day 01

Big O notation

> Landau notation
>
> Describes the limiting behavior of a function when the argument tends towards a particular value or infinity
>
> A description of a funtion in terms of big O notation usualy only provides an upper bound on the growth rate of the function
>
> There are several related notions, using the symbols θ to describe other kinds of bounds on asymptotic growth rates

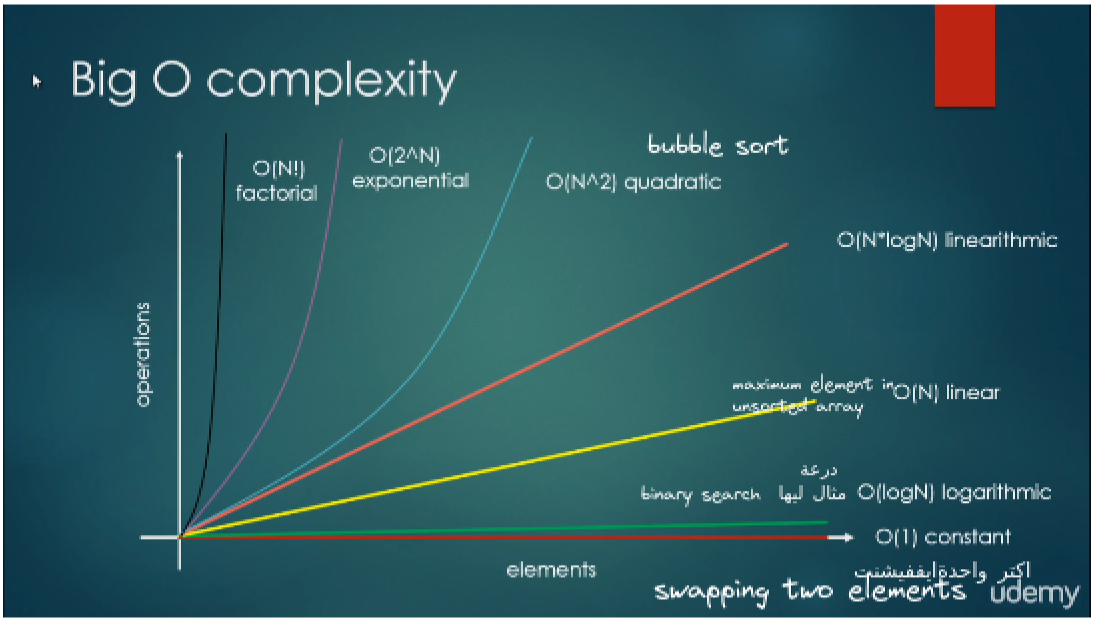
---

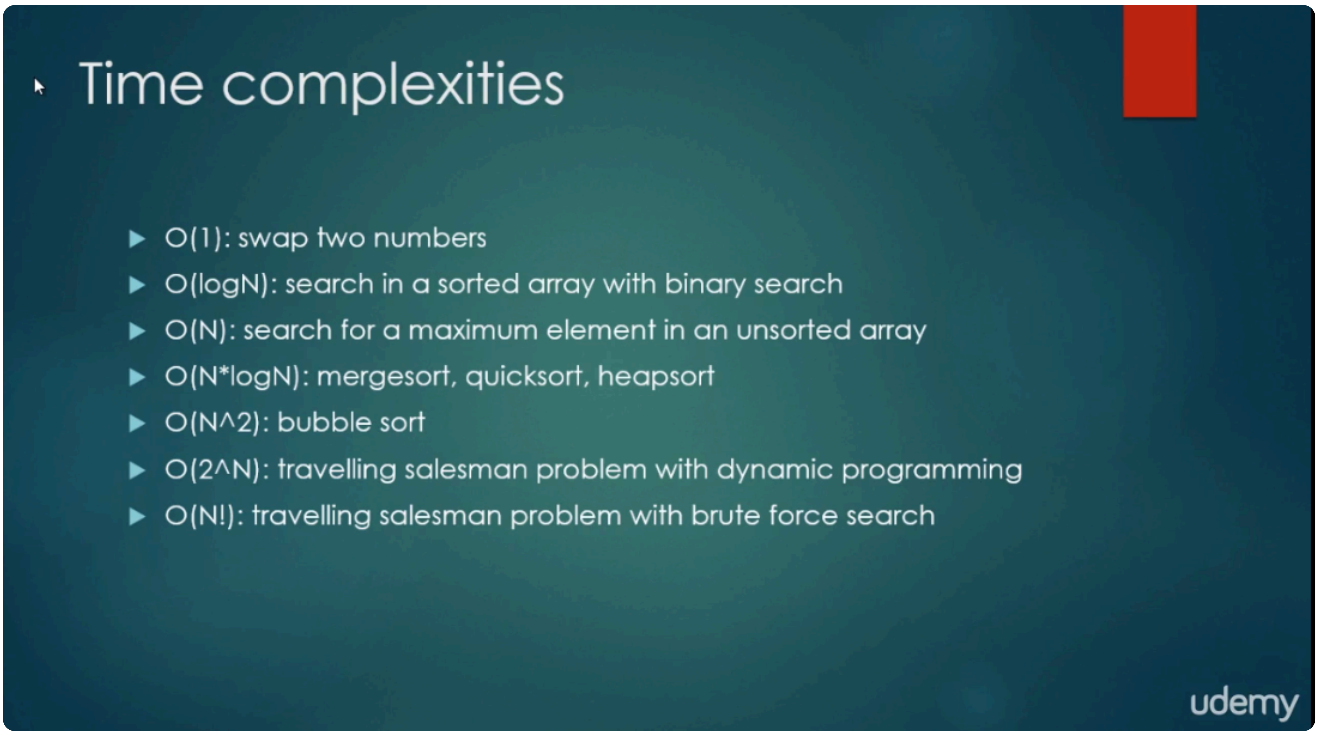
Let f and g be two functions on some subset of real numbers,
we say

$$f(x) = O(g(x)) \ \ as \ \ x \to \infty$$

if and only if there is a positive constant c such that for all sufficiently large values of x, the absolute value of $f(x)$ is *at most* (c * the absolute value of $g(x)$)

> if $f(x)$ is a sum of several terms, the one with the larges growth rate is kept, and all the others are omitted $O(N + \log N) = O(N)$
>
> if $f(x)$ is a product of several factors, any constants are omitted $O(c * N) = O(N)$





Complicity Classes

Polynomial (p)

One of the most fundamental complicity classes

Contains all the decision problems that can be solved by a deterministic Turing machine

Efficiently solvable

for example : Sorting algorithms

Non-Deterministic Polynomials (NP)

If you have a solution to a problem, we can verify this solution in a polynomial time by (a deterministic Turing machine)

P is in NP

Most important question (N = NP) is true ?

Examples : Factorization (RSA), Traveling sales man

NP - complete

A decision problem is NP-complete if it is both NP and NP-Hard

We usually look for an approximate soultion

Heuristics

Examples : Chinese postman problem, graph coloring, Hamiltonian cycle

NP - Hard

A class of problems that at least hard as the hardest problems in NP

A problem H is NP - Hard when every problem L in NP can be roduced in polynomial time to H

For example : The halting problem