UNIT-IV

- CODING AND TESTING:
  - CODING
  - CODE REVIEW
  - SOFTWARE DOCUMENTATION
  - TESTING
  - UNIT TESTING
  - BLACK-BOX TESTING
  - WHITE-BOX TESTING
  - DEBUGGING
  - PROGRAM ANALYSIS TOOL
  - INTEGRATION TESTING
  - TESTING OBJECT-ORIENTED PROGRAMS
  - SYSTEM TESTING
  - SOME GENERAL ISSUES ASSOCIATED WITH TESTING

• CODING INTRODUCTION:

→ Coding is one of the part in SDLC, which will come after designing phase.

→ Before going to build the code, choose the best technologies like C, C++, JAVA, .net etc.

→ Understand the basic design principles and concepts.

→ Each technologies have its own structures.

→ It is an english language, easy to readable, writable, and learn.

→ Nowadays, most of the software industries promote software engineers to reuse existing source codes rather than writing them from the scratch.

• coding principles:

The coding principles are closely related to the principles of design and modelling. These are

1. Information Hiding
2. Structured programming features
3. Maximize cohesion and minimize coupling.
4. code reusability
5. Simplicity
6. code verification
7. code documentation
8. Follow coding standards, guidelines and styles.

• CODE REVIEW:

→ code review and verification is the process of identify errors, failures and faults in the source code.

→ code review is a traditional method for the verification used in the software life cycle.

→ The following methods are used for code review:

1. code walkthrough
2. code inspection

code walkthrough:

code walkthrough is a technical and peer review process of finding mistakes in the source codes. i.e seminars.

code inspection:

It is similar to code walkthrough, it detect programming defects in the source codes. ~~ie~~

• SOFTWARE DOCUMENTATION:

→ Software development, Operation, and maintenance process include various kinds of documents.

→ Documents acts as a communication medium between different team members of development.

→ It helps user in understanding the system operations.

→ It is also known as "User manuel".

→ Documents are also designed for planning, managing, and implementation of development and maintenance activities.

→ The categories of documentation done in the System:

- Internal documentation
- System documentation
- User documentation
- process documentation
- Daily documentation

• TESTING:

"Software testing is the process of finding defects in the software", so that these can be debugged and the defect-free software can meet the customer needs and expectations.

• UNIT TESTING:

In unit testing we are going to test the individual component & individual units of software System.

The main aim of this unit testing is to find bugs by isolating an individual module using test stub and test drivers and by executing test cases on it.

- INTEGRATION TESTING:

→ In integration testing we are going to combine the individual units a group, then after testing is perform.

→ The main purpose of integration testing is to identify the faults in interaction between integrated units.

→ There are various approaches in integration testing.

1. Top down approach — It begins with the main module and move downwards.
2. Bottom up approach — It begins with the bottom modules to top module.

- SYSTEM TESTING:

→ The system testing is also known as "System wide event".

→ It checks the entire software systems.

→ The main purpose of this system testing is to evaluate the system is worked with the specified requirements or not.

- ACCEPTANCE TESTING

→ In acceptance testing, where a system is tested for acceptability.

- REGRESSION TESTING:

→ Regression testing is also known as program revalidation

→ Regression testing is performed whenever new functionality is added or the existing functionality is modified in the program. For example 1 ← 2 ← 3 ← 4 (Backtracking)

• BLACK BOX TESTING:

→ Black box testing is one of the software, which checks the functionality of an application without Knowing the internal structures of the application.

→ Black box testing is an external behaviour of the System or application.

→ The following are some of the methods used in black box testing.

1. Equivalence class partitioning.
2. Boundary value Analysis (BVA)
3. cause effect graph.

1. Equivalence class partitioning:

→ In equivalence class partitioning we are going to divide the input values into two parts.

ⓐ Valid input partitioning

ⓑ Invalid input partitioning.

→ Among these two partitions we are going to take the ~~responsible~~ representative values from each part as text data.

