as inheritance.

What is a unit?

A unit is the smallest portion of the program that can be executed. We may treat each class or a method with in each class as unit. Unit testing of a class with a super class may be impossible to do without the super classes methods or variables.

Levels of Testing

There are 3 or 4 levels of testing depending on our uproach.

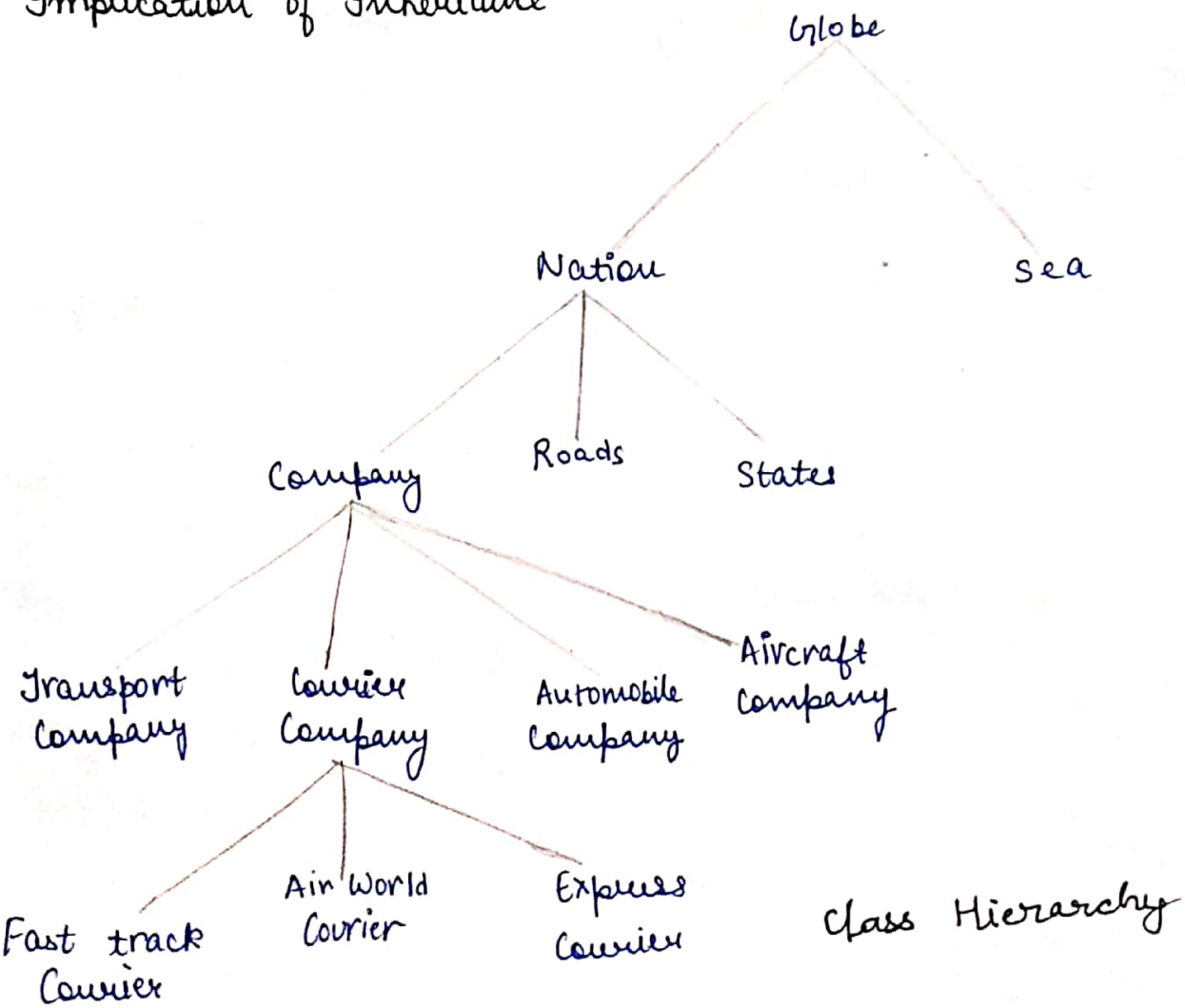
(i) Method testing
(ii) Class testing
(iii) interclass testing
(iv) System testing

In order to test a class we create an Object & pass the appropriate parameters to the constructor. In encapsulation plays an important role in class testing because data & function are combined in a class. We concentrate on each encapsulated class during unit testing but each function may be difficult to test independently. Interclass testing considers the parameter passes issues between 2 classes and is similar to integration testing. System testing considers the whole system and test cases are generated using functional testing techniques.

