DIVIDIR & CONQUISTAR

Juan Mendivelso

Presentation made by Juan Mendivelso. Contents and figures extracted from the book: Introduction to Algorithms, Third Edition. Cormen, Leiserson, Rivests and Stein. The MIT Press. 2009.

DIVIDIR & CONQUISTAR

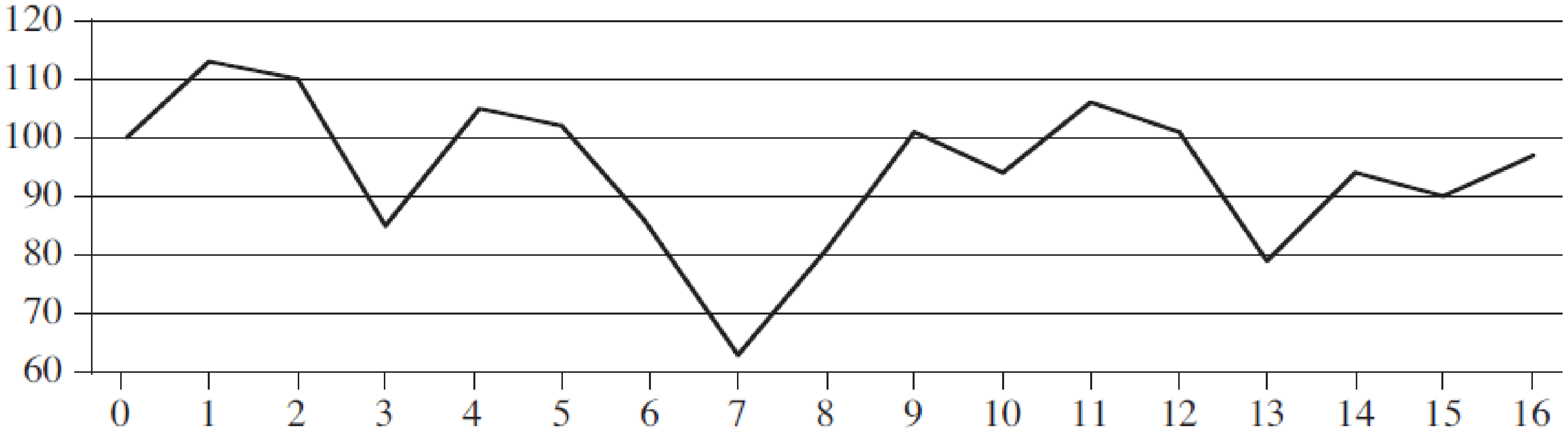
- Dividir: el problema en subproblemas.
- Conquistar: los subproblemas solucionándolos recursivamente.
- Combinar: la solución de los subproblemas para establecer una solución para el problema original.

CONTENIDO

1. Problema del subarreglo máximo
2. Multiplicación de Matrices

1. PROBLEMA DEL MÁXIMO SUBARREGLO

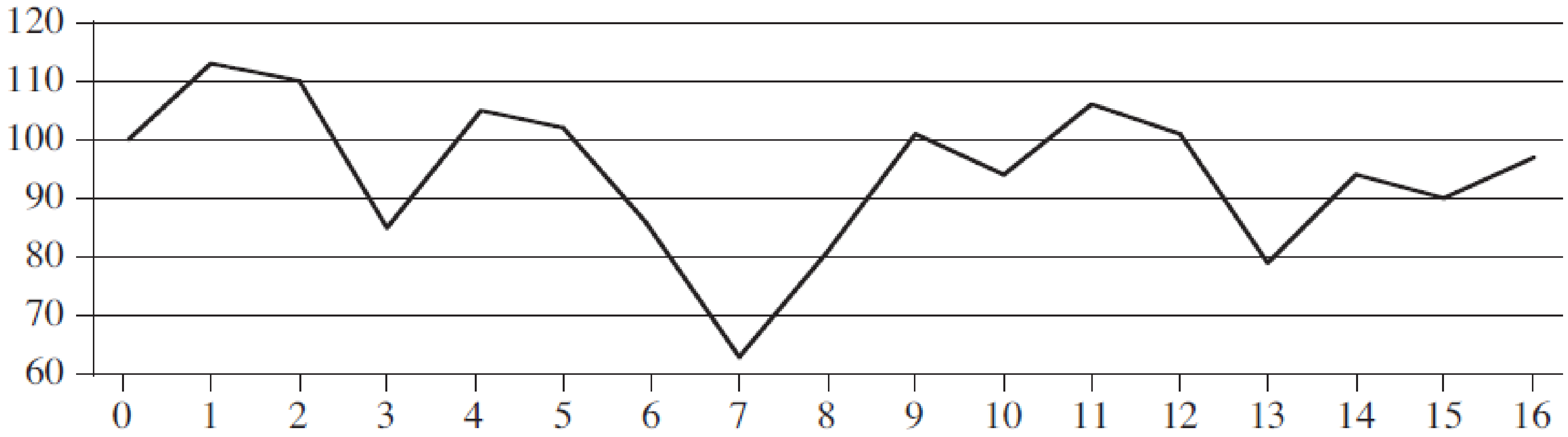
Problema de obtener la mayor ganancia en compra & venta de acciones



Day	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Price	100	113	110	85	105	102	86	63	81	101	94	106	101	79	94	90	97
Change		13	−3	−25	20	−3	−16	−23	18	20	−7	12	−5	−22	15	−4	7

