Jack Duncan
3.24.23

Project 3 Report

Project 3 was definitely the most difficult project so far, as the implementation of splaying in trees is the most complex algorithm I have written in any computer science class. The project started off with a BST class. Both my BST class and SplayTree class have insert, search, and remove methods written iteratively to save space, and destructors and print methods that use recursion for elegance. Insert and search was very straightforward for BST, but the remove function was more difficult due to multiple cases depending on the children of the node to be removed. If organized well, BST is O(logn) time complexity; however, BST is worst case O(n) for the insert, search, and remove functions because the tree may be skewed like in test 0 where the BST resembles more of a linked list than a tree.

The SplayTree class is based on the BST, with the destructor and print methods being exactly the same and the search and insert methods closely resembling the BST's search and insert methods. The SplayTree class has a splay function which iteratively brings up a node to the root. For the SplayTree's nodes, the structure sNode, I added a third pointer called parent, which points to the parent of a node. Although it adds to the space requirement of the tree, it makes bottom up splaying much faster as I did not have to search for my node every time the node moved up the tree. My splay function uses rotateLeft and rotateRight helper functions for the different zig and zag cases. The remove function works as shown in the Splay Tree Visualization website, splaying the to-be-deleted node to the root and then finding the max value in the left subtree. The SplayTree has the same time complexity as the BST, but because of the constant reorganization, the amortized time is Olog(n) as it tries to avoid the worst case.

I also ran valgrind for each test case and all heap blocks were freed and I had no errors. Results from the 5 test files are shown below.

	Traversal Count	Time (ns)
BST 0	193	47,746
SplayTree 0	99	21,296
BST 1	7,326	97,931
SplayTree 1	882	68,777
BST 2	49,629	307,490
SplayTree 2	3,841	165,522
BST 3	3,182	136,985
SplayTree 3	5,190	182,937
BST 4	120	47,747
SplayTree 4	119	16,143