

Path Testing & Basis Path Testing

What is Path Testing?

Path testing is a structural testing method that involves using the source code of a program in order to find every possible executable path. It helps to determine all faults lying within a piece of code. This method is designed to execute all or selected path through a computer program.

Any software program includes, multiple entry and exit points. Testing each of these points is a challenging as well as time consuming. In order to reduce the redundant tests and to achieve

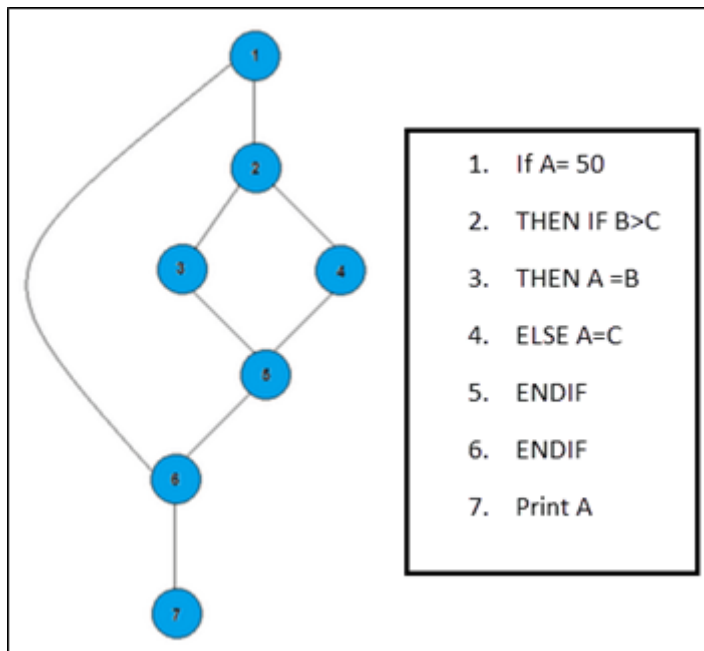
maximum test coverage, basis path testing is used.

What is Basis Path Testing?

The basis path testing is same, but it is based on a [White Box Testing \(/white-box-testing.html\)](/white-box-testing.html) method, that defines test cases based on the flows or logical path that can be taken through the program. Basis path testing involves execution of all possible blocks in a program and achieves maximum path coverage with least number of test cases. It is a hybrid of branch testing and path testing methods.

The objective behind basis path testing is that it defines the number of independent paths, thus the number of test cases needed can be defined explicitly (maximizes the coverage of each test case).

Here we will take a simple example, to get better idea what is basis path testing include



2016/032216_1047_LearnPathBa1.png)

In the above example, we can see there are few conditional statements that is executed depending on what condition it suffice. Here there are 3 path or condition that need to be tested to get the output,

- **Path 1:** 1,2,3,4,5,6, 7
- **Path 2:** 1,2,4,5,6, 7
- **Path 3:** 1, 6, 7

Steps for Basis Path testing

The basic steps involved in basis path testing include

- Draw a control graph (to determine different program paths)
- Calculate Cyclomatic complexity (</cyclomatic-complexity.html>) (metrics to determine the number of independent paths)
- Find a basis set of paths
- Generate test cases to exercise each path

Benefits of basis path testing

- It helps to reduce the redundant tests
- It focuses attention on program logic
- It helps facilitates analytical versus arbitrary case design

- Test cases which exercise basis set will execute every statement in program at least once

Conclusion:

Basis path testing helps to determine all faults lying within a piece of code.

