Sahar Alizadeh
Prof. Alex Groce
CS 569
Final Report
Email:alizades@oregonstate.edu

Introduction:

Nowadays, testing programs become substantial and noticeable for computer developers and programmers to find errors and faults. They need software's testing programs to be ensuring about quality and reliability of their programs. There are some techniques for testing depends on programs and testing algorithms. Software testing is most effort consuming phase in software development so minimizing the efforts and maximizing the number of faults detected and automated test case generation contributes to reduce cost and time effort. Software testing is the procedure of executing a program or system with the intent of finding faults. Testing is a process of confirming that product is working according to the specification and satisfying the customer needs. Software testing provides a means to reduce errors, cut maintenance and overall software costs. In random testing, programs are tested by generating random, independent inputs. Results of the output are compared against software specifications to verify that the test output is pass or fail. Random testing have some benefits such as having lower cost, is quick and it finds real bugs.
In this code I used mutation testing, which is used to design new software tests and evaluate the quality of existing software tests. Mutation testing involves modifying a program in small way. Each mutated version is called a *mutant* and tests detect and reject mutants by causing the behavior of the original version to differ from the mutant.

Related Work

Random testing has been used to find errors in many applications; a partial list includes Unix utilities, Windows GUI applications, Haskell programs, and Java programs [1]. JCrasher [3] creates test inputs by using a "parameter graph" to find method calls whose return values can serve as input parameters. RANDOOP does not explicitly create a parameter graph; instead it uses a component set of previously created sequences to find input parameters. Eclat creates tests that are likely to expose errors by performing random generation augmented by automatic pruning based on execution results. Eclat prunes sequences that appear to be illegal because they make the program behave differently than a set of correct training runs. Ferguson and Korel [1] proposed an input generation technique that begins by executing the program under test with a random input, and systematically modifies the input so that it follows a different path. Ferguson and Korel compared basic block coverage achieved by inputs generated using their chaining technique versus randomly generated inputs [1].

Implementation

Random testing selects inputs at random from the programmer's input space and

checks the program behaves correctly on each space. In this testing, program was tested by generating random test by independent inputs and the outputs are compered against to verify that the test output is pass or fail. Theoretical works indicated that random testing is as effective as more semantic input generation techniques but some empirical studies suggest that systematic testing is more effective than random. Besides, random Test is useful in wildly used and well-tested libraries and performs semantic and undirected test generation. Random testing is also used to find errors in many applications include Unix utilities, Windows GUI applications, Haskell programs and Java programs [1].

In this report, I worked on SUT test and Template Scripting Testing Language (TSTL) and presented how it can be used to produce coverage and find bugs. I would explain about improvement of implementation of test generation programming using TSTL API and find improvement in testing part. The program was presented with name tester.py, which used sut.py.
There are 5 important command lines with these ascriptions:
1- Timeout: is represented time to cover a test with second.
2- Seed: It is for Python random objects and it used random number generation in the code
3- Depth: It is presented maximum length of a test in the algorithm.
4- Width: It is used for maximum memory/ BFS queue with searching width.
5-Faults: It is used for checking faults in SUT either is 0 or 1. If it is true, it should be saved for each discovered failures in the current directory as failure1.test failure2.test and etc.
6- Coverage: It should be reported coverage of testing as a final coverage report by using TSTL's internalReport () function.
7- Running: It should produced branch of coverage after running either is 0 or 1 by using TSTl randomtester.py.
One method or multiple methods with arguments are randomly picked to generate a sequence. The randomly generated sequence is append to the previous sequence to create new sequence. This new sequence is checked if it is already present or not. Contracts are used to find errors in the system and filters are used to avoid bad tests. In milestones, I have added code for random testing of creation of new branches. This code would test sut.py for the time passed as an argument. In this code if the argument running is set to 1, then I am checking for new branches in sut. If it is not equal to null set, then I iterate over the branches and print the time taken along with total branch count and the new branch in every iteration. If it is null then I do nothing. After that, I added these functions as improvement for each of the following arguments as bellows:

1- Faults: First, I checked faults; if it is 1 it incremented bugs and write sut.failure to a file. 2- Second, if coverage is 1 then it called internal report and it printed results.

