Lab Report #1
ECE 322
<NAME>
<ID #>

Introduction:

The purpose of this lab was to introduce Linux Desktop Testing Project (LDTP), and provide experience in utilizing the cross platform application to perform automated user interface testing through the use of a python script. The completion of the exercises in the lab ensures that a person is exposed to various aspects of LDTP's API and the python language, while providing direct experience of developing automation scripts for testing using GUI events. The lab also provides an initial look at the process of developing and implementing a series of tests for the purpose of ensuring correctness of functionality.

Part 1:

Task one involved the creation of an automated script to test a set of functionalities within a text editor. The automation script was written in python, and used LDTP for GUI interactions. In the case of my implementation, the text editor was "Notepad" in a Windows 7 environment. The python script tested the following five separate functionalities:

1. Modify and save a document
2. Create a new document
3. Open the previously saved document
4. Cancel an open or save operation
5. Exit the application

The python script for part 1, "Part 1.py", completes the testing of this functionality in a different order than specified above. The script initially opens the open dialog and proceeds to cancel it - #4. The script then creates a new document, adds text to it and saves the file -#2. It then modifies the file by appending a piece of text to the file, and saving the file again (#1). The script proceeds to close the notepad application, and reopen it. The script then opens the previously saved file that created in the previous test (#3), and then closes the notepad application.-#5

Part 2:

The second part of the lab required basic functionality testing of a calculator application that was included in an operating system such as Windows or Linux. The implementation included with this report was completed for the calculator included with Window 7 - "calc.exe".

The basic functionalities tested by the script are the following:

1. Addition
2. Subtraction
3. Division
4. Multiplication

The tests for each of these functionalities involved the use of negatives, positives, and the basic use of floating point numbers. In total there are 33 separate tests that verify various aspects of the input domain. Some basic functionality, such as clearing the screen and the use of the various buttons, are also inadvertently verified in the tests. The final test of the script involves testing whether the calculator follows the operation priority rules set out by the acronym B.E.D.M.A.S. (Brackets, Exponents, Division, Multiplication, Addition, and Subtraction) The Windows 7 calculator has various modes such as standard and scientific. In standard mode this last test will fail since the calculation evaluates values as there are entered, but in scientific mode this test will succeed.

The script is composed of two separate files. "Part 2.py" contains the logic required for the actual tests, while "Part2_helper.py" contains the created framework that runs the logic of the tests through LDTP to the calculator. The script will run the various tests against their expected values, and will report the result of each test, success or failure, to the terminal. After all the tests are complete a final tally of tests, test failures, and test successes will be output to the terminal. These tests are located in the appendix of this report; Table 1: Calculator Test Cases.

Conclusion:

Linux Desktop Testing Project (LDTP) is a GUI application testing tool that allows the development of automation scripts through an event driven interface with an application's GUI. LDTP's ability to interface with an application's GUI allows for the implementation of automated functionality and regression testing. In general, LDTP worked exceptionally well for the development of the two automation scripts required by the lab. LDTP's API provided direct interactions with the notepad and calculator applications with no major issues. One minor issue experienced with LDTP, on the Windows operating system, was the excessive amount of time LDTP takes to process the elements in a menu. LDTP is a high quality testing framework that provides a foundation for building high quality testing and automation scripts, which will in turn lead to the development of higher quality software.

Appendix

Table 1: Calculator Test Cases

#	Test Case	Actual Case	Expected
1	0 + 0 = 0	0 + 0	0
2	0 * 0 = 0	0 * 0	0
3	0 - 0 = 0	0 - 0	0
4	a + 0 = a, pos + 0 = pos	1 + 0	1
5	a – 0 = a, pos – 0 = pos	1 – 0	1
6	pos * 0 = 0	1 * 0	0
7	0 + a = a, 0 + pos = pos	0 + 1	1
8	0 – a = -a, 0 - pos = neg	0 – 1	-1
9	0 * pos = 0	0 * 1	0
10	0 / pos = 0	0 / 1	0
11	neg + 0 = neg	-1 + 0	-1
12	neg - 0 = neg	-1 - 0	-1
13	neg * 0 = 0	-1 * 0	0
14	0 + neg = pos	0 + -1	-1
15	0 - neg = pos	0 - -1	1
16	0 * neg = 0	0 * -1	0
17	0 / neg = 0	0 / -1	0
18	pos + pos = pos	2 + 3	5
19	pos - bigger pos = neg	2 – 3	-1
20	bigger pos - pos = pos	3 – 2	1
21	neg * neg = pos	-1 * -1	1
22	neg * pos = neg	-2 * 3	-6
23	pos * neg = neg	2 * -3	-6
24	pos * pos = pos	2 * 3	6
25	pos / pos = pos	6 div 3	2
26	neg / pos = neg	-6 / 3	-2
27	pos / neg = neg	6 / -3	-2
28	large + large = larger	123456789 + 123456789	246913578
29	decimal + decimal = decimal	1.2 + 2.3	3.5
30	decimal * decimal = decimal	1.2 * 2.3	2.76
31	decimal / decimal = decimal	1.2 / 2.4	0.5
32	decimal - decimal = decimal	1.559 - 8.3473	-6.7883
33	BEDMASS	1 + 1 * 2	3