Shivani Wanjara

CS562

Applied Testing Software Engineering Project: Part 3

Library : https://github.com/nryoung/algorithms/tree/master/algorithms/data_structures

Using TSTSL we write test cases for the given library of program. I have used the library as mentioned above, where I have used the data structures section for testing. In which so far I have completed running test on few data structures like binary tree, stack and queue. Where I have concentrated currently on functionality of the codes. As provided in the previous reports are the test cases that test the basic functionality of the code and also checks if the program works with random input. Using Random tester as the required input for my test written in TSTL. The Random tester file provides inputs randomly to my test cases. However as we know when we use random tester will assign the values randomly at the same time randomly run the test coded as well.

Bugs Found :

While testing the code, I did not find any bugs as such. The functionality of the code so far seems correct and appropriate for the functions coded. Furthermore if bugs found, will update the same.

Binary tree:

Where each node has a key and the value of each key should be larger than the keys on the left side of the node and should be smaller than the keys on the right side of the node. This is tested, by assigning the key and value to the tree. I have tested put, contains, is_empty, delete, max and min functions of the binary search tree. The few cases I considered are as follows: Random tester assigns key and value to the tree, next it is checked if they contain the values which it would and hence not be empty, hence the condition would be not empty which is true, this is tested in case1.I have also tried to find the max and min node of the tree, using the key value of the Tree. Also deletion of the key once it has been inserted, to check if the condition is false or true. Testing of deletion of the key value in the tree is the deletion itself is successful or not.

Code Coverage :

Test.tstl :

44.1767068273 PERCENT COVERED

106 BRANCHES COVERED

73 STATEMENTS COVERED

Coverage.out

```
name                    Stmts   Miss Branch BrPart  Cover   Missing
-----------------------------------------------------------------
binary_search_tree.py     169     96     80      7    44%   14-19, 27-32, 35, 41, 51, 61, 63, 72, 82,
92, 111, 121, 128, 135, 145, 149, 156, 163, 173, 177-216, 231-304, 306, 325, 335, 343-395, 62-
>63, 127->128, 144->145, 155->156, 172->173, 224->231, 305->306
```

Test1 .tstl :

23.2492489815 TOTAL RUNTIME

21 BRANCHES COVERED

15 STATEMENTS COVERED

Coverage.out

```
Name                    Stmts   Miss Branch BrPart  Cover   Missing
-----------------------------------------------------------------
binary_search_tree.py     169    154     80      5     8%   14-19, 27-32, 35-92, 99-111, 121-304,
306, 308, 310, 315-325, 335-395, 95->99, 305->306, 307->308, 309->310, 313->315
```

Stack:

Similarly for stack testing I have added new values for the stack and check if the size of the stack has been increased or no after successful insertion of value in stack. The test will also increase the value of the size of stack. Function usability in Stack tested are stack.add() , stack.remove . stack.isempty() , stack.size() . These functions work as expected and no bugs were found in this

Code Coverage :

Test2.TSTL

25.0 PERCENT COVERED

6 BRANCHES COVERED

3 STATEMENTS COVERED

Coverage.out

```
Name      Stmts   Miss Branch BrPart  Cover   Missing
```

```
-----------------------------------------------------

stack.py      12      9      0      0    25%   13-16, 19, 27, 36-50
```

Queue:

With the queue I have tested the basic functionality of it. Addition of an element in the queue and is empty and remove functions. . Function usability in Queue tested are add,remove ,is_empty . These functions work as expected and no bugs were found in this function.

Code Coverage :

Test4.tstl :

```
23.0769230769 PERCENT COVERED

6 BRANCHES COVERED

3 STATEMENTS COVERED
```

Coverage.out

```
Name       Stmts   Miss Branch BrPart  Cover   Missing

-----------------------------------------------------

queue.py      13     10      0      0    23%   14-20, 23, 31, 40-54
```

Conclusion:

By the end of the project I will test few more data structures from the same library to test the functionality. I am also working on more complicated kind of test cases for the already tested data structures. Currently I am working on Linked List testing cases.