Final Improvement

I implemented a novel test generation algorithm using TSTL APIs. I improved my previous testing code with increasing the number of coverage and branches with mutation. I also added new branches covered with random member selection, which is working based on length. I executed the code and removed test after implementation. The results is shown as below:
Fault Coverage is False:

```
10-248-106-213:alizades sahar$ python finaltester.py  30 2 3 4 0 1 1
STARTING POP BRANCHCOV 0
STARTING POP STATEMENTCOV 0
Entered here
0.013514995575 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (46, -44))
0.0135631561279 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (259, -257))
0.0135700702667 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 56))
0.0135760307312 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (45, 46))
0.0135819911957 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 45))
0.0135869979858 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (258, 259))
0.013592004776 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (56, -55))
0.0135982036591 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 258))
NEW BRANCHES DUE TO MUTATION: set([(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (74, -70)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (-1,                35)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',               (259,              -257)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (71,                73)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (46,               -44)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (38,                40)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (73,                74)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (40,               -34)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (35,                36)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (-1,                56)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (36,                37)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (-1,                45)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (-1,                71)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (45,                46)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (-1,               258)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',               (258,               259)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (56,               -55)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (37, 38))])
```

NEW BRANCHES FOUND

```
set([(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (71,                73)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (-1,                71)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py',                (74,               -70)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (73, 74))])
Entered here
1.20192909241 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (188, 189))
1.20194220543 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (184, 193))
1.20194816589 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (258, 262))
1.20195317268 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (264, -257))
1.20195817947 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (262, 263))
1.20196318626 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (177, 179))
1.20196819305 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (175, 176))
1.20197319984 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (185, 186))
1.20197820663 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (181, -171))
1.20198321342 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (187, 188))
1.20198702812 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (186, 187))
1.20199203491 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (176, 177))
1.2019970417 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (173, 174))
1.20200204849 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (174, 175))
1.20200705528 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (263, 264))
1.20201206207 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (193, -183))
1.20201706886 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (172, 181))
1.20202207565 77  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (189, 191))
```

NEW BRANCHES DUE TO MUTATION:

set([(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (173, 179)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (12, 13)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (177, 179)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (264, -257)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (102, 103)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 86)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (86, 88)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (104, 105)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (100, 102)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 124)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (103, 104)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (17, -15)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (173, 174)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (174, 175)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (184, 193)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (172, 181)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (90, 92)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (176, 177)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (6, -5)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (92, 94)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (185, 186)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (191, -183)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (124, 125)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 6)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (126, -119)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (193, -183)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (16, 17)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (98, 100)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 172)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (117, -85)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (188, 189)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (258, 262)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (262, 263)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (181, -171)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 11)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (187, 188)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (185, 191)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (106, 117)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (105, 106)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (172, 173)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (179, -171)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 16)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (94, 98)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (88, 90)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (263, 264)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (175, 176)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (184, 185)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (11, 12)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (186, 187)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 184)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (125, 126)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (189, 191)),
(u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (13, -10))])

NEW BRANCHES FOUND set([])
TSTL INTERNAL COVERAGE REPORT:

/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py ARCS: 79 [(-1, 6), (-1, 11), (-1, 16), (-1, 35), (-1, 45), (-1, 56), (-1, 71), (-1, 86), (-1, 124), (-1, 172), (-1, 184), (-1, 197), (-1, 258), (-1, 267), (6, -5), (11, 12), (12, 13), (13, -10), (16, 17), (17, -15), (35, 36), (36, 37), (37, 38), (38, 40), (40, -34), (45, 46), (46, -44), (56, -55), (71, 73), (73, 74), (74, -70), (86, 88), (88, 89), (88, 90), (89, 100), (90, 92), (92, 94), (94, 98), (98, 100), (100, 102), (102, 103), (103, 104), (104, 105), (105, 106), (106, 117), (117, -85), (124, 125), (125, 126), (126, -119), (172, 173), (172, 181), (173, 174), (173, 179), (174, 175), (175, 176), (176, 177), (177, 179), (179, -171), (181, -171), (184, 185), (184, 193), (185, 186), (185, 191), (186, 187), (187, 188), (188, 189), (189, 191), (191, -183), (193, -183), (197, 227), (227, -195), (258, 259), (258, 262), (259, -257), (262, 263), (263, 264), (264, -257), (267, 268), (268, -266)]
/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py LINES: 59 [6, 11, 12, 13, 16, 17, 35, 36, 37, 38, 40, 45, 46, 56, 71, 73, 74, 86, 88, 89, 90, 92, 94, 98, 100, 102, 103, 104, 105, 106, 117, 124, 125, 126, 172, 173, 174, 175, 176, 177, 179, 181, 184, 185, 186, 187, 188, 189, 191, 193, 197, 227, 258, 259, 262, 263, 264, 267, 268]

