

ANÁLISIS Y DISEÑO DE ALGORITMOS I Periodo I – 2023

Jesús Aranda
jesus.aranda@correounivalle.edu.co

Universidad del Valle
Escuela de Ingeniería de Sistemas y Computación

Este documento es una adaptación del material original del profesor Oscar Bedoya



El i -ésimo orden estadístico

Selección en tiempo lineal esperado

Selección en tiempo lineal en el peor caso

Medianas y ordenes estadísticos

El i -ésimo orden estadístico

El i -ésimo orden estadístico de un conjunto de n elementos es el i -ésimo elemento más pequeño

Medianas y ordenes estadísticos

El i -ésimo orden estadístico

El i -ésimo orden estadístico de un conjunto de n elementos es el i -ésimo elemento más pequeño

¿Cuál es el primer orden estadístico?

¿Cuál es el n -ésimo orden estadístico?

Medianas y ordenes estadísticos

El i -ésimo orden estadístico

El i -ésimo orden estadístico de un conjunto de n elementos es el i -ésimo elemento más pequeño

Mínimo: primer orden estadístico de un conjunto

Máximo: n -ésimo orden estadístico de un conjunto de n elementos

Mediana: $(n+1)/2$ orden estadístico si n es impar
si n es par, se tienen dos medianas, $\lfloor (n+1)/2 \rfloor$ y $\lceil (n+1)/2 \rceil$

Medianas y ordenes estadísticos

El problema de encontrar el i -ésimo orden estadístico se conoce como el **problema de selección**

Problema de selección

Entrada: Un conjunto A de n números diferentes y un número i , $1 \leq i \leq n$

Salida: El elemento $x \in A$, tal que es mayor que $i-1$ elementos de A

Medianas y ordenes estadísticos

El problema de encontrar el i -ésimo orden estadístico se conoce como el **problema de selección**

Indique una solución al problema de selección. Incluya la estimación de complejidad de la solución dada

Medianas y ordenes estadísticos

El problema de encontrar el i -ésimo orden estadístico se conoce como el **problema de selección**

Ordenar los n números usando MergeSort o HeapSort ($O(n \lg n)$) y luego retornando como salida aquel en la posición i del arreglo ($O(1)$). La solución tendrá complejidad $O(n \lg n)$

Medianas y ordenes estadísticos

Selección en tiempo lineal esperado

En general, el problema de selección se puede realizar en tiempo $\Theta(n)$

Para solucionar el problema en tiempo lineal esperado, se utiliza la técnica de dividir y conquistar. Se utiliza el procedimiento RANDOMIZED-PARTITION de Quicksort

A diferencia del Quicksort no se procede sobre ambos lados de la partición, esto hace que se logre un tiempo de $\Theta(n)$ en lugar de $\Theta(n \lg n)$

Medianas y ordenes estadísticos

RANDOMIZED-SELECT(A, p, r, i)

if $p=r$

then return $A[p]$

$q \leftarrow \text{RANDOMIZED-PARTITION}(A, p, r)$

$k \leftarrow q - p + 1$

if $i \leq k$

then return **RANDOMIZED-SELECT**(A, p, q, i)

else return **RANDOMIZED-SELECT**($A, q+1, r, i-k$)

Medianas y ordenes estadísticos

RANDOMIZED-SELECT(A,p,r,i)

if $p=r$

then return $A[p]$

$q \leftarrow \text{RANDOMIZED-PARTITION}(A,p,r)$

$k \leftarrow q-p+1$

if $i \leq k$

then return **RANDOMIZED-SELECT**(A,p,q,i)

else return **RANDOMIZED-SELECT**(A,q+1,r,i-k)

RANDOMIZED-SELECT(A, 1, 8, 2)

4	5	1	9	8	10	3	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

RANDOMIZED-SELECT(A,p,r,i)

if $p=r$

then return $A[p]$

$q \leftarrow \text{RANDOMIZED-PARTITION}(A,p,r)$

$k \leftarrow q-p+1$

if $i \leq k$

then return **RANDOMIZED-SELECT**(A,p,q,i)

else return **RANDOMIZED-SELECT**(A,q+1,r,i-k)

PARTITION(A, 1, 8)

3	1	5	9	8	10	4	7
---	---	---	---	---	----	---	---

$q \leftarrow 2$

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q-p+1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

$K \leftarrow 2-1+1=2$

como $i=2 \leq k$

return RANDOMIZED-SELECT($A, 1, 2, 2$)

3	1	5	9	8	10	4	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT(A,p,r,i)
```

```
  if p=r
```

```
    then return A[p]
```

```
  q ← RANDOMIZED-PARTITION(A,p,r)
```

```
  k ← q-p+1
```

```
  if i ≤ k
```

```
    then return RANDOMIZED-SELECT(A,p,q,i)
```

```
    else return RANDOMIZED-SELECT(A,q+1,r,i-k)
```

$K \leftarrow 2 - 1 + 1 = 2$ \longrightarrow K indica la cantidad de elementos en la partición izquierda

como $i = 2 \leq k$

return RANDOMIZED-SELECT(A, 1, 2, 2)

3	1	5	9	8	10	4	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT(A,p,r,i)
```

```
  if p=r
```

```
    then return A[p]
```

```
  q ← RANDOMIZED-PARTITION(A,p,r)
```

```
  k ← q-p+1
```

```
  if i ≤ k
```

```
    then return RANDOMIZED-SELECT(A,p,q,i)
```

```
    else return RANDOMIZED-SELECT(A,q+1,r,i-k)
```

$K \leftarrow 2-1+1=2$

como $i=2 \leq k$ \longrightarrow Se evalua si el i-esimo orden estadístico está en la primera partición

return RANDOMIZED-SELECT(A, 1,2,2)

3	1	5	9	8	10	4	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

RANDOMIZED-SELECT(A,p,r,i)

if $p=r$

then return $A[p]$

$q \leftarrow \text{RANDOMIZED-PARTITION}(A,p,r)$

$k \leftarrow q-p+1$

if $i \leq k$

then return **RANDOMIZED-SELECT**(A,p,q,i)

else return **RANDOMIZED-SELECT**(A,q+1,r,i-k)

RANDOMIZED-SELECT(A, 1,2,2)

3	1	5	9	8	10	4	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q - p + 1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

PARTITION($A, 1, 2$)

1	3	5	9	8	10	4	7
---	---	---	---	---	----	---	---

$q \leftarrow 1$

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q - p + 1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

$K \leftarrow 1 - 1 + 1 = 1$

como $i=2 > k$

return RANDOMIZED-SELECT($A, 2, 2, 2$)

1	3	5	9	8	10	4	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q-p+1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

RANDOMIZED-SELECT($A, 2, 2, 2$) retorna el valor 3
que corresponde al segundo orden estadístico

1	3	5	9	8	10	4	7
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q - p + 1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

Utilice el algoritmo RANDOMIZED-SELECT para determinar el 3er y 5º orden estadístico del siguiente arreglo

7	6	8	5	9	10	2	3
---	---	---	---	---	----	---	---

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q - p + 1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

¿Cuál es la complejidad del algoritmo en el peor caso?

