Page No:
Date: / /

SEQA :- UNIT - 4

Q.2 Explain Unit Testing with neat & clear diagram.

→ Unit testing is the process of checking small pieces of code to ensure that the individual parts of a program work properly on their own.

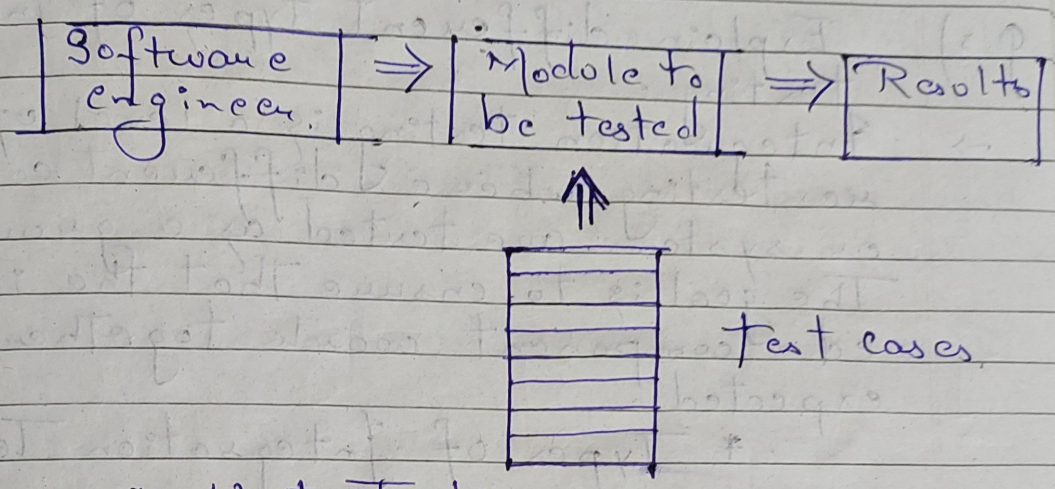
→ Unit tests are used to test individual blocks (units) of functionality.

→ Unit testing is done by developers. It is done during the coding phase by the developer.

→ The purpose of unit testing is to test the correctness of isolated code. A unit component is an individual f^n or code of application. White box testing approach is used for unit testing & usually done by developers.

* Objectives

1) To isolate section of code.
2) To verify correctness of code.
3) To test every function & procedure.
4) To fix bugs early in the development cycle & to save costs.
5) To help with code reuse



* Unit Testing

* Advantages :-

1) Unit testing decreases total testing cost.
2) It can be automated.
3) It makes debugging easier.
4) It helps in finding debugging earlier.

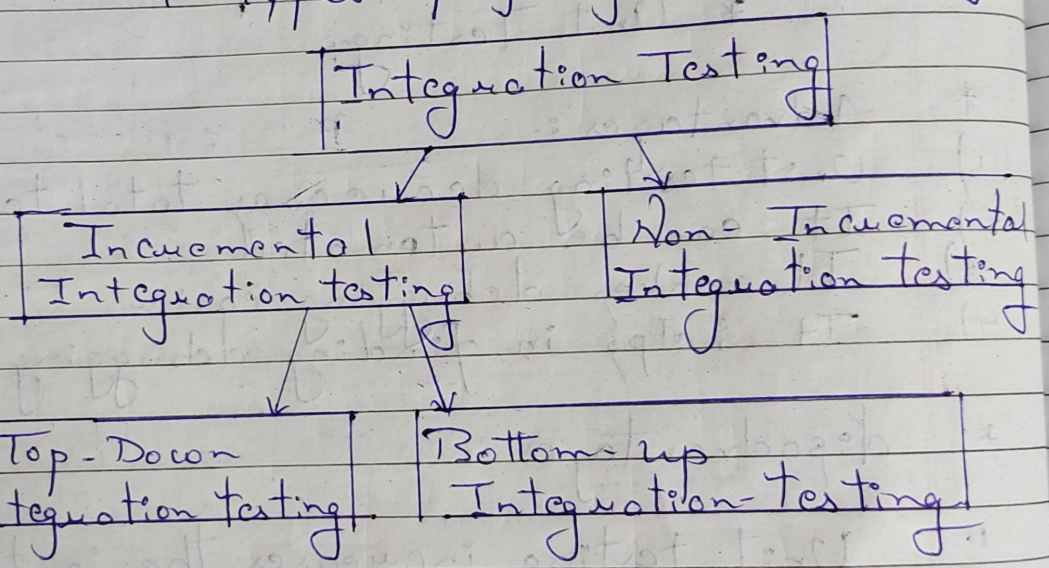
* disadvantages :-

1. Unit testing increases the initial development time.
2. It cannot identify integration or broad level error.
3. In the unit testing, evaluation of all execution path is not possible.

