Jameson Burroughs, Christopher Lewis
Megan O'Neal, Alex Wray

SilverBullets

Objective:
Our team chose the open source project Sugar. Our goal was to evaluate one of the built-in activities, test it, collect the results, and compare them to an expected output in order to confirm that the activity works properly. We chose the Calculator activity inside of Sugar, and within the activity, five different functions to test. One of our later objectives was to inject faults into the activity's code, in order to have the functions return some incorrect results. We wanted to see what it took to break the code in some ways, but not all. One of our overall goals was also to meticulously document our experiences throughout the processs.

All Twenty five of our test cases in a table resulting from the script, before any faults have been implemented.

Test Report

Test Number	Tested Method	Inputs	Tested Executable file	Test Results	Oracle	Pass/Fail
1	add(x,y)	6 , 2	TestCase1.py	8	8	Pass
2	add(x,y)	1 , "testString"	TestCase2.py	TypeError	TypeError	Pass
3	add(x,y)	-1 , -1	TestCase3.py	-2	-2	Pass
4	add(x,y)	0 , 7	TestCase4.py	7	7	Pass
5	add(x,y)	.99999999999999999999 , .00000000000000000001	TestCase5.py	1.0	1.0	Pass
6	sub(x,y)	-1 , 3	TestCase6.py	-4	-4	Pass
7	sub(x,y)	-5 , -10	TestCase7.py	5	5	Pass
8	sub(x,y)	.1911 , .9998089	TestCase8.py	-0.8087089	-0.8087089	Pass
9	sub(x,y)	"silver" , "bullets"	TestCase9.py	TypeError	TypeError	Pass
10	sub(x,y)	"00099" , 66	TestCase10.py	TypeError	TypeError	Pass
11	mul(x,y)	0 , .55555	TestCase11.py	0.0	0.0	Pass
12	mul(x,y)	-7 , -7	TestCase12.py	49	49	Pass
13	mul(x,y)	8 , -8	TestCase13.py	-64	-64	Pass
14	mul(x,y)	.000001 , .011111111	TestCase14.py	1.1111111e-08	1.1111111e-08	Pass
15	mul(x,y)	"silver" , 4	TestCase15.py	silversilversilversilver	TypeError	Fail
16	div(x,y)	4 , 0	TestCase16.py	ValueError	ValueError	Pass
17	div(x,y)	0 , 4	TestCase17.py	0	0	Pass
18	div(x,y)	"hhh8" , 100	TestCase18.py	TypeError	TypeError	Pass
19	div(x,y)	5.18 , 1.65	TestCase19.py	3.13939393939	3.13939393939	Pass
20	div(x,y)	1 , .999999999999	TestCase20.py	1.0	1.0	Pass
21	mod(x,y)	-340 , 60	TestCase21.py	20	20	Pass
22	mod(x,y)	340 , 60	TestCase22.py	40	40	Pass
23	mod(x,y)	0 , 10000	TestCase23.py	0	0	Pass
24	mod(x,y)	4 , 3.14	TestCase24.py	ValueError	.86	Fail
25	mod(x,y)	-100 , -23.14	TestCase25.py	ValueError	-7.44	Fail

Sugar Info:
Sugar is a learning platform that reinvents how computers are used for education. Collaboration, reflection, and discovery are integrated directly into the user interface. Sugar promotes "studio thinking" and "reflective practice". Through Sugar's clarity of design, children and teachers have the opportunity to use computers on their own terms. Students can reshape, reinvent, and reapply both software and content into powerful learning activities. Sugar's focus on sharing, criticism, and exploration is grounded in the culture of free software (FLOSS).

Source: http://wiki.sugarlabs.org/go/What_is_Sugar%3F

Test Case Example:
TestNumber: 1
TestedRequirement: Returns the sum of two dat values will return their sum
TestedComponent: functions.py
TestedMethod: add(x,y)
TestInputs: 6 , 2
Test Executable file: TestCase1.py
ExpectedOutcome : testCase1oracle.txt

This is our table after faults were injected into SugarLabs code, we did such things as changing addition to return the absolute value of the two numbers this not only resulted in the expected positive outcomes only but also added a floating point decimal to our numbers. Another interesting change was making it so mod didn't check for integers which allowed us to mod decimal numbers giving the code more funtionality.

Technical Overview:
Within sugarLabs we tested the funtcions for Addition, Subtraction, Multiplication, Division, and the Modulo operator. Within these operators we tested the use of regular integer numbers, large and extremely small numbers, as well as incorrect data types such as strings.

Project Contents (Folder Names & Descriptions):
/scripts/ - contains main python running script, runAllTests.py
/testCases/ - contains text files detailing each test case. Holds information to be passed
/project/src/ - location of the python Calculator activity that is being tested
/oracles/ - contains text files with expected outcomes of each test case
/testCaseExecutables/ - contains the python executable files to run each test case
/reports/ - contains the HTML file containing the aggregated result information
/Outputs/ - contains text files of test case executable results after the program is run

Test Report

Test Number	Tested Method	Inputs	Tested Executable file	Test Results	Oracle	Pass/Fail
1	add(x,y)	6 , 2	TestCase1.py	8.0	8	Fail
2	add(x,y)	1 , "testString"	TestCase2.py	TypeError	TypeError	Pass
3	add(x,y)	-1 , -1	TestCase3.py	2.0	-2	Fail
4	add(x,y)	0 , 7	TestCase4.py	7.0	7	Fail
5	add(x,y)	.99999999999999999999 , .00000000000000000001	TestCase5.py	1.0	1.0	Pass
6	sub(x,y)	-1 , 3	TestCase6.py	-4	-4	Pass
7	sub(x,y)	-5 , -10	TestCase7.py	5	5	Pass
8	sub(x,y)	.1911 , .9998089	TestCase8.py	-0.8087089	-0.8087089	Pass
9	sub(x,y)	"silver" , "bullets"	TestCase9.py	TypeError	TypeError	Pass
10	sub(x,y)	"00099" , 66	TestCase10.py	TypeError	TypeError	Pass
11	mul(x,y)	0 , .55555	TestCase11.py	None	0.0	Fail
12	mul(x,y)	-7 , -7	TestCase12.py	None	49	Fail
13	mul(x,y)	8 , -8	TestCase13.py	None	-64	Fail
14	mul(x,y)	.000001 , .011111111	TestCase14.py	None	1.1111111e-08	Fail
15	mul(x,y)	"silver" , 4	TestCase15.py	None	TypeError	Fail
16	div(x,y)	4 , 0	TestCase16.py	4	ValueError	Fail
17	div(x,y)	0 , 4	TestCase17.py	ValueError	0	Fail
18	div(x,y)	"hhh8" , 100	TestCase18.py	TypeError	TypeError	Pass
19	div(x,y)	5.18 , 1.65	TestCase19.py	3.13939393939	3.13939393939	Pass
20	div(x,y)	1 , .999999999999	TestCase20.py	1.0	1.0	Pass
21	mod(x,y)	-340 , 60	TestCase21.py	20	20	Pass
22	mod(x,y)	340 , 60	TestCase22.py	40	40	Pass
23	mod(x,y)	0 , 10000	TestCase23.py	0	0	Pass
24	mod(x,y)	4.1 , 3	TestCase24.py	1.1	.86	Fail
25	mod(x,y)	-100 , -23.14	TestCase25.py	-7.44	-7.44	Pass

Information and picutres used from our github and from the sugarlabs website https://www.sugarlabs.org/