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q-p+1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

En el peor caso $T(n) = T(n-1) + \Theta(n)$, por lo que $T(n) = O(n^2)$

Medianas y ordenes estadísticos

```
RANDOMIZED-SELECT( $A, p, r, i$ )
```

```
  if  $p=r$ 
```

```
    then return  $A[p]$ 
```

```
   $q \leftarrow$  RANDOMIZED-PARTITION( $A, p, r$ )
```

```
   $k \leftarrow q - p + 1$ 
```

```
  if  $i \leq k$ 
```

```
    then return RANDOMIZED-SELECT( $A, p, q, i$ )
```

```
    else return RANDOMIZED-SELECT( $A, q+1, r, i-k$ )
```

En el caso promedio $T(n) = O(n)$

Aplicando análisis de esperanza utilizando variables aleatorias para los posibles subarreglos que se generan con las particiones

Medianas y ordenes estadísticos

Selección en tiempo lineal en el peor caso

Utiliza `PARTITION` modificado de tal forma que reciba como parámetro el valor sobre el cual se hará la partición

Medianas y ordenes estadísticos

Algoritmo:

1. Dividir los n elementos del arreglo en $\lfloor n/5 \rfloor$ grupos de 5 elementos y 1 grupo de $n \bmod 5$ elementos
2. Calcular la mediana de cada uno de los $\lfloor n/5 \rfloor$ grupos. (Ordenar cada grupo y tomar el de la mitad)
3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lfloor n/5 \rfloor$ medianas calculadas en 2
4. Dividir el arreglo de entrada en 2 subarreglos utilizando PARTITION modificado, para que el "pivote" sea x . La parte izquierda de la partición tendrá k elementos y la derecha $n-k$
5. Utilice el algoritmo recursivamente para calcular el i -ésimo orden estadístico de la parte izquierda de la partición si $i \leq k$ o el $(i-k)$ -ésimo de la parte derecha si $i > k$

Medianas y ordenes estadísticos

Algoritmo:

1. Dividir los n elementos del arreglo en $\lfloor n/5 \rfloor$ grupos de 5 elementos y 1 grupo de $n \bmod 5$ elementos
2. Calcular la mediana de cada uno de los $\lceil n/5 \rceil$ grupos. (Ordenar cada grupo y tomar el de la mitad)
3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lceil n/5 \rceil$ medianas calculadas en 2
4. Dividir el arreglo de entrada en 2 subarreglos utilizando PARTITION modificado, para que el "pivote" sea x . La parte izquierda de la partición tendrá k elementos y la derecha $n-k$
5. Utilice el algoritmo recursivamente para calcular el i -ésimo orden estadístico de la parte izquierda de la partición si $i \leq k$ o el $(i-k)$ -ésimo de la parte derecha si $i > k$

El algoritmo termina cuando se haga un llamado para seleccionar un orden estadístico sobre un conjunto que tiene un solo elemento

Medianas y ordenes estadísticos

Obtener el 2º orden estadístico en {3,1,2,6,9,10,15,4,5,8,7,13,12}

Medianas y ordenes estadísticos

Obtener el 2º orden estadístico en {3,1,2,6,9,10,15,4,5,8,7,13,12}

1. Dividir los n elementos del arreglo en $\lfloor n/5 \rfloor$ grupos de 5 elementos y 1 grupo de $n \bmod 5$ elementos

3

1

2

6

9

10

15

4

5

8

7

13

12

Medianas y ordenes estadísticos

2. Calcular la mediana de cada uno de los $\lceil n/5 \rceil$ grupos. (Ordenar cada grupo y tomar el de la mitad)

1

2

3

6

9

4

5

8

10

15

7

12

13

Medianas y ordenes estadísticos

2. Calcular la mediana de cada uno de los $\lceil n/5 \rceil$ grupos. (Ordenar cada grupo y tomar el de la mitad)

1

2

3

6

9

4

5

8

10

15

7

12

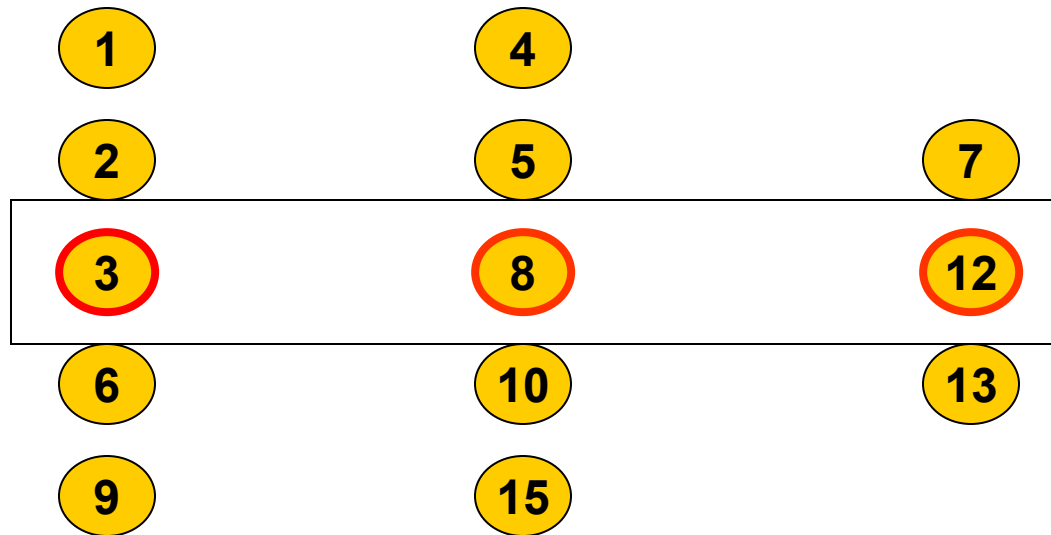
13

i

: i es una mediana

Medianas y ordenes estadísticos

3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lceil n/5 \rceil$ medianas calculadas en 2

