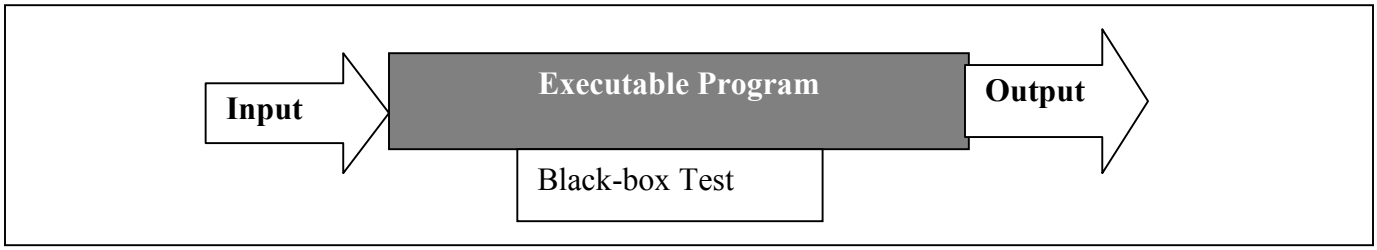
Black Box Testing Defined

Steven Bucksbaum, March 20, 2011

Black box testing, also called functional testing and behavioral testing, focuses on determining whether or not a program does what it is supposed to do based on its functional requirements. Black box testing attempts to find errors in the external behavior of the code in the following categories:

1. Incorrect or missing functionality;
2. Interface errors;
3. Errors in data structures used by interfaces;
4. Behavior or performance errors; and
5. Initialization and termination errors.



Develop a list of tests to conduct. The list is a "check-off" so no test is accidentally by-passed. The list also allows for regression testing, testing older functionality to be sure that new features did not create new "bugs" in older application features.

Format your test cases:

Column 1 Test ID	Column 2 Description	Column 3 Expected Results	Column 4 Actual Results

1. Col 1-Give each test a unique identifier
2. Col 2-Describe steps or input for the condition you want to test.
3. Col 3-Expected Results for the input or output
4. Col 4-Actual results and if the test succeeded or failed. If "PASSES" record PASS. If the test FAILS, add a description of the failure and "what came out".

Test case should be written in clear and understandable language and the test case execution is repeatable.

1. Note any necessary pre-conditions for the test case.
2. If helpful, provide "screen shots" of the user interface, noting what to look for.
3. Expected results should be very specific.
4. Test cases based on customer requirements
5. Begin writing test cases with the most used path of execution or functionality.
6. A good test case uncovers a different class of errors. Test cases should not repeat tests.
7. Bugs lurk in corners and congregate at boundaries." Programmers often make mistakes on the boundaries of the equivalence classes/input domain.
8. Boundary Value Analysis (BVA) and guides you to create test cases at the "edge" of the equivalence classes. Boundary value is defined as a data value that corresponds to a minimum or maximum input, internal, or output value specified for a system or component
9. Decision tables are used to record complex business rules that must be implemented in the program, and therefore tested.