Posibles soluciones

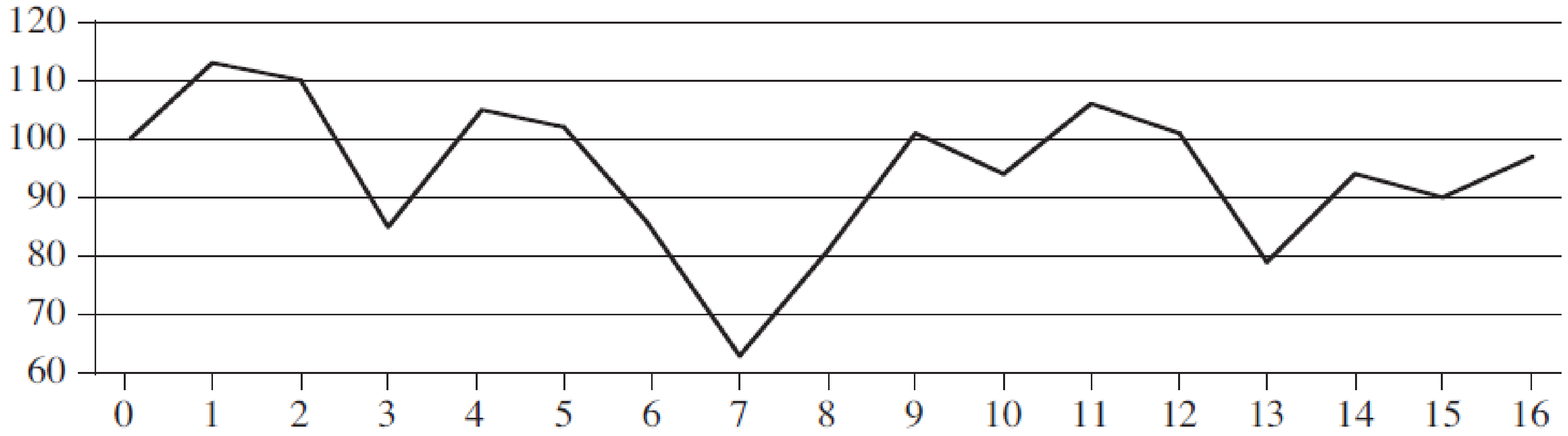
1. Comprar en el punto del precio más bajo y vender en el más alto.



Day	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Price	100	113	110	85	105	102	86	63	81	101	94	106	101	79	94	90	97
Change		13	−3	−25	20	−3	−16	−23	18	20	−7	12	−5	−22	15	−4	7

Posibles soluciones

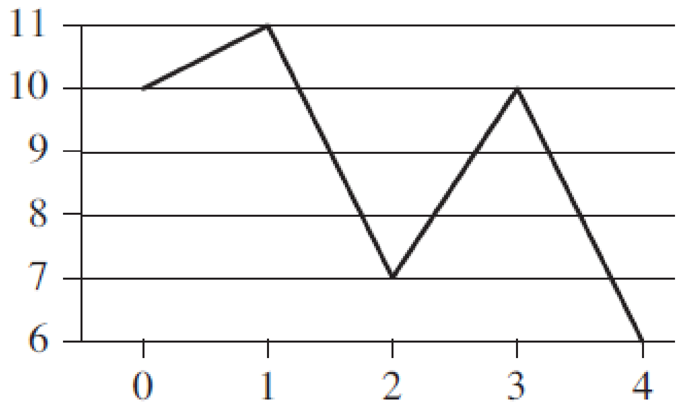
2. Escoger la opción que produzca mayor ganancia entre:
   - Tomar el valor máximo global y el valor mínimo local del periodo anterior a este.
   - Tomar el valor mínimo global y el valor máximo local del periodo posterior a este.



Day	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Price	100	113	110	85	105	102	86	63	81	101	94	106	101	79	94	90	97
Change		13	−3	−25	20	−3	−16	−23	18	20	−7	12	−5	−22	15	−4	7

Posibles soluciones

2. Escoger la opción que produzca mayor ganancia entre:
    - Tomar el valor máximo global y el valor mínimo local del periodo anterior a este.
    - Tomar el valor mínimo global y el valor máximo local del periodo posterior a este.



