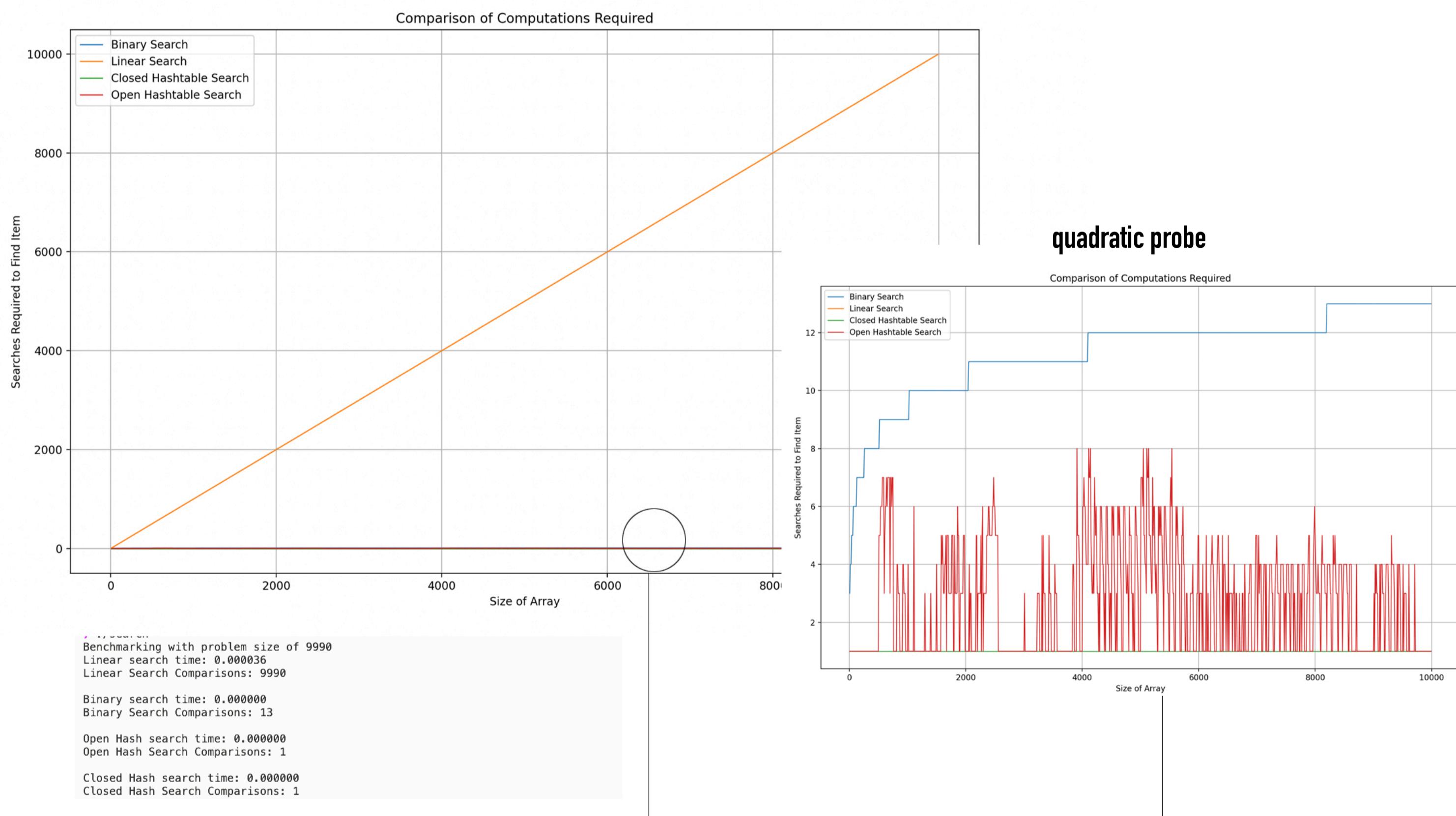


Item at end

When searching for an item at the end of an array, the linear search algorithm will always perform its worst at $O(N)$ time, traversing the entirety of a list before finding its item. Binary search will perform at its average case, $O(\log(N))$, as shown by its logarithmic curve here. Ideally, both maps will perform near $O(1)$. Unfortunately, my open map's algorithm (red) seems to sometimes cluster particular sets of larger valued keys, in terms of hashing strings of digits that are longer. As shown in this graph, the open map has clusters around certain key values, degrading its search time to $O(N)$ in those areas (though clearly still outperforming linear, even at its worst. The closed map continues to perform at near constant time.



As seen here, quadratic probing resolved the clustering and performance degradation issues with the open map over these hashed key values. This was the case for all value ranges (front, middle, back, item not found).