

INSERTION-SORT(A, n)

```

1 for  $i = 2$  to  $n$ 
2    $key = A[i]$ 
3   // Insert  $A[i]$  into the sorted subarray  $A[1:i-1]$ 
4    $j = i - 1$ 
5   while  $j > 0$  and  $A[j] > key$ 
6      $A[j+1] = A[j]$ 
7      $j = j - 1$ 
8    $A[j+1] = key$ 
    
```

cost	times
c_1	n
c_2	$n-1$
c_3	$n-1$
c_4	$n-1$
c_5	$\sum_{i=2}^n t_i$
c_6	$\sum_{i=2}^n (t_i - 1)$
c_7	$\sum_{i=2}^n (t_i - 1)$
c_8	$n-1$

Son n veces ya que se cuenta la última iteración para salir del bucle
 $\sum_{i=2}^n 1 = n-1 \rightarrow n+1$

t_i : Representa la cantidad de ejecuciones de la línea 5 para una cierta iteración i
 IMPORTANTE DEFINIR NUESTRAS VARIABLES

El tiempo de ejecución será esa sumatoria ponderada

Que tiene variables que no solamente dependen de n sino de i

Podemos sacar cual es el peor y mejor caso

1° t_i está acotado
 mínimo hacemos 1 ejecución (ya que tenemos el while)

$$1 \leq t_i \leq i$$

ya que el while retrocede como máximo $i-1$ hasta 0 (salto de 1 en 1)

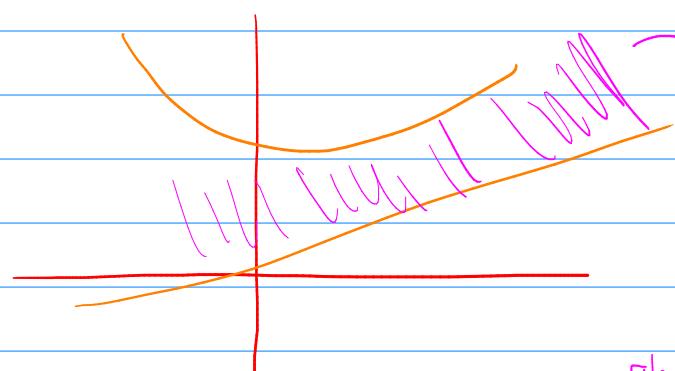
2° Sacando calculos podemos

concluir que tiempo de ejecución $T(n)$ está acotado superiormente por alguna expresión cuadrática y una expresión lineal

$$c \underset{\text{cte}}{\bigcirc} + n \underset{\text{cte}}{\bigcirc} \leq T(n) \leq \underset{\text{cte}}{\bigcirc} n^2 + \underset{\text{cte}}{\bigcirc} n + \underset{\text{cte}}{\bigcirc} c$$

Obs: A nosotros no nos interesa ser específicos con estas constantes porque nos queda claro como oscila la función T

T :



Oscila entre alguna cuadrática y lineal

Pero es muy largo escribir todo esto:

$$c \underset{\text{cte}}{\bigcirc} + n \underset{\text{cte}}{\bigcirc} \leq T(n) \leq \underset{\text{cte}}{\bigcirc} n^2 + \underset{\text{cte}}{\bigcirc} n + \underset{\text{cte}}{\bigcirc} c$$

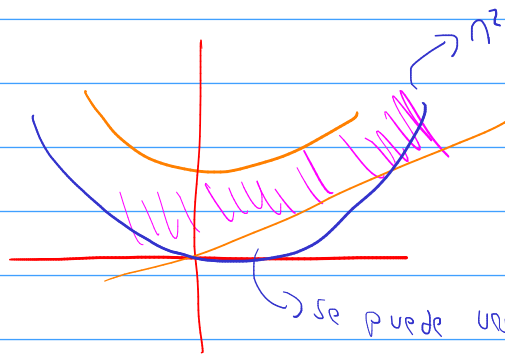
alguna notación que me diga que está acotado superiormente por una cuadrática

alguna notación que me diga que está acotado inferiormente por una lineal

Ahora sabemos que está acotado por esto:

$$T(n) \leq \underbrace{C_1}_{cte} n^2 + \underbrace{C_2}_{cte} n + \underbrace{C_3}_{cte}$$

Si quisiéramos reducirlo a n^2
y quisiéramos acotarlo a n^2
Hay un problema



Se puede ver que n^2 es menor a n en cierto intervalo lo cual NO CUMPLE

Ahora la solución es REESCALAR
Para que ahí sí esté acotada para un cierto n

Entonces lo que queremos es:

Decir que:

→ $T(n)$ está acotado por un reescalamiento de n^2 a partir de un cierto punto superiormente inicial
 $\Rightarrow T(n) = O(n^2)$ o sea $T(n)$ vive en ese conjunto n^2

La Notación a esta expresión es O-Notation (Big O)

$$O(g(n)) = \{ f(n) : \text{Existe una constante } C \text{ tal que}$$

$$0 \leq f(n) \leq C g(n) \quad \forall n \geq n_0 \}$$

En otras palabras, $O(g(n))$ son todas las funciones $f(n)$ que pueden ser acotadas superiormente por un Reescalar de cierta función $g(n)$ a partir de un cierto punto → puede ser cualquier escalar y valor inicial

Con tal que cumpla eso se dice que es Big O

traducción más reducida:

$$T(n) \leq C n^2 \quad \forall n \geq n_0 \quad C, n_0 > 0$$

$\in_j m:$

Probar que: $n^2 + 10n + 2 = O(n^2)$

Esto es lo mismo que decir que:

