

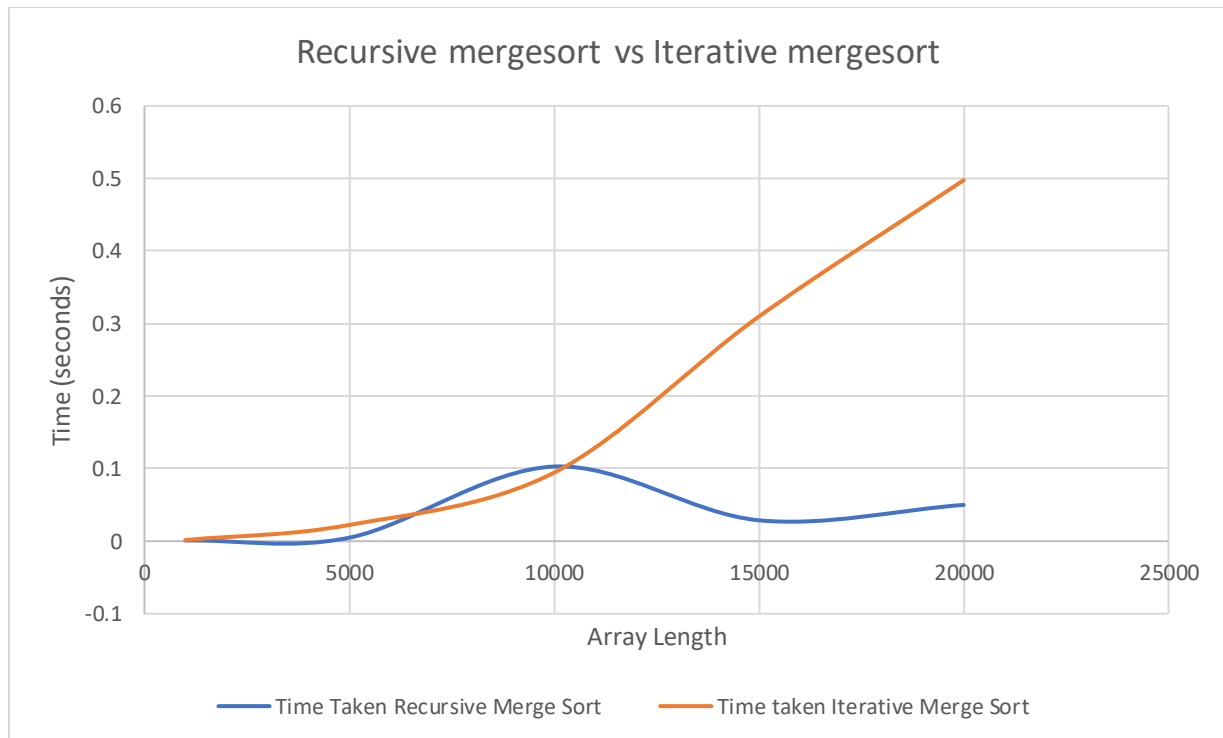
CS2023 - DATA STRUCTURES AND ALGORITHMS IN-CLASS LAB EXERCISE