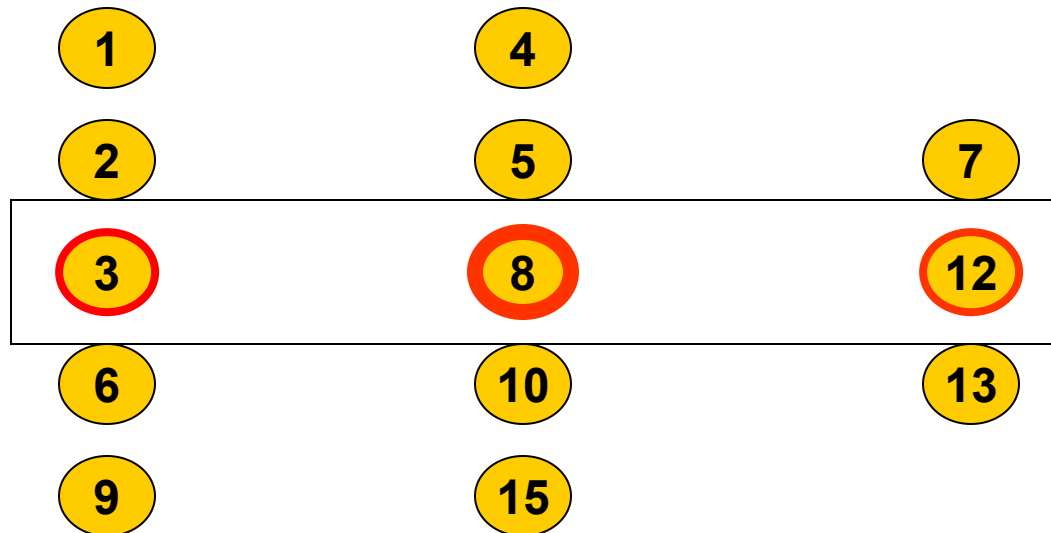


Se calcula la mediana de las medianas

i : i es una mediana

Medianas y ordenes estadísticos

3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lceil n/5 \rceil$ medianas calculadas en 2



Se calcula la mediana de las medianas

$x \leftarrow 8$

Medianas y ordenes estadísticos

4. Dividir el arreglo de entrada en 2 subarreglos utilizando PARTITION modificado, para que el "pivote" sea x . La parte izquierda de la partición tendrá k elementos y la derecha $n-k$

3	1	2	6	9	10	15	4	5	8	7	13	12
---	---	---	---	---	----	----	---	---	---	---	----	----

PARTITION usando $x=8$ como "pivote"

7	1	2	6	3	5	4	15	10	9	8	13	12
---	---	---	---	---	---	---	----	----	---	---	----	----

$k=7$

Medianas y ordenes estadísticos

5. Utilice el algoritmo recursivamente para calcular el i -ésimo orden estadístico de la parte izquierda de la partición si $i \leq k$ o el $(i-k)$ -ésimo de la parte derecha si $i > k$

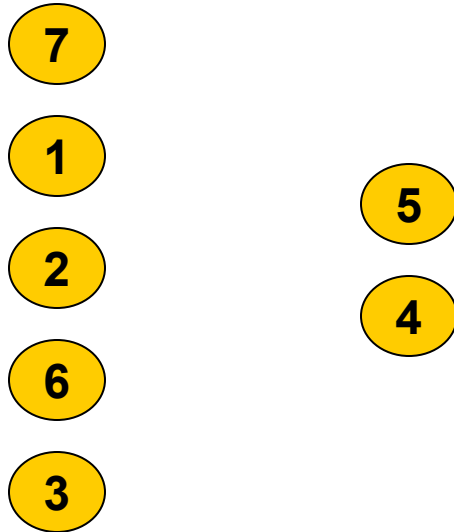
7	1	2	6	3	5	4	15	10	9	8	13	12
---	---	---	---	---	---	---	----	----	---	---	----	----

Obtener el 2º orden estadístico en {7,1,2,6,3,5,4}

Medianas y ordenes estadísticos

Obtener el 2º orden estadístico en $\{7,1,2,6,3,5,4\}$

1. Dividir los n elementos del arreglo en $\lfloor n/5 \rfloor$ grupos de 5 elementos y 1 grupo de $n \bmod 5$ elementos



Medianas y ordenes estadísticos

2. Calcular la mediana de cada uno de los $\lceil n/5 \rceil$ grupos. (Ordenar cada grupo y tomar el de la mitad)

1

2

3

6

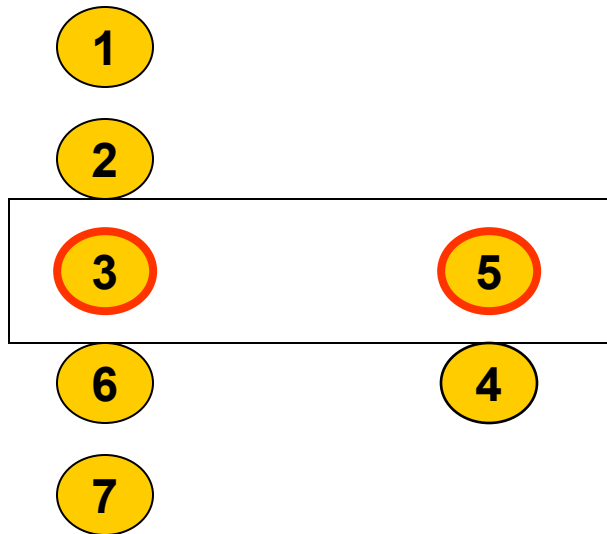
7

5

4

Medianas y ordenes estadísticos

3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lceil n/5 \rceil$ medianas calculadas en 2



Se calcula la
mediana de las
medianas

$x \leftarrow 5$

Medianas y ordenes estadísticos

4. Dividir el arreglo de entrada en 2 subarreglos utilizando PARTITION modificado, para que el "pivote" sea x . La parte izquierda de la partición tendrá k elementos y la derecha $n-k$

7	1	2	6	3	5	4
---	---	---	---	---	---	---

PARTITION usando $x=5$ como "pivote"

4	1	2	3	6	7	5
---	---	---	---	---	---	---

$k=4$

Medianas y ordenes estadísticos

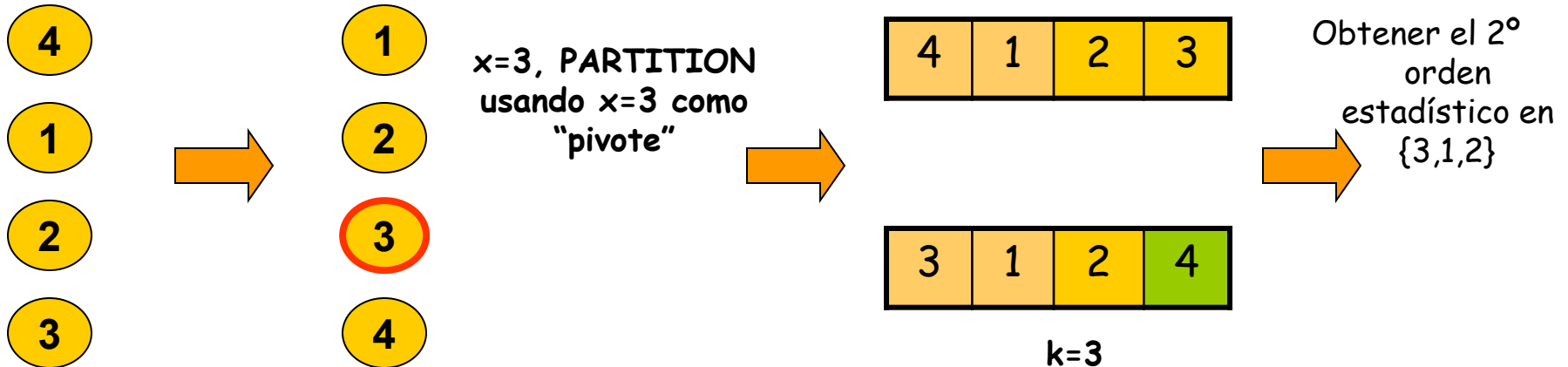
5. Utilice el algoritmo recursivamente para calcular el i -ésimo orden estadístico de la parte izquierda de la partición si $i \leq k$ o el $(i-k)$ -ésimo de la parte derecha si $i > k$

