Akshay Jain
HPC, Assignment 2

Q2. Parallel sample sort:

Here are the timings from running my sample sort on 64 stampede cores:

Value of N	Time taken (seconds)
1,000	0.180856
10,000	0.180608
100,000	0.207191
1,000,000	0.516526
10,000,000	3.773809
100,000,000	39.287584
1,000,000,000	seg fault
10,000,000,000	seg fault

Q3) Pitch your final project:
(Doing project with Mark Andrew Ward)

Parallel Partitioning of N-dimensional Data

For a wide variety of applications, it is often necessary to distribute large datasets amo ngst multiple machines. For certain applications the partitioning of the data must be carefully chosen such that nearby points in the given metric space will be placed on the same machine. In the scientific computing community it is often of interest to partition two and three dimensional data, but in other areas, such as information retrieval, one may be interested in the partitioning of much higher dimensional data. In our project we hope to explore this problem of how to efficiently and effectively partition higher dimensional data in a given metric space. We plan to make use of some existing libraries such as p4est and/or pcl that will provide useful data structures and algorithms including octrees, kd-trees, approximate nearest neighbor search, and k-means clustering. We will implement multiple methods, which will

include exact and approximate methods, for the data partitioning and compare each method. First, we look at the runtime required for each method. Next, we examine the quality of the data clusters found by each algorithm by comparing intra- and inter-cluster distances. The distribution of data to the appropriate machines will be facilitated by MPI. The calculation of intra-cluster distances or other metrics will likely take advantage of OpenMP while the required measures between distinct data clusters will make use of MPI as the data will likely sit on different physical machines.