

SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE NAME : 19CS511 SOFTWARE TESTING

III YEAR / V SEMESTER

Unit 2- TEST CASE DESIGN STRATEGIES

Topic 8 : Using White Box Approach to Test design





Using White Box Approach to Test design - **Problem**



- White box testing can be quite complex and expensive.
- Developers who usually execute white box test cases detest it. The white box testing by developers is not detailed can lead to production errors.
- White box testing requires professional resources, with a detailed understanding of programming and implementation.

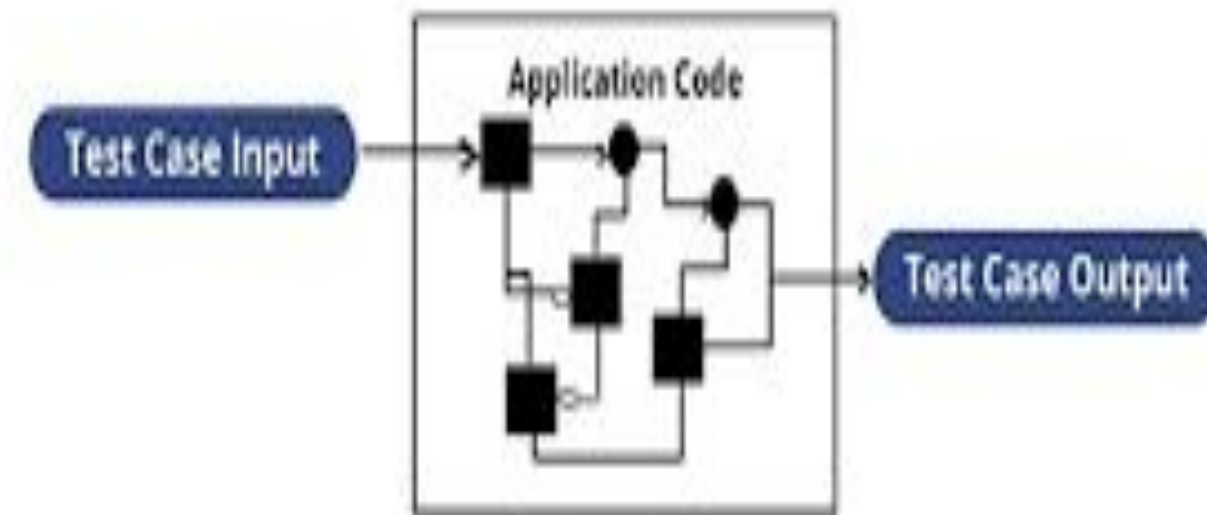


White Box Testing



White Box Testing is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

WHITE BOX TESTING APPROACH





What do you verify in White Box Testing?



White box testing involves the testing of the software code for the following:

- Internal security holes
- Broken or poorly structured paths in the coding processes
- The flow of specific inputs through the code
- Expected output
- The functionality of conditional loops
- Testing of each statement, object, and function on an individual basis



White Box Testing Techniques



Following are important White Box Testing Techniques:

- Statement Coverage
- Decision Coverage
- Branch Coverage
- Condition Coverage
- Multiple Condition Coverage
- Finite State Machine Coverage
- Path Coverage
- Control flow testing
- Data flow testing



Test adequacy criteria



- The goal for white box testing is to ensure that the internal components of a program are working properly. A common focus is on structural elements such as statements and branches.
- Program-based adequacy criteria are commonly applied in white box testing. They use either logic and control structures, data flow, program text, or faults as the focal point of an adequacy evaluation

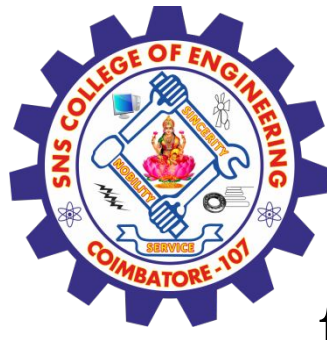


Test adequacy criteria-Cont..



The criteria can be viewed as representing minimal standards for testing a program. The application scope of adequacy criteria also includes:

- Helping testers to select properties of a program to focus on during test;
- Helping testers to select a test data set for a program based on the selected properties;
- Supporting testers with the development of quantitative objectives for testing;
- Indicating to testers whether or not testing can be stopped for that program.



Test adequacy criteria-Cont..



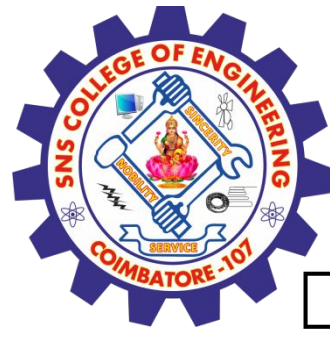
the system is performing as per the required specifications.

- **Test case** :It is a set of inputs, execution conditions, and also contains pass/fail criteria.
- **Test case specification** :It is a requirement that one or more test case must satisfy.
- **Test Suite** : A collection of test cases.
- **Test execution** :The process of execution of test cases.
- **Adequacy Criteria** :This is the *condition/predicate* which is either true or false for a specific pair of program & test suite.



Test case selection and adequacy criteria aims to full fill the following

1. **Statement Coverage** - A test case is designed with an aim to cover all the statements in a program. The test adequacy criteria in this case would be to execute all the statements.
2. **Branch Coverage** - The test adequacy criteria is to make sure the test cases cover the branches or the condition statements.
3. **Path Coverage** - Here, the adequacy criteria ensures that a specific path is covered, that is, from the entry point to exit.
4. **Mutation adequacy** - A program is injected with a fault. This is called the mutant program. Now the original and the mutant program are executed



Categories of test data adequacy criteria



□ Categories of test data adequacy criteria :

- **Specification-based** - Based on the requirement specification, test adequacy criteria is set. Testing is based on the features of the specification. Test criteria thus ensures that the requirements have been fully exercised.
- **Program-based** - Determines testing requirements as per the program under test.
- **Combined specification and program based criteria** - Combines both the concepts to carry out testing.
- **Interface Based Criteria** - Testing requirements are fixed based on the type of input and the range of inputs to be given to a program.



Activity



Advantages

1. Code optimization by finding hidden errors.
2. White box tests cases can be easily automated.
3. Testing is more thorough as all code paths are usually covered.
4. Testing can start early in SDLC even if GUI is not available.

Disadvantages

1. White box testing can be quite complex and expensive.
2. Developers who usually execute white box test cases detest it. The white box testing by developers is not detailed can lead to production errors.
3. White box testing requires professional resources, with a detailed understanding of programming and implementation.
4. White-box testing is time-consuming, bigger programming applications take the time to test fully.



Assessment 1



1. List out the Advantages of white box test

- a) _____
- b) _____
- c) _____
- d) _____

2. Identify the Disadvantages of white box test

- a) _____
- b) _____
- c) _____
- d) _____





TEXT BOOKS:

1. Ricardo Baeza-Yates and Berthier Ribeiro-Neto, —Modern Information Retrieval: The Concepts and Technology behind Search, Second Edition, ACM Press Books, 2011.
2. Ricci, F, Rokach, L. Shapira, B.Kantor, —Recommender Systems Handbook, First Edition, 2011.

REFERENCES:

1. C. Manning, P. Raghavan, and H. Schütze, —Introduction to Information Retrieval, Cambridge University Press, 2008.
2. Stefan Buettcher, Charles L. A. Clarke and Gordon V. Cormack, —Information Retrieval: Implementing and Evaluating Search Engines, The MIT Press, 2010.

THANK YOU