4	1	2	3	6	7	5
---	---	---	---	---	---	---

Obtener el 2º orden estadístico en $\{4,1,2,3\}$

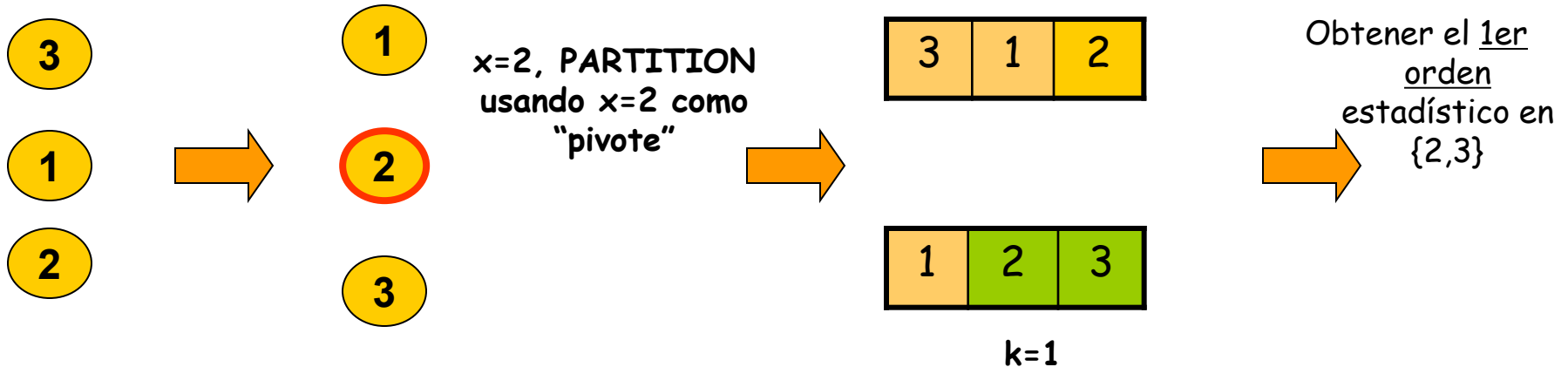
Medianas y ordenes estadísticos

Obtener el 2º orden estadístico en {4,1,2,3}



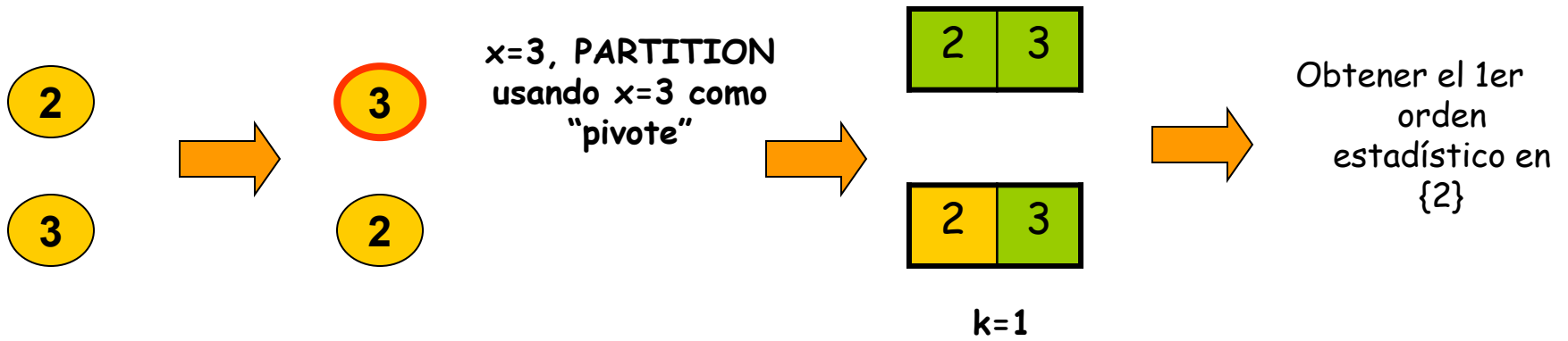
Medianas y ordenes estadísticos

Obtener el 2º orden estadístico en {3,1,2}



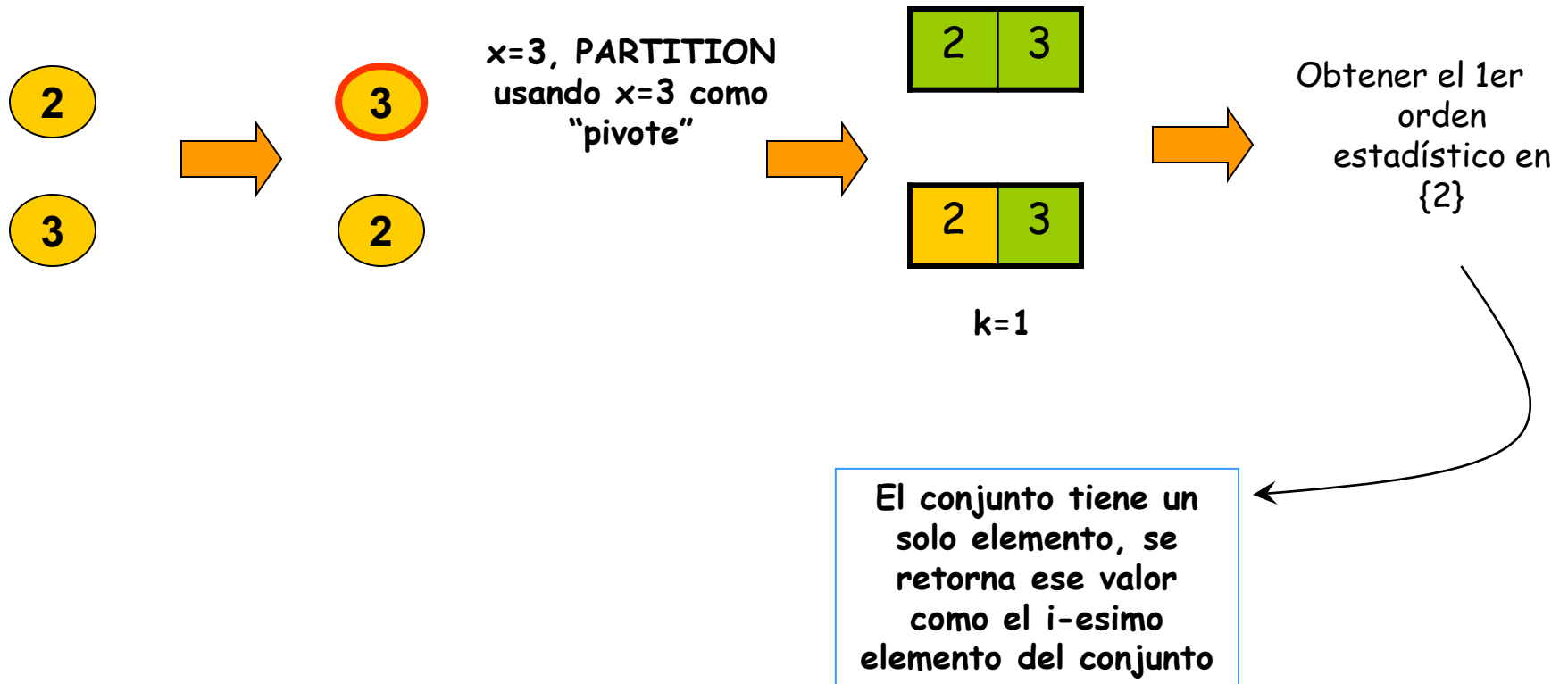
Medianas y ordenes estadísticos

Obtener el 1er orden estadístico en $\{2,3\}$



Medianas y ordenes estadísticos

Obtener el 1er orden estadístico en {2,3}



Medianas y ordenes estadísticos

Algoritmo:

1. Dividir los n elementos del arreglo en $\lfloor n/5 \rfloor$ grupos de 5 elementos y 1 grupo de $n \bmod 5$ elementos
2. Calcular la mediana de cada uno de los $\lceil n/5 \rceil$ grupos. (Ordenar cada grupo y tomar el de la mitad)
3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lceil n/5 \rceil$ medianas calculadas en 2
4. Dividir el arreglo de entrada en 2 subarreglos utilizando PARTITION modificado, para que el "pivote" sea x . La parte izquierda de la partición tendrá k elementos y la derecha $n-k$