Q.3) Explain different types of Integration testing.

→ Integration testing is a level of software testing where different components or system are tested as a group. The goal is to ensure that the integrated component work together as expected.

* Types of Integration Testing



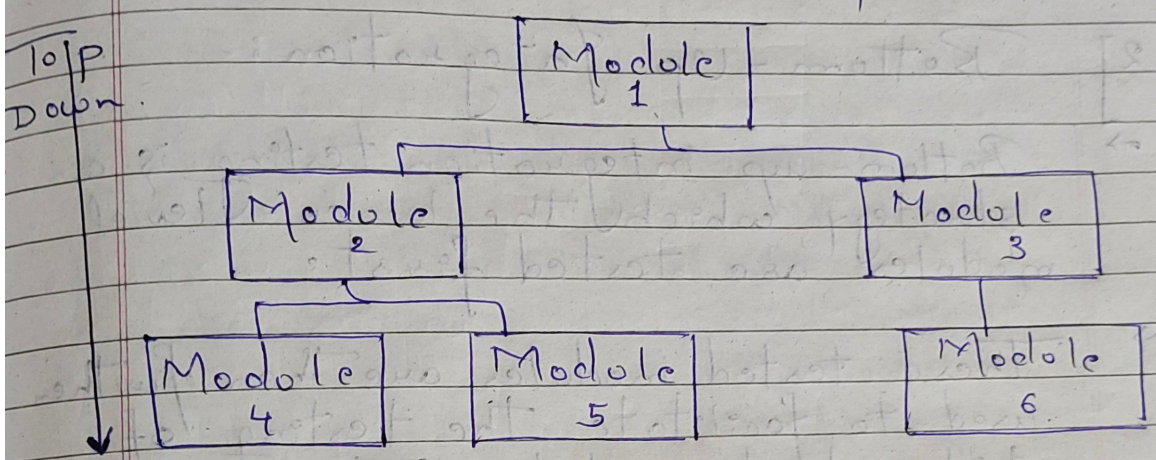
1] Top-Down Integration Testing :-

→ Top Down Integration Testing is a method in which integration testing takes place from top to bottom following the control flow of software system.

→ The higher level modules are tested first & then lower level modules are

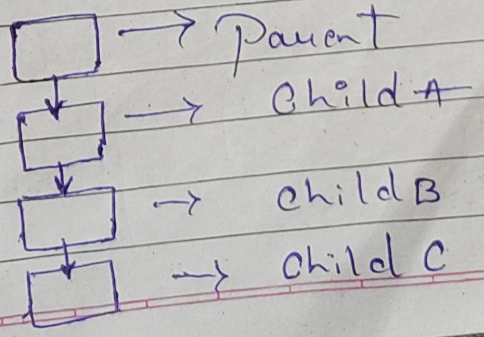
tested & integrated in order to check the software functionality.

→ Stubs are used for testing if some modules are not ready.



Advantages :- 1) Fault localization is easier.
2) Possibility to obtain early prototype.
3) Critical modules are tested on priority.

Disadvantages :- 1) Need many Stubs.
2) Modules at a lower level are tested inadequately.

