**İlker Keklik**

**Project 2 Report**

There are several differences between AVL Tree and Splay Tree. They both have advantages and drawbacks. If I talk about the working logic, AVL Tree is a sellf-balanced binary tree and after every operation it checks the balance factor of each node in it and if it find the unnormal difference it balances the tree and always keep it balanced. Splay Tree have a different way to store data efficiently.There is a rule in Splay Tree. Inserted node, parent of the deleted node or previously searched root always must be the root of the tree . Thus, the data that is used or will be used continuously is stored in the parts of the tree close to the root and allows us to spend less time when we want to reach them. In AVL Tree search time is always O(logn) but in Splay Tree it depends on the input to be used.

Now, I will try several search cases to understand in which situation which tree have better performance. Firstly, all elements in input 2(from Project file) is inserted both trees.

**Case1:** I have searched some values from input2 which is given in the Project file and these are last inserted elements mostly.

78,87,4,26,47,50,50,50,50,50

The search factor of the Splay Tree is 29.

The search factor of the AVL Tree is 55.

**Case2:** I have searched some values from input 2 which is given in the Project file and these are random selected elements.

13,96,2,45,91,77,14,50,17,49

The search factor of the Splay Tree is 72.

The search factor of the AVL Tree is 52.

**Case3:** I have searched some values which are not in input 2.

52,99,41,6,55,100,101,102,103,104

The search factor of the Splay Tree is 69.

The search factor of the AVL Tree is 69.

**Graphs of Cases**

Performance of the trees in each cases can be considered as inversely proportional to the search factors. This applies to all cases.