TSTL BRANCH COUNT: 79
TSTL STATEMENT COUNT: 59
Total Number of bugs 0

Total number of actions 103404
Total Runtime 30.0030620098
FINAL POP BRANCHCOV 79
FINAL POP STATEMENTCOV 59
10-248-106-213:alizades sahar$
Fault Coverage is True:

```
10-248-106-213:alizades sahar$ python finaltester.py  30 2 3 4 1 1 1
STARTING POP BRANCHCOV 0
STARTING POP STATEMENTCOV 0
Entered here
0.0093560218811 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (46, -44))
0.00940203666687 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (259, -257))
0.00940895080566 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 56))
0.00941491127014 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (45, 46))
0.00942087173462 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 45))
0.00942587852478 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (258, 259))
0.00943088531494 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (56, -55))
0.00945591926575 14  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (-1, 258))
Entered here
0.421013832092 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (188, 189))
0.421026945114 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (184, 193))
0.421032905579 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (258, 262))
0.421037912369 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (264, -257))
0.421043872833 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (262, 263))
0.421049833298 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (177, 179))
0.421054840088 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (175, 176))
0.421059846878 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (185, 186))
0.421065807343 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (181, -171))
0.421070814133 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (187, 188))
0.421077013016 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (186, 187))
0.4210820198
06 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (176, 177))
0.42108798027 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (173, 174))
0.421092987061 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (174, 175))
0.421097993851 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (263, 264))
0.421103954315 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (193, -183))
0.421108961105 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (172, 181))
0.421115875244 73  branch (u'/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py', (189, 191))
```

TSTL INTERNAL COVERAGE REPORT:

```
/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py ARCS: 79 [(-1, 6), (-1, 11), (-1, 16), (-1, 35), (-1, 45), (-1, 56), (-1, 71), (-1, 86), (-1, 124), (-1, 172), (-1, 184), (-1, 197), (-1, 258), (-1, 267), (6, -5), (11, 12), (12, 13), (13, -10), (16, 17), (17, -15), (35, 36), (36, 37), (37, 38), (38, 40), (40, -34), (45, 46), (46, -44), (56, -55), (71, 73), (73, 74), (74, -70), (86, 88), (88, 89), (88, 90), (89, 100), (90, 92), (92, 94), (94, 98), (98, 100), (100, 102), (102, 103), (103, 104), (104, 105), (105, 106), (106, 117), (117, -85), (124, 125), (125, 126), (126, -119), (172, 173), (172, 181), (173, 174), (173, 179), (174, 175), (175, 176), (176, 177), (177, 179), (179, -171), (181, -171), (184, 185), (184, 193), (185, 186), (185, 191), (186, 187), (187, 188), (188, 189), (189, 191), (191, -183), (193, -183), (197, 227), (227, -195), (258, 259), (258, 262), (259, -257), (262, 263), (263, 264), (264, -257), (267, 268), (268, -266)]
/Users/sahar/cs569_new_alex/cs569sp16/projects/alizades/avl.py LINES: 59 [6, 11, 12, 13, 16, 17, 35, 36, 37, 38, 40, 45, 46, 56, 71, 73, 74, 86, 88, 89, 90, 92, 94, 98, 100, 102, 103, 104, 105, 106, 117, 124, 125, 126, 172, 173, 174, 175, 176, 177, 179, 181, 184, 185, 186, 187, 188, 189, 191, 193, 197, 227, 258, 259, 262, 263, 264, 267, 268]
```

TSTL BRANCH COUNT: 79
TSTL STATEMENT COUNT: 59
Total Number of bugs 0
Total number of actions 106428
Total Runtime 30.0021879673
FINAL POP BRANCHCOV 79
FINAL POP STATEMENTCOV 59

MyTester.py

In this code, I have used parameter declaration like in randomtester.py and improved my code. I can run the code without any parameters and the parameters are built by the code itself using the default values declared in the code.
This has improved the branch coverage and statement coverage.

```
Total number of actions 70000
Total Runtime 64.9264240265
FINAL POP BRANCHCOV 185
FINAL POP STATEMENTCOV 139
```

Reference

[1] Feedback directed random test generation, Carlos Pacheco, Shuvendu K. Lahiri, Micheal D. Ernst and Thomas Ball

[2] Model cheching programs, W. Visser, K. Havelund, G. Brat, S. Park and F. Lerda

[3] Test input generation for Java containers using state matching. W. Visser and R. Pelanek .

[4] Alex Groce coverTester.py. Retrived May 5, 2016, from https://github.com/agroce/cs569sp16/blob/ master/SUTs/coverTester.py.

[5] Alex Groce NewCover.py. Retrived May 5, 2016, from https://github.com/agroce/cs569sp16/blob/ master/SUTs/newCover.py