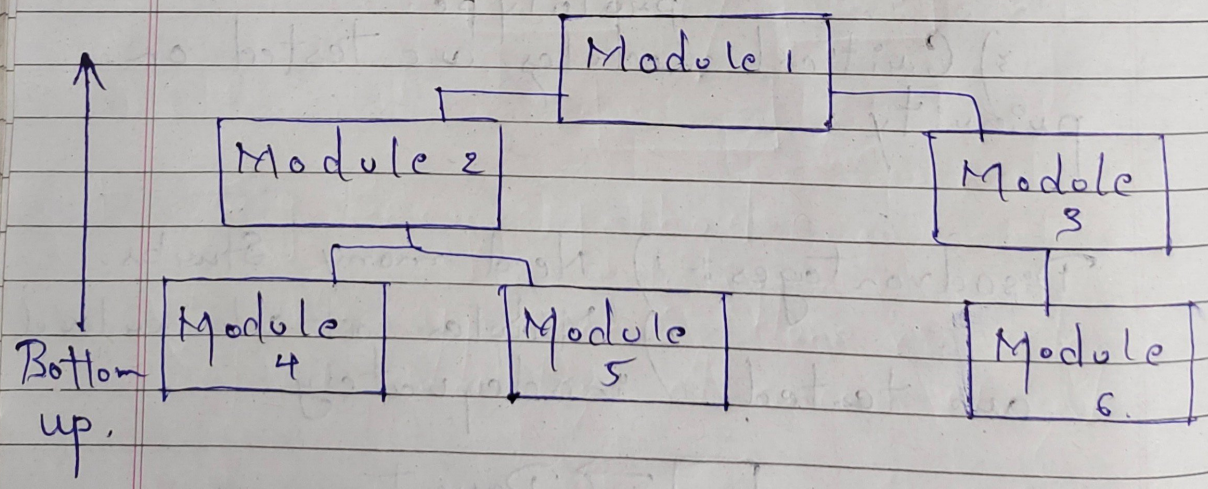


In the top-down approach, we will be ensuring that the module we are adding is the child of previous one like child C is the child of Child B & soon.

2] Bottom-Up Integration:-

→ Bottom-up integration testing is a strategy which the lower-level modules are tested first.

→ These tested modules are then further used to facilitate the testing of higher level modules. The process continues untill all modules at top level are tested.

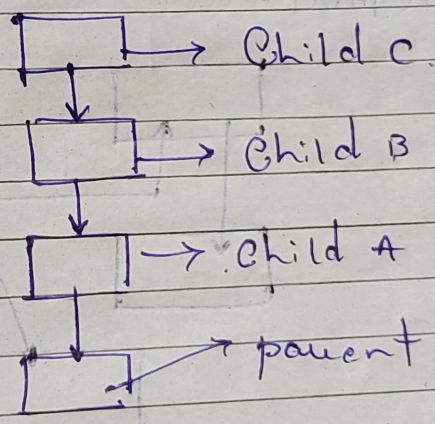


Page No.:
Date. / /

* Advantages :- 1) Fault localization is easier
2) No time is wasted waiting for all modules to be developed unlike Big-bang approach.

* Disadvantages :-

1) Critical modules are tested last due to which the defects can occur.
2) There is no possibility of an early prototype.

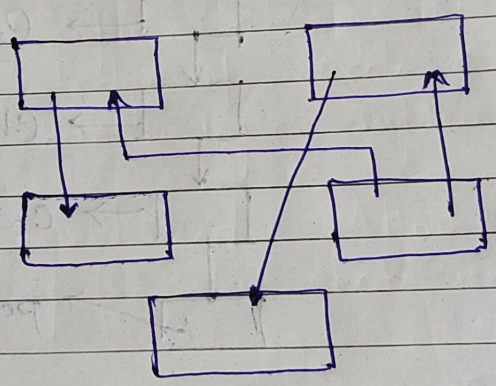


Page No:
Date:

3) Non - incremental integration Testing

→ When the data flow is very complex & when it is difficult to find who is a parent & who is a child.

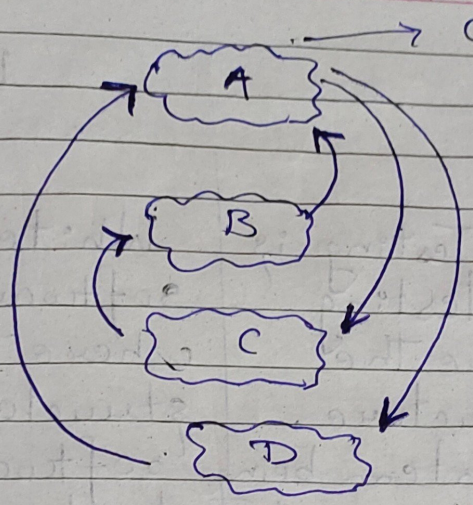
→ And in such case, we will create the data in any module bang on all other existing modules & check if the data is present. Hence, it is also known as Big bang method.



* Big Bang Method.

→ In this approach, testing is done via integration of all modules at once. It is

→ It is convenient for small software systems, if used for large software systems identification of defects is difficult.



Advantages :- 1) It is convenient for small size software systems.

