Data Structures:

1. It organizes and store data

Example: Array, Tree

1. Each has its own strengths and weakness

Example: Array is great when we know the index to access, when we don’t know the index we cant access the array

An algorithm defines the steps needed to perform task

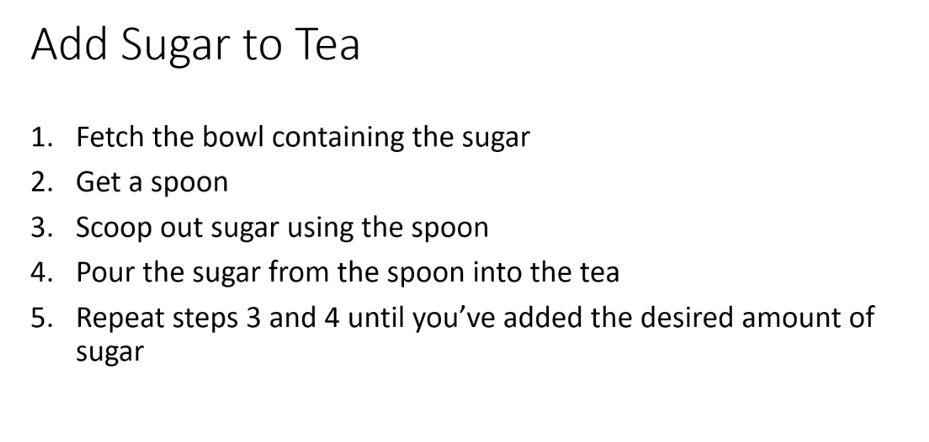
An algorithm is not the implementation, it just describes the steps

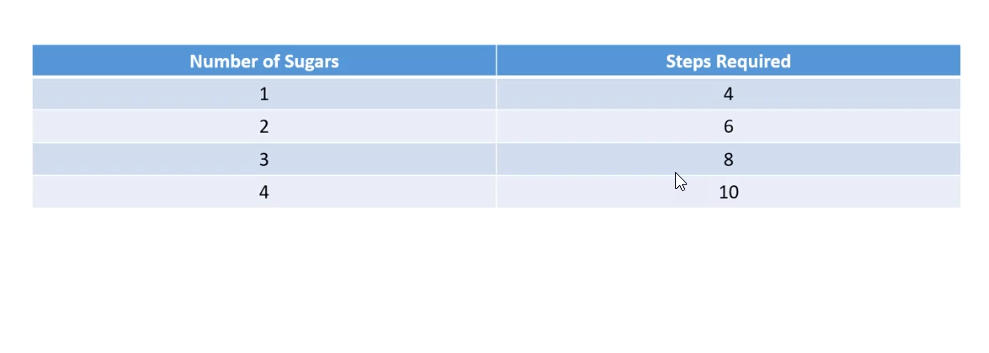
Big O Notation : Way of measuring, how an algorithm performs

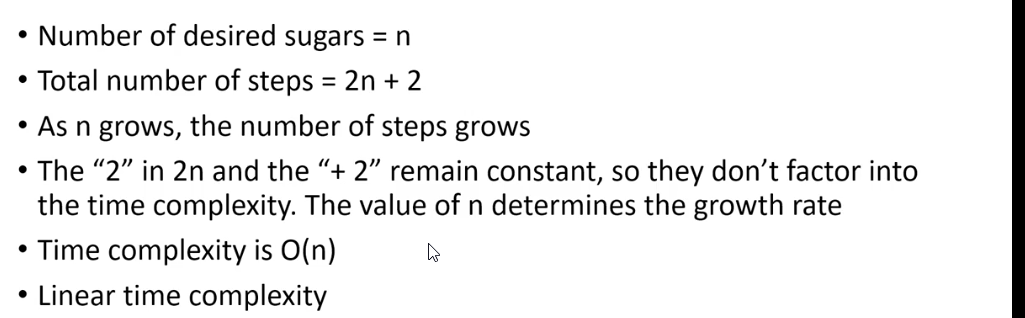
Time Complexity: Number of steps involved to run an alg

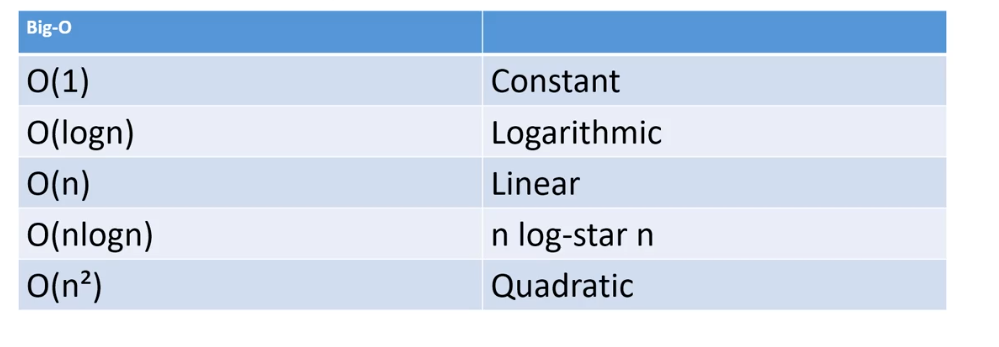
Memory Complexity: Amount of memory requires to run an alg