5. Utilice el algoritmo recursivamente para calcular el i -ésimo orden estadístico de la parte izquierda de la partición si $i \leq k$ o el $(i-k)$ -ésimo de la parte derecha si $i > k$

El algoritmo termina cuando se haga un llamado para seleccionar un orden estadístico sobre un conjunto que tiene un solo elemento

Obtener el 5° orden estadístico en $\{4, 13, 1, 5, 11, 7, 2, 6, 9, 20, 21, 8, 3, 10\}$

Medianas y ordenes estadísticos

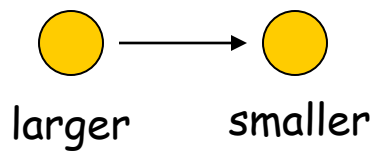
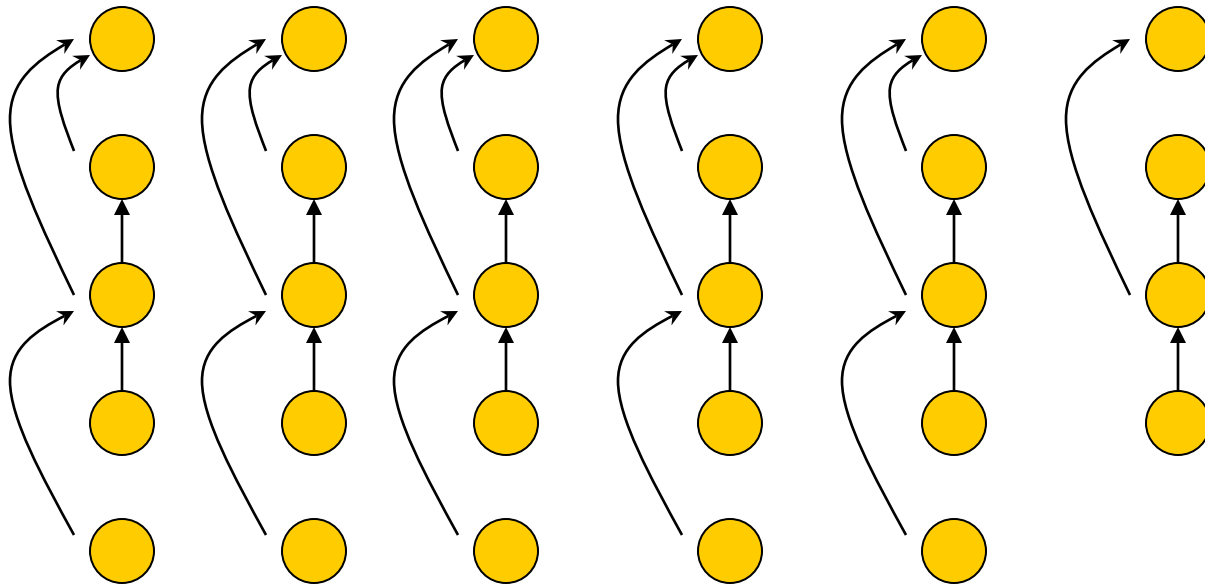
Algoritmo:

1. Dividir los n elementos del arreglo en $\lfloor n/5 \rfloor$ grupos de 5 elementos y 1 grupo de $n \bmod 5$ elementos
2. Calcular la mediana de cada uno de los $\lceil n/5 \rceil$ grupos. (Ordenar cada grupo y tomar el de la mitad)
3. Usar el algoritmo de selección recursivamente para calcular la mediana x , de las $\lceil n/5 \rceil$ medianas calculadas en 2
4. Dividir el arreglo de entrada en 2 subarreglos utilizando PARTITION modificado, para que el "pivote" sea x . La parte izquierda de la partición tendrá k elementos y la derecha $n-k$
5. Utilice el algoritmo recursivamente para calcular el i -ésimo orden estadístico de la parte izquierda de la partición si $i \leq k$ o el $(i-k)$ -ésimo de la parte derecha si $i > k$

