

# CS618-Indexing and Searching Techniques in Databases

## Assignment -2

Ayush Mittal (11183)

### 1 System Configurations:

Processor: 1.3 GHz Intel Core i5

Memory: 4 GB 1600 MHz DDR3

Processor Name: Intel Core i5

### 2 Results

Query	Minimum	Maximum	Average	Standard Deviation
Insertion	0.0	8.5e-05	2.20266033965e-06	1.50794191755e-06
Point Seach	0.0	3e-06	1.416e-06	6.42607189502e-07
k-NN Search	0.000124	0.023558	0.0092651	0.00550378192882
Range Search	9e-06	2.27136	1.174607946	0.685951293947
Window Search	1.1e-05	0.480596	0.011643917	0.0479948604511

Table 1: Results for Quad Tree

Query	Minimum	Maximum	Average	Standard Deviation
Insertion	0.0	7.1e-05	2.12585914085e-06	1.30632849697e-06
Point Seach	0.0	4e-06	1.672e-06	6.91676224834e-07
k-NN Search	0.000102	0.024142	0.009187103	0.0054756540775
Range Search	1.9e-05	3.63044	1.410143583	0.833216275723
Window Search	1e-05	0.000228	2.6817e-05	1.18172547997e-05

Table 2: Results for KD-Tree

### 3 Conclusion

- The lower mean and standard deviation values in the above results show that **KD Trees outperforms Quad Trees** in all the cases except Point Queries in which it is a bit slower as compared to Quad Trees.
- The average time is a bit less for Point Queries in Quad Trees. This may be because of the fact that in case of Point Queries we just need to traverse one branch of the tree in both the trees and since the fanout of the Quad Tree is more, the quad tree can be smaller for same sequence of queries.
- Large time in Range, Window and K-NN Queries as compared to Point Queries indicates that a larger part of the tree is traversed for these queries.
- Coefficient of variation(=Standard Deviation/Mean) is very high for both the trees in all the cases indicating that the trees are not balanced. This is expected as the structure of the trees depend on the order in which the data is inserted. This fact can also be observed by looking at the high difference between the minimum and maximum values for all the cases.
- KD-Trees performs better than quad trees for range, window and kNN queries. This may be because in case of all these queries less number of comparisions are to be made in case of KD-Trees because of a smaller fanout value. Also, because of a smaller fanout value less number of nodes are explored for kNN Search and hence the time for best first search improves.