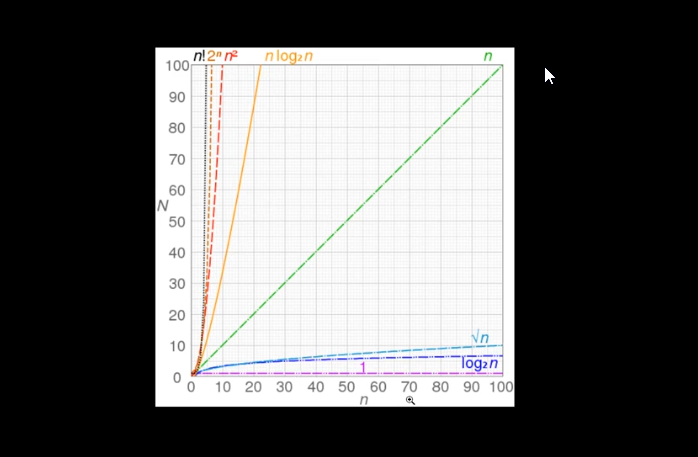
Example:









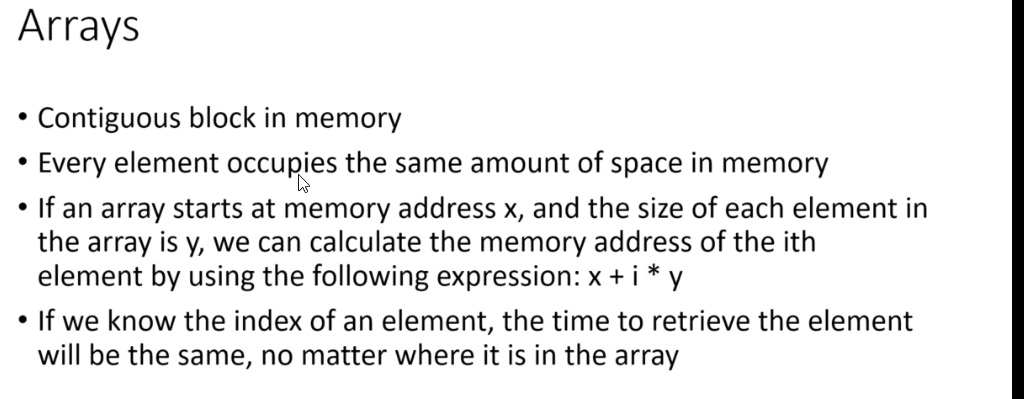


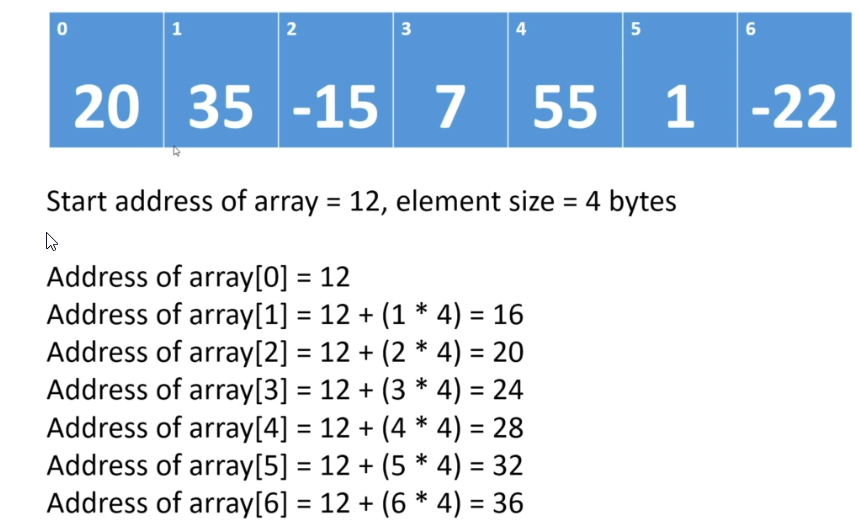
<https://en.wikipedia.org/wiki/Big_O_notation#/media/File:Comparison_computational_complexity.svg>

Arrays:

Contiguous block in memory – It is one huge block – means size needs to be defined

Every element occupies the same amount of space in memory





Here 12 is the Bogus memory address space

Why Array index starts at 0

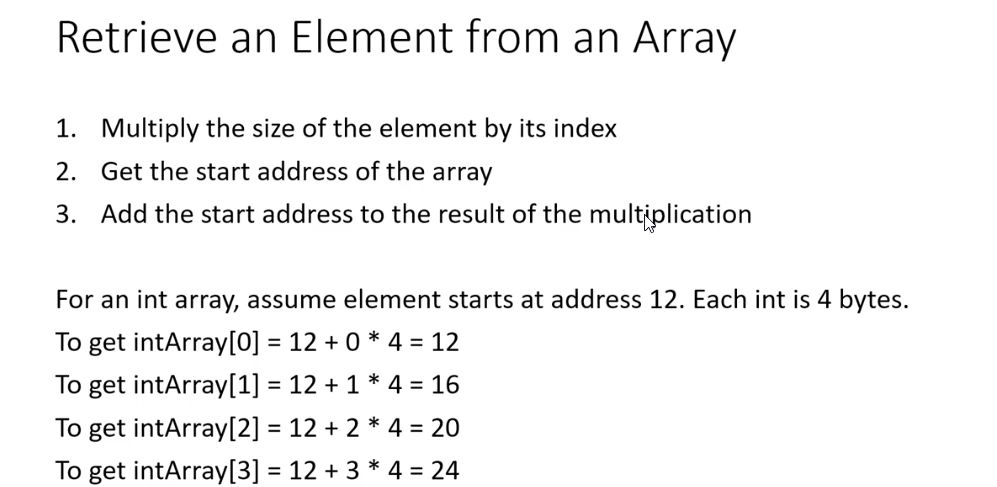
If we look in above image, the formula Is x + (I \* y)

X = some memory address

I = index element

Y = size of each element

Arrays are efficient when we know the index value from the array



Retrieval in array when you know the index value can be done in O(1) times

Retrieval in array when we not know the index value is 0(n) times – Linear Complexity

