

Time Complexity

Time Complexity	Name	Comparison
1	Constant	افزایش کل اعداد الگوریتم
$\log n$	logarithmic	افزایش شیبی به $\log n$
n	linear	افزایش شیبی به n
$n \log n$	linear logarithmic	افزایش n^2 و $n \log n$
n^2	Quadratic	افزایش n^3 و $n \log n$
n^3	cubic	افزایش n^2 و 2^n
2^n	exponential	افزایش 2^n و $n!$
$n!$	Factorial	افزایش 2^n

$$P(n) = 3 + 5n + 4n^2$$

$$P(n) = O(n^2)$$

مثال
افزایش n^2 و n و 3
همه با هم $O(n^2)$

حساب ال $Big O(P(n))$

$$P(n) = 2n^3 + 3n^2 + 100n$$

$$P(n) = O(n^3) \rightarrow$$

عناصر n^3 و n^2 و n

بزرگترین
order
عند n

- used to measure the Performance of any algorithm.
- By providing the order of growth of the function
- upper bound \rightarrow Worst case اسوأ حاله

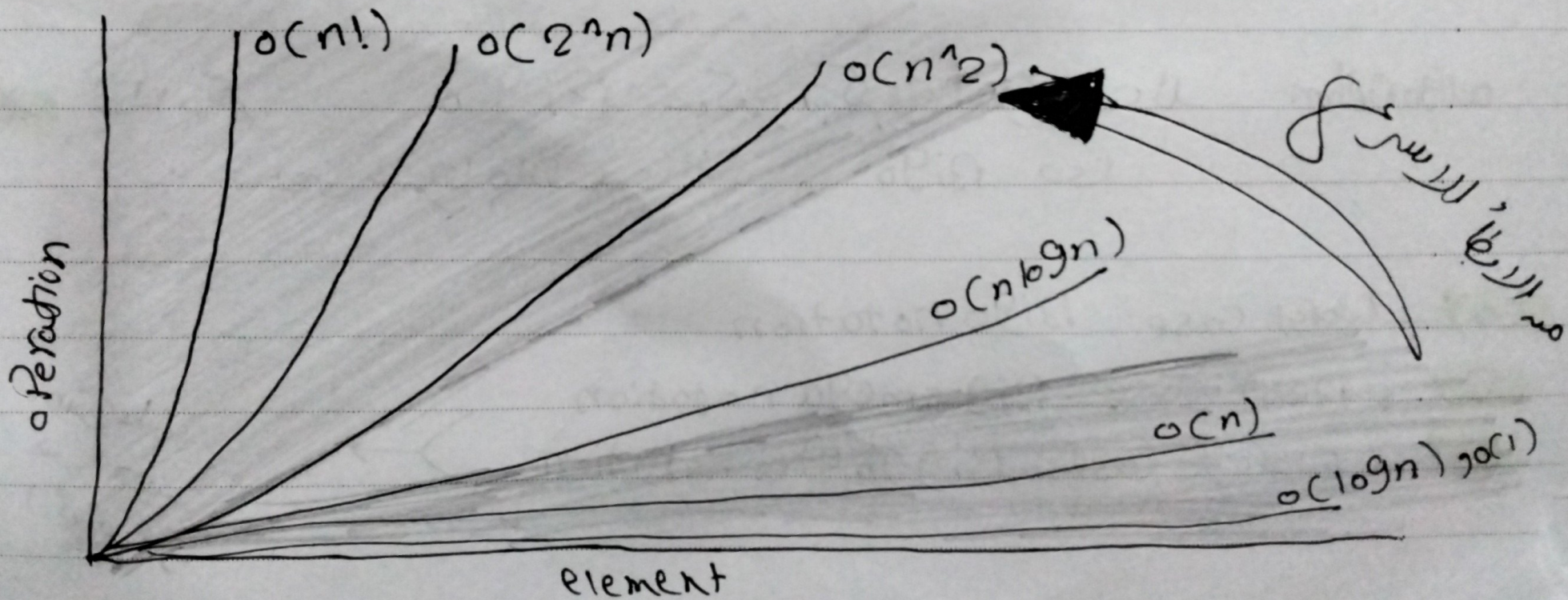
لذا ذکر $Big O$ - که بزرگترین $Big O$ الگوریتم
یعنی عمرها ما را می دهد $Big O$ دی

O \rightarrow Worst case $Big O$ notation

Ω \rightarrow Best case $Big \Omega$ notation

Θ \rightarrow Average case $Big \Theta$ notation

همه چیز
بزرگترین
بزرگترین



من الارتفاع للسرعة

Complexity algorithm

Running algorithm → Time

بمختلف machine للتانيه

- Factors
- ① single Processor or multi-Processor
 - ② Core Speed GHz 2.5
 - ③ 32/64 bit
 - ④ Ram
 - ⑤ Programming language
 - ⑥ Compiler
 - ⑦ Input →

العوامل التي يتوقف عليها

وعدد المدخلات

مثال ١

Pre quency count method

Function sum(a,b)

Result = a+b

Return Result

end Function

Cost , Return , ...

$P(n) = 3$

$O(Pn) = 3$ → عدد الخطوات

Arithmetic operation

+ - / * 1 unit of time

Assignment operator

x = 5 1 unit of time

Comparison

x < y 1 unit of time

Return statement

1 unit of time

مثال ٢

① Function Arraysum(arr,n)

② Sum = 0 → 1 unit of time

③ For(i=0; i < n; i++) → loop

④ Sum = Sum + arr[i];

Return Sum;

في حالة بيغير وال

Comparison , increment

لكن i = 0 هو اوله

1 unit

لكن المقارنة لا تتكرر time ← n+1 بالصفحة على بتسا

time ← n ← i++ increment

unit of time ← 1 ← Return وال

السطر 4 ← عملية ← 2n/unit

$P(n) = 1 + 1 + n + 1 + n + 1 + 2n = 4n + 4$

$O(Pn) = n$

لوعايز

الخطوة

Function Print2Darray (arr, n)

مثال ٢

① For (i=0; i<n; i++)

i=0 → مرة واحدة

② { For (j=0; j<n; j++)

i<n → n+1

{ Print (arr[i][j]);

i++ → n

??

* الطريقة ① $2n+2$ بها كود ؟

من Zero تعتبرها n (طول

دوم) أي فاهمين

لكن لو مبدأ من الصفر يكتبها $2n+2$

مهم هنا بالتوازي والعمل

السطر (برود) $n \times n$ ← loop

السطر (م) $n \times n$ ← loop
loop

$$P(n) = 2n^2 + n$$

$$O(f(n)) = n^2$$

مثال ٣

$$2^k \geq n$$

$$k = \log_2(n)$$

$$1 \leftarrow i = 1$$

$$1 \leftarrow n = 10;$$

$$\log_2(n) \leftarrow \text{While } (i < n) \{$$

$$\log_2(n) \leftarrow i *= 2 \}$$

تكرار

$$\begin{array}{r} i \\ 1 \times 2 = 2^1 \end{array}$$

$$2 \times 2 = 2^2$$

$$2^2 \times 2 = 2^3$$

$$O(f(n)) = \log n$$