Parafraseo

Probar que: $\exists c, n_0 > 0$ tq $n > n_0$

escalar \uparrow punto

Probar que $n^2 + 10n + 2$ está acotado superiormente por un reescalamiento de n^2

$$\rightarrow n^2 + 10n + 2 \leq c n^2$$

Barridos:

Truco: separar los términos y acotar cada uno

$$n^2 \leq n^2$$

$$10n \leq n^2, \text{ para } \forall n \geq 10 \rightarrow n_0$$

$$2 \leq n^2, \forall n \geq \sqrt{2}$$

$$n^2 + 10n + 2 \leq 3n^2 \rightarrow c$$

Limpio:

Usaremos $C=3$ y $n_0=10$

Para probar que:

probaremos

$$\text{Como: } n \geq 10 \rightarrow \text{que } \underbrace{n^2 + 10n + 2}_{f(n)} \leq 3n^2$$

$$\begin{array}{l} + \\ \downarrow \\ \left(\begin{array}{l} n \geq 10 \\ n^2 \geq n^2 \quad \checkmark \\ n^2 \geq 10n \\ n^2 \geq 10n \geq 2 \end{array} \right. \end{array}$$

$$n^2 + 10n + 2 \leq 3n^2$$

$$\Rightarrow \forall n \geq 10 \rightarrow f(n) \leq 3n^2$$

$$\underline{f(n) = O(n^2)} \quad \square =$$

Ω -Notation

↳ Notación que me permite acotar inferiormente

Es decir

Función $T(n)$ está acotada inferiormente por un reescalamiento de $g(n)$
a partir de un instante

$$\Omega(g(n)) = \{ f(n), \text{ existe una constante } c, n_0 \geq 0 \text{ t.q.} \\ 0 \leq c g(n) \leq f(n) \forall n \geq n_0 \}$$

Hay una notación que acota inferior y superiormente por escalares de la misma función a partir de un punto inicial

↳ O sea:

$$\exists c_1, c_2, n_0 > 0 \text{ t.q. } \forall n \geq n_0 \Rightarrow c_1 g(n) \leq f(n) \leq c_2 g(n)$$

Esto se llama Big-theta \Rightarrow Big- Θ

Θ -Notation: Son todas las funciones que pueden ser acotadas tanto por arriba como por abajo de escalares de $g(n)$ a partir de un cierto punto

$$\Theta(g(n)) = \{ f(n), \text{ donde existen constantes } c_1, c_2, n_0 > 0 \text{ t.q.} \\ 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n); \forall n \geq n_0 \}$$

↳ Esto también es lo mismo a decir que:

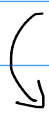
$$\Theta(g(n)) \equiv \underset{\substack{\uparrow \\ \text{acotada} \\ \text{superiormente} \\ \text{por un} \\ c_1 g(n)}}{O(g(n))} \wedge \underset{\substack{\uparrow \\ \text{acotada} \\ \text{inferiormente} \\ \text{por } c_2 g(n)}}{\Omega(g(n))}$$

misma $g(n)$

Otra forma de definir Θ es que cumpla O y Ω a la vez

$\in_j m_i$

$$\frac{1}{2}n^2 - 3n = \Theta(n^2)$$



$$\exists c_1, c_2, n_0 \text{ t.q. } \forall n \geq n_0 \xrightarrow[\text{que}]{\text{queremos}} c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2$$

→ En este tipo de problemas primero buscar la mas sencilla

1°

$$\frac{1}{2}n^2 - 3n \leq n^2 \quad \checkmark \quad \forall n \geq 1$$

$$\text{o sea } c_2 = 1$$

2°

$$c_1 n^2 \leq \left(\frac{1}{2}n^2 - 3n \right) \quad \text{torta}$$

TRUCCO: buscar el mas grande
sea mi torta y los demas
limitarlos a mitad de torta
ya que asi al quitarlos,
lo que tengas sea MAYOR
a lo limitado

$$3n \leq \frac{1}{2} \left(\frac{1}{2}n^2 \right)$$

$$n \leq \frac{1}{4}n^2 \rightarrow \forall n \geq 12$$

$$\frac{1}{2}n^2 - 3n \geq \frac{1}{2}n^2 - \frac{1}{2} \left(\frac{1}{2}n^2 \right) = \frac{1}{4}n^2$$

$$\frac{1}{2}n^2 - 3n \geq \left(\frac{1}{4}n^2 \right)$$

$$c_1$$

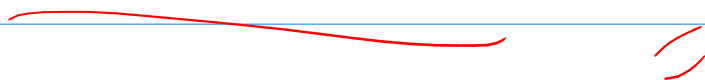
$$\text{Con } c_1 = \frac{1}{4}, c_2 = 1, n_0 = 12$$

o.o

$$\frac{1}{4}n^2 \leq \frac{1}{2}n^2 - 3n \leq 1n^2, \quad \forall n \geq 12$$

$$\text{o.o} \quad \frac{1}{2}n^2 - 3n = \Theta(n^2)$$

□



V y W , tamaño $n = 6$

$t = 0$

$i = 1$

$V \rightarrow$

5	6	7	8	9	10
---	---	---	---	---	----

while $i \leq n$

$t = t + 1$

$m = i + 1$

while ($m \leq n$ & $V[m] < W[i]$)

$m = m + 1$

$i = m$

$W \rightarrow$

11	12	13	14	15	16
----	----	----	----	----	----

$i = 1$

$t = 1$

$m = 2$

$2 \leq 6$