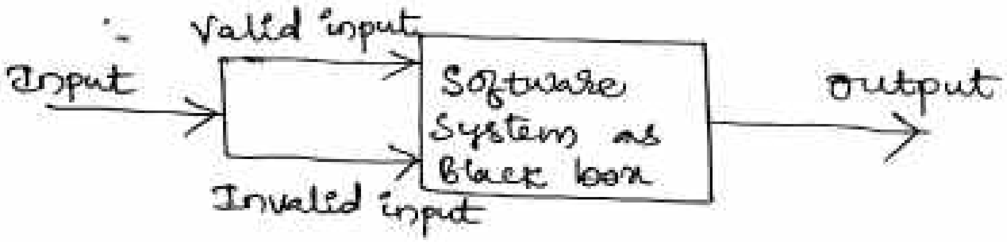


fig: Equivalence class partitioning.

Example :

1. A number start with an optional sign
2. The optional sign can be followed by any no. of digits.
3. After the decimal point then there should two digits after the decimal

Applying equivalence class partitioning by using Black box testing technique.

Sol :

Test case id	number	expected result
1.	1234	Valid number
2.	−10	valid number
3.	− 567 . 24	valid number
4.	567 . 24	valid number .
5 .	568 . 2	Invalid number.

2. Boundary Value Analysis (BVA)

→ It is one of the software test design technique, that involves determining the boundaries for input values and selecting values.

→ In boundary value Analysis inside boundary ranges is applicable.

→ The boundary value Analysis follows some ~~[illegible]~~ categories.

(i) Min value

(ii) Min+ value

(iii) Max value

(iv) Max_value

(v) Nominal value

Example: A program reads an integer number with range [1,100] and determine whether it is a even number or odd number, design test cases Using BVA?

Sol: Total number of test cases ⇒ = 4n+1 (where n=1)
= 5

Min value = 1
Min+ value = 2
Max value = 100
Max_value = 99
Nominal value = 50.

Test cases:

Test Case Id	Integer variable	Expected output
1	1	odd number
2	2	Even number
3	100	Even number
4	99	odd number
5.	50	Even number.

3. Cause effect graph:

→ This technique is used to identify the cause (input condition) and effect (output condition).

→ This techniques produce a cause effect graph generate test cases accordingly.

Example: 'cash withdrawal' depends upon the 'valid pin' 'valid amount' and 'cash availability' in the account. Using cause effect graph techniques.

causes:

C1: Enter valid amount

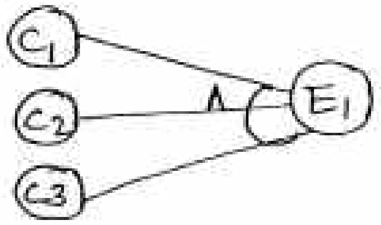
C2: Enter valid pin

C3: cash availability in the account

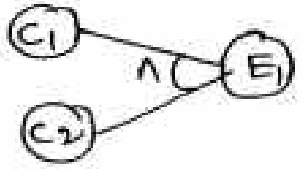
Effects: E1: cash withdrawal

The cause effect graph using notations logical functions AND, OR, NOT.

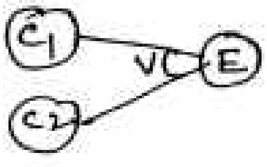
Step① E1: will be successful if C1, C2, C3 are true.



Step② E1: will be successful if C1, C2 are true.



step③ E1: will be unsuccessful if C1, C2 are fail.



step④: E1: is unsuccessful if C1 is fail

• WHITE BOX TESTING :

→ White box testing is also known as "internal behaviour of System".

→ It is also called "glass box testing".

→ White box testing is one of the testing method that will test the internal structure of an application or programs.