Disadvantages :- 1) Time provided for testing is very less.

2) We may miss to test some of the interfaces.

Q.4) Differentiate

Block Box Testing	White Box Testing
Def'n:- Black box Testing is a software testing method where the internal structure, design of system being tested is not known to the tester.	White Box tesing is a software testing meth. where the internal structure, & code of software being tested. It is also known as clear box testing.
Working:- Testers focus on input & outputs of software without knowledge of its internal code or logic.	Testers have knowledge of the internal code & logic of the software.
Advantage:- Tester does not need knowledge of the internal code.	Allows thorough of each line of code.
Disadvantages:- Limited coverage of code paths.	Requires a detailed understanding of the internal code.
Example:- 1. Functional testing 2. Acceptance testing 3. System testing	1. Unit testing 2. Integration testing 3. Code coverage testing.

Page No.:
Date. / /

Q. 6 Explain different types of Testing used in System Testing

→ 1. Load Testing :-

→ Load testing is a type of software testing which is carried out to determine ~~product or application~~ the behaviour of a system or software product under extreme load.

2. ~~Recovery Testing~~ :- ~~It is system that forces the software to fail in variety of ways~~

5. Performance Testing :-

→ Performance Testing is a type of software testing that is carried out to test the speed, scalability, stability & reliability of the software product or application.

3) Regression Testing :-

This type of testing checks to make sure that changes made to the system do not introduce new problems.

Page No.
Date. / /

4) Smoke testing :- This type of checks the basic functionality of a system to make sure it is working properly.

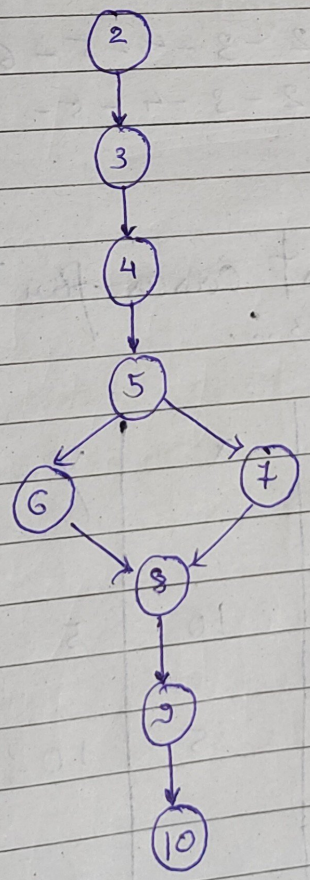
2) Recovery Testing :-
This type of testing checks how well a system recovers from a failure.

Page No.:
Date. / /

Q.11) Consider the pseudo code for simple subtraction

1. Program 'Simple Subtraction'
2. Input (x, y)
3. Output (x)
4. Output (y)
5. ~~x = y =~~ if x > y then Do
6. x - y = z
7. Else y - x = z
8. ~~Else~~ End If
9. Output (z)
10. Output "End Program"

→



Page No:
Date: / /

Q. 10)

* Step 1) We will first draw the flow graph for given pseudo code. This flow graph is as shown

* Step 2) We will compute cyclomatic complexity using following formula = e - v + 2 where e is total no. of edges, v is total no. of vertices.

9 - 9 + 2 = 2

Now we will find two independent path testing

* Step 3)

path 1 -> 2 - 3 - 4 - 5 - 6 - 8 - 9 - 10

path 2 => 2 - 3 - 4 - 5 - 7 - 8 - 9 - 10.

* Step 4) The test cases for these paths are as :-

Independent path	x	y	Expected Result
path 1			
2-3-4-5-6-8-9-10	10	5	5 (End program)
path 2			
2-3-4-5-7-8-9-10	5	10	5 (-"-)
path 2			
2-3-4-5-7-8-9-10	5	5	0 (-"-)

Page No.:
Date. / /

Q.10) Analyze the Significance of Cyclomatic Complexity.

→ Cyclomatic complexity is a measure of how complex a program is, & it is important to keep it low to write reliable, maintainable & understandable code.

The cyclomatic complexity can be computed by three ways.

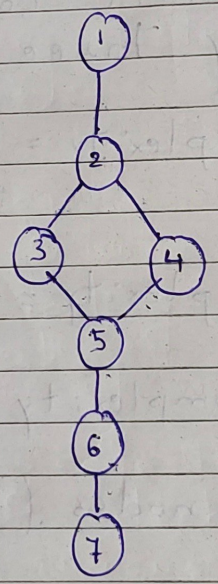
1. Cyclomatic Complexity = Total no. of regions in the flow graph.

