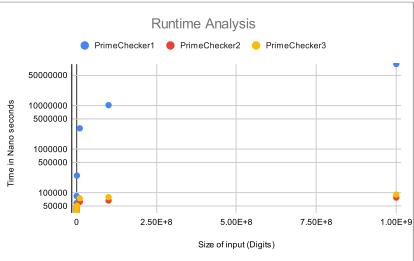


Digits	PrimeChecker1	PrimeChecker2	PrimeChecker3
1000	40120	34630	34990
10000	58400	41890	41500
100000	84490	46950	46070
1000000	249010	51280	50740
10000000	3029500	62230	74970
100000000	10301660	66720	79820
1000000000	90568820	78080	91390



Overall PrimeChecker2 and 3 would be the most efficient algorithms. This is because PrimeChecker2 and 3 will not grow as much as PrimeChecker1 because their big $O(\sqrt{n})$ will not grow as fast as PrimeChecker1's big $O(n)$ which is linear. Although the data shows 2 is faster than 3 at certain data points, and vice versa, this is because the data is averages taken from different inputs.