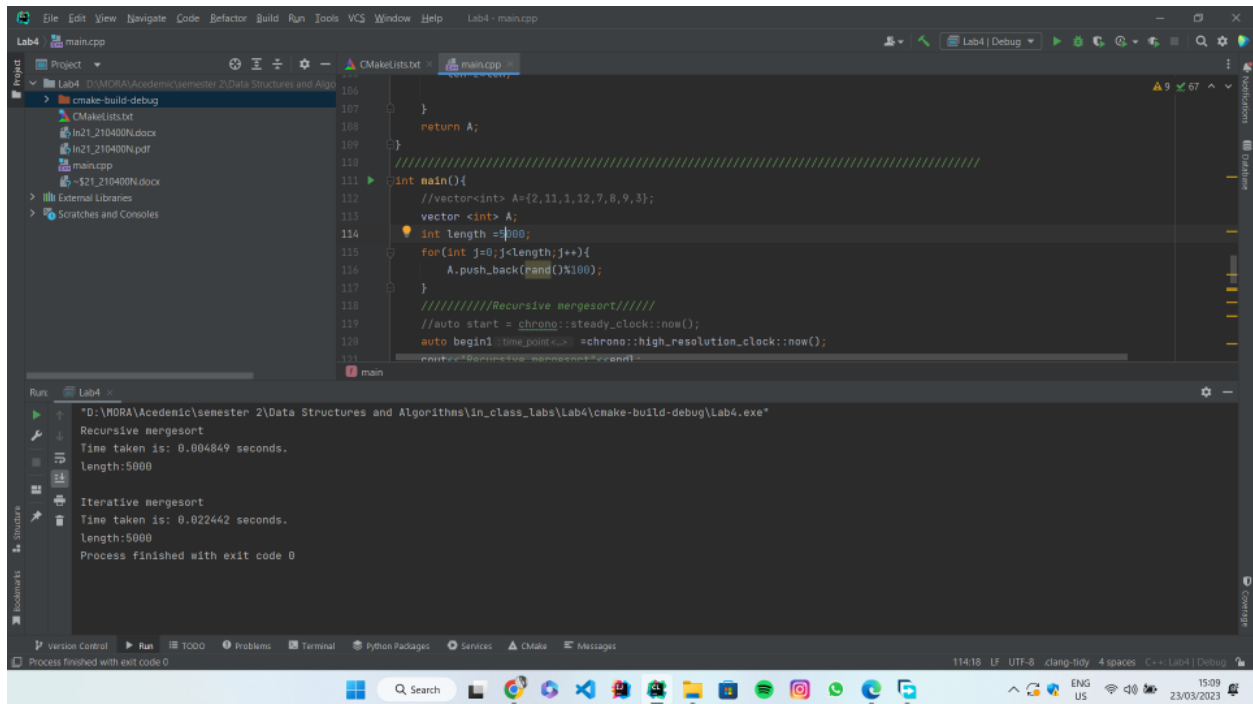
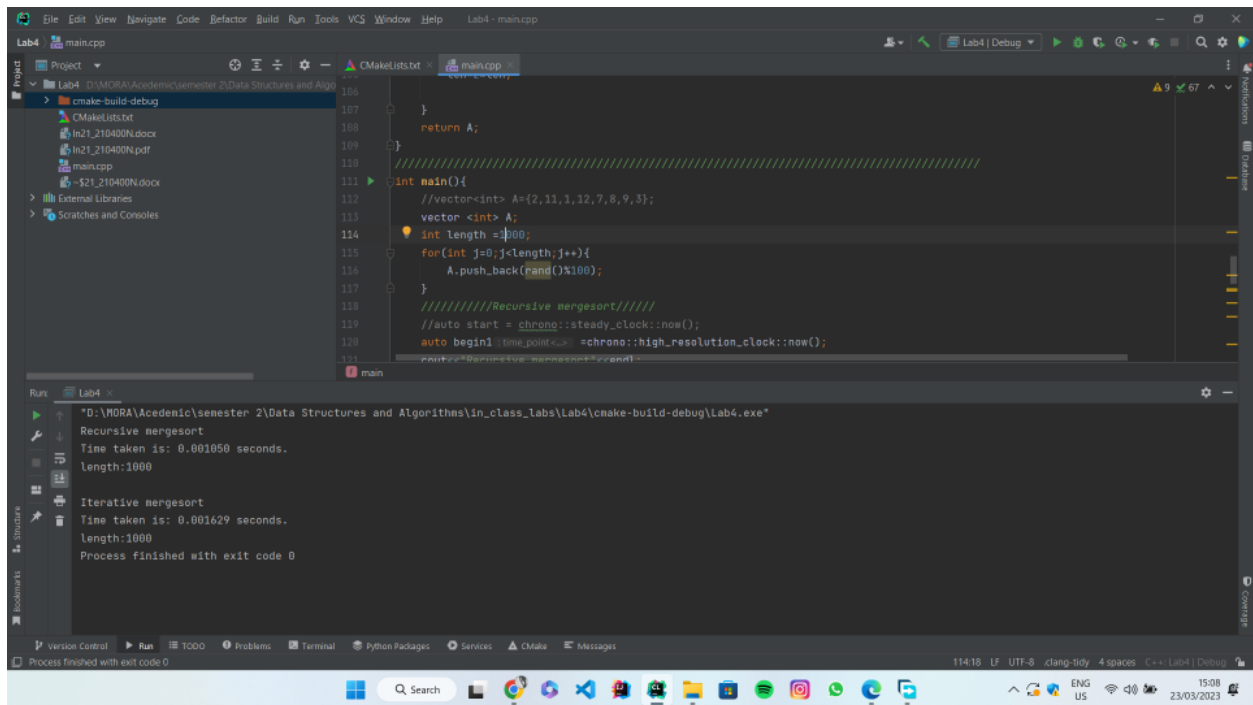
NAME:MUTHUWANA M.A.N.R