Análisis de complejidad

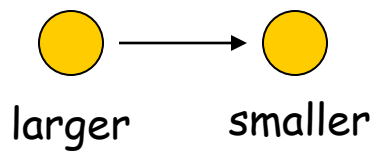
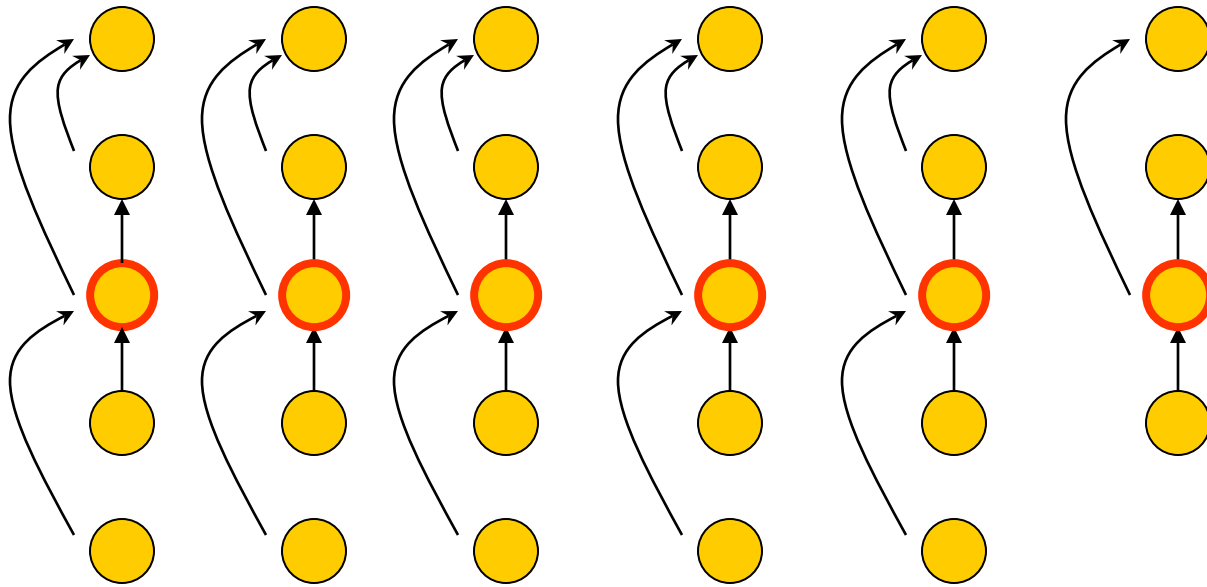
Medianas y ordenes estadísticos

Análisis de complejidad



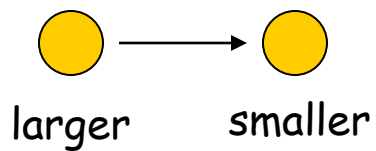
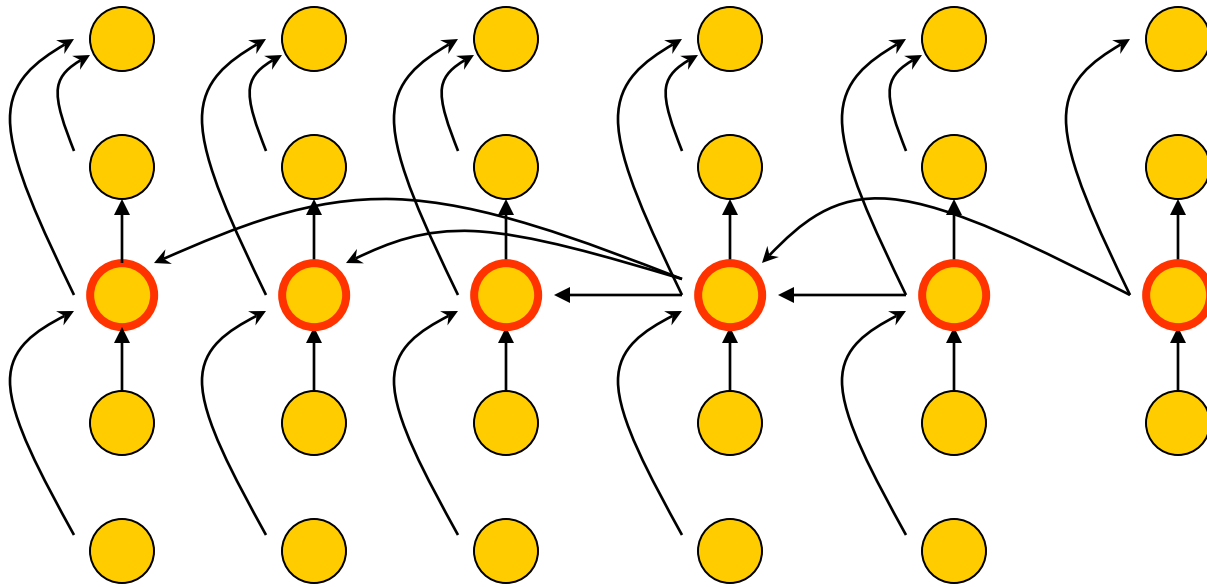
Medianas y ordenes estadísticos

