**Bisecting KMeans Clustering**

**Implementations Detail:**

* Generated Random n data points and select a random point p as centroid of cluster C1.
* Select the farthest point q from p as centroid of Cluster C2.
* Assign data points to clusters based on min distance.
* Calculate midpoint of clusters as new centroids.
* Repeat steps 3 and 4 until cluster centroid does not change.
* Calculate SSE for Clusters.
* Bi-portion the cluster with highest SSE.
* Repeat step 3 to 7 until K clusters are formed.
* Sort the data based on cluster assignment.
* Search cluster i’s data point in which search data points falls within cluster i’s radius.

**Instructions to run the code:**

**How to compile**

gcc -c "cs5331-HW3-(Arun kumar Jegarkal)-(Jaichandra Sesetty).c"

gcc "cs5331-HW3-(Arun kumar Jegarkal)-(Jaichandra Sesetty).o" -o "cs5331-HW3-(Arun kumar Jegarkal)-(Jaichandra Sesetty).exe"

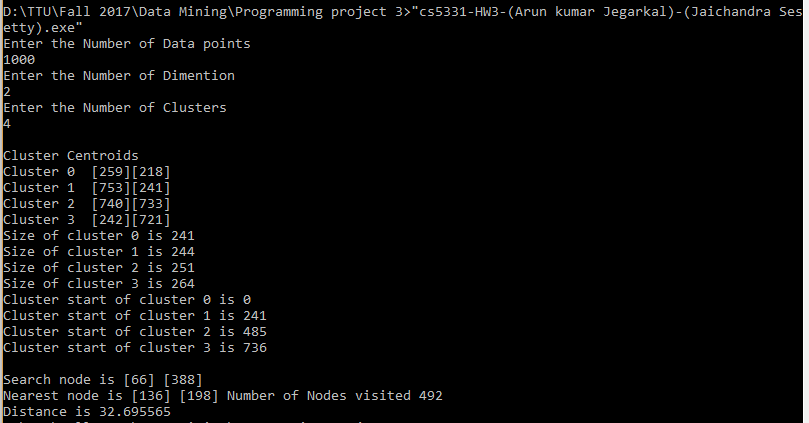
**How to execute**

"cs5331-HW3-(Arun kumar Jegarkal)-(Jaichandra Sesetty).exe"

**Sample Run**

1000 Data Points with 2 Dimensions and 4 Clusters

* Here you can observer once the cluster are formed data array is sorted
* cluster 0 data points start at 0th location till 240th, Cluster 1 from 241st to 484th , cluster 2 from 485th to 735th and cluster 3 from 736th to 1000th



10000 Data Points with 6 Dimensions and 10 clusters

