Common Runtimes

Name	Big O Notation	Description	Example
Constant time	O(1)	Same amount of time, regardless of the number of elements	Random access array
Logarithmic	O(log n)	When doubling the number of elements doesn't double the time (binary trees)	Search algorithms
Linear	O(n)	Adding element increases runtime linearly	Looping an array / list
Quasilinear	O(n log n)	Every element has to be compared with every other element. Lots of comparisons.	Sorting algorithms
Quadratic	$O(n^2)$	2,4,8,16,32,64 Increasing quadratically	Nested loops
Exponential	$O(2^n)$	Recursion	Fibonacci Series