LAB-2

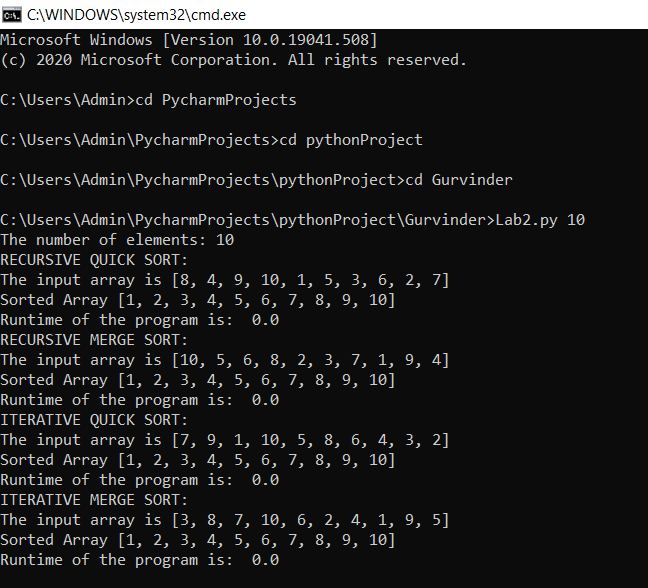
AIM:

Write a program to implement quick sort and merge sort. The implementation would be carried using iterative and recursive methods.

EXPERIMENT:

A function that calls itself is called **Recursive** while a set of instructions which are being executed repeatedly are called **Iterative**. Recursion reduces the size of code while iteration makes it longer.

OUTPUT (For 10 numbers):



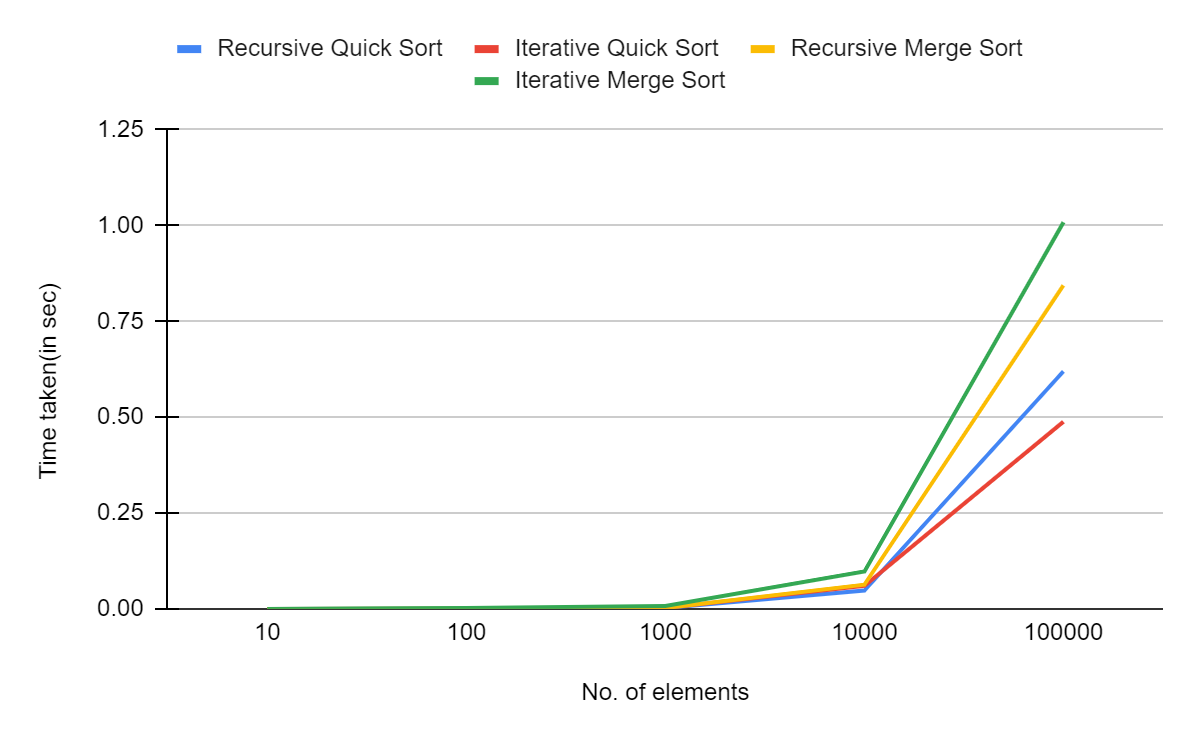
The memory taken by:

1. Recursive Quick Sort - 12.1796875
2. Iterative Quick Sort - 12.1875
3. Recursive Merge Sort - 12.25390625
4. Iterative Merge Sort - 12.23828125

The memory was calculated separately for each function.

**GRAPH:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| n | Recursive Quick Sort | Iterative Quick Sort | Recursive Merge Sort | Iterative Merge Sort |
| 10 | 0 | 0 | 0 | 0 |
| 100 | 0.0009932518005 | 0 | 0.0009965896606 | 0.002032995224 |
| 1000 | 0.002472400665 | 0.002022981644 | 0.002024412155 | 0.00760602951 |
| 10000 | 0.04779243469 | 0.05993270874 | 0.06314253807 | 0.09755468369 |
| 100000 | 0.6199605465 | 0.4883248806 | 0.8441109657 | 1.00895977 |



**CONCLUSION:**

Recursive and iterative quick and merge sorts were implemented and their behavior was observed. From the graph, it seems that iterative merge sort is slower than recursive merge sort and recursive quick sort is slower than iterative quick sort.