Day	0	1	2	3	4
Price	10	11	7	10	6
Change		1	–4	3	–4

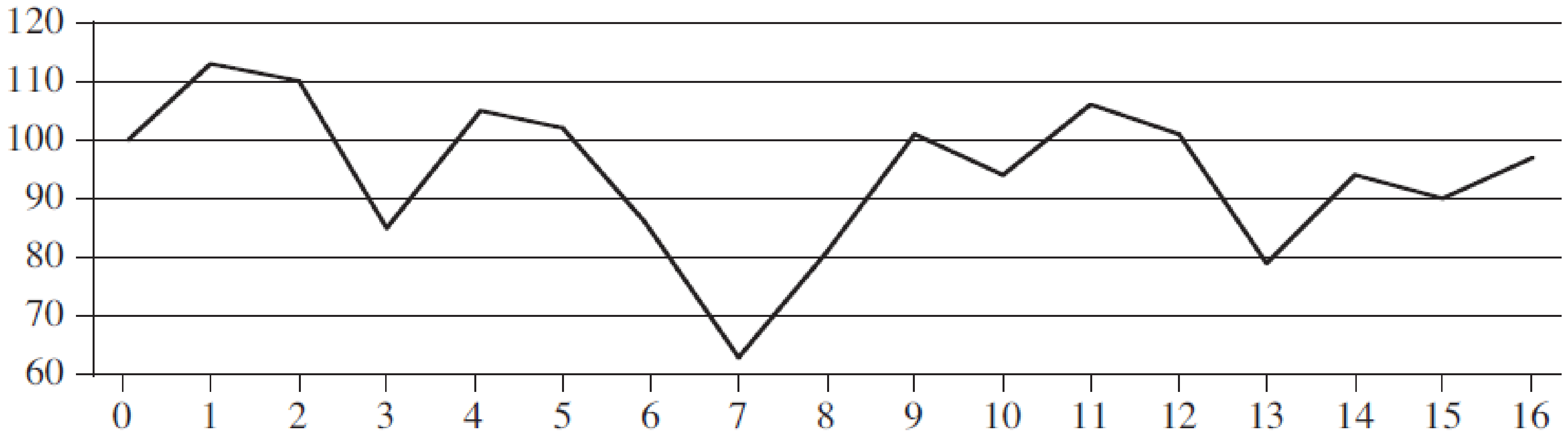
Posibles soluciones

3. Fuerza Bruta:

- Considerar todas las posibles parejas de fecha inicio y fecha de fin y para cada combinación hallar la ganancia.
- Si son n días, hay $\theta(n^2)$ parejas.
- Asumiendo que cada una se pueda determinar en tiempo constante, se requeriría $\Omega(n^2)$.
- ¿Es posible?

Transformación del problema

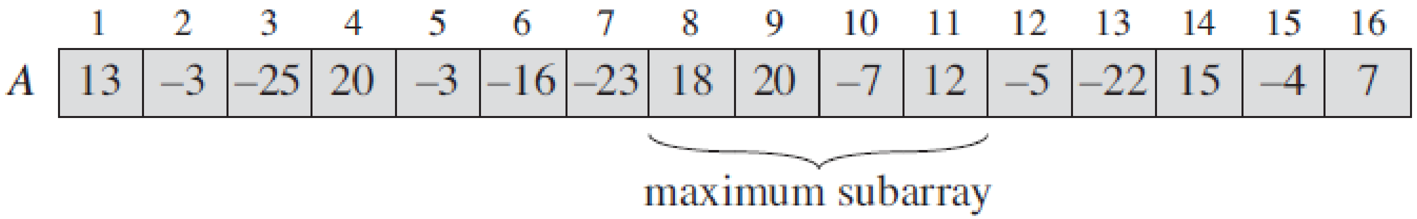
- Sea P[0..n] el arreglo de los precios de la acción de cada día.
- Sea A[1..n] el arreglo de los cambios del precio del día anterior al actual.
- El problema se convierte en encontrar <u>un</u> subarreglo con mayor suma (subarreglo máximo).



Day	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Price	100	113	110	85	105	102	86	63	81	101	94	106	101	79	94	90	97
Change		13	−3	−25	20	−3	−16	−23	18	20	−7	12	−5	−22	15	−4	7

Transformación del problema

- Computar la suma del subarreglo podría tomar la longitud de este.
- Pero se puede organizar el cómputo de la suma de cada subarreglo de manera que el costo de dicho cómputo sea constante.
- Esto basado en las sumas de otros subarreglos. ¿Cómo?
- Entonces, encontrar la suma de los $\theta(n^2)$ subarreglos tomaría $\theta(n^2)$.
- ¿Es necesario encontrar la suma de todos los subarreglos?



Posibles soluciones

4. Dividir y Conquistar:

- Usar la transformación del problema descrita anteriormente.
- No hallar la suma para todos los $\theta(n^2)$ subarreglos, hallar la suma de un subarreglo máximo con dividir y conquistar.

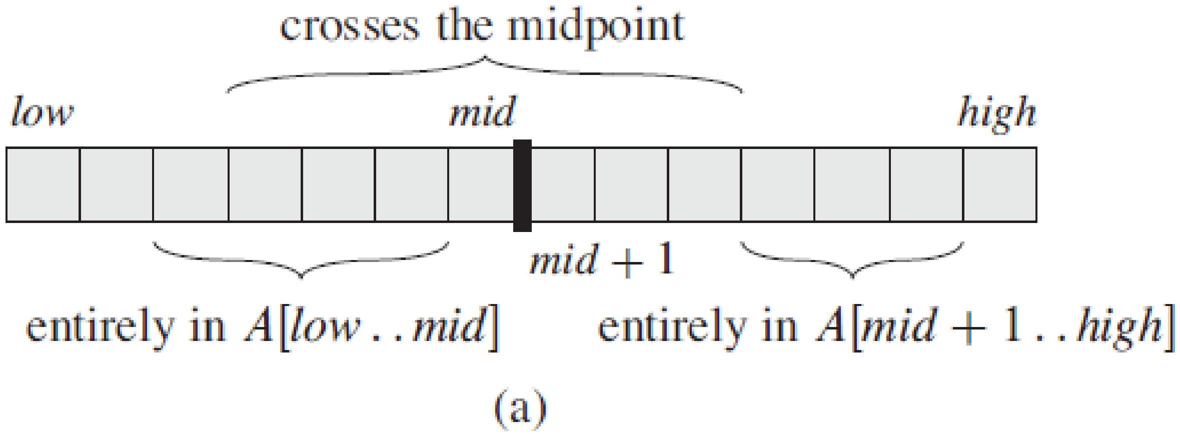
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A	13	–3	–25	20	–3	–16	–23	18	20	–7	12	–5	–22	15	–4	7

