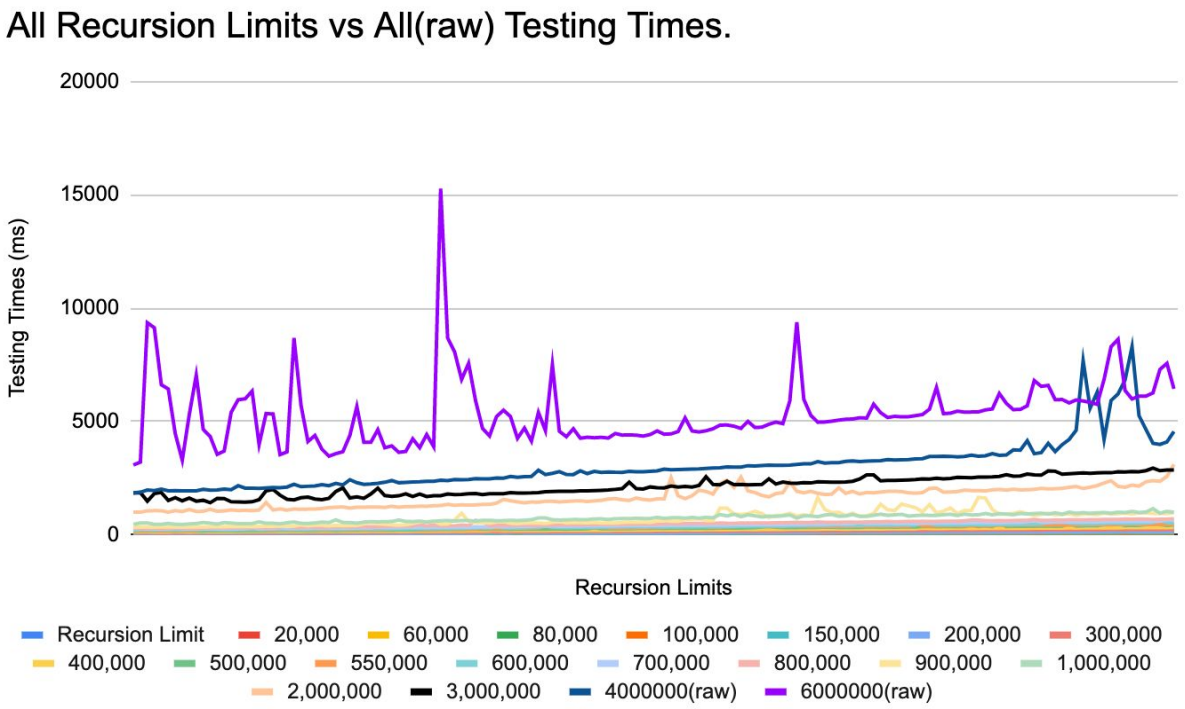
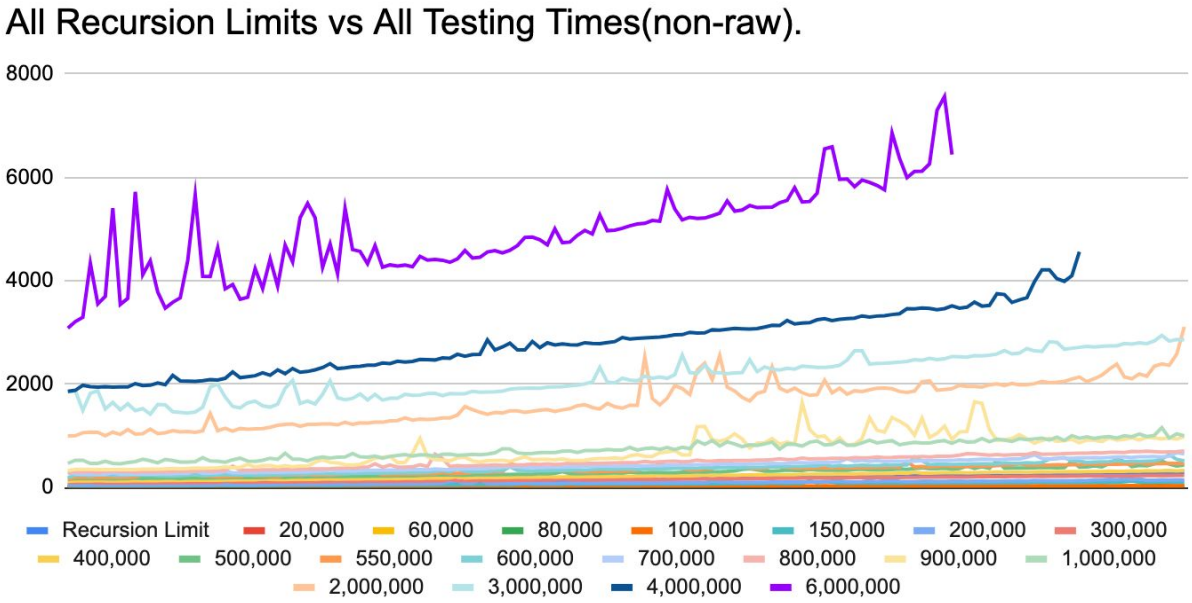