Análisis de complejidad



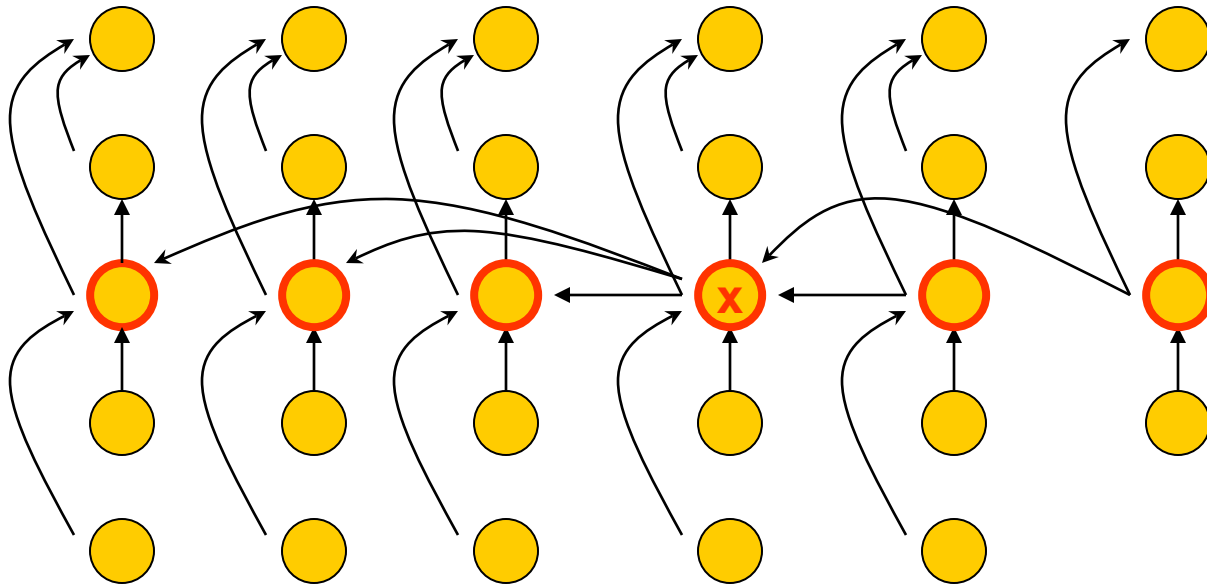
Medianas y ordenes estadísticos

Análisis de complejidad



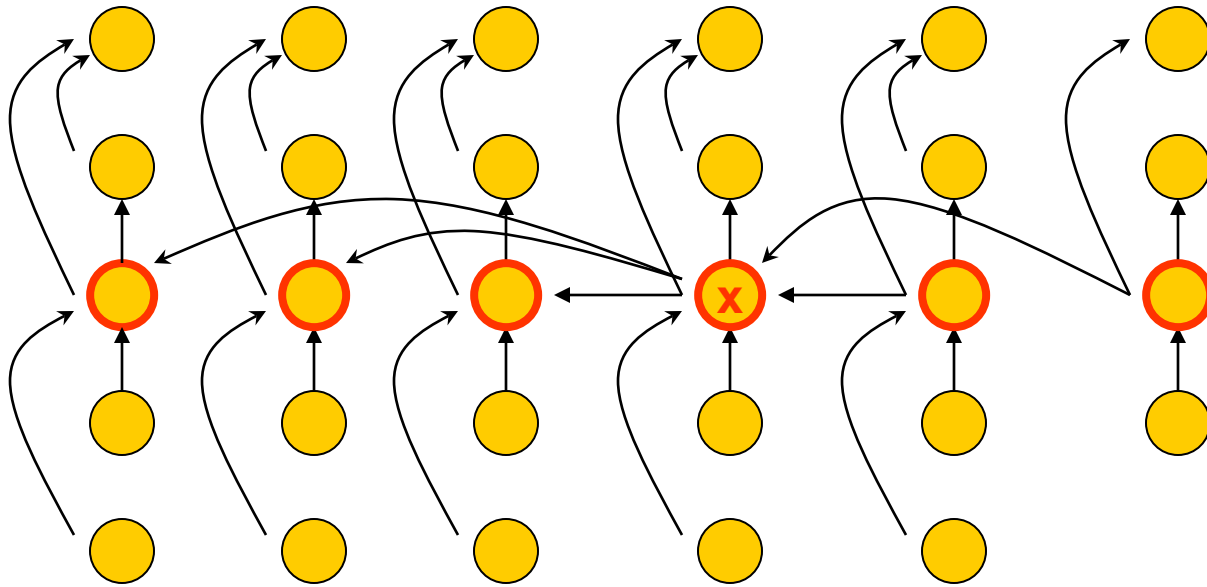
Medianas y ordenes estadísticos

Análisis de complejidad



Medianas y ordenes estadísticos

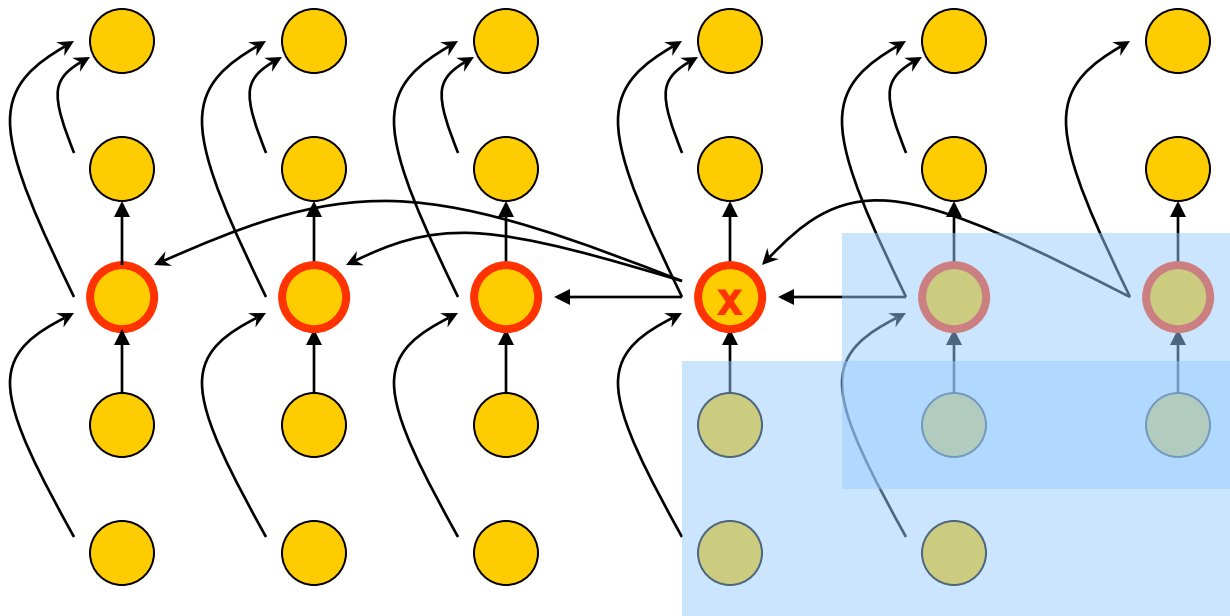
Análisis de complejidad



¿Qué valores son
mayores que x?

Medianas y ordenes estadísticos

Análisis de complejidad

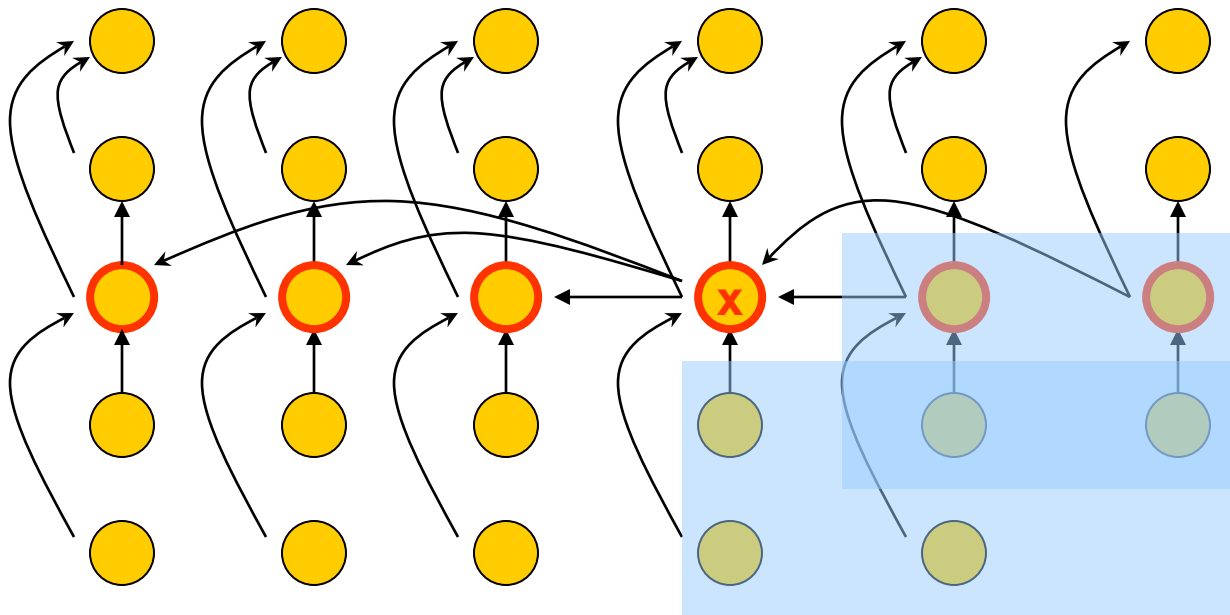


Al menos la mitad de las medianas son menores o iguales a x

Por lo tanto, al menos la mitad de los grupos contribuyen 3 elementos que son mayores que x , exceptuando el grupo donde está x y el último grupo que puede no estar completo

