CS562 – Applied Software Engineering.

Bhargav Pandya, 932690891.

Progress Report

I am using TSTL to test a Binary Search Tree library and till now I have tested several functions of the library. The basic structure of the binary search tree in the system under test is that each node holds a key/value pair and each node on the left side of the root is smaller and the nodes on the right side of the root are bigger than the root. From the functions which I have tested until now there are no bugs in the System Under Test and the tester as well. The functions work and produce output as expected.

There are several other functions which are yet to be tested, however I have made sure that I thoroughly cover various cases in the functions which I have tested till now. Some of the functions which I have tested are: insertion, deletion, check if binary tree contains keys, if it is empty, checking the max node and the max key, checking the minimum node and minimum node. I tested these functionalities as follows:

1. `<bstree>.put(<key>,<value>) ; <bstree,1>.contains(<key,1>) == True ; <bstree,1>.delete(<key,1>) => (<bstree,1>.contains(<key,1>) == False)`

a) Inserting a key/value pair in the tree and then checking if that key exists in the tree followed by deleting the key and then again checking if the key exists in the tree;

2. `<bstree>.put(<key>,<value>) ; ((<bstree,1>._max_node().key) == (<bstree,1>.max_key()))`

b) Inserting keys in the tree and checking if the max node function gives a node which has the same key as the key returned by the max key function;

c) The minimum node and minimum key functions are tested on similar lines as the max node and max key function;

3. `<bstree>.put(<key>,<value>) ; <bstree,1>.contains(<key,1>) => (<bstree,1>.is_empty() == False)`

d) Inserting a key/value pair in an empty tree and checking if the tree is empty after performing the insertion.

What I have tried to achieve till now is to check each of these functionalities thoroughly, testing each of the function in different ways. This is to ensure that the functions work fine under a wide number of cases. Testing these functions produced no bugs till now. The code is written very well and it is very unlikely to find bugs. While testing the other functions I might also include new methods to aid in the testing. I intend to test each of the functions thoroughly by the end of the term.

Currently I am testing other functions in the Binary Search Tree library like get, floor node, floor key, rank, select_key, delete max key, delete min key etc. And by the end of the term I intend to include two new libraries for testing. The first one is a linked list library and the second one is stack library. For these libraries also I will be testing the functions and

checking if they have any bugs. Some of the functionalities I am testing for stack are add, remove, check if stack is empty.

Currently I am writing more complex tests for the above libraries and by the end of the term I intend to these three libraries thoroughly and report if I find any bugs.

Code Coverage:

I achieved approximately 40% code coverage and covered 102 branches and 71 statements.

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