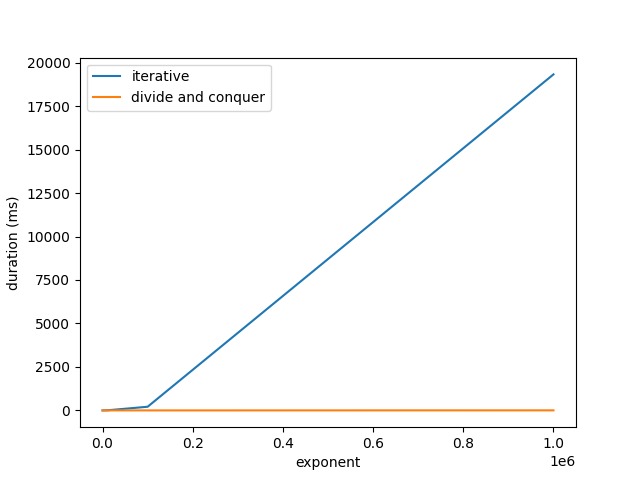
1. B) **Asymptotic running time complexity:**

* Naïve method: O (n), as it involves a loop that runs n times; n = exponent.
* Divide-and-conquer: O (log n); using recurrence relation: T (n) = T (n/2) + O (1).

C)

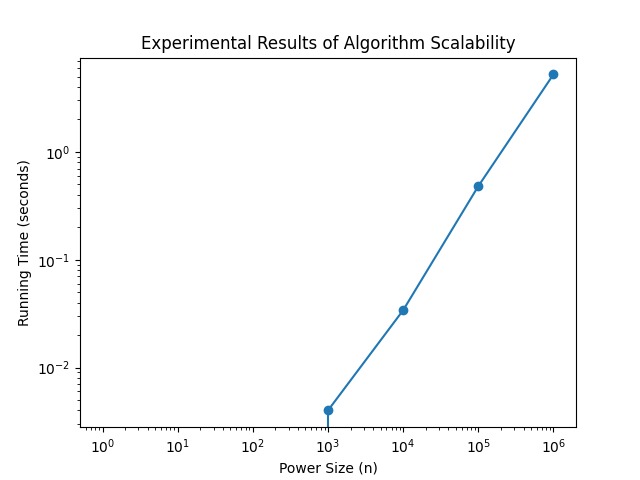


D) From the graph, we can confirm the scalability of the algorithm and validates that the experimental results are similar to the theoretical analysis.

1. B) **Asymptotic running time complexity:**

* Algorithm made: O (n log n)
* Merge sort: O (n log n)
* Binary search: O (log n)

C)



By examining the plot of running times for different input sizes, we can see if the algorithm maintains an expected growth rate. If the empirical and theoretical analyses align, then it confirms the scalability of the algorithm. As input increases, running time grows in a logarithmic way.