maximum subarray

Dividir & Conquistar

- **Dividir:** Dividir el arreglo en dos subarreglos.
- **Conquistar:** Para cada uno de estos subarreglos, encontrar recursivamente la suma de un subarreglo máximo de estos.
- **Combinar:** Determinar la solución del problema original escogiendo entre tres opciones:
  1. Solución del primer subproblema.
  2. Solución del segundo subproblema.
  3. Una solución que empiece en el primer subarreglo y termine en el segundo subarreglo.

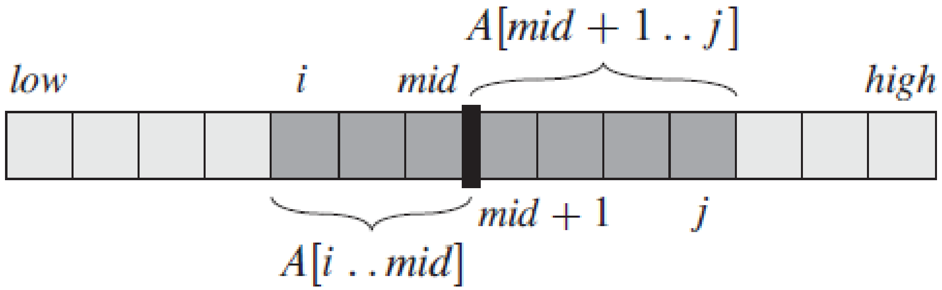
Dividir & Conquistar



Combinar

FIND-MAX-CROSSING-SUBARRAY(*A*, *low*, *mid*, *high*)

```
left-sum = -∞
sum = 0
for i = mid downto low
    sum = sum + A[i]
    if sum > left-sum
        left-sum = sum
        max-left = i
right-sum = -∞
sum = 0
for j = mid + 1 to high
    sum = sum + A[j]
    if sum > right-sum
        right-sum = sum
        max-right = j
return (max-left, max-right, left-sum + right-sum)
```



Algoritmo completo

```
FIND-MAXIMUM-SUBARRAY(A, low, high)
if high == low
    return (low, high, A[low])          // base case: only one element
else mid = ⌊(low + high)/2⌋
    (left-low, left-high, left-sum) =
        FIND-MAXIMUM-SUBARRAY(A, low, mid)
    (right-low, right-high, right-sum) =
        FIND-MAXIMUM-SUBARRAY(A, mid + 1, high)
    (cross-low, cross-high, cross-sum) =
        FIND-MAX-CROSSING-SUBARRAY(A, low, mid, high)
    if left-sum ≥ right-sum and left-sum ≥ cross-sum
        return (left-low, left-high, left-sum)
    elseif right-sum ≥ left-sum and right-sum ≥ cross-sum
        return (right-low, right-high, right-sum)
    else return (cross-low, cross-high, cross-sum)
```

Algoritmo completo - Recurrencia

```
FIND-MAXIMUM-SUBARRAY(A, low, high)
if high == low
    return (low, high, A[low])            // base case: only one element
else mid = ⌊(low + high)/2⌋
    (left-low, left-high, left-sum) =
        FIND-MAXIMUM-SUBARRAY(A, low, mid)
    (right-low, right-high, right-sum) =
        FIND-MAXIMUM-SUBARRAY(A, mid + 1, high)
    (cross-low, cross-high, cross-sum) =
        FIND-MAX-CROSSING-SUBARRAY(A, low, mid, high)
    if left-sum ≥ right-sum and left-sum ≥ cross-sum
        return (left-low, left-high, left-sum)
    elseif right-sum ≥ left-sum and right-sum ≥ cross-sum
        return (right-low, right-high, right-sum)
    else return (cross-low, cross-high, cross-sum)
```

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \,, \\ 2T(n/2) + \Theta(n) & \text{if } n > 1 \,. \end{cases}$$

¿Complejidad?

Algoritmo completo - Recurrencia

```
FIND-MAXIMUM-SUBARRAY(A, low, high)
if high == low
    return (low, high, A[low])                // base case: only one element
else mid = ⌊(low + high)/2⌋
    (left-low, left-high, left-sum) =
        FIND-MAXIMUM-SUBARRAY(A, low, mid)
    (right-low, right-high, right-sum) =
        FIND-MAXIMUM-SUBARRAY(A, mid + 1, high)
    (cross-low, cross-high, cross-sum) =
        FIND-MAX-CROSSING-SUBARRAY(A, low, mid, high)
    if left-sum ≥ right-sum and left-sum ≥ cross-sum
        return (left-low, left-high, left-sum)
    elseif right-sum ≥ left-sum and right-sum ≥ cross-sum
        return (right-low, right-high, right-sum)
    else return (cross-low, cross-high, cross-sum)
```

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 , \\ 2T(n/2) + \Theta(n) & \text{if } n > 1 . \end{cases}$$

