**Report of Empirical Analysis**

**1. System and Language Information:**

**Language Used:** Java

**Compiler Used:** JDK 17

**Runtime Environment Used:** JRE 8

**2. Evaluation:**

**Case # 1:**

Inserted Values = n = 100,000

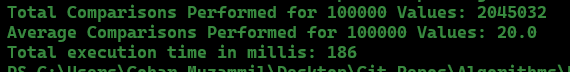
**Binary Search Tree**

Execution Time (including insertion and searching) = 186 Milliseconds

Total Comparisons (for entire data points) = 2045032

Average Comparisons (per search) = 20.0

**Output:**



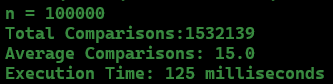
**AVL Tree**

Tree Execution Time = 125 Milliseconds

Tree Total Comparisons = 1532139

Tree Average Comparisons = 15.0

**Output:**

****

**Splay Tree**

Execution Time = 195 Milliseconds

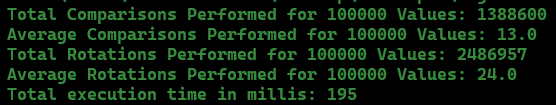
Total Comparisons = 1388600

Average Comparisons = 13.0

Total Rotations = 24869557

Average Rotations = 24.0

**Output:**

****

**Case # 2:**

Inserted Values = n = 1,000,000

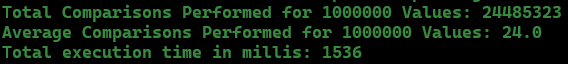
**Binary Search Tree**

Execution Time (including insertion and searching) = 1536 Milliseconds

Total Comparisons (for entire data points) = 24485323

Average Comparisons (per search) = 24.0

**Output:**



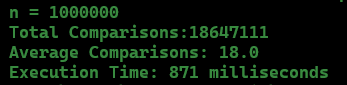
**AVL Tree**

Execution Time = 871 Milliseconds

Total Comparisons = 18647111

Average Comparisons = 18.0

**Output:**

****

**Splay Tree**

Execution Time = 1915 Milliseconds

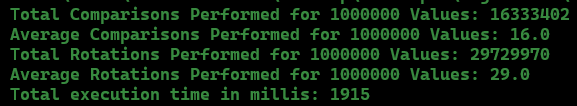
Total Comparisons = 16333402

Average Comparisons = 16.0

Total Rotations = 29729970

Average Rotations = 29.0

**Output:**

****

**Case # 3:**

Inserted Values = n = 10,000,000

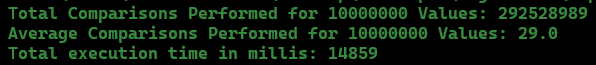
**Binary Search Tree**

Execution Time (including insertion and searching) = 14859 Milliseconds

Total Comparisons (for entire data points) = 292528989

Average Comparisons (per search) = 29.0

**Output**



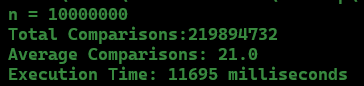
**AVL Tree**

Execution Time = 11695 Milliseconds

Total Comparisons = 219894732

Average Comparisons = 21.0

**Output:**

****

**Splay Tree**

Execution Time = 26868 Milliseconds

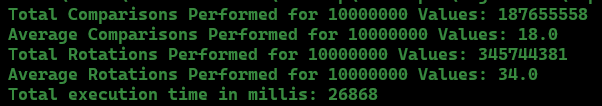
Total Comparisons = 187655558

Average Comparisons = 18.0

Total Rotations = 345744381

Average Rotations = 34.0

**Output:**

****

**4. Comparisons**

**Splay Trees vs Binary Search Trees:**

For each value of n, Splay Trees took more time to execute as compared to BST. But the average and total comparisons of Splay Trees were much less than the BST.

These were the results for all 3 values of n:

|  |  |  |
| --- | --- | --- |
|  | **Average Execution Time**  **(for n = 100,000 and n = 1,000,000 and**  **n = 10,000,000)** | **Average of Total Number of Comparisons for Each n** |
| **Splay Trees** | 9,659 Milliseconds | 68,459,186 |
| **BST** | 5,506 Milliseconds | 106,353,114 |

**Splay Trees vs AVL Trees:**

For all values of n, the execution time of Splay Trees was much higher than AVL Trees. But the average comparisons performed by Splay Trees were always less than AVL Trees. The difference wasn’t very high but it was still notable.

These were the results for all 3 values of n:

|  |  |  |
| --- | --- | --- |
|  | **Average Execution Time**  **(for n = 100,000 and n = 1,000,000 and**  **n = 10,000,000)** | **Average of Total Number of Comparisons for Each n** |
| **Splay Trees** | 9,659 Milliseconds | 68,459,186 |
| **AVL Trees** | 4,230 Milliseconds | 80,024,660 |

**5. Overall Rankings**

|  |  |
| --- | --- |
| **In terms of Execution Time** | 1. AVL Trees 2. BST 3. Splay Trees |
| **In terms of Comparisons** | 1. Splay Trees 2. AVL Trees 3. BST |