

Homework 3

1. Problem 1

Test Classes

Number of inputs

Values: 0, 1, 2, 3

Input integer value

Values: Non-negative

Test Cases

1. Test Condition: 0 inputs
Test Input:
Test Output: USAGE message
2. Test Condition: 1 input
Test Input: 4!
Test Output: 12
3. Test Condition: 2 inputs
Test Input: 10, 2
Test Output: 90
4. Test Condition: 3 inputs
Test Input: 8 5 -
Test Output: 56

Justification

Minimal testing would test only the explicit requirements defined for the program, serving as verification that the system changes have met them. Even though we are skipping over testing implicit requirements, as long as each one is met, then we have “sufficiently” tested the program.

Here, we utilize Weak EC testing in order to minimally test the program, essentially creating one test for each variable in each parameter. We are also only looking to test the explicit requirements listed in the USAGE description. Any implicit requirements such as how to handle error situations where the number of inputs is greater than 4, or if the first two inputs are not integers, will not be tested.

Also, in this case we don't need a specific test for the non-negative value input integer type since the other 5 tests already cover it.

2. Problem 2

$$(((a + b) < c) \wedge \neg p) \vee (r > s)$$

- Let $a + b < c$ be the atomic condition **x**.
- Let $r > s$ be the atomic condition **y**.
- Thus, the compound condition with its atomic conditions is:
 $(x \wedge \neg p) \vee y$

Atomic Condition	Test Specification			
	Spec 1	Spec 2	Spec 3	Spec 4
x	T	F	T	T
p	F	F	T	T
y	F	F	F	T
Result	T	F	F	T

Result for x covered by: Spec 1, Spec 2

Result for p covered by: Spec 1, Spec 3

Result for y covered by: Spec 3, Spec 4

Test Cases

(a, b, c, p, r, s)

Spec: 1

Test Input: (1, 2, 5, F, 3, 7)

Test Output: True

Spec: 2

Test Input: (1, 2, 1, F, 3, 7)

Test Output: False

Spec: 3

Test Input: (1, 2, 5, T, 3, 7)

Test Output: False

Spec: 4

Test Input: (1, 2, 5, T, 7, 3)

Test Output: True