Medianas y ordenes estadísticos

Análisis de complejidad

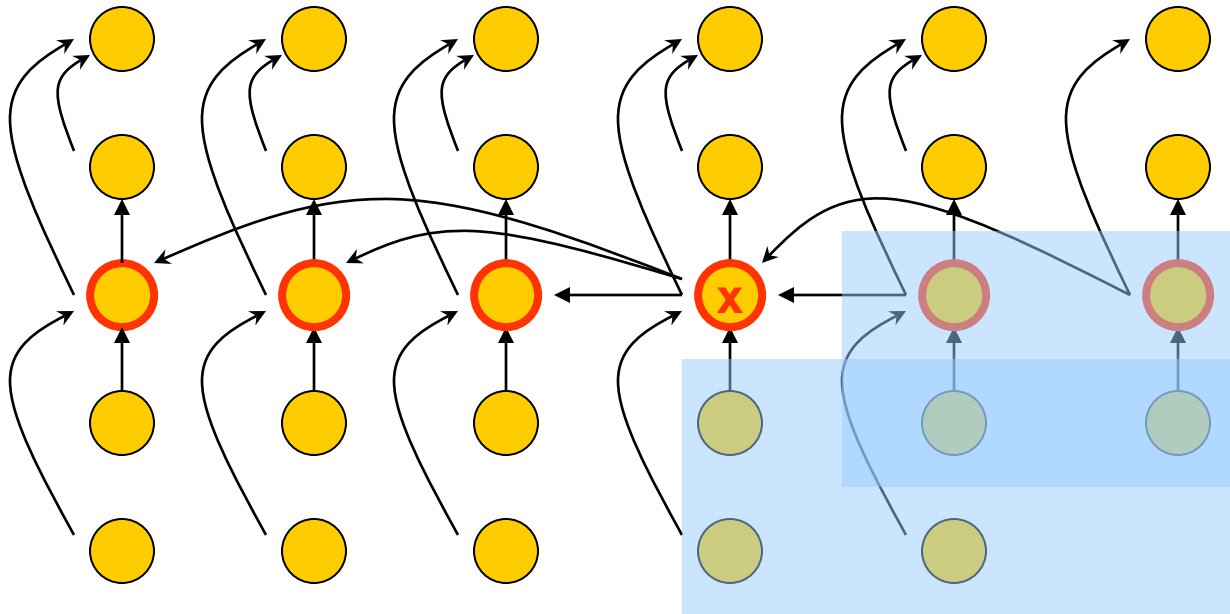


Se tiene entonces que el número de elementos mayores que x es:

$$3(\lceil 1/2 \rceil \lceil n/5 \rceil - 2) \geq 3n/10 - 6$$

Medianas y ordenes estadísticos

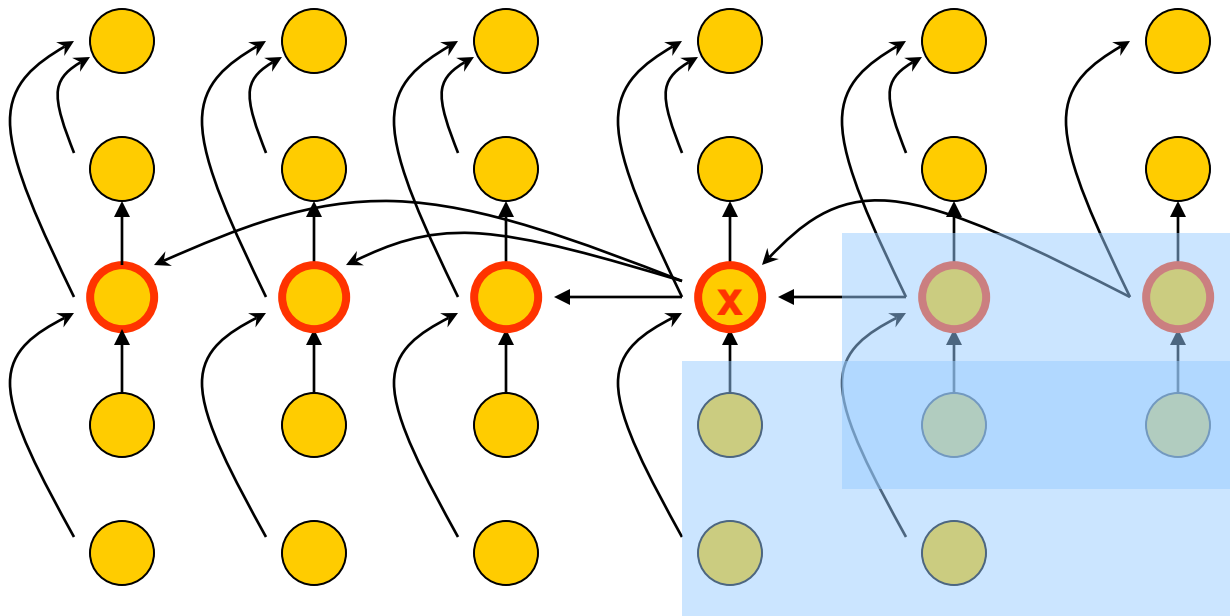
Análisis de complejidad



En el peor de los casos, el llamado recursivo se hace sobre a lo sumo $n - (3n/10 - 6) = 7n/10 + 6$ elementos

Medianas y ordenes estadísticos

Análisis de complejidad



$$T(n) = T(\lceil n/5 \rceil) + T(7n/10 + 6) + O(n)$$

↑
Calcular la mediana
de medianas

↑
Llamado recursivo

↑
PARTITION

Medianas y ordenes estadísticos

Análisis de complejidad

$T(n)=O(n)$, probar por sustitución

$$T(n) \leq c\lceil n/5 \rceil + c(7n/10+6) + O(n)$$

$$\leq cn/5 + 7cn/10 + 6c + O(n)$$

$$\leq 9cn/10 + 7c + O(n)$$

$$\leq cn$$