Complejidad: $\Theta(n \lg n)$

2. MULTIPLICACIÓN DE MATRICES

Multiplicación de matrices

$$c_{ij} = \sum_{k=1}^{n} a_{ik} \cdot b_{kj}$$

$$\begin{pmatrix} -3 & 0 & 2 \\ -1 & 0 & 1 \\ 2 & 5 & -2 \end{pmatrix} \begin{pmatrix} 3 & 1 & 5 \\ 0 & -2 & 6 \\ 3 & -3 & 7 \end{pmatrix} =$$

$$= \begin{pmatrix} -9+0+6 & -3+0-6 & -15+0+14 \\ -3+0+3 & -1+0-3 & -5+0+7 \\ 6+0-6 & 2-10+6 & 10+30-14 \end{pmatrix} =$$

$$= \begin{pmatrix} -3 & -9 & -1 \\ 0 & -4 & 2 \\ 0 & -2 & 26 \end{pmatrix}$$

Multiplicación de matrices

$$c_{ij} = \sum_{k=1}^{n} a_{ik} \cdot b_{kj}$$

$$\begin{pmatrix} -3 & 0 & 2 \\ -1 & 0 & 1 \\ 2 & 5 & -2 \end{pmatrix}\begin{pmatrix} 3 & 1 & 5 \\ 0 & -2 & 6 \\ 3 & -3 & 7 \end{pmatrix} =$$

$$= \begin{pmatrix} -9+0+6 & -3+0-6 & -15+0+14 \\ -3+0+3 & -1+0-3 & -5+0+7 \\ 6+0-6 & 2-10+6 & 10+30-14 \end{pmatrix} =$$

$$= \begin{pmatrix} -3 & -9 & -1 \\ 0 & -4 & 2 \\ 0 & -2 & 26 \end{pmatrix}$$

Algoritmos

1. Näive
2. Dividir & Conquistar
3. Strassen

1. Näive

$$c_{ij} = \sum_{k=1}^{n} a_{ik} \cdot b_{kj}$$

SQUARE-MATRIX-MULTIPLY(A, B)

```
n = A.rows
let C be a new n × n matrix
for i = 1 to n
    for j = 1 to n
        c_ij = 0
        for k = 1 to n
            c_ij = c_ij + a_ik · b_kj
return C
```

$$\begin{pmatrix} -3 & 0 & 2 \\ -1 & 0 & 1 \\ 2 & 5 & -2 \end{pmatrix}\begin{pmatrix} 3 & 1 & 5 \\ 0 & -2 & 6 \\ 3 & -3 & 7 \end{pmatrix} =$$

$$= \begin{pmatrix} -9+0+6 & -3+0-6 & -15+0+14 \\ -3+0+3 & -1+0-3 & -5+0+7 \\ 6+0-6 & 2-10+6 & 10+30-14 \end{pmatrix} =$$

$$= \begin{pmatrix} -3 & -9 & -1 \\ 0 & -4 & 2 \\ 0 & -2 & 26 \end{pmatrix}$$

¿Complejidad?

1. Näive - Complejidad

$$c_{ij} = \sum_{k=1}^{n} a_{ik} \cdot b_{kj}$$

```
SQUARE-MATRIX-MULTIPLY(A, B)
n = A.rows
let C be a new n × n matrix
for i = 1 to n
    for j = 1 to n
        c_ij = 0
        for k = 1 to n
            c_ij = c_ij + a_ik · b_kj
return C
```

$\Theta(n^3)$

$$\begin{pmatrix} -3 & 0 & 2 \\ -1 & 0 & 1 \\ 2 & 5 & -2 \end{pmatrix} \begin{pmatrix} 3 & 1 & 5 \\ 0 & -2 & 6 \\ 3 & -3 & 7 \end{pmatrix} =$$

$$= \begin{pmatrix} -9+0+6 & -3+0-6 & -15+0+14 \\ -3+0+3 & -1+0-3 & -5+0+7 \\ 6+0-6 & 2-10+6 & 10+30-14 \end{pmatrix} =$$

$$= \begin{pmatrix} -3 & -9 & -1 \\ 0 & -4 & 2 \\ 0 & -2 & 26 \end{pmatrix}$$

2. Dividir & Conquistar

Problema: Hallar C = AB, donde A,B,C son matrices de n X n.

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{aligned} C_{11} &= A_{11} \cdot B_{11} + A_{12} \cdot B_{21} \\ C_{12} &= A_{11} \cdot B_{12} + A_{12} \cdot B_{22} \\ C_{21} &= A_{21} \cdot B_{11} + A_{22} \cdot B_{21} \\ C_{22} &= A_{21} \cdot B_{12} + A_{22} \cdot B_{22} \end{aligned}$$

2. Dividir & Conquistar- Recurrencia

SQUARE-MATRIX-MULTIPLY-RECURSIVE(A, B)

$n = A.rows$
let C be a new $n \times n$ matrix
if $n == 1$
    $c_{11} = a_{11} \cdot b_{11}$
else partition A, B, and C as in equations (4.9)
    $C_{11} =$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{11}, B_{11})
        $+$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{12}, B_{21})
    $C_{12} =$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{11}, B_{12})
        $+$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{12}, B_{22})
    $C_{21} =$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{21}, B_{11})
        $+$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{22}, B_{21})
    $C_{22} =$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{21}, B_{12})
        $+$ SQUARE-MATRIX-MULTIPLY-RECURSIVE(A_{22}, B_{22})
return C

$$
\begin{aligned}
C_{11} &= A_{11} \cdot B_{11} + A_{12} \cdot B_{21} \\
C_{12} &= A_{11} \cdot B_{12} + A_{12} \cdot B_{22} \\
C_{21} &= A_{21} \cdot B_{11} + A_{22} \cdot B_{21} \\
C_{22} &= A_{21} \cdot B_{12} + A_{22} \cdot B_{22}
\end{aligned}
$$

¿Recurrencia?