→ White box testing techniques are

1. control flow graph
2. cyclomatic complexity of the program
3. Independant paths

Consider the following program segment :

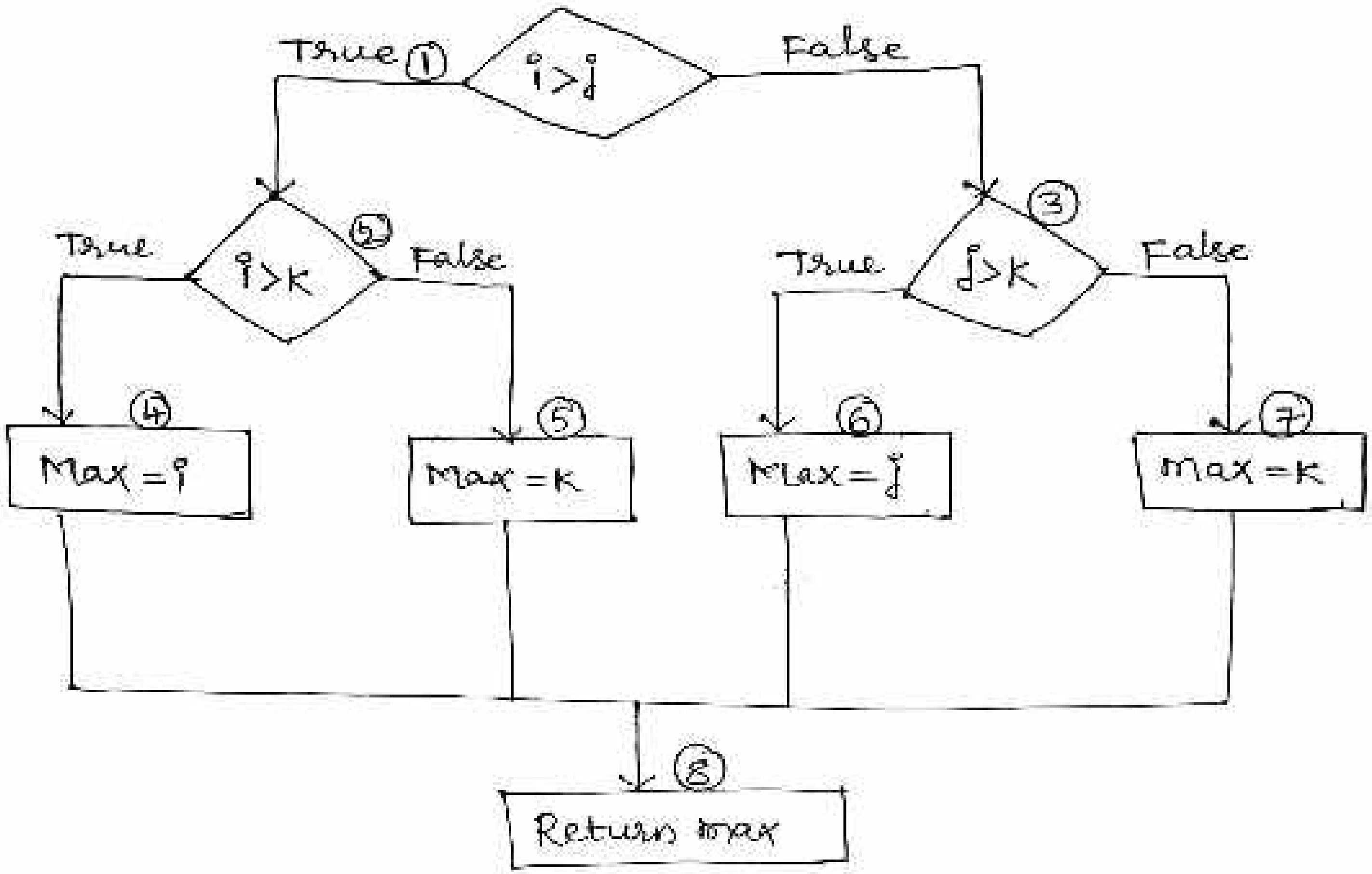
```
1. int max(int i, int j, int k)
2. {
3. int max;
4. if(i>j) then
5. if(i>k) then max=i;
6. else max=k;
7. else if (j>k) max=j;
8. else max=k
9. return(max);
10. }
```

ⓐ Draw control flow graph for this program segment

ⓑ Determine the cyclomatic complexity for this program.

