# **BIG O**

# **INTRODUCTION**

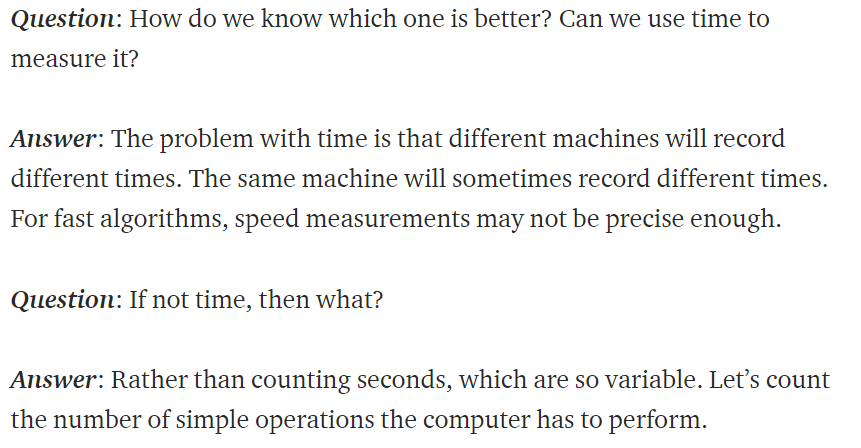
* How much time does this algorithm need to complete?
* How much space does this algorithm need for computing?

We are trying to understand how quickly the runtime of algorithm grows if a size of inputs grows

Big O is the measure that mean how long time takes to run an algorithm in the worst case

Big O notation is the measure of scalability of algorithm (code). It’s an upper measure

# **SPEED MEAUREMENT**



# **TWO TYPES OF COMPLEXITY**

There are 2 types complexity:

* Time complexity – amount of time taken to run an algorithm
* Space complexity – amount of resources(usually space) taken to run an algorithm

# **COMPLEXITY**

O(1) Constant time- no loops

O(log N) Logarithmic time- normally searching algorithms have log n if the input is sorted (Binary Search)

O(n) Linear time- for loops, while loops through n items

O(n log(n)) Log Linear — usually sorting operations

O(n²) Quadratic- every element in a collection needs to be compared to every other element. Two nested loops

O(2^n) Exponential- recursive algorithms that solve a problem of size N

O(n!) Factorial- we are adding a loop for every element

# 

# **FIND THE BIG O COMPLEXITY OF AN ALGORITHM**

Rule 1: Worst case

Rule 2: Remove the leading constants

Rule 3: Different terms for inputs

Rule 4: Drop nondominant terms

# **BEST CASE SCENARIO, AVERAGE-CASE SCENARIO, AND WORST-CASE SCENARIO**

**Best case — being represented as Big Omega — Ω(n)**.

**Average case — being represented as Big Theta — Θ(n)**.

**Worst case — being represented as Big O Notation — O(n)**.

Note: programmers typically assess the worst case O(n)

|  |  |  |  |
| --- | --- | --- | --- |
| ID | DATA STRUCTURE | TIME COMPLEXITY | SPACE COMPLEXITY |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| ID | ALGHORITHMS | TIME COMPLEXITY | SPACE COMPLEXITY |
|  | SLIDING WINDOW | O(N) | O(1) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