2. Dividir & Conquistar- Recurrencia

```
SQUARE-MATRIX-MULTIPLY-RECURSIVE(A, B)
  n = A.rows
  let C be a new n × n matrix
  if n == 1
      c11 = a11 · b11
  else partition A, B, and C as in equations (4.9)
      C11 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A11, B11)
            + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A12, B21)
      C12 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A11, B12)
            + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A12, B22)
      C21 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A21, B11)
            + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A22, B21)
      C22 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A21, B12)
            + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A22, B22)
  return C
```

$$
\begin{aligned}
C_{11} &= A_{11} \cdot B_{11} + A_{12} \cdot B_{21} \\
C_{12} &= A_{11} \cdot B_{12} + A_{12} \cdot B_{22} \\
C_{21} &= A_{21} \cdot B_{11} + A_{22} \cdot B_{21} \\
C_{22} &= A_{21} \cdot B_{12} + A_{22} \cdot B_{22}
\end{aligned}
$$

$$
T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ 8T(n/2) + \Theta(n^2) & \text{if } n > 1 \end{cases}
$$

¿Complejidad?

2. Dividir & Conquistar- Recurrencia

```
SQUARE-MATRIX-MULTIPLY-RECURSIVE(A, B)
n = A.rows
let C be a new n × n matrix
if n == 1
    c11 = a11 · b11
else partition A, B, and C as in equations (4.9)
    C11 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A11, B11)
          + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A12, B21)
    C12 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A11, B12)
          + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A12, B22)
    C21 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A21, B11)
          + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A22, B21)
    C22 = SQUARE-MATRIX-MULTIPLY-RECURSIVE(A21, B12)
          + SQUARE-MATRIX-MULTIPLY-RECURSIVE(A22, B22)
return C
```

$$
\begin{aligned}
C_{11} &= A_{11} \cdot B_{11} + A_{12} \cdot B_{21} \\
C_{12} &= A_{11} \cdot B_{12} + A_{12} \cdot B_{22} \\
C_{21} &= A_{21} \cdot B_{11} + A_{22} \cdot B_{21} \\
C_{22} &= A_{21} \cdot B_{12} + A_{22} \cdot B_{22}
\end{aligned}
$$

$$
T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ 8T(n/2) + \Theta(n^2) & \text{if } n > 1 \end{cases}
$$

$\Theta(n^3)$

3. Método de Strassen

Similar, pero evita una multiplicación aumentando un número constante de sumas.

1. Divide A y B de la misma manera. $A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$
2. Calcula 10 matrices $S_1,\ldots,S_{10}$ de n/2 X n/2 cada una, a través de sumas.
3. Calcula 7 matrices $P_1,\ldots, P_7$ de n/2 X n/2 cada una, a través de productos.
4. Calcula la solución a través de C_{11}, C_{12}, C_{21}, C_{22} de n/2 X n/2 cada una, a través de sumas y restas de las P_i.

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

Detalles del algoritmo

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \qquad \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{aligned}
S_1 &= B_{12} - B_{22} , \\
S_2 &= A_{11} + A_{12} , \\
S_3 &= A_{21} + A_{22} , \\
S_4 &= B_{21} - B_{11} , \\
S_5 &= A_{11} + A_{22} , \\
S_6 &= B_{11} + B_{22} , \\
S_7 &= A_{12} - A_{22} , \\
S_8 &= B_{21} + B_{22} , \\
S_9 &= A_{11} - A_{21} , \\
S_{10} &= B_{11} + B_{12} .
\end{aligned}$$

$$\begin{aligned}
P_1 &= A_{11} \cdot S_1 \\
P_2 &= S_2 \cdot B_{22} \\
P_3 &= S_3 \cdot B_{11} \\
P_4 &= A_{22} \cdot S_4 \\
P_5 &= S_5 \cdot S_6 \\
P_6 &= S_7 \cdot S_8 \\
P_7 &= S_9 \cdot S_{10}
\end{aligned}$$

$$C_{11} = P_5 + P_4 - P_2 + P_6$$

$$C_{12} = P_1 + P_2$$

$$C_{21} = P_3 + P_4$$

$$C_{22} = P_5 + P_1 - P_3 - P_7$$

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \qquad \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{aligned}
S_1 &= B_{12} - B_{22} , \\
S_2 &= A_{11} + A_{12} , \\
S_3 &= A_{21} + A_{22} , \\
S_4 &= B_{21} - B_{11} , \\
S_5 &= A_{11} + A_{22} , \\
S_6 &= B_{11} + B_{22} , \\
S_7 &= A_{12} - A_{22} , \\
S_8 &= B_{21} + B_{22} , \\
S_9 &= A_{11} - A_{21} , \\
S_{10} &= B_{11} + B_{12} .
\end{aligned}$$

