**PRACTICAL NO 4**

**Description:** Dynamic testing (or dynamic analysis) is a term used in software engineering to describe the testing of the dynamic behavior of code. That is, dynamic analysis refers to the examination of the physical response from the system to variables that are not constant and change with time.

**AIM: To study of White Box Testing in software testing.**

**Theory:**

**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 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

The testing can be done at system, integration and unit levels of software development. One of the basic goals of white box testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug.

**White Box Testing Techniques**

**I. Statement Coverage:**

Statement Coverage is a white box testing technique in which all the executable statements in the source code are executed at least once. It is used for calculation of the number of statements in source code which have been executed. The main purpose of Statement Coverage is to cover all the possible paths, lines and statements in source code. Statement coverage is used to derive scenario based upon the structure of the code under test.

**Statement coverage = (Number of executed statements / Total number of statements) x 100**

**The value of this Technique:**

* Unreachable code can be detected.
* Empty ELSE-parts are not considered.
* Coverage is measured using test tools.

**II. Branch Coverage:**

Branch Coverage is a white box testing method in which every outcome from a code module (statement or loop) is tested. The purpose of branch coverage is to ensure that each decision condition from every branch is executed at least once. It helps to measure fractions of independent code segments and to find out sections having no branches. For example, if the outcomes are binary, you need to test both True and False outcomes. The formula to calculate Branch Coverage:

Branch Coverage = Number of Executed Branches / Total Number of Branches

**The value of this Technique:**

* More test cases are necessary
* Tools often support determining coverage
* Inadequate for object-oriented systems
* It is used to detect uncalled methods or program parts

**III. Path Coverage:**

All possible paths through the test object. Path coverage requires the execution of all

different paths through the test object. An execution path is a set of nodes and directed edges in a flow graph that connects the start node to a terminal node.

The number of Basic paths is E – N + 2 (Linear Complexity)

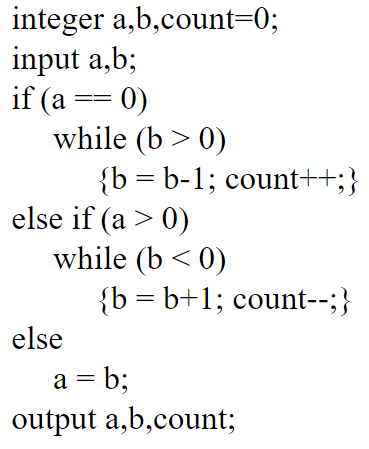
**Advantages of White Box Testing**

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

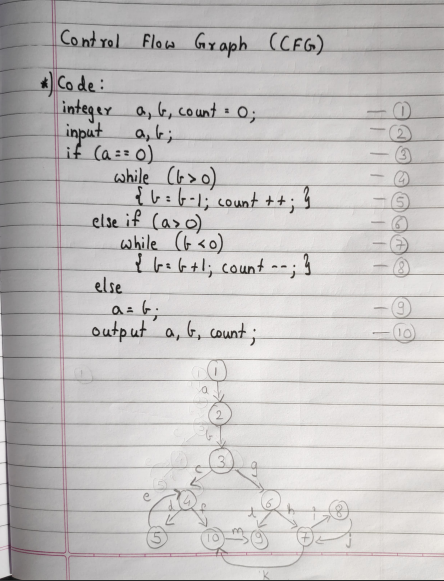
**Disadvantages of White Box Testing**

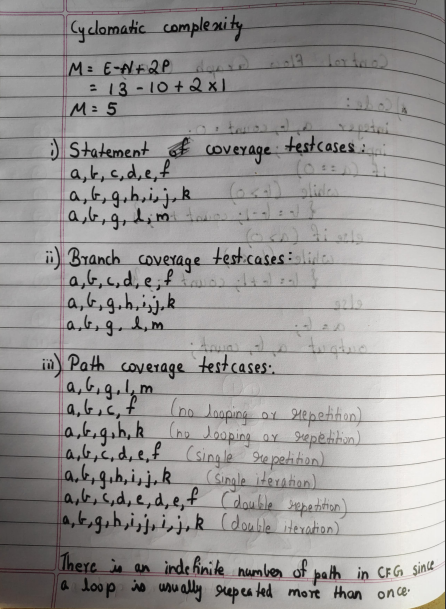
* White box testing can be quite complex and expensive.
* Developers who usually execute white box test cases detest it. The white box testing by developers is not detailed can lead to production errors.
* White box testing requires professional resources, with a detailed understanding of programming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

**Draw CFG for the code given. Find Cyclomatic complexity, Statement coverage, branch coverage and path coverage testcases.**



**Control Flow Graph (CFG):**





**CONCLUSION:** We have studied the White Box Testing in software testing.