

# Complexity Analysis [1/2] :- operations vs Input

الهدف من عدد الـ operations الكور، ببساطة والهدف الثاني هو ان يكون الـ input  
 في انما هو مهم هو قد قدي يتغير اي كان الـ input (بما ان الـ input هو المهم ولكن الـ input  
 الذي يتغير عدد الـ operations مع تغير عدد الـ input. يعني مثلا لو كان الـ input 5 فليكن الـ input  
 10 فلو غيرت الـ input بـ 20 فكم عدد الـ operations مختلف عن القابل ولا لا؟  
 كل الـ input اضعف عدد الـ operations بـ 2 ولا يتغير الـ input!  
 (فرا حذر كونه ممكنة لا الـ input)

Cases:   
 - Best Case   
 - Average Case   
 - Worst Case

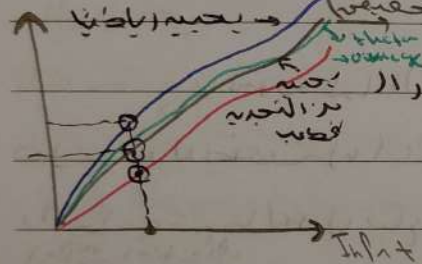
الـ input لو قيمته كذا ده اوضحتر حاجة والكور فنوضح  
 اكبر عدد الـ operations فوال (الـ input)

## Notations: - 1) O Notation

## 2) Omega Notation

## 3) Theta Notation

n. of operations



لوانا المدة الفعلية لوقت انجرب عدد الـ input الحرفي معينه (الحقيقي)  
 على ان يبقا من وايضا فنحذف رياضيا نحسب العلاقة بين الـ input وعدد الـ operations  
 لانه حسب. ففي تقوية Notation بتساوي  
 O Notation - انما نطابق من يتوافق عدد الـ input و O Notation

الـ input الحقيقى (الحقيقي) ايضا لو ايرى الـ input انما هو الـ input الحقيقى

Omega - من في معظم الاحيان تكون الـ input الحقيقى يعني بياخذ عدد الـ operations اقل


Linear Relation -  $O(n)$   
 operation عدد الـ input

من الوضع الحقيقى (ببني تطور انصاف الحقيقى)  
 theta - يتعامل نظاير الـ input (متر بمر انصاف)

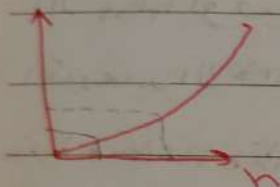
الـ input بتساوي الـ input لـ  $O(n)$  لان صاير الـ input الـ theta

- Rule 1) the running time is the sum of all (consecutive blocks)   
 2) the time complexity of next block  $\rightarrow$  multiplication of its time   
 3) drop the lower terms (  $O(n^2 + n)$  )   
 (المؤثرين الوحيد على)

- 4) لو عندك الـ input بتساوي الـ input وناظرا   
 5) Drop constants (addition, subtraction, multiplication, Division)


 }  $O(n)$  size  
 drop complexity  
 ( $O(2n)$ )

$n^2 \in O(n^2)$   $n^2 \in \Omega(n^2)$   
 $n \log n \in O(n^2)$   $n \log n \in \Omega(n)$   
 $n^2 \in O(n^2)$   $n^2 \in \Omega(n^2)$



$(100 \times 100)^2 \leftarrow \text{halving}$   
 $(100 \times 1000)^2 \leftarrow \text{halving}$   
 $h=2 \rightarrow p=50$   
 $h=10 \rightarrow 1250$

الزيتون في ١١ (١٩٤٤) أكبر بكثير من المستحيل

الخلافة ينتهون لما ارادوا زاد للظف بملا عدد الاستعدادات في سيرها اي الميصول فيه  
حوييريه للظف (مان) ولا حل يريه اربع اوقات (ثمان) ولا اربال الصلح وحرر بال  
رأسه يهضم بالعدل بين الزبارة وفي نقاطيل حيم نوصد بوزر حاصه نكلوا  
في سنة ١٢٨٤ هـ

(ج) صيغة الكسرة تكون في جملتها وفائتو لساكنات ولا ينام

الخطوات للحل :- (١) هين المشير الى كونك تريد كما حصل قيمة تكرار (٢) اختيار التوزيع

لواله الله وحده لا شريك له، قل عدد السطور التي تنمض في حطمان (الخطات) صفا حيا

$\frac{1}{1}, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5} \rightarrow$  harmonic series

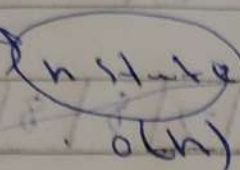
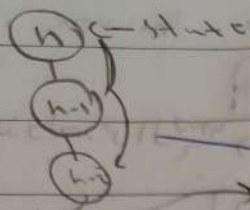
حقن شرط انكور:  $(h_1) \rightarrow (h_2)$  معرف  $(h_1)$  و  $(x)$  انكر من على حسب اليمين انكره  
البيان في صورة  $(h_1) \rightarrow (h_2)$  كبر الزمان قد

خطوات الرقم حسب قسما لحد بحد للواحد  
logarithm function

# Complexity Analysis [2/2]

من أجل تحليل التعقيد نحسب دوماً

time complexity و space complexity  
 recursive هو طريقة ال tree



لحساب تعقيد واحد

الوقت و المساحة ال relative هو عدداً states \* time complexity in one state

الصور time complexity تعاليم في حالة ال  $h$ !

$$10^3 = 1000$$

لوحظ ال recursive الواحد يتكرر على نفس مدخلين  $(2^h)$  complexity

كل عدد  $n$  يقل كل ما القيمة الجسدية زادت لنفس الرقم  $10^3 > 10^2$

لو كل مرة نكرر واحد  $(h-1)$  و  $h$  لو  $h$  عدد فعدد الخطوات تعتمد عليها  $\frac{h}{100}$

$$h \rightarrow \frac{h}{2} + \frac{h}{3} + \frac{h}{4} \rightarrow h(h) \rightarrow O(h \log h)$$

Function اكبر من الحقيقه فمثال عاين

$$h = (h-1) + (h-2) + (h-3) + \dots + 1 = \frac{h(h+1)}{2} = \frac{h^2 + h}{2} \rightarrow O(h^2)$$

نفسه عاين ال  $h$  time complexity بقا  $h$  ليس، السارة كدة هو ال كوررة

$max$  ال مقدار يستعمله قد اي  $O(1)$  constant

بما ان يستعمل ال  $T_0$  انما يتكرر على مستوى ال  $APIC$

متصور في ال  $max$  ال  $max$  ال  $max$  ال  $max$

روح يكون اكبر  $h$  بوجودة في ال  $max$  ال  $max$

نكون لو عدت  $10^8$  يكون طريقه عاين