$$\begin{aligned}
P_1 &= A_{11} \cdot S_1 &&= A_{11} \cdot B_{12} - A_{11} \cdot B_{22} , \\
P_2 &= S_2 \cdot B_{22} &&= A_{11} \cdot B_{22} + A_{12} \cdot B_{22} , \\
P_3 &= S_3 \cdot B_{11} &&= A_{21} \cdot B_{11} + A_{22} \cdot B_{11} , \\
P_4 &= A_{22} \cdot S_4 &&= A_{22} \cdot B_{21} - A_{22} \cdot B_{11} , \\
P_5 &= S_5 \cdot S_6 &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{22} + A_{22} \cdot B_{11} + A_{22} \cdot B_{22} , \\
P_6 &= S_7 \cdot S_8 &&= A_{12} \cdot B_{21} + A_{12} \cdot B_{22} - A_{22} \cdot B_{21} - A_{22} \cdot B_{22} , \\
P_7 &= S_9 \cdot S_{10} &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{12} - A_{21} \cdot B_{11} - A_{21} \cdot B_{12} .
\end{aligned}$$

$$C_{11} = P_5 + P_4 - P_2 + P_6$$

$$\begin{array}{l}
A_{11} \cdot B_{11} + A_{11} \cdot B_{22} + A_{22} \cdot B_{11} + A_{22} \cdot B_{22} \\
\qquad\qquad\qquad\qquad - A_{22} \cdot B_{11} \qquad\qquad\qquad + A_{22} \cdot B_{21} \\
\qquad\quad - A_{11} \cdot B_{22} \qquad\qquad\qquad\qquad\qquad\qquad\qquad - A_{12} \cdot B_{22} \\
\qquad\qquad\qquad\qquad\qquad - A_{22} \cdot B_{22} - A_{22} \cdot B_{21} + A_{12} \cdot B_{22} + A_{12} \cdot B_{21} \\
\hline
A_{11} \cdot B_{11} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad + A_{12} \cdot B_{21} ,
\end{array}$$

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \qquad \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{aligned}
S_1 &= B_{12} - B_{22} , \\
S_2 &= A_{11} + A_{12} , \\
S_3 &= A_{21} + A_{22} , \\
S_4 &= B_{21} - B_{11} , \\
S_5 &= A_{11} + A_{22} , \\
S_6 &= B_{11} + B_{22} , \\
S_7 &= A_{12} - A_{22} , \\
S_8 &= B_{21} + B_{22} , \\
S_9 &= A_{11} - A_{21} , \\
S_{10} &= B_{11} + B_{12} .
\end{aligned}$$

$$\begin{aligned}
P_1 &= A_{11} \cdot S_1 &&= A_{11} \cdot B_{12} - A_{11} \cdot B_{22} , \\
P_2 &= S_2 \cdot B_{22} &&= A_{11} \cdot B_{22} + A_{12} \cdot B_{22} , \\
P_3 &= S_3 \cdot B_{11} &&= A_{21} \cdot B_{11} + A_{22} \cdot B_{11} , \\
P_4 &= A_{22} \cdot S_4 &&= A_{22} \cdot B_{21} - A_{22} \cdot B_{11} , \\
P_5 &= S_5 \cdot S_6 &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{22} + A_{22} \cdot B_{11} + A_{22} \cdot B_{22} , \\
P_6 &= S_7 \cdot S_8 &&= A_{12} \cdot B_{21} + A_{12} \cdot B_{22} - A_{22} \cdot B_{21} - A_{22} \cdot B_{22} , \\
P_7 &= S_9 \cdot S_{10} &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{12} - A_{21} \cdot B_{11} - A_{21} \cdot B_{12} .
\end{aligned}$$

$$C_{12} = P_1 + P_2$$

$$\begin{array}{lr}
A_{11} \cdot B_{12} - A_{11} \cdot B_{22} & \\
\quad + A_{11} \cdot B_{22} & + A_{12} \cdot B_{22} \\
\hline
A_{11} \cdot B_{12} & + A_{12} \cdot B_{22} ,
\end{array}$$

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{aligned}
S_1 &= B_{12} - B_{22} , \\
S_2 &= A_{11} + A_{12} , \\
S_3 &= A_{21} + A_{22} , \\
S_4 &= B_{21} - B_{11} , \\
S_5 &= A_{11} + A_{22} , \\
S_6 &= B_{11} + B_{22} , \\
S_7 &= A_{12} - A_{22} , \\
S_8 &= B_{21} + B_{22} , \\
S_9 &= A_{11} - A_{21} , \\
S_{10} &= B_{11} + B_{12} .
\end{aligned}$$

$$\begin{aligned}
P_1 &= A_{11} \cdot S_1 &&= A_{11} \cdot B_{12} - A_{11} \cdot B_{22} , \\
P_2 &= S_2 \cdot B_{22} &&= A_{11} \cdot B_{22} + A_{12} \cdot B_{22} , \\
P_3 &= S_3 \cdot B_{11} &&= A_{21} \cdot B_{11} + A_{22} \cdot B_{11} , \\
P_4 &= A_{22} \cdot S_4 &&= A_{22} \cdot B_{21} - A_{22} \cdot B_{11} , \\
P_5 &= S_5 \cdot S_6 &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{22} + A_{22} \cdot B_{11} + A_{22} \cdot B_{22} , \\
P_6 &= S_7 \cdot S_8 &&= A_{12} \cdot B_{21} + A_{12} \cdot B_{22} - A_{22} \cdot B_{21} - A_{22} \cdot B_{22} , \\
P_7 &= S_9 \cdot S_{10} &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{12} - A_{21} \cdot B_{11} - A_{21} \cdot B_{12} .
\end{aligned}$$

$$C_{21} = P_3 + P_4$$

$$\begin{array}{l}
A_{21} \cdot B_{11} + A_{22} \cdot B_{11} \\
\qquad\qquad - A_{22} \cdot B_{11} + A_{22} \cdot B_{21} \\
\hline
A_{21} \cdot B_{11} \qquad\qquad\quad + A_{22} \cdot B_{21} ,
\end{array}$$