ⓒ Determine the independent paths.

ⓐ Draw control flow graph for this program segment.



(ii) Cyclomatic complexity

$V(G) = E - n + P$

$= 10 - 8 + 2$

$= 4$

∵ E → no. of edges
n → no. of nodes
P → no. of components ie
True (or) False.

(iii) Independant paths

1 — 2 — 4 — 8

1 — 2 — 5 — 8

1 — 3 — 6 — 8

1 — 3 — 7 — 8 .

(iv) Test cases

Test Case Id	Input Values			Expected output	Independent paths	Remarks
	i	j	k			
TC 01	5	3	2	5	1-2-4-8	max i
TC 02	5	3	6	6	1-2-5-8	max k
TC03	3	5	2	5	1-3-6-8	max j
TC 04	3	5	6	6	1-3-7-8	max k

• DEBUGGING:

→ Debugging is nothing but, a process of identifying or analyzing the errors and "removing the errors".

→ Debugging is done at all the levels of testing.

→ The following are some of the methods which are used in debugging approaches.

1. Brute force method
2. Back tracking method

1. Brute force method:

It is the simplest method of debugging. It uses memory dumps or output statements for debugging.
i.e snapshot of execution sequence.

2. Backtracking method:

Backtracking is the refinement of brute force method and it is one of the successful methods of debugging.
" It begins from where the bug is discovered and the source code is traced out backtrack.

• PROGRAM ANALYSIS TOOL:

→ program analysis tools are automated tools, which provide additional information in addition to the output produced by the translators.

→ These tools takes source code or executable code as input and produce required information. These tools are useful to find program size, complexity.

→ program analysis tools are classified into two categories.

1. Static analysis tool

2. Dynamic analysis tool.

1. Static analysis tool:

Static program analysis tools look at the source code without executing it and detect possible defects before running the program.

Example: Snapshots are considered as static program analysis techniques, and code walkthrough and code inspection review techniques are also considered.

2. Dynamic analysis tool:

Dynamic analysis tools look at the flow of execution of the program. It analyses the dynamic behaviour of the program at run time.

Example: In runtime environment the library files, data structured are considered.

• TESTING OBJECT-ORIENTED PROGRAMS:

Object-Oriented programming has several benefits for developing real world problems. But Object-Oriented software testing has to deal with new problems introduced by Object Oriented features such as

1. Encapsulation
2. Inheritance
3. polymorphism
4. message passing
5. Dynamic binding.

These features encounter new Kinds of bugs and issues, solved by unit testing, integration testing, system testing.

• SOME GENERAL ISSUES ASSOCIATED WITH TESTING:

In this section, we will discuss usability testing, regression testing and smoke testing.

Usability testing:

→ Usability testing is concerned with user interaction.

→ Usability refers to the ease of use and comfort that users have while working with software.

→ There are three types of usability tests

1. scenario test
2. prototype test
3. product test.

Regression testing:

→ Regression testing is performed after program revision.

→ There are various techniques ① test all ② test minimization ③ test prioritization ④ Random selection.

Smoke Testing: Smoke testing is carried out to check major functionalities of the system. Smoke testing taken Hardware devices

Important Questions :

1. What is meant by testing? And what are different testing strategies?
2. Differentiate between Black box testing and White box testing?
3. What is meant by Debugging?
4. What are the Black box testing techniques? Explain briefly?
5. What are the White box testing techniques? Explain briefly?
6. What are the principles of Coding?

Error Guessing :

The error guessing technique is a black box testing techniq It is done by programmer. Error guessing is an intuitive and adhoc process of testing. The possible errors or error-prone situations are listed and then ~~tested~~ test cases are written for such errors.

The boolean variables have the values true (1) or false (0). The test cases are written for particular input values which have a 0 or 1.

The NULL position is another error-prone are in program.