**Distributed Computation via MPI: using Microarray Analysis for Genome Wide Association Studies**

**Objective:**

To develop an MPI-based message-passing application which analyzes raw microarray data to identify differentially expressed genes between two user-defined groups of patient samples.

**Introduction:**

Microarrays may be used to measure gene expression in many ways, but one of the most popular applications is to compare expression of a set of genes from a cell maintained in a particular condition to the same set of genes from a reference cell maintained under normal conditions .For example, cDNA from cells grown in one condition may be labelled with a red dye and from cells grown in another condition with a green dye. Once the samples have been differentially labelled, they are allowed to hybridize onto the same glass slide. At this point, any cDNA sequence in the sample will hybridize to specific spots on the glass slide containing its complementary sequence. The amount of cDNA bound to a spot will be directly proportional to the initial number of RNA molecules present for that gene in both samples.

**Statistical Analysis:**

Microarrays are analyzed by pure statistical analysis to differentiate the genes of a specific function or disease. The very basic analysis used is Student T-test or Petmutative T-test. As each and every gene data consists of two groups, for ex. Disease and control. Initial T-test of the sample would be generated considered as Tsamp . However, gene data is not statistically significant until and unless it has been normalized. For that, every gene data has been formed with more combinations on shuffling and forming a normalized data and discriminating the score based on dscore. Finally, basing on the dscore all the genes are sorted to find the highly expressed genes.

**Assumptions:**

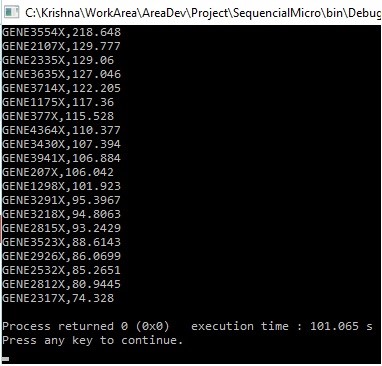
We all know that, more the data for normalizing, more accurate it is. Every gene has been shuffled **100** times in this experiment, for the best fitted curve. However, it can be more data like 1000 times, but for the performance on laptop, the count has been kept as 100. According, to the statistics, any data above 30 number would give a least normalized curve.

**Programming:**

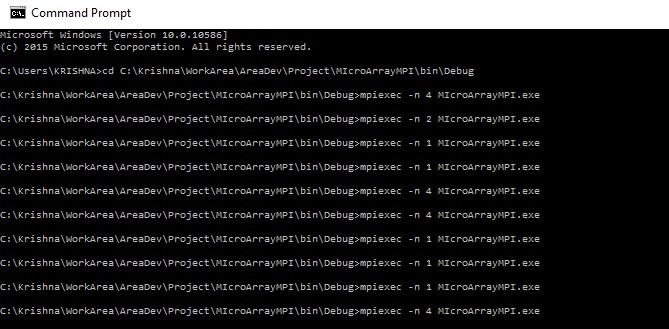
**Sequential Version:**

File NCI60.csv has been taken for analyzing the microarray Data. The sequential program has been developed on C++, with an extensive use of BOOST Libraries. **Boost Token Iterators** were used to spit the CSV file into tokens required and pushing the data in the specified vectors. **BOOST Accumulators** were used to calculate the Mean and Variance of the accumulated data and using those values to calculate actual t-score and d-score. The obtained d-score has been paired with **std::Pair** and finally sorted against other genes to differentiate using **std::sort** algorithm.

The problem is to identify genes that best discriminants for renal cancer (“RE”). There are eight diseased samples, out of 60 total patients. There are 4550 total genes to be considered.

**Output:**

**MPI Programming:**

MPI\_SCATTER has been used to calculate the t-score values at every point. The Project got build successfully, unfortunately not revealing any errors and output too.

**Results:**  The top 20 observed genes are displayed on the console out of which the Top5 are

GENE3554X,GENE2107X, GENE2335X, GENE3635X,GENE3714X.