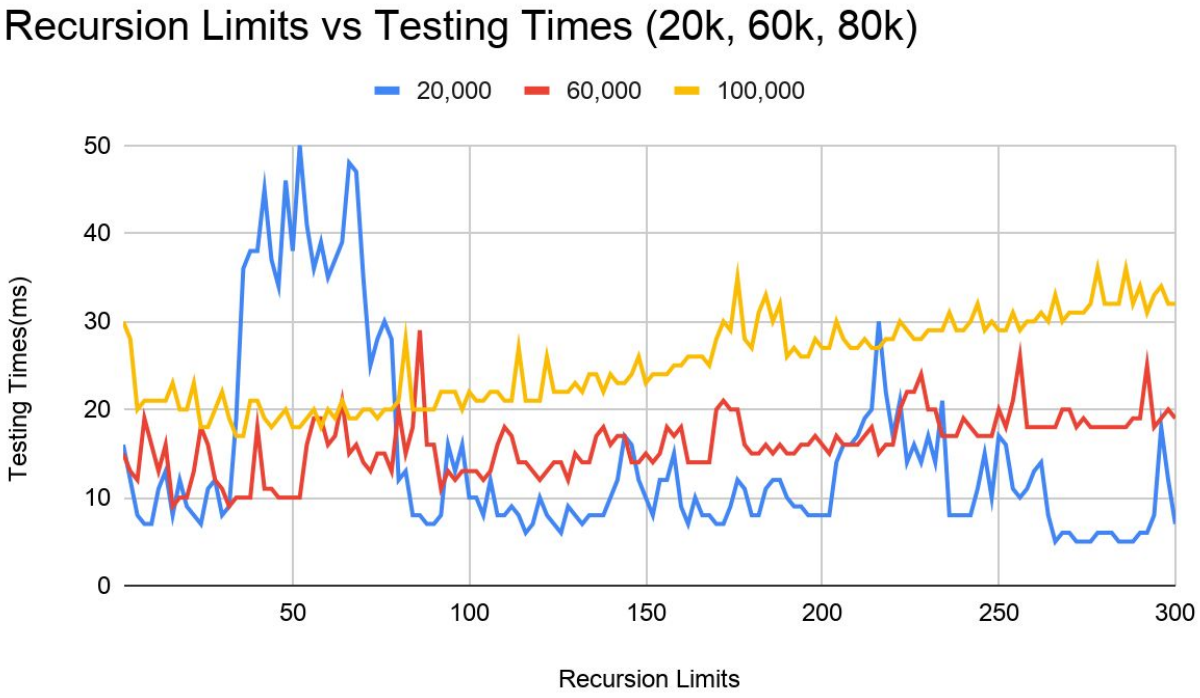
- Comments: One interesting observation would be the inconsistency of the 6 million array size. While the other array sizes have inconsistency as well, the 6 million array size or the largest array has the most inconsistency or outliers. I think the reason for the high number of outliers could be the processor of the computer. As the processor tries to find the testing times for the larger arrays, the lags, which would result in more outliers.



