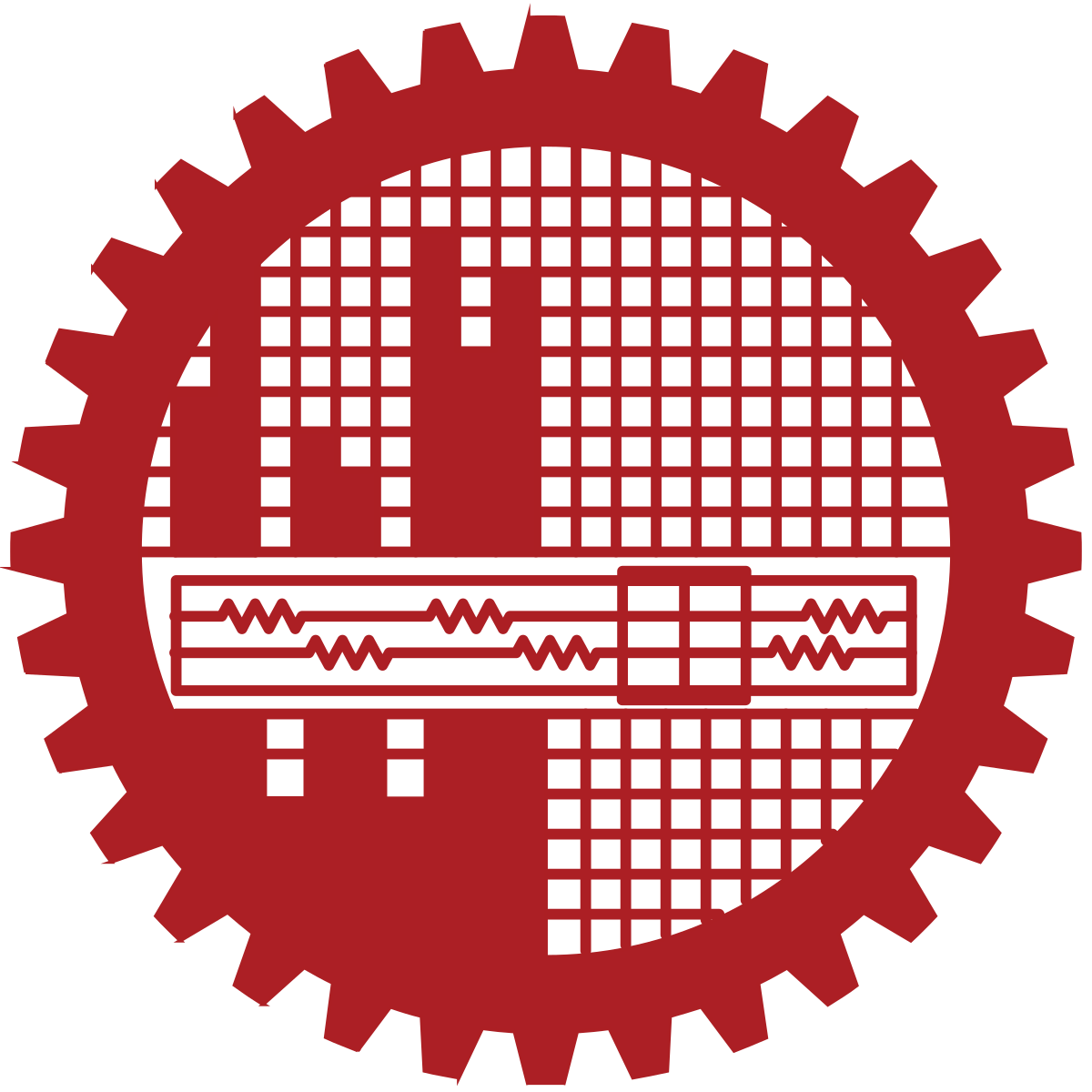
**CSE 204**

**DATA STRUCTURES AND ALGORITHM**

**SESSIONALS**

****

**OFFLINE NO : SEVEN**

**TITTLE : Comparison of Merge Sort and Quicksort**

**SUBMITTED BY :**

**FAHMID - AL-RIFAT**

**STUDENT NO : 1705080**

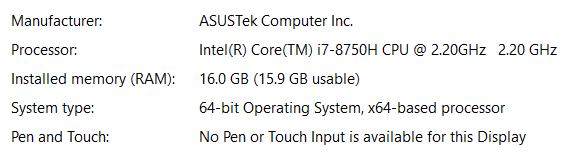
LEVEL-2 ,TERM-1

DEPARTMENT OF CSE

BUET

**OBJECTIVE:** The objective of offline is to compare the time needed to execute the two famous algorithms of sorting merge sort and quicksort where various size of array are randomly generated in different order for each time of execution where merge sort data are sorted by dividing the array into equal parts and merging them recursively where in quick sort data are sorted taking a pivot element and placing them properly where the divided parts are also sorted same way . Between two sort quick sort is inplace sorting algorithm

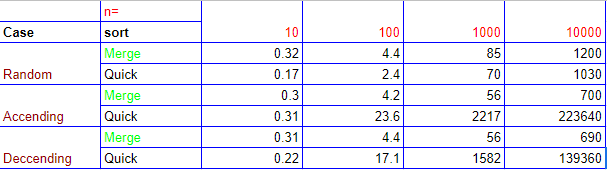
**MACHINE CONFIGURATION:**



Compiler : Code Blocks 17.12

**DATA SET :**

**Table of sort algorithm time in millisecond and array size:**



**DISCUSSION :**

Merge sort and quicksort are algorithm for sorting elements in the list of finites elements in array . In the code arrays elements are generated randomly . In the merge sort each element of the array are divided into two equal parts recursively until the array have one element, then each step of the recursion the divided parts are merged from bottom to up and Thus the array is sorted and complexity of the algorithm become **O(n log n)** . In the best case the algorithm complexity **O(nlogn )** and the worst case **O(nlogn)** also as whatever the array elements serial is the whole process remains same . On the other hand in Quicksort each element of the array are divided into two parts according to pivot and setting it in proper place. In similar recursive way is run in the before and after part of the pivot.Every time we take a pivot and place into its place and so it's called in place algorithm.

Thus, running time complexity of selection sort is **O(nlogn)** in average and best case. But in the best case complexity here decreased a lot , the complexity then **O(n2)** as in this case we also have to traverse whole as always when we pick the pivot the parts become unbalanced which mean in one part all the elements gathered keeping other parts empty

Among both of the sorting algorithms, both sorts are fast, efficient, stable while quick sort is more space efficient and less time efficient in it’s worst case