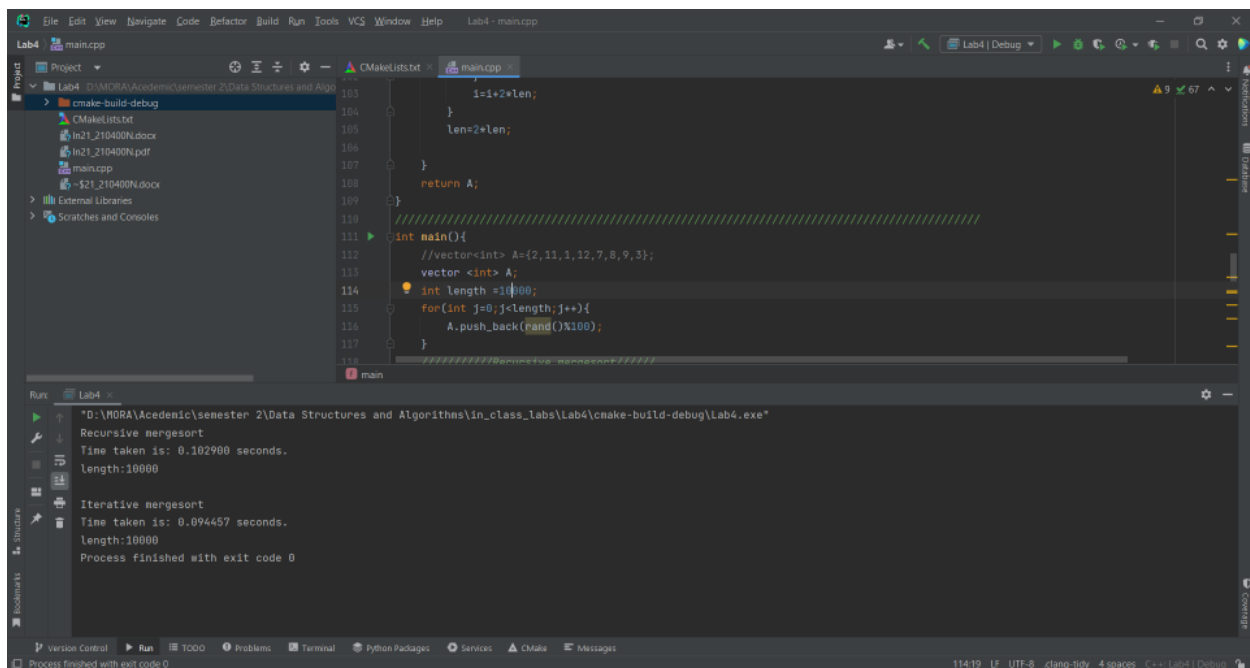
INDEX:210400N

- Iterative and non-iterative merge sort

Array length	Time Taken Recursive Merge Sort	Time taken Iterative Merge Sort
1000	0.00105	0.001629
5000	0.004849	0.022442
10000	0.1029	0.094457
15000	0.028837	0.309653
20000	0.050037	0.497652
25000	0.073977	0.704183