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \qquad \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

$$\begin{aligned}
S_1 &= B_{12} - B_{22}, \\
S_2 &= A_{11} + A_{12}, \\
S_3 &= A_{21} + A_{22}, \\
S_4 &= B_{21} - B_{11}, \\
S_5 &= A_{11} + A_{22}, \\
S_6 &= B_{11} + B_{22}, \\
S_7 &= A_{12} - A_{22}, \\
S_8 &= B_{21} + B_{22}, \\
S_9 &= A_{11} - A_{21}, \\
S_{10} &= B_{11} + B_{12}.
\end{aligned}$$

$$\begin{aligned}
P_1 &= A_{11} \cdot S_1 &&= A_{11} \cdot B_{12} - A_{11} \cdot B_{22}, \\
P_2 &= S_2 \cdot B_{22} &&= A_{11} \cdot B_{22} + A_{12} \cdot B_{22}, \\
P_3 &= S_3 \cdot B_{11} &&= A_{21} \cdot B_{11} + A_{22} \cdot B_{11}, \\
P_4 &= A_{22} \cdot S_4 &&= A_{22} \cdot B_{21} - A_{22} \cdot B_{11}, \\
P_5 &= S_5 \cdot S_6 &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{22} + A_{22} \cdot B_{11} + A_{22} \cdot B_{22}, \\
P_6 &= S_7 \cdot S_8 &&= A_{12} \cdot B_{21} + A_{12} \cdot B_{22} - A_{22} \cdot B_{21} - A_{22} \cdot B_{22}, \\
P_7 &= S_9 \cdot S_{10} &&= A_{11} \cdot B_{11} + A_{11} \cdot B_{12} - A_{21} \cdot B_{11} - A_{21} \cdot B_{12}.
\end{aligned}$$

$$C_{22} = P_5 + P_1 - P_3 - P_7$$

$$\begin{array}{llllllll}
& A_{11} \cdot B_{11} + A_{11} \cdot B_{22} + A_{22} \cdot B_{11} + A_{22} \cdot B_{22} \\
& \qquad - A_{11} \cdot B_{22} & & & + A_{11} \cdot B_{12} \\
& \qquad\qquad - A_{22} \cdot B_{11} & & & \qquad - A_{21} \cdot B_{11} \\
- A_{11} \cdot B_{11} & & & & - A_{11} \cdot B_{12} + A_{21} \cdot B_{11} + A_{21} \cdot B_{12} \\
\hline
& \qquad\qquad\qquad A_{22} \cdot B_{22} & & & \qquad\qquad + A_{21} \cdot B_{12},
\end{array}$$

3. Método de strassen - Análisis

Similar, pero evita una multiplicación aumentando un número constante de sumas.

1. Divide A y B de la misma manera. **θ(1)**

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

2. Calcula 10 matrices $S_1,\ldots,S_{10}$ de n/2 X n/2 cada una, a través de sumas. **θ(n^2)**
3. Calcula 7 matrices $P_1,\ldots, P_7$ de n/2 X n/2 cada una, a través de productos. **7T(n/2)**
4. Calcula la solución a través de C_{11}, C_{12}, C_{21}, C_{22} de n/2 X n/2 cada una, a través de sumas y restas de las P_i. **θ(n^2)**

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

3. Método de strassen - Recurrencia

Similar, pero evita una multiplicación aumentando un número constante de sumas.

1. Divide A y B de la misma manera. **θ(1)**

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

2. Calcula 10 matrices $S_1,\dots,S_{10}$ de n/2 s. **θ(n^2)**
3. Calcula 7 matrices $P_1,\dots, P_7$ de n/2 X n/2 cada una, a través de productos. **7T(n/2)**
4. Calcula la solución a través de C_{11}, C_{12}, C_{21}, C_{22} de n/2 X n/2 cada una, a través de sumas y restas de las P_i. **θ(n^2)**

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1\,, \\ 7T(n/2) + \Theta(n^2) & \text{if } n > 1\,. \end{cases}$$

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

3. Método de Strassen - Análisis

- Recurrencia

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 , \\ 7T(n/2) + \Theta(n^2) & \text{if } n > 1 . \end{cases}$$

- ¿Complejidad?

3. Método de Strassen - Análisis

- Recurrencia

$$T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 , \\ 7T(n/2) + \Theta(n^2) & \text{if } n > 1 . \end{cases}$$

- Complejidad: $\Theta(n^{\lg 7}) \approx \Theta(n^{2.80})$

BIBLOGRAFÍA

- Cormen TH, Leiserson CE, Rivest RL, Stein, C. Introduction to Algorithms, 3rd Edition. The MIT Press. 2009.