**K-Means Algorithm Documentation**

**Hadar Pur - 308248533**

1. The master read and obtain all the points from the input file.
2. The master process does initial to the clusters according to the first k points.
3. The master process calculates the amount of points that each process (include him) will handle with, also, the master handle with the rest of the points (in case N%numproc!=0).
4. The master process broadcast to the other processes the number of clusters and number of points that the individual process will be handle.
5. The master process sends to each slave process that appropriate segment of points to each slave process.
6. The all processes together (master and slaves) start the Algorithm.
7. The master process sends to each slave process the current time.
8. Each process calculates his own points coordinates according to the time from step 7.
9. Each process activate the K-Means function.
10. The master obtains the current quality and checks if the quality is less than QM and check:
    * If the current quality is less than QM or the time is T/dt the master sends to all slave processes Final-Termination-tag and return the quality.
    * If the current quality is greater than QM, the master saves the less quality between the current quality and the previous quality.
11. The slaves’ processes are finished and finalize.
12. The master is writing to the output file the time, quality and the clusters centers to a output file and finalize

* **The program will be more efficient if I would use MPI\_Allreduce instead of MPI\_Reduce and then each process would check by himself the termination tag and to avoid extra MPI communication.**

**O((T/dt) \* Limit \* ( (N+K)/numOfProcs))**