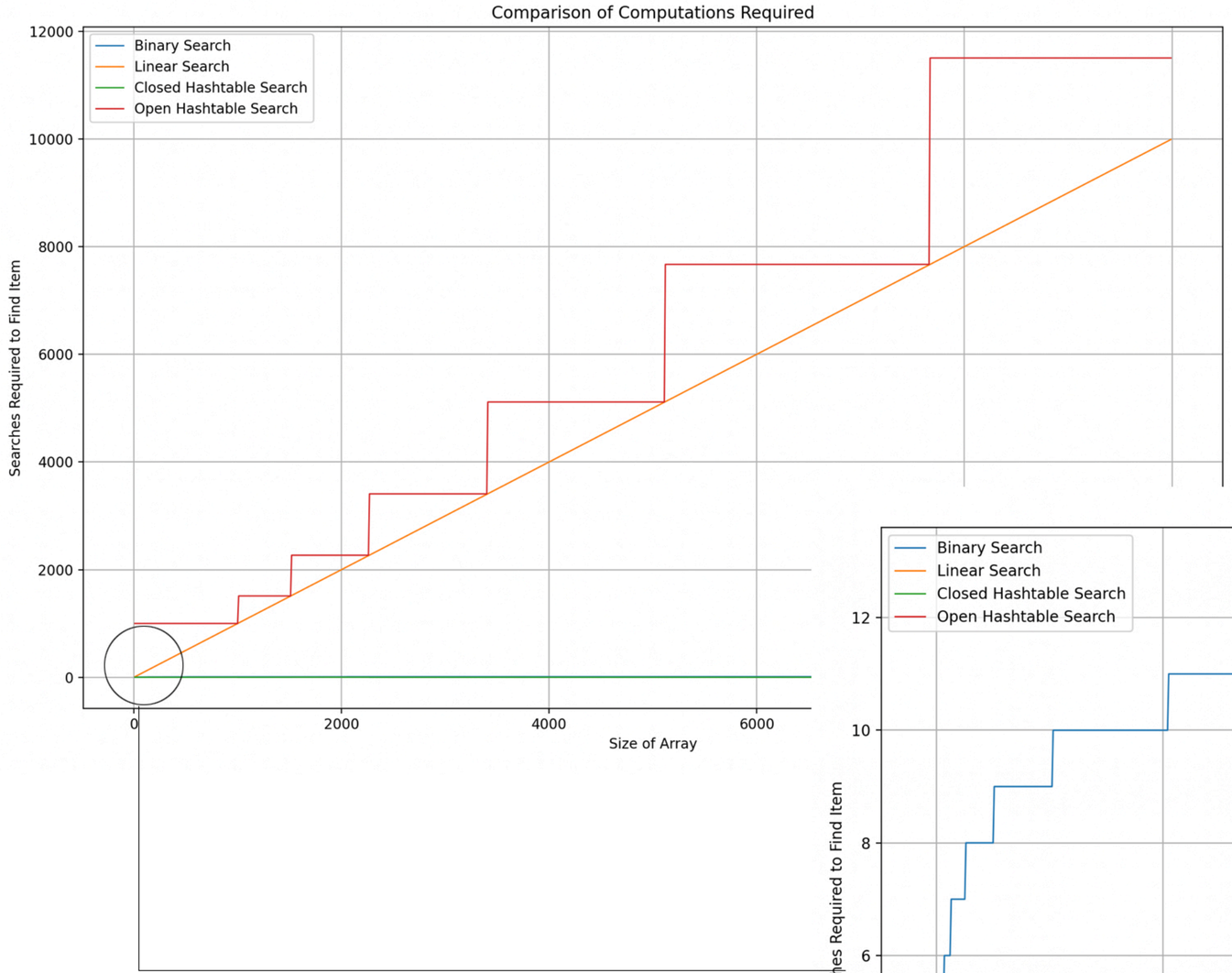
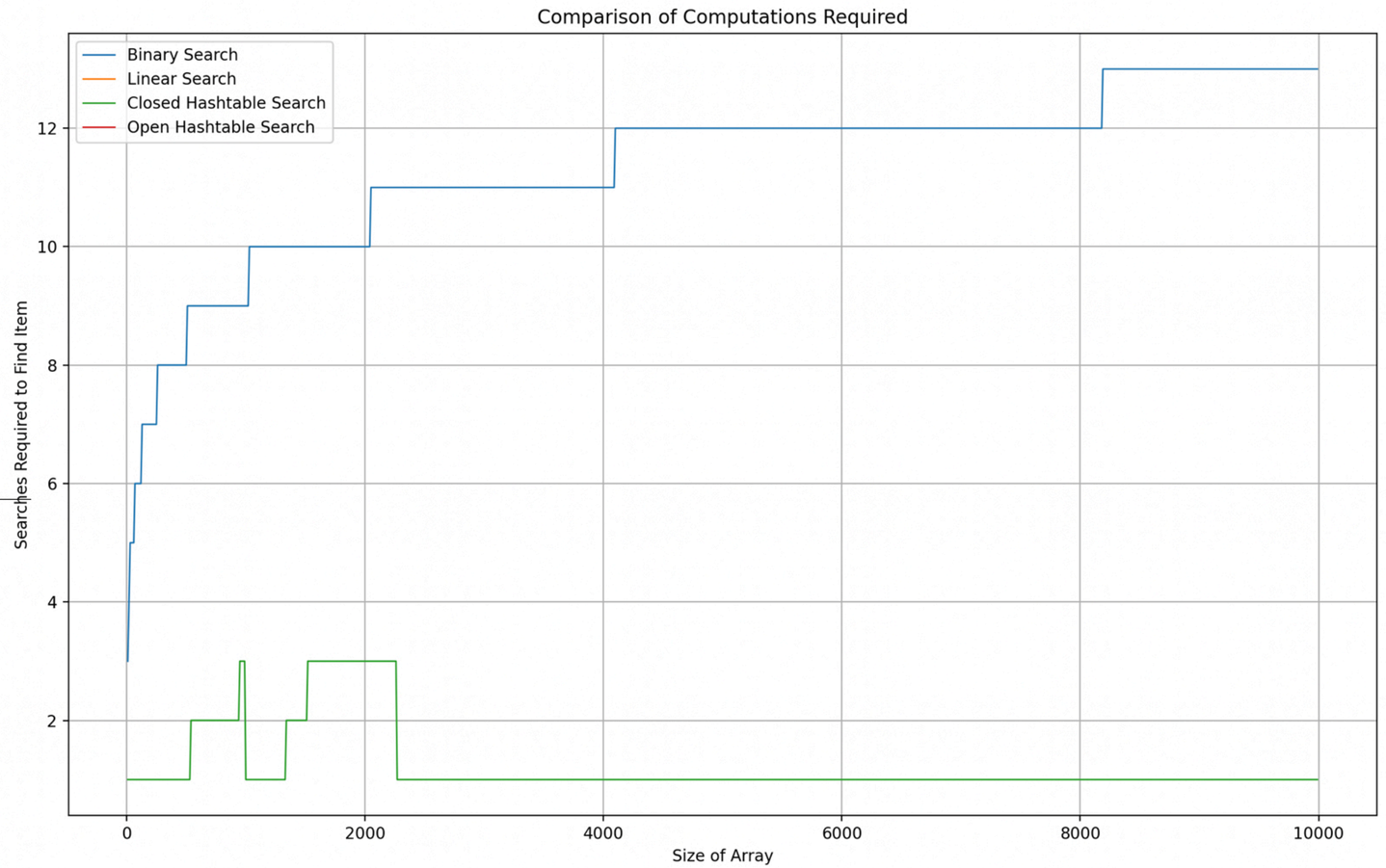


Item not found

When searching for an item not present in an array of an ordered array, the linear search algorithm will always perform at its worst case O(N) , traversing the entirety of a list before seeing that the item is not present. Binary search will perform at its it's average case, O(log(N)), as shown by its logarithmic curve here. For the maps, this is where the closed map significantly outperforms the open map. The open map (at least with how I coded the 'contains' method) will start searching for the item at its hash index, then continue to search every 'bucket' in the map via wrap around. This is why the open map has the O(N) stepwise appearance, and even take more steps than the linear algo, since the open map will always have more buckets in the map than actual items in the map. On the other end of the spectrum, the closed map will only search whatever nodes are present at that hash address, still maintaining near constant time look up in this scenario.




```
Benchmarking with problem size of 9990
Linear search time: 0.000036
Linear Search Comparisons: 9990

Binary search time: 0.000000
Binary Search Comparisons: 13

Open Hash search time: 0.000161
Open Hash Search Comparisons: 11504

Closed Hash search time: 0.000000
Closed Hash Search Comparisons: 1
```