**1. Organization of Experimental Profiling**

I have performed an experiment by running all BST, Min 5-Heap and Max 5-Heap using different random numbers ranging from 1m to 5m where m is equal to 1,000,000.

The CPU clock start and end time were noted for each build. deleteMin, and deleteMax operation for each tree structure and then finally their average was computed to evaluate the result. This operation is executed 5 times for each number set at different seed.

**2. Input data generating using random number generator**

To perform this I have used the standard Rand() function that generates random number and made it within the range of 1 to 5m using the mod function of 5m. For every number set (1m to 5m) I have used different seeding. So overall my experiment was performed for 5 numbers set 1m to 5m where each number or data was limited to 1 to 5m only and then for each number set 5 times the operation was performed for different seeding.

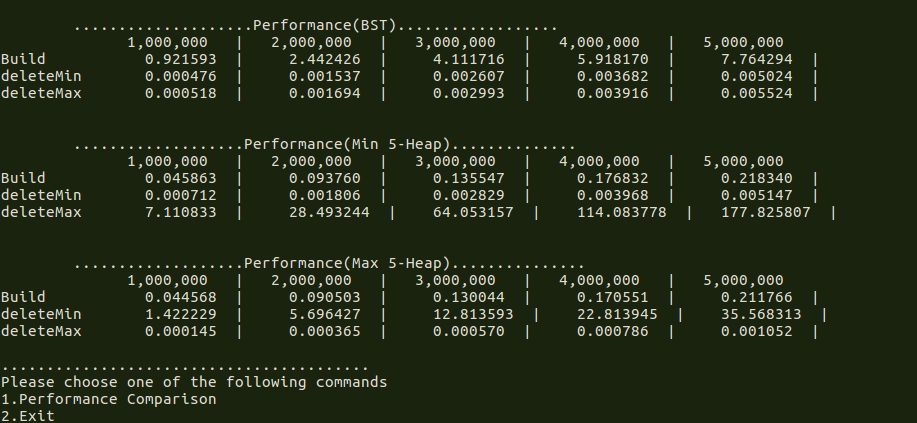
**3. CPU time recording in C++.**

CPU time was recorded using the provided TimerClass where start and end time duration was calculated in seconds and nanoseconds.

**4. Data recording and analysis**

Time duration for each table is noted and shown in the program's output for build, deleteMin and deleteMax.

**5. Performance comparison, observations and summary.**



P.S As already stated above each operation was performed on the same number set for each table.

**6. Conclusion**

Hence from the above experiment I can conclude that the build operation, deleteMin in Min Heap and deleteMax in Max Heap can be performed in less time but when a deleteMax operation is required the Min Heap is not an ideal data structure and similarly Max Heap is not ideal for deleteMin operation.