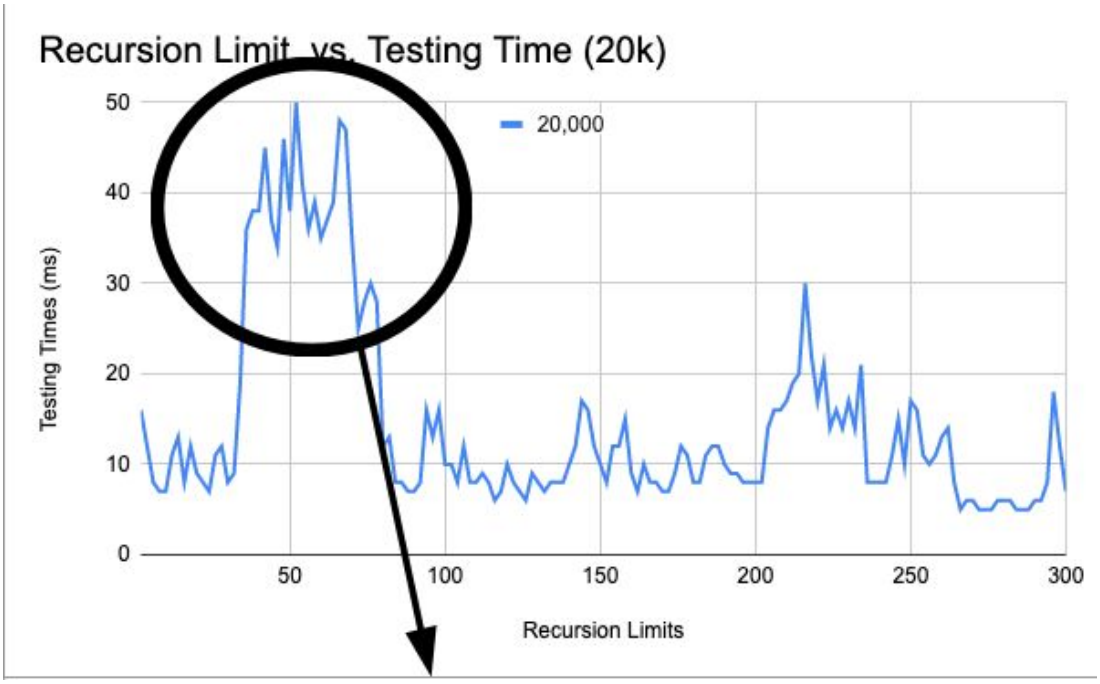
Comments:

- Even after the outliers are taken out of large array, the graph has more outliers than the other array sizes.

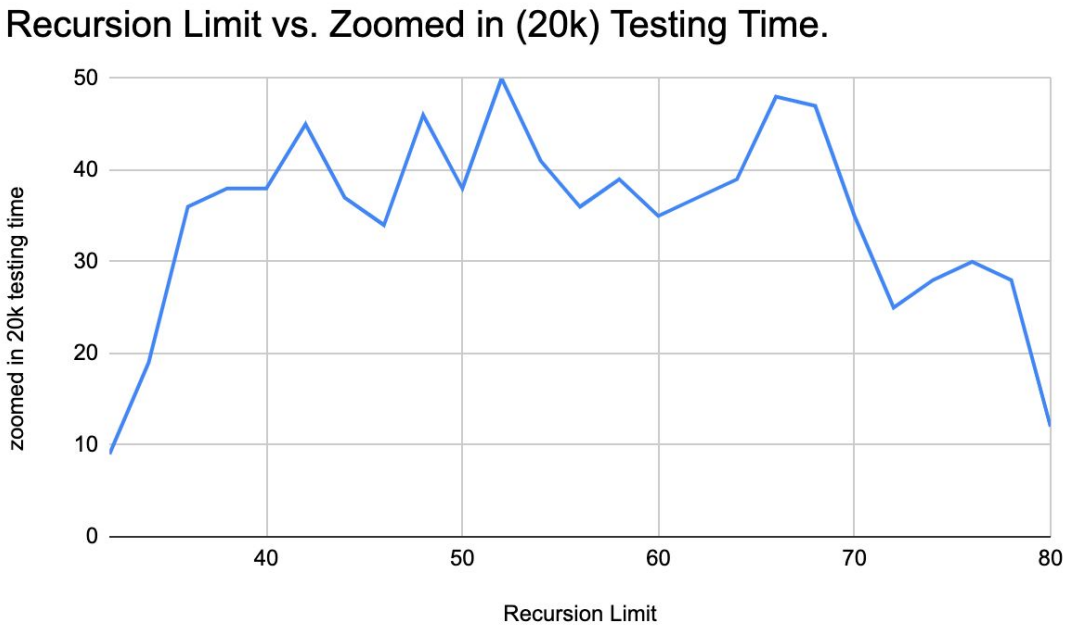
- One obvious observation in these graphs would be as we increase the array size, there would be an increase in the processing times as well. We can examine that as we move on to greater array sizes, the testing times would gradually increase. After this observation, it can be concluded that the smaller recursion limits would result in faster testing times.



Comments: In this graph the testing times of 20k, 60k, and 80k are compared. As seen in the image, the graph of 20k has a sudden increase that could be reaching its optimal limit between the recursion limits of 44 to 76. When we compare the 20k graph to the 60k and the 80k, we can examine that the 60k and 100k graph are more flat compared to the 20k graph. There is an unexpected rise in the testing time for the 20k array size graph.



- Comments: This graph just compares the recursion limits to the 20k array size without comparing it to the 60k or 100k array size. This graph is not flat and there's constant change in the testing times. Furthermore, there is a sudden increase and then a sudden decrease.



- Comments: In the zoomed version of the graph, we can examine a sudden spike in the testing time and upon further examination and testing of the 20k array size, there was always a sudden spike between the fifty recursion point to the 100 recursion point. Similar to this graph, the sudden spike in the could be due to a lang in the processor as we are about to run the code.