2. Cyclomatic Complexity = E − N + 2

3. Cyclomatic complexity = P + 1

P → predicate nodes (decision making nodes).

```
1.  int temp
2.  if (a>b)
3.      temp = a
4.  else temp = b
5.      ~~temp = b~~
5.  if (c > temp)
6.      temp = c
7.  return temp.
```



Step 1) We will first draw the flow graph for given pseudo code. This flow graph is as shown.

Step 2) We will compute cyclomatic complexity using following formula $e - v + 2$, where e is edges & v is the vertices.

$= E - V + 2 = 7 - 7 + 2 = 2$

Step 3)

path 1 -> 1 - 2 - 3 - 5 - 6 - 7
path 2 -> 1 - 2 - 4 - 5 - 6 - 7

Step 4) The test cases for the paths as follows.

Independent path	x	y	z	Expected Result
path 1				
1 - 2 - 3 - 5 - 6 - 7.	10	5	8	10
path 2				
1 - 2 - 3 - 4 - 6 - 7	5	10	8	10
path 2				
1 - 2 - 3 - 4 - 6 - 7	5	6	8	8.

Page No.
Date. / /

Q.1) Explain the Testing Strategies in SDLC

→ Software Testing is the process of evaluating a software applications to identify if it meets specified requirements & to identify any defects.

→ We begin by 'testing-in-the-small' & move toward 'testing-in-the-large'.

→ Various testing strategies for conventional software are:-

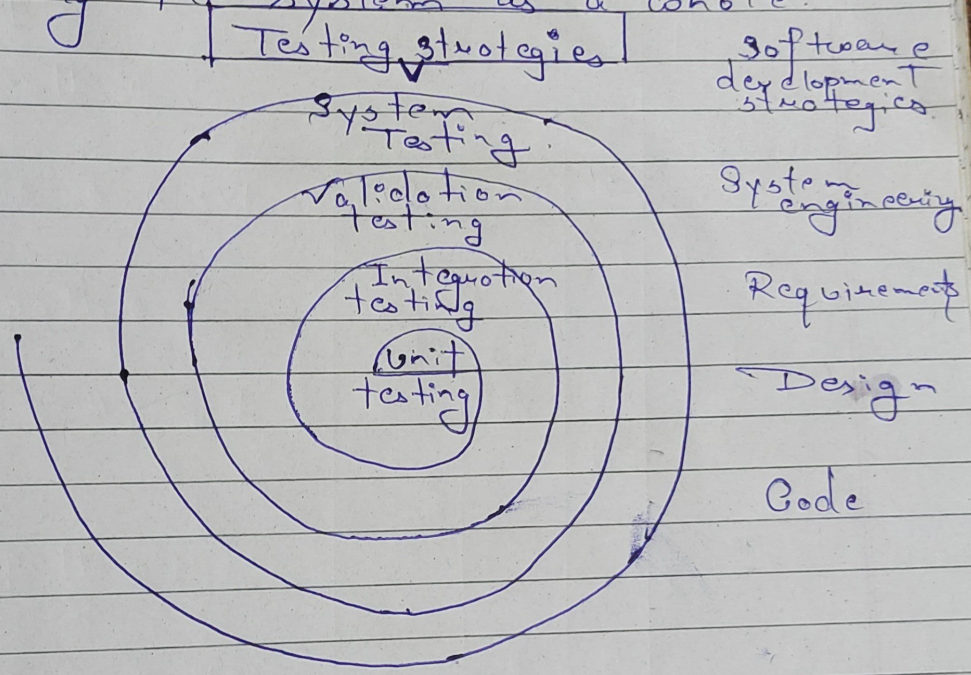
1. Unit testing
2. Integration testing
3. Validation testing
4. System testing

1. Unit testing :- In this type of testing techniques are applied to detect errors from each software component individually.

2. Integration testing :- Integration testing is a level of software testing where different component or system are tested as a group.

3. Validation Testing :- It provides assurance that the software validation criteria meets all functional, behavioral, & performance requirements.

4. System Testing :- In system testing all system elements forming the system as a whole.



→ Write extra theory from Unit 4 ppt if you want.