◀ [Prev \(/python-unit-testing-guide.html\)](/python-unit-testing-guide.html) [Report a Bug](#) [Next > \(/performance-testing.html\)](/performance-testing.html)

YOU MIGHT LIKE:

BLOG



(/testng-tutorial.html)
(/testng-tutorial.html)

TestNG Tutorial: Project Workspace, Library, Test Cases

(/testng-tutorial.html)

BLOG



(/zephyr-agile-jira.html)
(/zephyr-agile-jira.html)

Zephyr Agile for JIRA Tutorial

(/zephyr-agile-jira.html)

BLOG



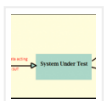
(/what-is-rad-rapid-software-development-model-advantages-disadvantages.html)
(/what-is-rad-rapid-software-

development-model-advantages-disadvantages.html)

What is RAD Model? Advantages & Disadvantages

(/what-is-rad-rapid-software-development-model-advantages-disadvantages.html)

SOFTWARE TESTING

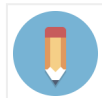


(/fuzz-testing.html)
(/fuzz-testing.html)

Fuzz Testing Tutorial: Fuzzing Strategy & Tools

(/fuzz-testing.html)

SOFTWARE TESTING



(/test-coverage-in-software-testing.html) (/test-coverage-in-software-testing.html)

Test coverage in software testing

(/test-coverage-in-software-testing.html)

SOFTWARE TESTING



(/defect-density-software-testing-terminology.html)
(/defect-density-software-testing-terminology.html)

Defect Density: Software Testing Terminology

(/defect-density-software-testing-terminology.html)

Testing Tutorials

[40\) Mainframe Testing \(/mainframe-testing.html\)](#)

[41\) Testing for Retail POS System \(/testing-for-retail-pos-point-of-sale-system.html\)](#)

[42\) Healthcare Application Testing \(/healthcare-application-testing-with-sample-test-cases.html\)](#)

[43\) Testing Ecommerce Applications \(/testing-e-commerce-applications.html\)](#)

[44\) Testing Insurance Applications \(/testing-insurance-applications-with-sample-testcases.html\)](#)

[45\) Testing Telecom Application \(/testing-telecom-application-with-sample-testcases.html\)](#)

[46\) Business Intelligence Testing \(/business-intelligence-testing-sample-test-cases.html\)](#)

[47\) White Box Testing \(/white-box-testing.html\)](#)

[48\) Static Testing \(/testing-review.html\)](#)

[49\) Cyclomatic Testing \(/cyclomatic-complexity.html\)](#)

[50\) Python Unit Testing \(/python-unit-testing-guide.html\)](#)

[51\) Basis Path Testing \(/basis-path-testing.html\)](#)

[52\) Performance Testing \(/performance-testing.html\)](#)

[53\) Load Testing \(/load-testing-tutorial.html\)](#)

[54\) Stress Testing \(/stress-testing-tutorial.html\)](#)

[55\) Volume Testing \(/volume-testing.html\)](#)

[56\) Scalability Testing \(/scalability-testing.html\)](#)

[57\) Soak Testing \(/soak-testing.html\)](#)

[58\) Stability Testing \(/stability-testing.html\)](#)

[59\) Software Testing Metrics \(/software-testing-metrics-complete-tutorial.html\)](#)

[60\) Create Test Strategy Document \(/how-to-create-test-strategy-document.html\)](#)

[61\) Testing as a Service \(/what-is-testing-as-a-service-taas.html\)](#)

[Contact US \(/contact-us.html\)](/contact-us.html)

[About US \(/about-us.html\)](/about-us.html)

[Advertise with Us \(/advertise-us.html\)](/advertise-us.html)

[Write For Us \(/become-an-instructor.html\)](/become-an-instructor.html)

Career Suggestion

[SAP Career Suggestion Tool \(/best-sap-module.html\)](/best-sap-module.html)

[Software Testing as a Career \(/software-testing-career-complete-guide.html\)](/software-testing-career-complete-guide.html)

Interesting

[Books to Read! \(/books.html\)](/books.html)

[Suggest a Tutorial](#)

[Blog \(/blog/\)](/blog/)

[Quiz \(/tests.html\)](/tests.html)

Execute online

[Execute Java Online \(/try-java-editor.html\)](/try-java-editor.html)

[Execute Javascript \(/execute-javascript-online.html\)](/execute-javascript-online.html)

[Execute HTML \(/execute-html-online.html\)](/execute-html-online.html)

[Execute Python \(/execute-python-online.html\)](/execute-python-online.html)

© Copyright - Guru99 2017

[Privacy Policy \(/privacy-policy.html\)](/privacy-policy.html)



