# What is a Big O

# What is a Big O

**What is Big O ?**

**Big O** notation is used in Computer Science to describe the performance or complexity of an algorithm. **Big O** specifically describes the worst-case scenario, and can be used to describe the execution time required or the space used (e.g. in memory or on disk) by an algorithm

[A beginner's guide to Big O notation - Rob Bell (Links to an external site.)Links to an external site.](https://rob-bell.net/2009/06/a-beginners-guide-to-big-o-notation/" \t "_blank)

https://rob-bell.net/2009/06/a-beginners-guide-to-big-o-notation/

Things That Are O(n2)

* Nested for-loop sort of an array or linked list
* "Insertion sort" of a randomly ordered array or linked list
* "Bubble sort" of an array or linked list
* Using an array or linked list for tracking for unique values

O(n2) is also referred to as **"quadratic** **timing complexity"**.

Things That Are O(n)

* successful search of a randomly arranged array or linked list
* unsuccessful search of a randomly arranged array or linked list
* counting the number of nodes in a linked list
* finding the last node in a linked list
* queue pop operation when using an array and shifting values
* queue pop operation when removing from the tail of a singly-linked list

O(n) is also referred to as **linear** timing complexity.

Things That Are O(1)

* stack push operation when adding to the end of an array or head of a linked list
* stack pop operation when removing from the end of an array or head of a linked list
* queue push operation when adding to the end of an array or head of a linked list
* queue pop operation when removing from the head of a singly-linked list
* opening a data file for input

O(1) is also referred to as **constant** timing complexity.