Varun Rajgopal
CS 569
Dr. Alex Groce

Bug/Progress Report

Bugs:
The python library that I have tested using TSTL is Bintrees. The library contains three trees that are Binary trees, AVL trees and Red black trees. As part of the first proposal I have tested the insertion and deletion functions for the Binary Trees. As part of my testing till now, I didn't find any bugs in the library. The code is neat and well tested.

The progress till date is as follows:

1. Generated tests for inserting a key and a value in to the binary Tree and check if the insertion is working as expected. During insertion the key is also appended to a list. This list helps during deletion.
2. Generated tests for removing a key from the binary Tree. Deleting an element from the binary Tree checks if the length of the binary tree is greater than 5 and key is already present in the list and then proceeds with the deletion from the binary Tree.
3. Implemented functions to display the tree structure during insertion and deletion.
4. Tested __init__ , __setitem__ and __getitem__ functions indirectly during insertion and deletion operations.
5. Tested Node and BinaryTree classes as part of the library.

Progress by end of term:

1. Implement python functions to test and check if the tree is a binary tree during Insertion and deletion operations.
2. Test the binary tree with different types of data types such as characters and check if the tree is a binary tree.
3. Test the heap methods such as max_items(), max_key(), min_item(), min_key(), pop_min(), pop_max().
4. Indirectly I would also be testing functions as __contains__(), __len__(), clear() etc.

Quality of the SUT:
For the quality of the SUT, I didn't find any bugs from the library. Based on my examination, the library is of high quality. It is written and maintained by the same author Manfred Moitzi. It is frequently updated and has a set of high quality test cases for each method. Following are some of the software quality factors that I want to highlight about the code.

1. Maintainability and readability:
    a. The code is well maintained and well documented, it helps the user to understand the code flow easily.
2. Performance and efficiency:
    a. The response time of the library is acceptable and efficiency of resource utilization is pretty good.

Varun Rajgopal
CS 569
Dr. Alex Groce

Code coverage:
The code coverage so far is 4.12% of the bintree.py. The library is about 1000 lines of code and 62 functions. I would be adding more functions to the TSTL file and would improve code coverage. I am planning to cover all the heap methods such as max_items(), max_key(), min_item(), min_key(), pop_min(), pop_max().

Following is the output for the code coverage:

```
4.12371134021 PERCENT COVERED
240.015151024 TOTAL RUNTIME
8497 EXECUTED
849627 TOTAL TEST OPERATIONS
201.237780094 TIME SPENT EXECUTING TEST OPERATIONS
30.9966840744 TIME SPENT EVALUATING GUARDS AND CHOOSING ACTIONS
1.13518619537 TIME SPENT CHECKING PROPERTIES
202.37296629 TOTAL TIME SPENT RUNNING SUT
0.575242996216 TIME SPENT RESTARTING
0.0 TIME SPENT REDUCING TEST CASES
35 BRANCHES COVERED
24 STATEMENTS COVERED
```