**Big O notation**

A mathematical notation used to describe an algorithms runtime and space requirements

Representation: O( )

O(1) – Constant Time

O(log n) – Logarithmic Time

O(n) – Linear Time

O(n log n) – Linearithmic Time

O(n^2) – Quadratic Time

O(2^n) – Exponential Time

O(n!) – Factorial Time

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Best case** | **Average case** | **Worst case** |
| Linear Search | O(1) | O(n) | O(n) |
| Binary Search | O(1) | O(log n) | O(log n) |

**Which is Better and Why?**

* **For Small or Unsorted Data**: Linear search is acceptable and simple to implement.
* **For Large, Sorted Data**: Binary search is much faster (logarithmic time) and ideal for scalability.