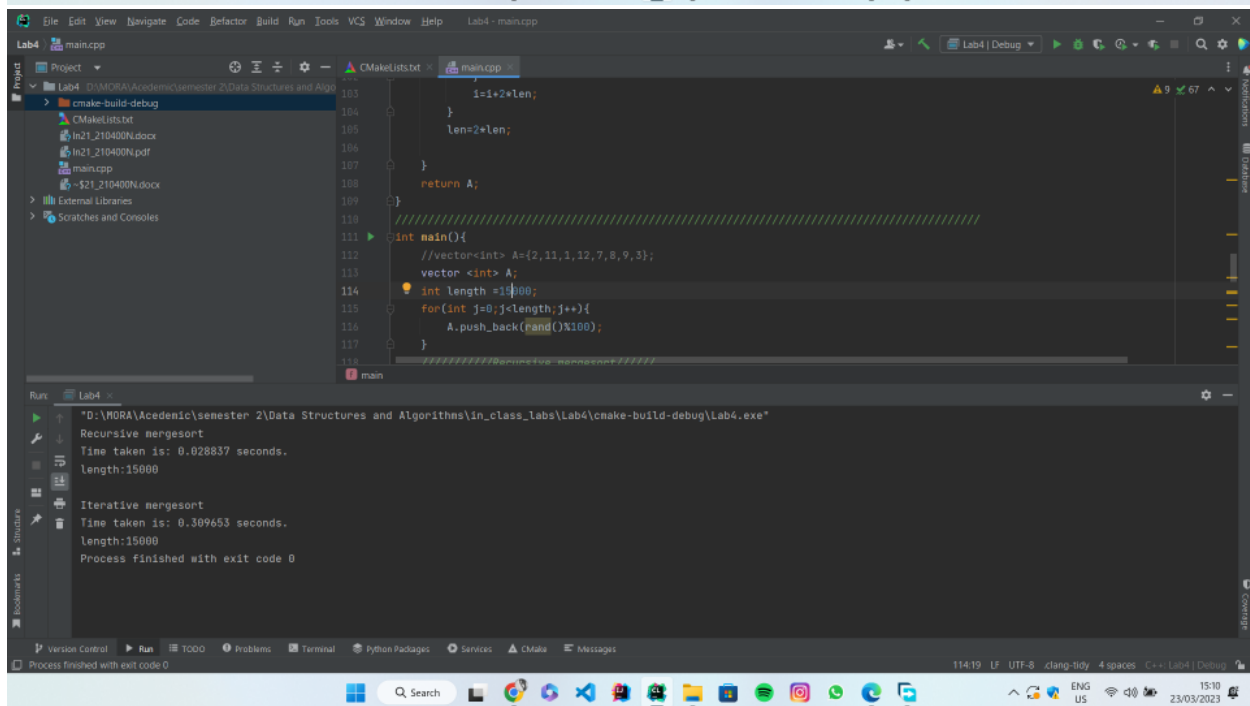
```
113     i=i*2*len;
114     }
115     len=2*len;
116     }
117     return A;
118 }
119
120 ///////////////////////////////////////////////////
121 int main(){
122     //vector<int> A={2,11,1,12,7,8,9,3};
123     vector<int> A;
124     int length=1000;
125     for(int j=0;j<length;j++){
126         A.push_back(rand()*100);
127     }
128     ///////////////////////////////////////////////////Recursive mergesort////////////////////////////////////
129     main
130 }
```

Run: Lab4

"D:\MORA\Academic\semester 2\Data Structures and Algorithms\in_class_labs\Lab4\cmake-build-debug\Lab4.exe"

Recursive mergesort
Time taken is: 0.102900 seconds.
Length:10000

Iterative mergesort
Time taken is: 0.094457 seconds.
Length:10000
Process finished with exit code 0



```
113     i=i*2*len;
114     }
115     len=2*len;
116     }
117     return A;
118 }
119
120 ///////////////////////////////////////////////////
121 int main(){
122     //vector<int> A={2,11,1,12,7,8,9,3};
123     vector<int> A;
124     int length=15000;
125     for(int j=0;j<length;j++){
126         A.push_back(rand()*100);
127     }
128     ///////////////////////////////////////////////////Recursive mergesort////////////////////////////////////
129     main
130 }
```

Run: Lab4

"D:\MORA\Academic\semester 2\Data Structures and Algorithms\in_class_labs\Lab4\cmake-build-debug\Lab4.exe"

Recursive mergesort
Time taken is: 0.028837 seconds.
Length:15000

Iterative mergesort
Time taken is: 0.309653 seconds.
Length:15000
Process finished with exit code 0