Advantages

* Early generation of test cases in the process.
* Helps designers to understand & express requirements.
* Used cases use to generate many test cases.
* May also generate test cases from SDD documents.

Implication of Inheritance



Class Hierarchy

Implication of Polymorphism

It means many forms. According to jaco98, Polymorphism means that the sender of the message does not need to receiving instances class. The receiving instance can belong any arbitrary class. It is considered to be an important concept. In object oriented language, arithmetic Operators like +, =, - are used to Operate on primary data type such as INT, FLOAT etc. We may overload these Operators so that they may Operate in the same way on Object as they Operate on primary

data types. Thus, the same Operator will have multiple forms.
When we abstract the interface of an Operation and leave the implemation details to subclasses, this activity polymorphic Operation.
We can create a super class by pulling out important states, behaviours and interfaces of the class. This may simplify the complexity of a problem. An Object may not need to know the class of another Object when it sends the message.

Implication of ENCAPSULATION

It is also known as information hiding concept. It is a way in which both data and function that Operate on data are combined in a single unit. The only way to access the data is through functions, which Operate on data. The data is hidden from the external world.
If data of any Object needs to be modified it may be done to the specified functions only. The process of encapsulating the data and functions into a single unit simplifies the activities of writing, modifying & maintaining the program. It also protects from accidental modifications.

Issues and Testing

- We may tested independently, as a unit or as a group of a System the decision is dependent on the amount of effort required to develop a test driver, severity of class in the system and associated risk with it.
- If a class has been developed to be a part of a class library, thorough testing is essential even if cost of development is high.
- Classes should be tested by developers who are familiar with internal design complexities and other critical issues of a class under test.

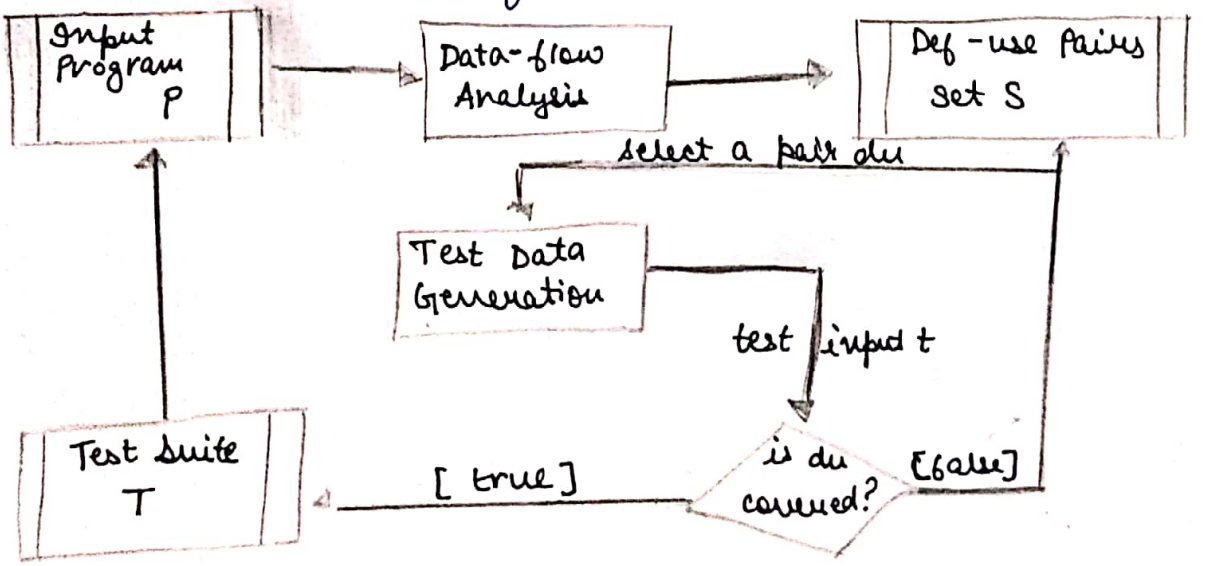
- Class should be tested with respect to specifications if some unspecified behaviour have been implemented we may not be able to test them.

- A test plan with a test suite made disciplin the testers to follow a predefined path.

Data Flow Testing

It is a specific strategy of testing that focusses on data variables and their values. It makes use of control flow graph. It is considered as a type of white box testing and structural testing. It keeps a check at the data receiving point by the variables and its usage points. It is done to cover the path testing and branch testing gap.

The process is conducted to detect the bugs because of the incorrect usage of data variable on data variables. For e.g. Initialization of data variables in programming code, etc.



Advantages

* Variables used but never defined
* Variables defined but never used,
* Variables defined multiple times before actually used,
* DE allocating variables before using

Limitation

* Testers require good knowledge of programming.
* Time consuming
* Costly process.