

Program Structures and Algorithms  
Spring 2023(SEC –1)

NAME: Prem Kumar Raghava Manoharan  
NUID: 002726784

**Task:**

To Solve 3-SUM using the Quadrithmic, Quadratic, and (bonus point) quadraticWithCalipers approaches, as shown in skeleton code in the repository

**Conclusion:**

The quadratic method, also known as the "two-pointer" method, works well for solving the 3-sum problem because **it takes advantage of the ordered nature of the input array since the array is sorted.** By using two pointers, one starting at the beginning of the array and the other at the end, the algorithm can quickly move towards the desired sum by incrementing or decrementing the pointers as needed.

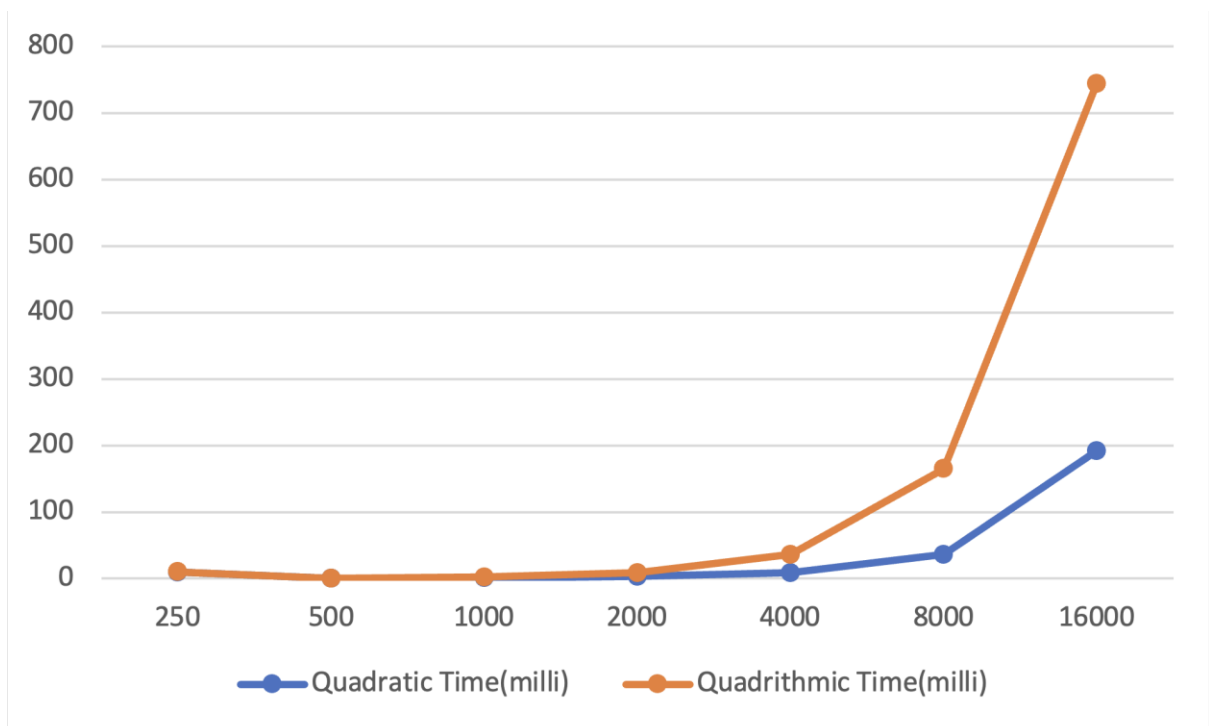
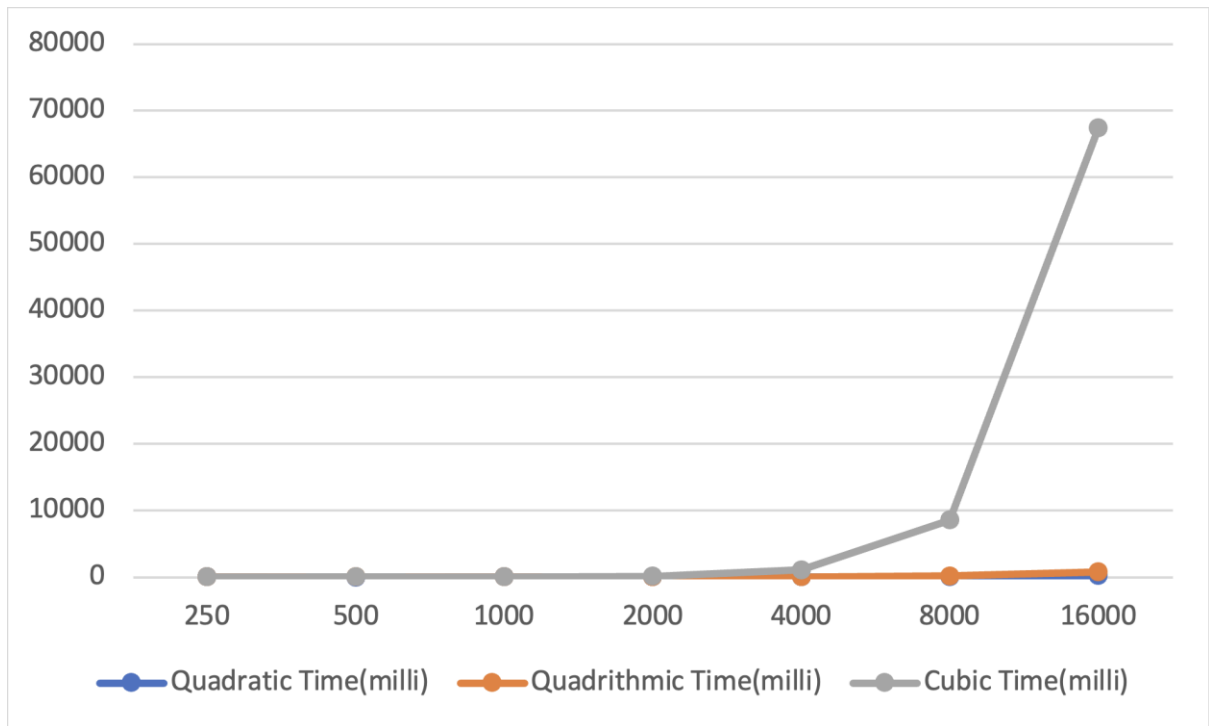
In contrast, a cubic or  $O(n^2 \log n)$  approach would involve nested loops, resulting in a higher time complexity and slower performance. The quadratic method can solve the problem in  $O(n^2)$  time, which is more efficient.

