Programing Assignment Report

CSI2110 - Data Structures and Algorithms

Fall 2023

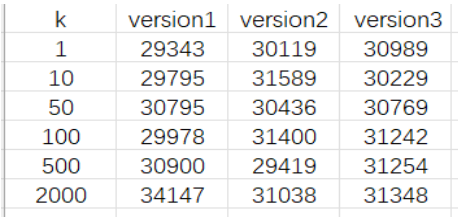
School of Electrical Engineering

&

Computer Science University of Ottawa

Hengjing Zhang - 300288003

Submission Date:21/10/2023



k	version1	version2	version3
1	29343	30119	30989
10	29795	31589	30229
50	30795	30436	30769
100	29978	31400	31242
500	30900	29419	31254
2000	34147	31038	31348

Figure1: The table for running time.

Version1 used the arraylist to implement priority queue. The execution time for version1 decreases initially as the value of k increases, reaching a minimum point between k values of 10 and 100. After this point, as k continues to rise, the execution time for version1 starts to increase.

Version2 used the max heap to implement priority queue. The execution time for version2 initially rises with increasing k values, peaking around k=100. After this peak, the execution time starts to decrease and then remains relatively stable until k reaches 2000.

The execution time for version3 consistently shows a gentle upward trend across the entire range of k values. However, this increase is very gradual.

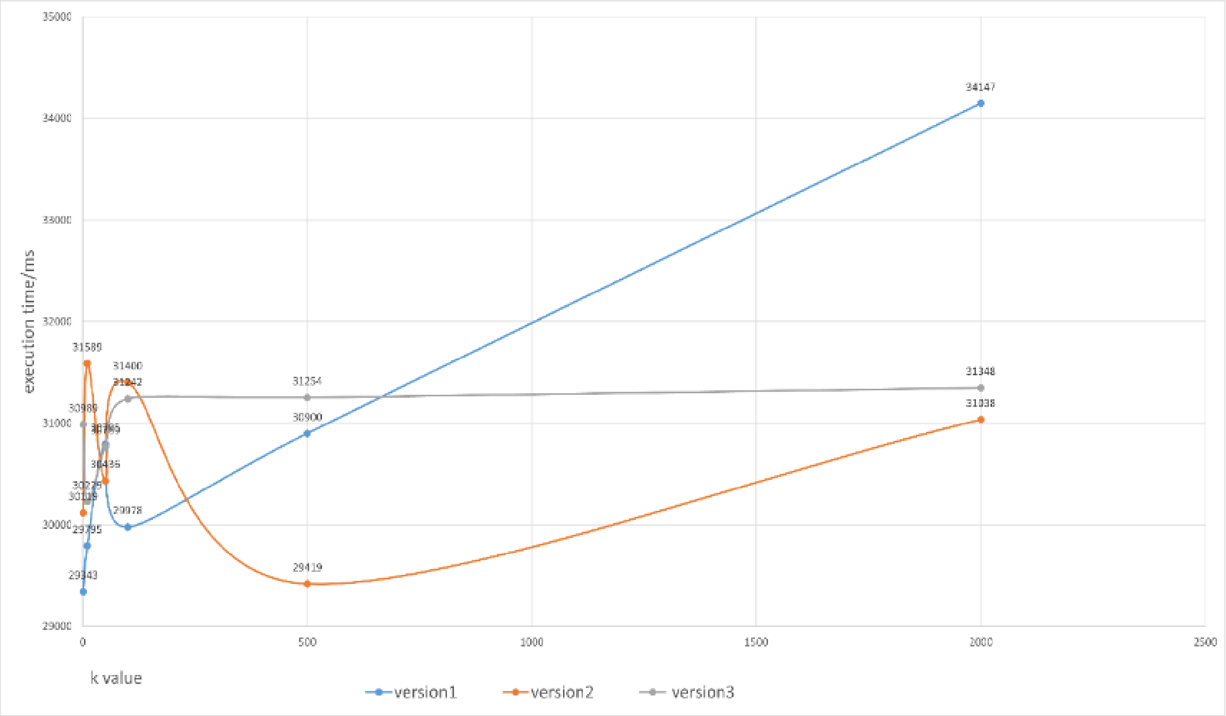


Figure2: The graph for running time of V1 to V3

Conclusion

For smaller k values, version1 appears to be the most optimal in terms of execution time, but its performance degrades as k increases. Version2 exhibits poorer performance in the mid-range of k values but shows improved and stable performance at larger k values. Version3, due to its very gradual increase in execution time across the entire range of k values, might be considered slightly better overall when compared to the other versions, especially for larger k values.