

Name: Daniel Coblantz

Date: 2/5/2025

Professor: Li

Analysis:

As the array size n grows, the total time for linear search increases correspondingly, while binary search remains relatively fast. In larger inputs (e.g., $n \geq 1,000,000$) linear search times become notably longer, but binary search stays efficient due to its $\log(n)$ time complexity. Even though in most cases the odds of getting a match are slim due to randomness. The difference in search times in linear search is dramatically larger than binary search due to how they handle searching for elements.

Results table:

Data size (n)	Search entries (m)	# of found	Avg time spent per linear search	Avg time spent per binary search
100	50	0	8.6666E-7 seconds	2.2916E-7 seconds
10,000	1,000	0	8.850709E-6 seconds	2.21542E-7 seconds
1e6	1000	0	1.75220625E-4 seconds	2.3454199999999998E-7 seconds
1e8	1000	0	0.017319459833 seconds	8.57416E-7 seconds

***note** I used a long data type for generating the values so the odds of finding a target is very slim, not sure if this was the right approach for the assignment.