**Evidence to support that conclusion:**

N	Quadratic Time(milli)	Quadrithmic Time(milli)	Cubic Time(milli)	QuadraticCaliper Time(milli)
250	0.6	0.3	1	0.5
500	0.4	0.4	2.5	0.2
1000	1.2	1.3	18.7	1.3
2000	3.2	5.7	132.8	1.3
4000	8.7	27.7	1049	6.8
8000	36.6	128.5	8370.9	31.8
16000	192.2	551.9	66621.6	174.5

- For N=250 which is comparatively lesser input which show  $O(n^2 \log n)$  is efficient in this case.
- But when N increases like
  - For N=500 Quadratic and Quadrithmic take same time and Cubic take more time than the Quadratic and Quadrithmic.
  - For N=1000 to N=16000 Quadratic take very less time and it is more efficient than other 2 algorithms.
- It is clearly demonstrated using two graphs
  - First one for comparing Quadratic, Quadrithmic and Cubic, since cubic has larger value compared to other two. The difference between Quadratic and Quadrithmic is not clearly visible.
  - So, the second Graph is for comparing the Quadratic and Quadrithmic alone and it is clearly visible that Quadratic is more efficient than Quadrithmic.
  - This is because **it takes advantage of the ordered nature of the input array since the array is sorted.**

### Graphical Representation:



## Unit Test Screenshots:

