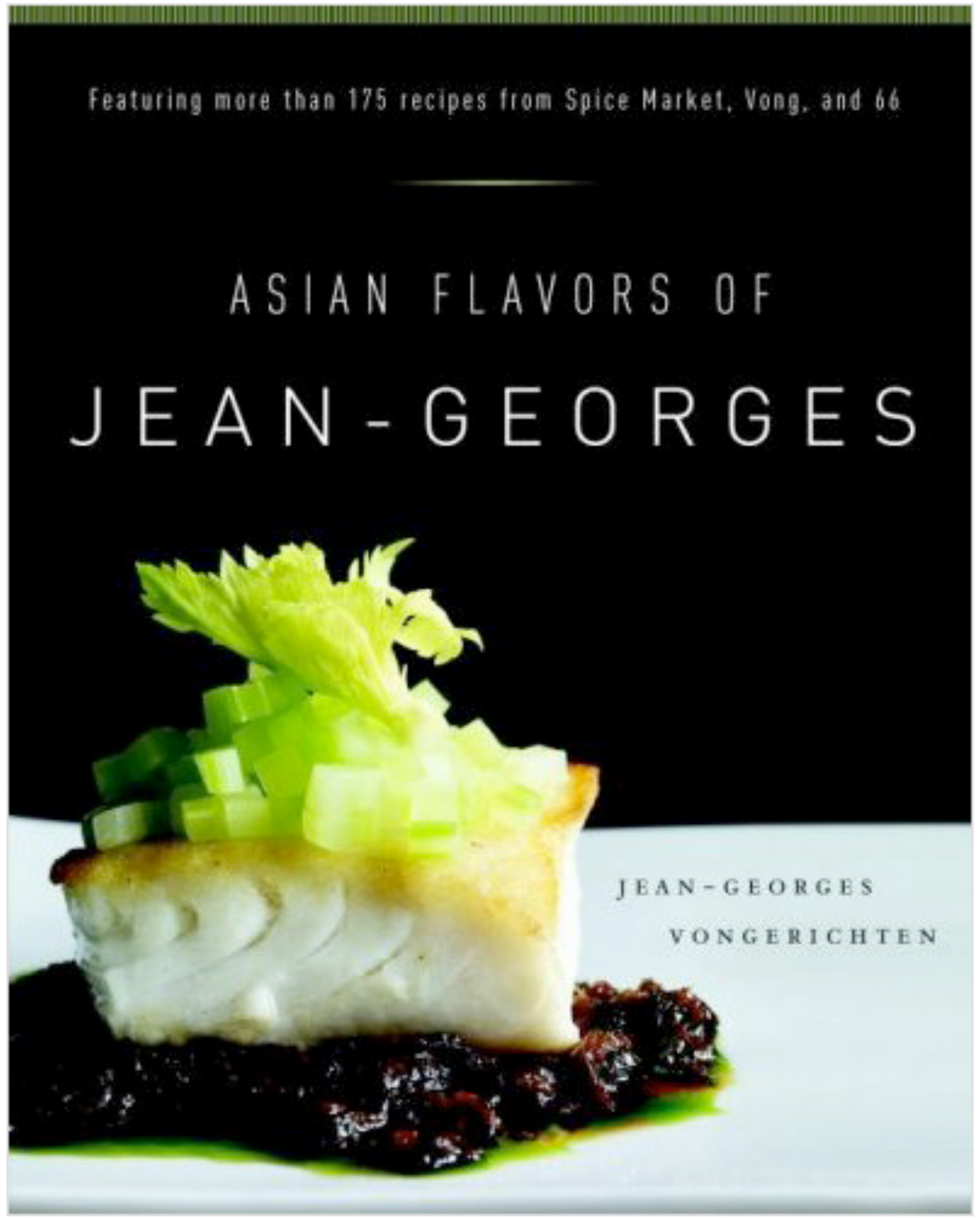
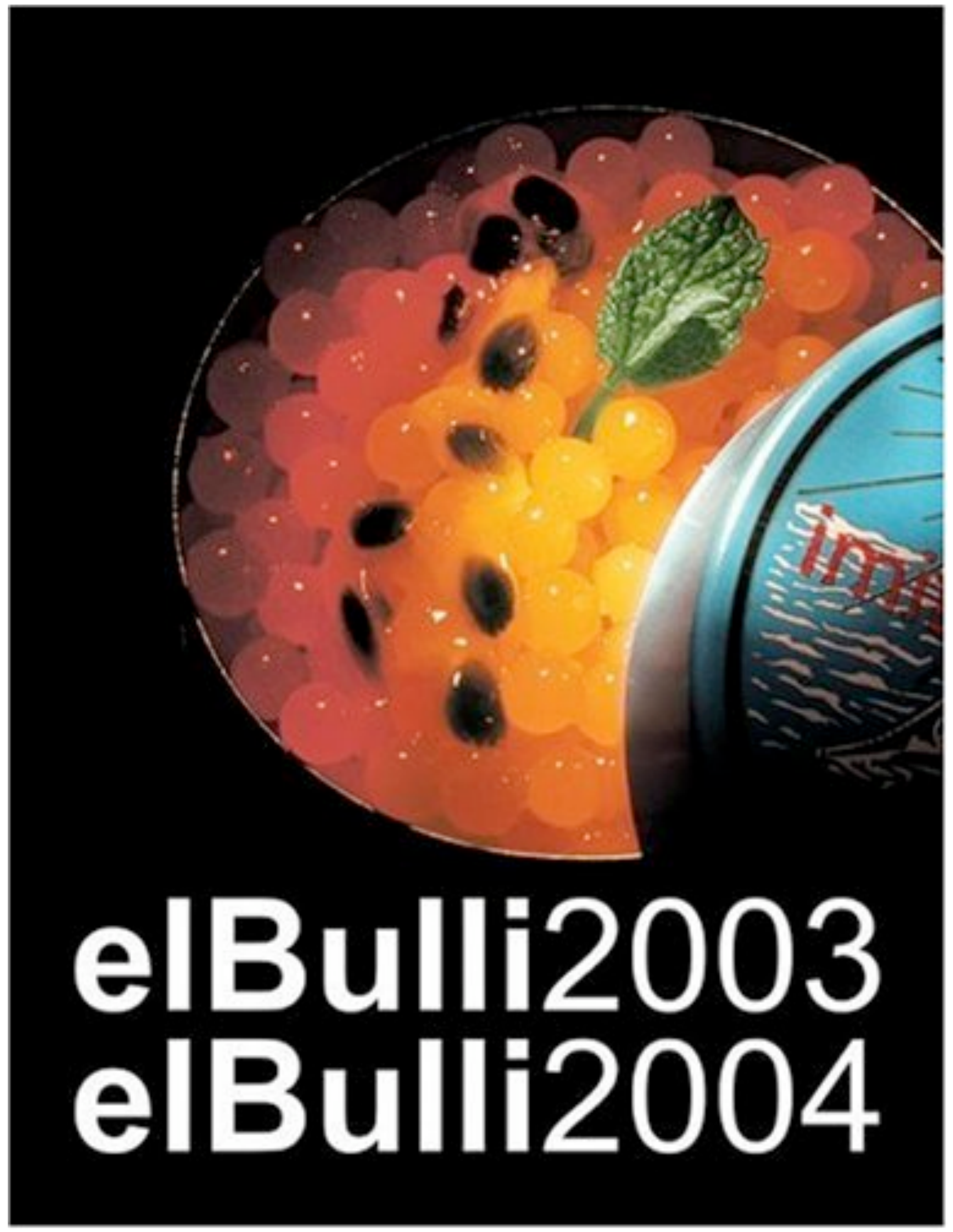
L5

sep 10 2013

4102
shelat

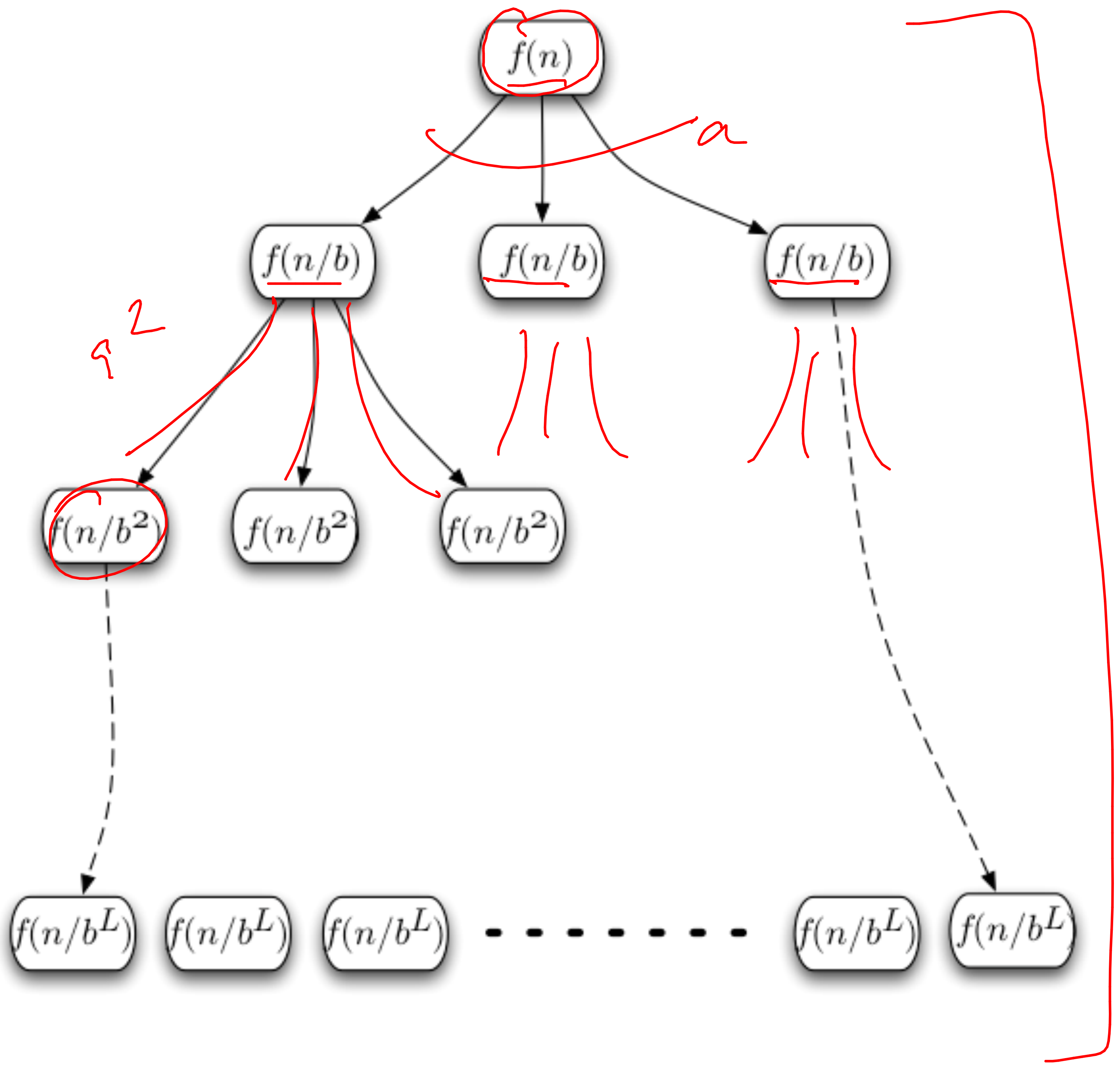
cookbook


Featuring more than 175 recipes from Spice Market, Vong, and 66
ASIAN FLAVORS OF
JEAN-GEORGES
JEAN-GEORGES
VONGERICHTEN

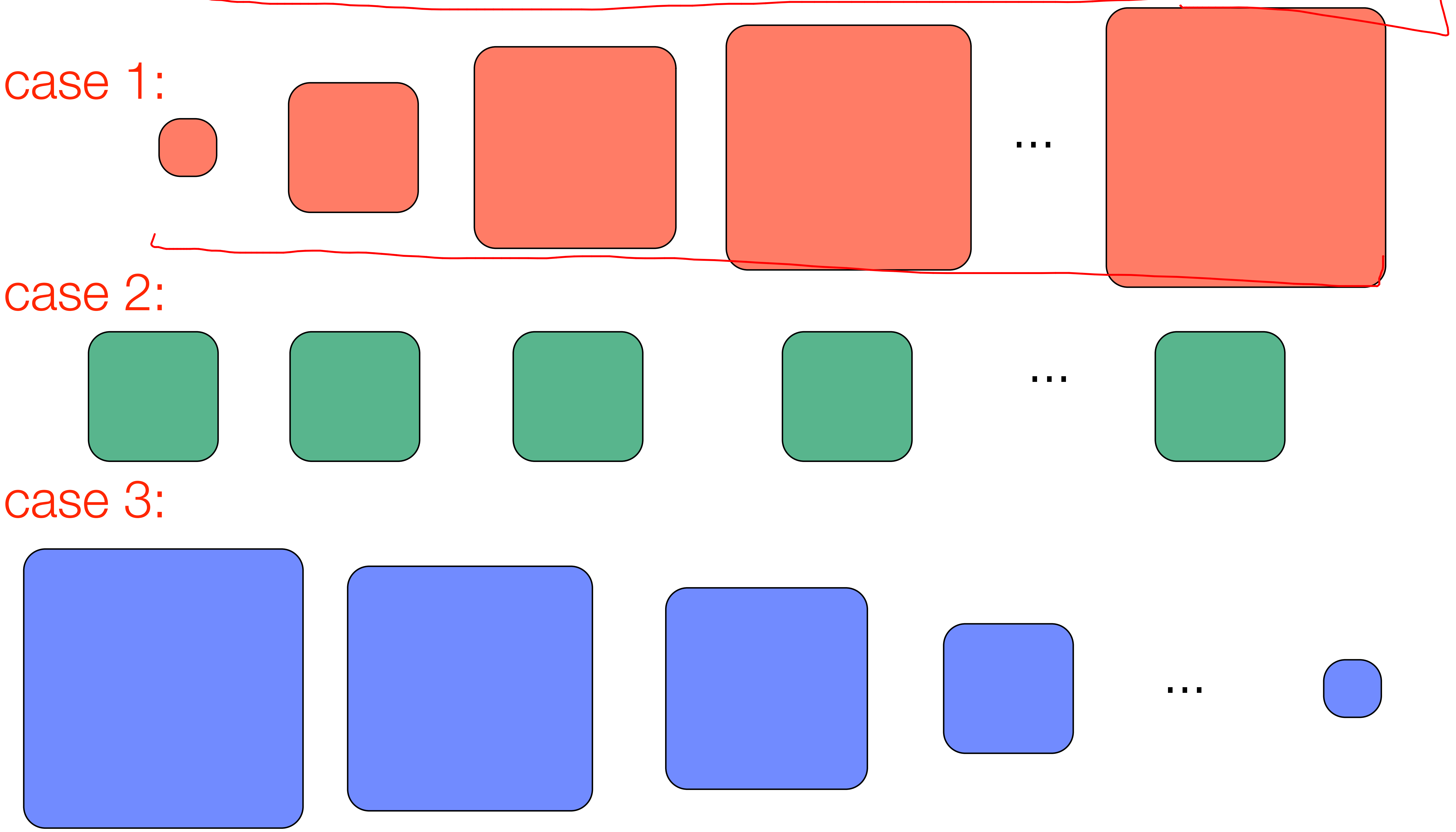

elBulli2003
elBulli2004

$$T(n) = aT(n/b) + f(n)$$

$$T(n) = aT(n/b) + f(n)$$



$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$



$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1:

$$f(n) = O(n^{\log_b a - \epsilon}), \epsilon > 0$$

Then:

$$T(n) = \Theta(n^{\log_b a})$$

case 2:

$$f(n) = \Theta(n^{\log_b a})$$

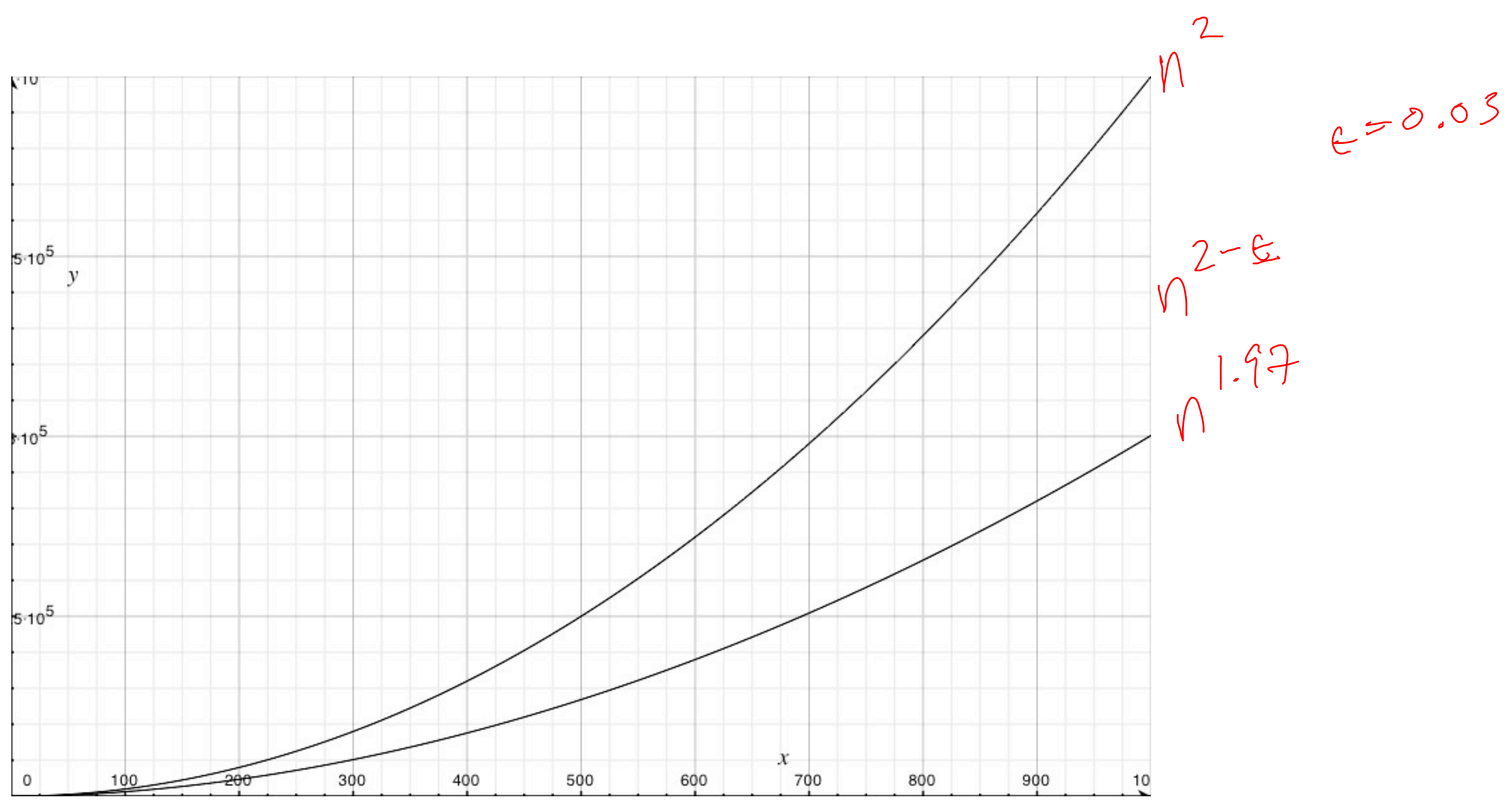
$$T(n) = \Theta(n^{\log_b a} \log n)$$

case 3:

$$f(n) = \Omega(n^{\log_b a + \epsilon}), \epsilon > 0$$

and c<1 s.t $af(n/b) < cf(n)$

$$T(n) = \Theta(f(n))$$



n^2
$\epsilon = 0.03$
$n^{2-\epsilon}$
$n^{1.97}$
y
x
0
100
200
300
400
500
600
700
800
900

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1: Since $f(n) < cn^{\log_b a - \epsilon}$

We have:

$$T(n) < cn^{\log_b a - \epsilon} + a \cdot c\left(\frac{n}{b}\right)^{\log_b a - \epsilon} + a^2 \cdot c\left(\frac{n}{b^2}\right)^{\log_b a - \epsilon} + \cdots + a^{L-1} c \left(\frac{n}{b^{L-1}}\right)^{\log_b a - \epsilon} + c \cdot a^L$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1: Since $f(n) < cn^{\log_b a - \epsilon}$

We have:

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + \frac{a}{b^{\log_b a - \epsilon}} + \frac{a^2}{(b^2)^{\log_b a - \epsilon}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a - \epsilon}}\right] + n^{\log_b a}$$

$\epsilon \sim 0.01$

$$< cn^{\log_b a - \epsilon}\left[1 + b^{\epsilon} + b^{2\epsilon} + b^{3\epsilon} + \cdots + b^{(L-1)\cdot\epsilon}\right] + n^{\log_b a}$$

$$\frac{(b^{\epsilon})^L - 1}{b^{\epsilon} - 1}$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1: Since $f(n) < cn^{\log_b a - \epsilon}$

We have:

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + \frac{a}{b^{\log_b a - \epsilon}} + \frac{a^2}{(b^2)^{\log_b a - \epsilon}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a - \epsilon}}\right] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + b^{\epsilon} + b^{2\epsilon} + \cdots + b^{\epsilon(L-1)}\right] + n^{\log_b a}$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1: Since $f(n) < cn^{\log_b a - \epsilon}$

We have:

$$T(n) \le cn^{\log_b a - \epsilon}\left[1 + \frac{a}{b^{\log_b a - \epsilon}} + \frac{a^2}{(b^2)^{\log_b a - \epsilon}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a - \epsilon}}\right] + n^{\log_b a}$$

$$T(n) \le cn^{\log_b a - \epsilon}\left[1 + b^{\epsilon} + b^{2\epsilon} + \cdots + b^{\epsilon(L-1)}\right] + n^{\log_b a}$$

$$T(n) \le cn^{\log_b a - \epsilon}\left[\frac{b^{\epsilon L} - 1}{b^{\epsilon} - 1}\right] + n^{\log_b a}$$

$$b^{\epsilon L} \sim \left(b^{\log_b n}\right)^{\epsilon} = n^{\epsilon}$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1: Since $f(n) < cn^{\log_b a - \epsilon}$

We have:

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + \frac{a}{b^{\log_b a - \epsilon}} + \frac{a^2}{(b^2)^{\log_b a - \epsilon}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a - \epsilon}}\right] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + b^{\epsilon} + b^{2\epsilon} + \cdots + b^{\epsilon(L-1)}\right] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a - \epsilon}\left[\frac{b^{\epsilon L} - 1}{b^{\epsilon} - 1}\right] + n^{\log_b a}$$

$$T(n) \leq c' n^{\log_b a - \epsilon}\left[n^{\epsilon} - 1\right] + n^{\log_b a}$$

$< c'' n^{\log_b a} = O(n^{\log_b a})$

→ a similar argument can be used to show a lower bound $\Omega(n^{\log_b a}) \Rightarrow \Theta$ bound

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1: Since $f(n) < cn^{\log_b a - \epsilon}$

We have:

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + \frac{a}{b^{\log_b a - \epsilon}} + \frac{a^2}{(b^2)^{\log_b a - \epsilon}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a - \epsilon}}\right] + c \cdot n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a - \epsilon}\left[1 + b^{\epsilon} + b^{2\epsilon} + \cdots + b^{\epsilon(L-1)}\right] + c \cdot n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a - \epsilon}\left[\frac{b^{\epsilon L} - 1}{b^{\epsilon} - 1}\right] + c \cdot n^{\log_b a}$$

$$T(n) \leq c' n^{\log_b a - \epsilon}\left[n^{\epsilon} - 1\right] + c \cdot n^{\log_b a} = O(n^{\log_b a})$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 2 : When $f(n) \le cn^{\log_b a}$

$f(n) = \Theta(n^{\log_b a})$

$T(n) \le cn^{\log_b a} + a \cdot c\left(\frac{n}{b}\right)^{\log_b a} + a^2 c\left(\frac{n}{b^2}\right)^{\log_b a} + \dots + a^L \left(\frac{n}{b^L}\right)^{\log_b a}$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 2 : When $f(n) < cn^{\log_b a}$

$$T(n) \leq cn^{\log_b a}\left[1 + \frac{a}{b^{\log_b a}} + \frac{a^2}{(b^2)^{\log_b a}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a}}\right] + cn^{\log_b a}$$

how many ones ??

L ones

$L = \log_b n$

$$\leq c \cdot n^{\log_b a} \cdot \log_b n + c \cdot n^{\log_b a}$$

$$= O(n^{\log_b a} \log n)$$

notice no b ??

difference in bases for logarithms ~ constants

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 2 : When $f(n) < cn^{\log_b a}$

$$T(n) \leq cn^{\log_b a}\left[1 + \frac{a}{b^{\log_b a}} + \frac{a^2}{(b^2)^{\log_b a}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a}}\right] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a}\,[1 + 1 + \cdots + 1] + n^{\log_b a}$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 2 : When $f(n) < cn^{\log_b a}$

$$T(n) \leq cn^{\log_b a}\left[1 + \frac{a}{b^{\log_b a}} + \frac{a^2}{(b^2)^{\log_b a}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a}}\right] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a}\,[1 + 1 + \cdots + 1] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a} \log_b(a)$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 2 : When $f(n) < cn^{\log_b a}$

$$T(n) \leq cn^{\log_b a}\left[1 + \frac{a}{b^{\log_b a}} + \frac{a^2}{(b^2)^{\log_b a}} + \frac{a^{L-1}}{(b^{L-1})^{\log_b a}}\right] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a}[1 + 1 + \cdots + 1] + n^{\log_b a}$$

$$T(n) \leq cn^{\log_b a}\log_b(a) \quad = O(n^{\log_b a}\log n)$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 3: $f(n) > n^{\log_b a+\epsilon}$ and c<1 s.t $af(n/b) < cf(n)$ $\rightarrow$ $a \cdot c \cdot f\left(\frac{n}{b^2}\right) < c \cdot f\left(\frac{n}{b}\right)$

$T(n) \le f(n) + c \cdot f(n) + c^2 \cdot f(n)$

$a^2 f\left(\frac{n}{b^2}\right) = a\; a \cdot f\left(\frac{n}{b^2}\right) < a\, c\, f\left(\frac{n}{b}\right) < c\, c \cdot f(n)$

$T(n) \le f(n) + c \cdot f(n) + c^2 f(n) + c^3 f(n) + \cdots + c^L f(n)$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 3: $f(n) > n^{\log_b a + \epsilon}$ and c<1 s.t $af(n/b) < cf(n)$

$$T(n) \leq f(n) + cf(n) + c^2 f(n) + \cdots + c^L f(n)$$

$$\leq f(n)\left[1 + c + c^2 + \cdots + c^L\right]$$

$$\frac{c^{L+1} - 1}{c - 1}$$

$c < 1$

so this term will be a small constant

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

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 3: $f(n) > n^{\log_b a + \epsilon}$ and c<1 s.t $af(n/b) < cf(n)$

$$T(n) \leq f(n) + cf(n) + c^2 f(n) + \cdots + c^L f(n)$$

$$T(n) \leq f(n)[1 + c + c^2 + \cdots + c^L]$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 3: $f(n) > n^{\log_b a + \epsilon}$ and c<1 s.t $af(n/b) < cf(n)$

$$T(n) \leq f(n) + cf(n) + c^2 f(n) + \cdots + c^L f(n)$$

$$T(n) \leq f(n)[1 + c + c^2 + \cdots + c^L]$$

$$= O(f(n))$$

$$T(n) = f(n) + af\left(\frac{n}{b}\right) + a^2 f\left(\frac{n}{b^2}\right) + a^3 f\left(\frac{n}{b^3}\right) + \cdots + a^L f\left(\frac{n}{b^L}\right)$$

case 1:

$$f(n) = O(n^{\log_b a - \epsilon}), \epsilon > 0$$

Then:

$$T(n) = \Theta(n^{\log_b a})$$

case 2:

$$f(n) = \Theta(n^{\log_b a})$$

$$T(n) = \Theta(n^{\log_b a} \log n)$$

case 3:

$$f(n) = \Omega(n^{\log_b a + \epsilon}), \epsilon > 0$$

$$T(n) = \Theta(f(n))$$

and c<1 s.t $af(n/b) < cf(n)$

example 2: $T(n) = 8T(n/2) + \Theta(n^2)$

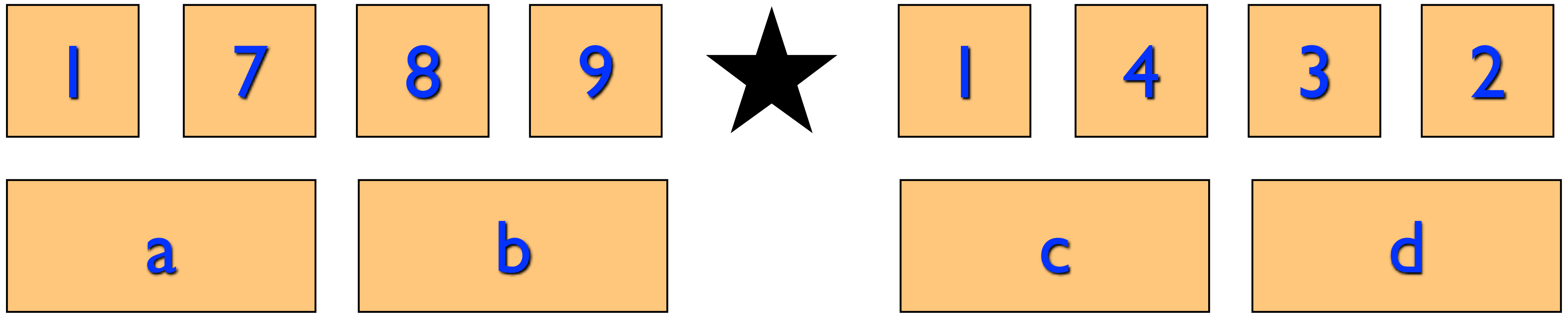
a (8), b (2), $f(n)$ ($\Theta(n^2)$)

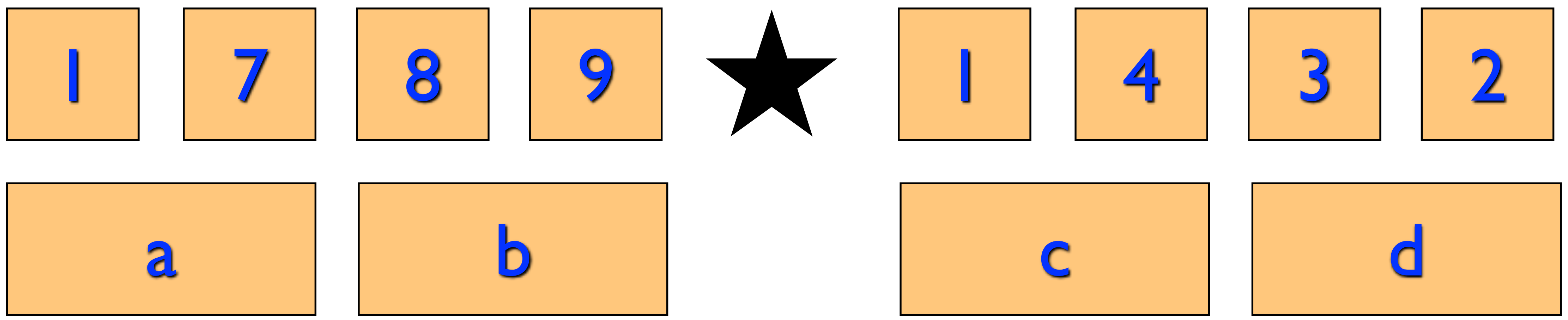
$f(n) = \Theta(n^2)$ $\qquad n^{\log_2 8} = n^3$

since $f(n) = O(n^{\log_b a - \epsilon})$ for $\epsilon = 0.1$, case 1 applies

$O(n^{2.99})$

$T(n) = \Theta(n^{\log_2 8}) = \Theta(n^3)$. ✓


1
7
8
9
1
4
3
2
a
b
c
d



$$T(n) = 4T(n/2) + 3O(n)$$

$4 \rightarrow a$, $2 \rightarrow b$, $3O(n) \rightarrow f(n)$

$0.01 \rightarrow \epsilon$

since $f(n) = O(n)$ is in $O(n^{\log_2 4 - 0.01}) = O(n^{1.99})$, then case 1 applies, and

$$T(n) = \Theta(n^{\log_b a}) = \Theta(n^2)$$

example 2:

$$T(n) = {}_a T\left(\frac{14}{17}n\right) + 24$$

$a = 1$ $\quad b = \frac{17}{14}$ $\quad f(n) = 24$

case 1: is $f(n) = 24$ in $O(n^{\log_{17/14} 1 - \epsilon}) \stackrel{??}{=} O(n^{0 - \epsilon})$ N

becaus $O(n^{-\epsilon})$ grows smaller as $n \uparrow$

case 2: is $f(n) = 24$ in $\Theta(n^{\log_b a}) = \Theta(n^0) = \Theta(1)$

Yes

case 2 applies & $T(n) = \Theta(n^{\log_b a} \cdot \log n) = \Theta(\log n)$

$$T(n) = 2T(n/2) + n^3$$

a (the 2), b (the 2 in $n/2$), $f(n)$ (the n^3)

case 3: $f(n) = n^3 = \Omega(n^{\log_2 2 + 0.01}) = \Omega(n^{1.01})$

additionally if we set $c = 0.9$ ($c = 0.1$), then

$$2 \cdot f\left(\frac{n}{2}\right) = 2 \cdot \left(\frac{1}{8}\right) \cdot n^3 \le \underbrace{0.9}_{c} \cdot n^3$$

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

$$0.25n^3 \le 0.9 \cdot n^3$$

$$0.25n^3 \not\le 0.1n^3$$

$$T(n) = 16T(n/4) + n^2$$

(todo @ home)

$$T(n) = 7T(n/2) + O(n^2)$$

Self check yourself

substitution

$$T(n) = 2T(\sqrt{n}) + \lg n$$

Let $2^m = n$

$m = \log n$

$$T(2^m) = 2T(\sqrt{2^m}) + \log(2^m)$$

$$= 2T(2^{m/2}) + m$$

$$S(m) = T(2^m)$$

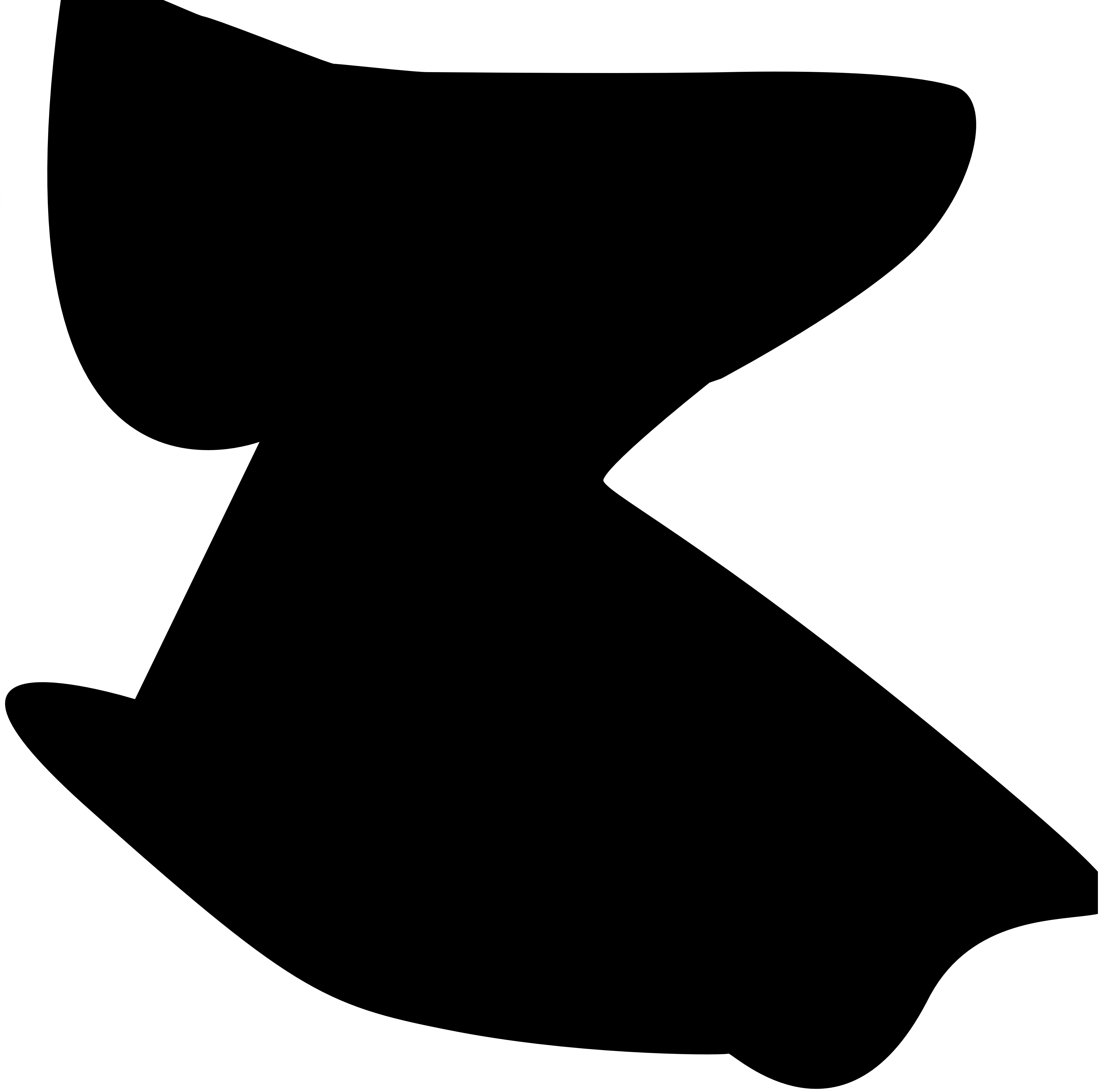
$$S(m) = 2S(m/2) + m$$

$$S(m) = \Theta(m \cdot \log m)$$

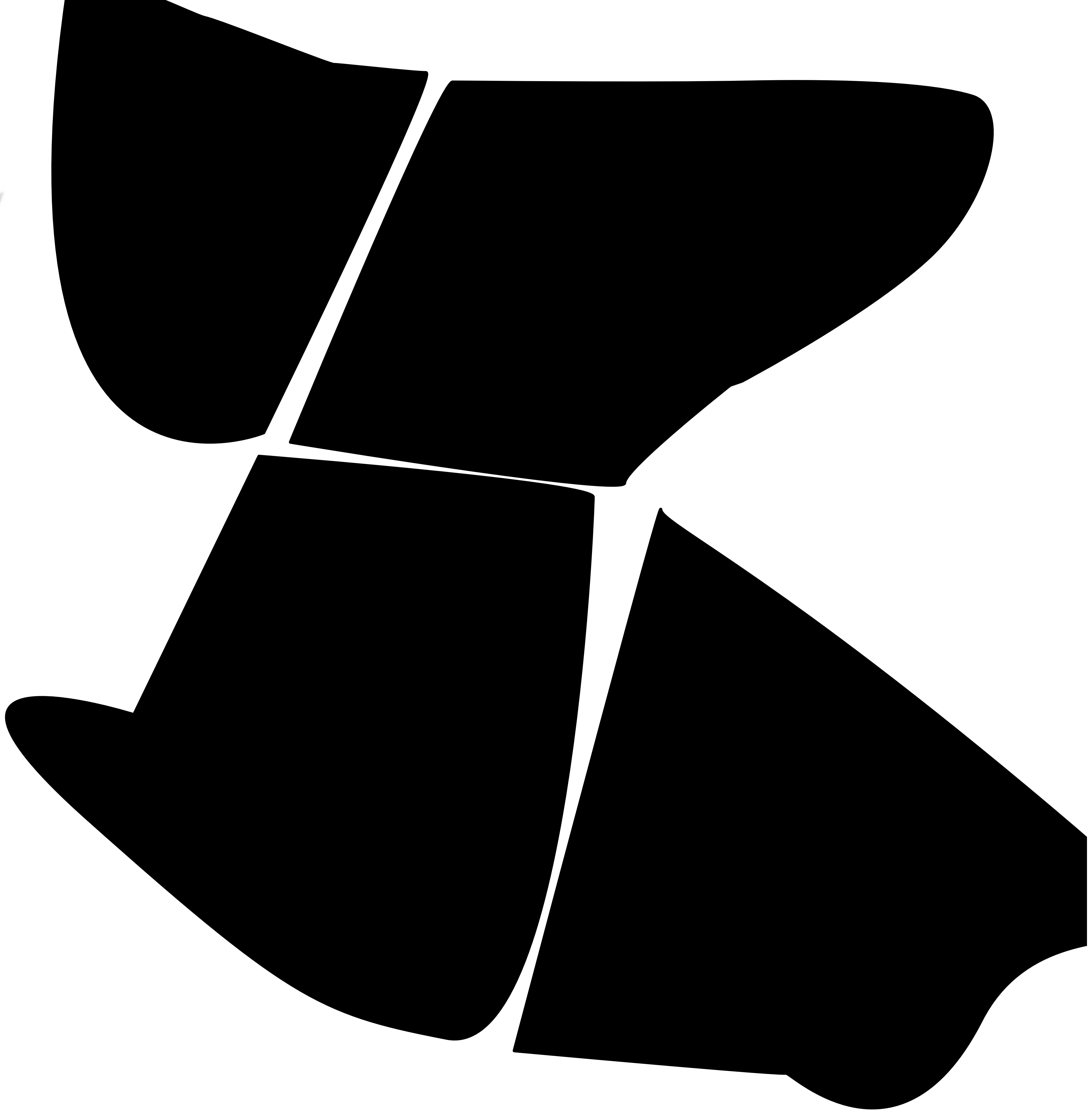
$$T(n) = \Theta(\log n \cdot \log\log n)$$

divide

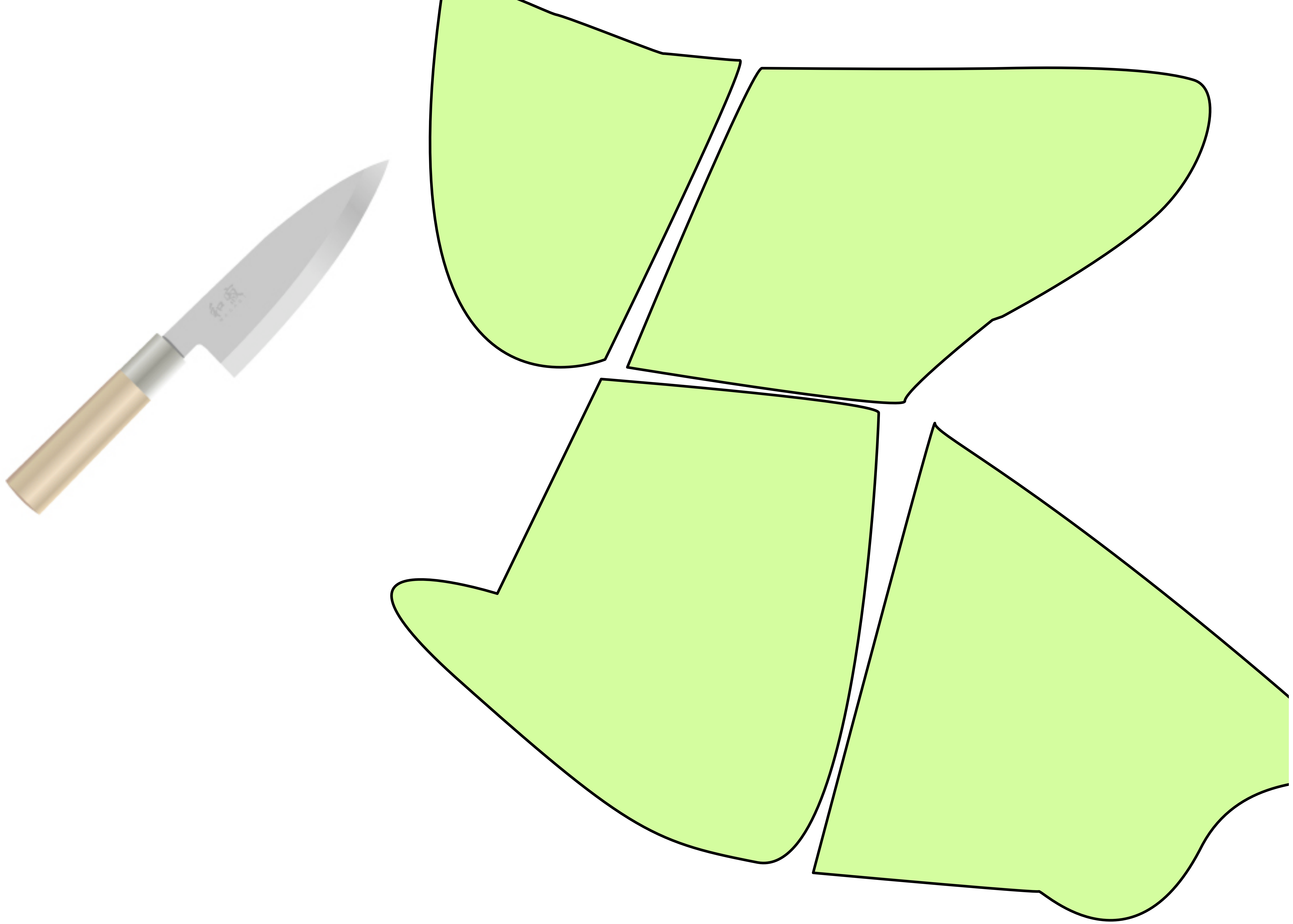
& conquer



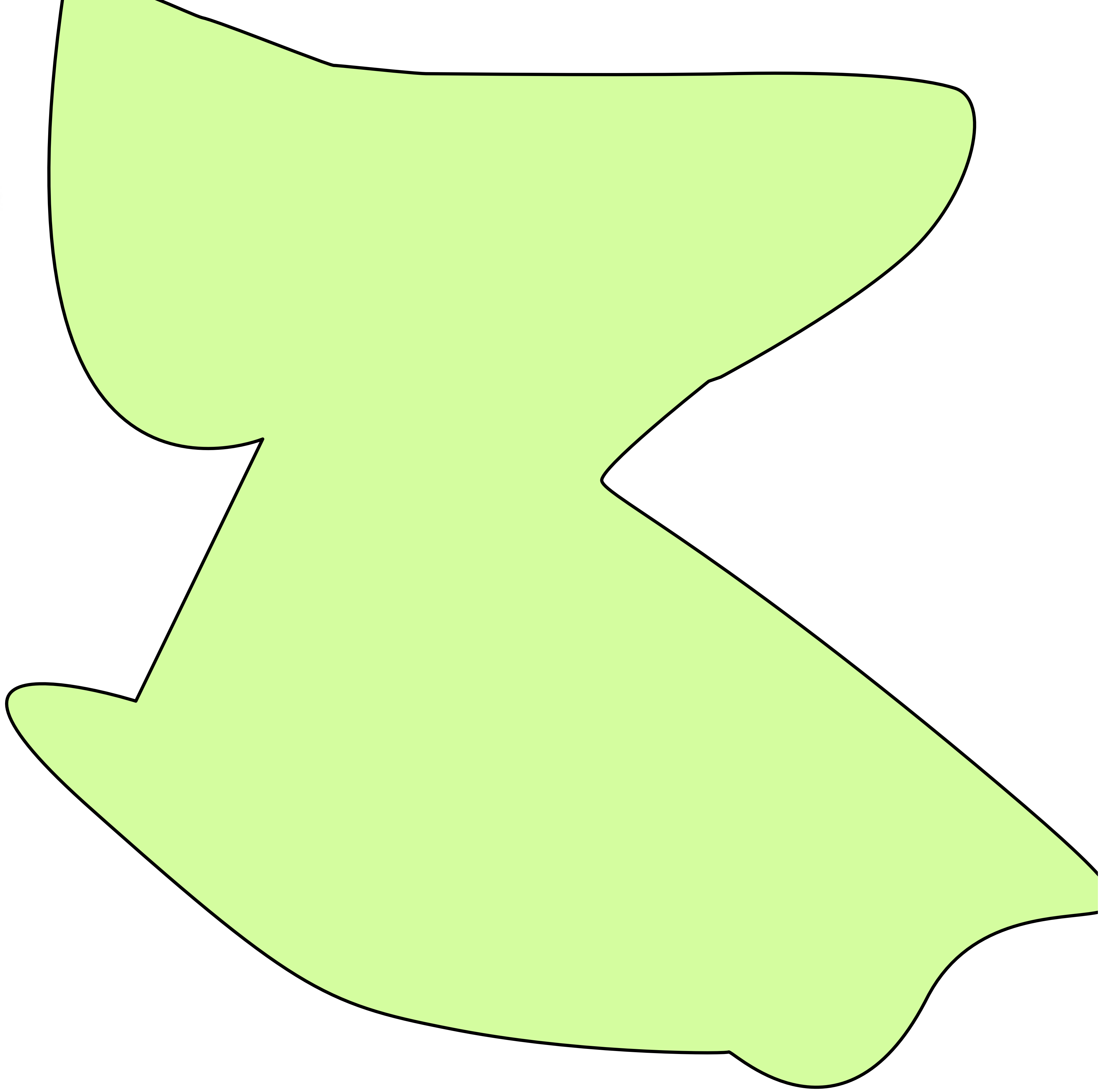
http://www.kitchenknifedrawer.com/files/1696205/uploaded/K6615D.jpg



http://www.kitchenknifedrawer.com/files/1696205/uploaded/K6615D.jpg



http://www.kitchenknifedrawer.com/files/1696205/uploaded/K6615D.jpg



http://www.kitchenknifedrawer.com/files/1696205/uploaded/K6615D.jpg

examples

merge sort

Karatsuba

closest-point

matrix mult

FFT

* MEDIAN

Merge

merge-sort (A, p, r)
  if $p < r$
    $q \leftarrow \lfloor (p+r)/2 \rfloor$
    merge-sort (A, p, q)
    merge-sort $(A, q+1, r)$
    merge(A, p, q, r)

```
MERGE(A[1..n], m):
  i ← 1; j ← m + 1
  for k ← 1 to n
    if j > n
      B[k] ← A[i]; i ← i + 1
    else if i > m
      B[k] ← A[j]; j ← j + 1
    else if A[i] < A[j]
      B[k] ← A[i]; i ← i + 1
    else
      B[k] ← A[j]; j ← j + 1
  for k ← 1 to n
    A[k] ← B[k]
```

jeff erickson

5	2	4	7	1	3	2	6

merge-sort (A, p, r)
    if $p < r$
        $q \leftarrow \lfloor (p+r)/2 \rfloor$
        merge-sort (A, p, q)
        merge-sort $(A, q+1, r)$
        merge(A, p, q, r)

```
MERGE(A[1..n], m):
  i ← 1; j ← m + 1
  for k ← 1 to n
      if j > n
          B[k] ← A[i]; i ← i + 1
      else if i > m
          B[k] ← A[j]; j ← j + 1
      else if A[i] < A[j]
          B[k] ← A[i]; i ← i + 1
      else
          B[k] ← A[j]; j ← j + 1
  for k ← 1 to n
      A[k] ← B[k]
```

jeff erickson

5	2	4	7	1	3	2	6

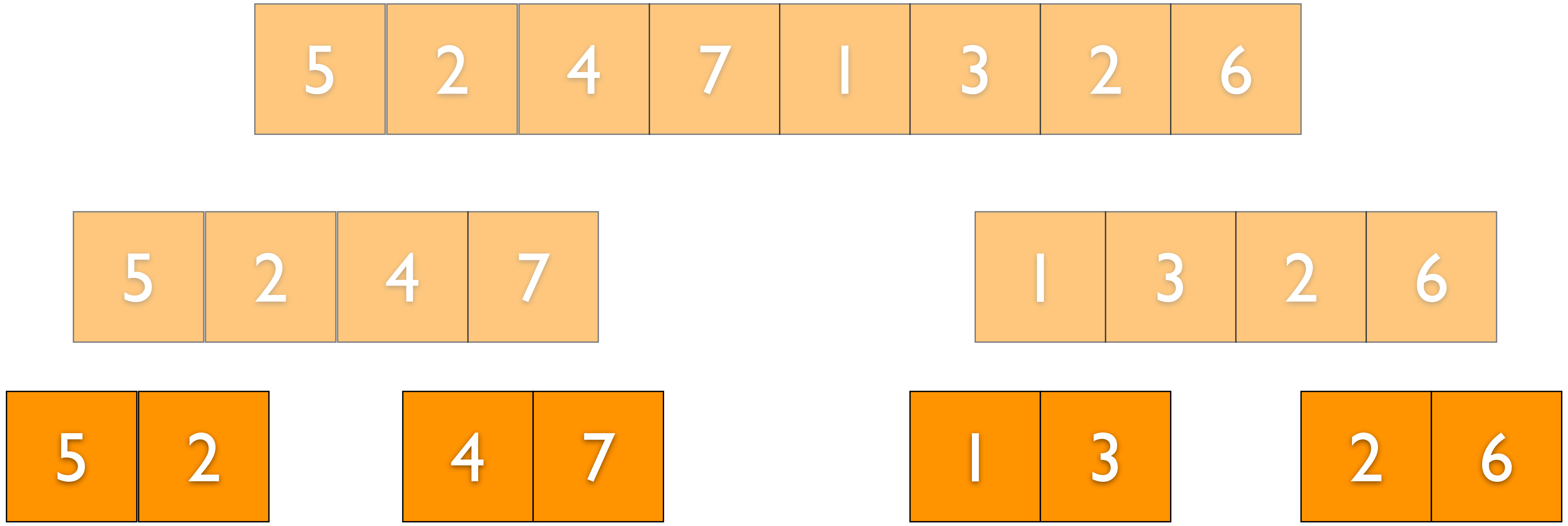
5	2	4	7

1	3	2	6

merge-sort (A, p, r)
if $p < r$
$q \leftarrow \lfloor (p + r)/2 \rfloor$
merge-sort (A, p, q)
merge-sort $(A, q+1, r)$
merge(A, p, q, r)

MERGE$(A[1..n], m)$:
$i \leftarrow 1$; $j \leftarrow m+1$
for $k \leftarrow 1$ to n
if $j > n$
$B[k] \leftarrow A[i]$; $i \leftarrow i+1$
else if $i > m$
$B[k] \leftarrow A[j]$; $j \leftarrow j+1$
else if $A[i] < A[j]$
$B[k] \leftarrow A[i]$; $i \leftarrow i+1$
else
$B[k] \leftarrow A[j]$; $j \leftarrow j+1$
for $k \leftarrow 1$ to n
$A[k] \leftarrow B[k]$

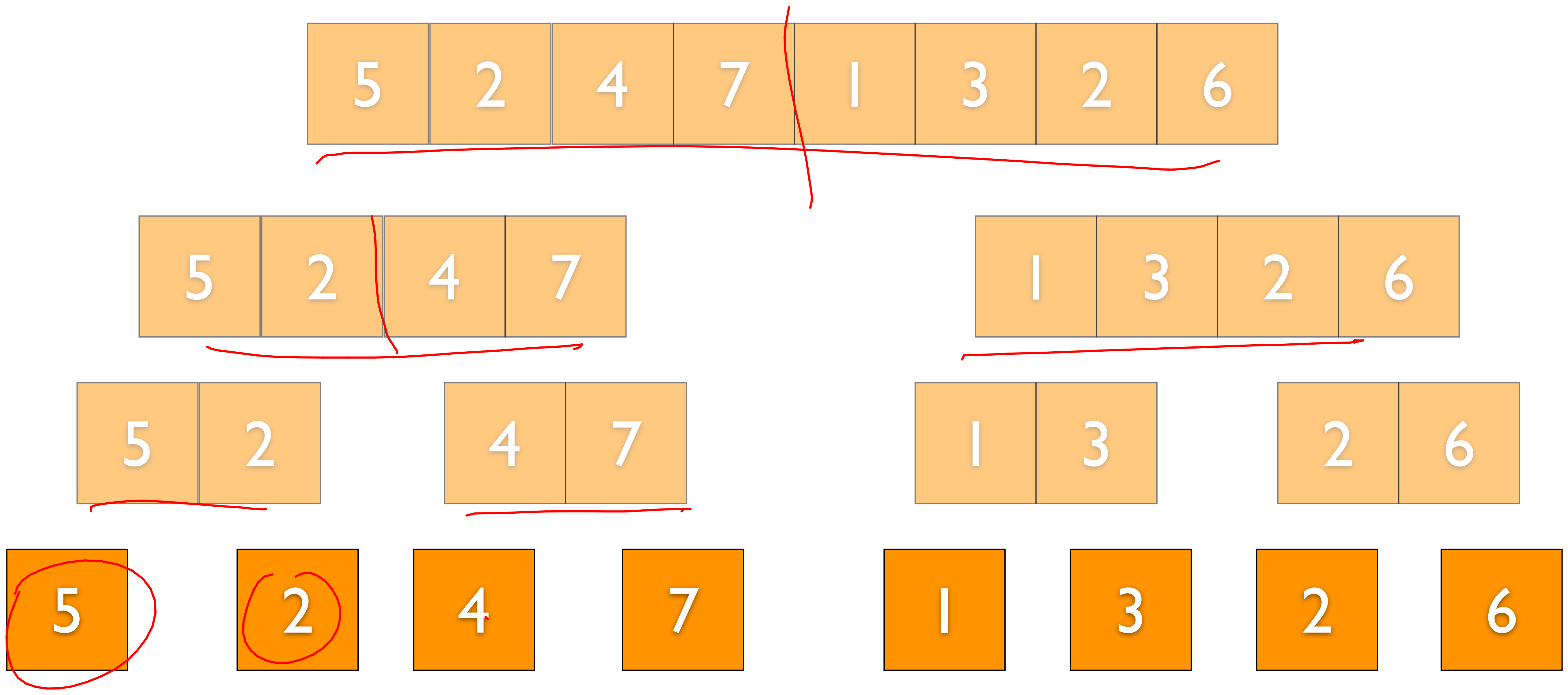
jeff erickson



merge-sort (A, p, r)
if $p < r$
$q \leftarrow \lfloor (p + r)/2 \rfloor$
merge-sort (A, p, q)
merge-sort $(A, q+1, r)$
merge(A, p, q, r)

MERGE$(A[1..n], m)$:
$i \leftarrow 1$; $j \leftarrow m+1$
for $k \leftarrow 1$ to n
if $j > n$
$B[k] \leftarrow A[i]$; $i \leftarrow i+1$
else if $i > m$
$B[k] \leftarrow A[j]$; $j \leftarrow j+1$
else if $A[i] < A[j]$
$B[k] \leftarrow A[i]$; $i \leftarrow i+1$
else
$B[k] \leftarrow A[j]$; $j \leftarrow j+1$
for $k \leftarrow 1$ to n
$A[k] \leftarrow B[k]$

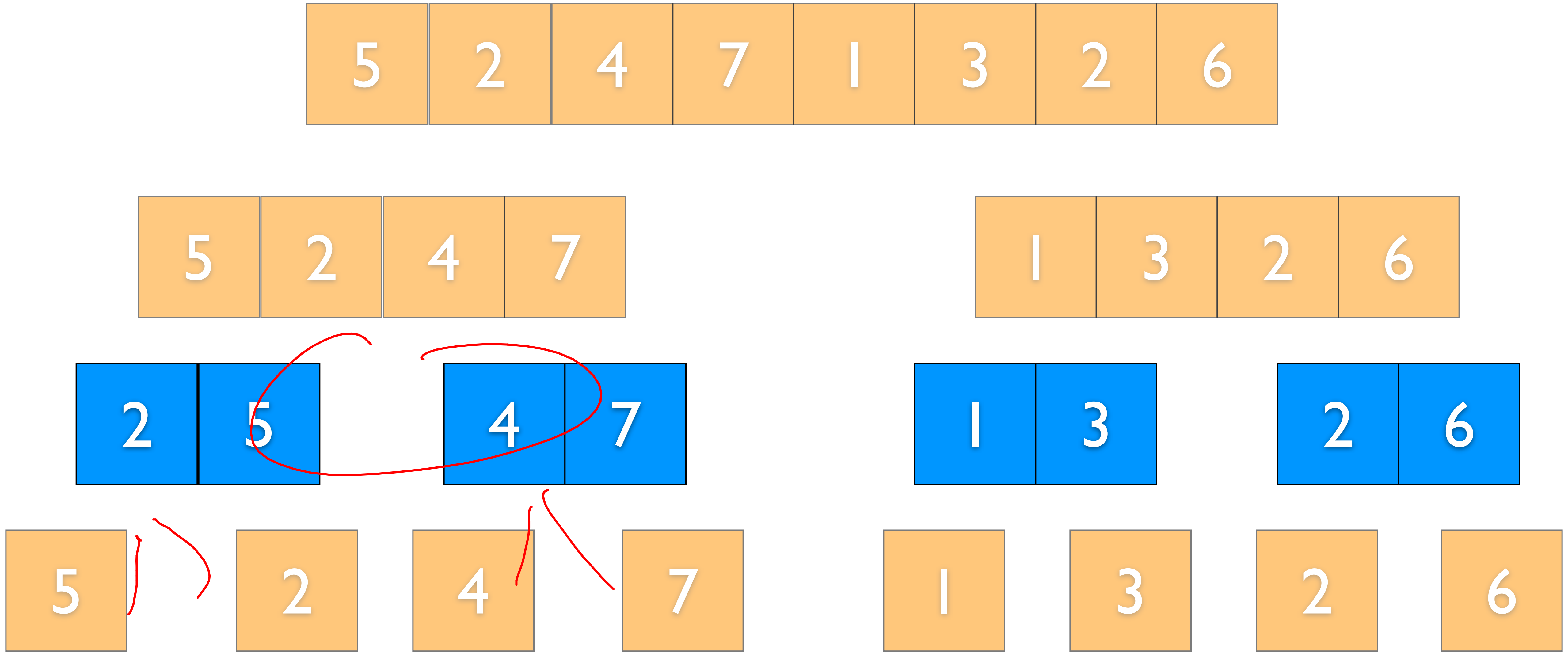
jeff erickson



merge-sort (A, p, r)
  if $p < r$
    $q \leftarrow \lfloor (p+r)/2 \rfloor$
    merge-sort (A, p, q)
    merge-sort $(A, q+1, r)$
    merge(A, p, q, r)

```
MERGE(A[1..n], m):
  i ← 1; j ← m + 1
  for k ← 1 to n
      if j > n
          B[k] ← A[i]; i ← i + 1
      else if i > m
          B[k] ← A[j]; j ← j + 1
      else if A[i] < A[j]
          B[k] ← A[i]; i ← i + 1
      else
          B[k] ← A[j]; j ← j + 1
  for k ← 1 to n
      A[k] ← B[k]
```

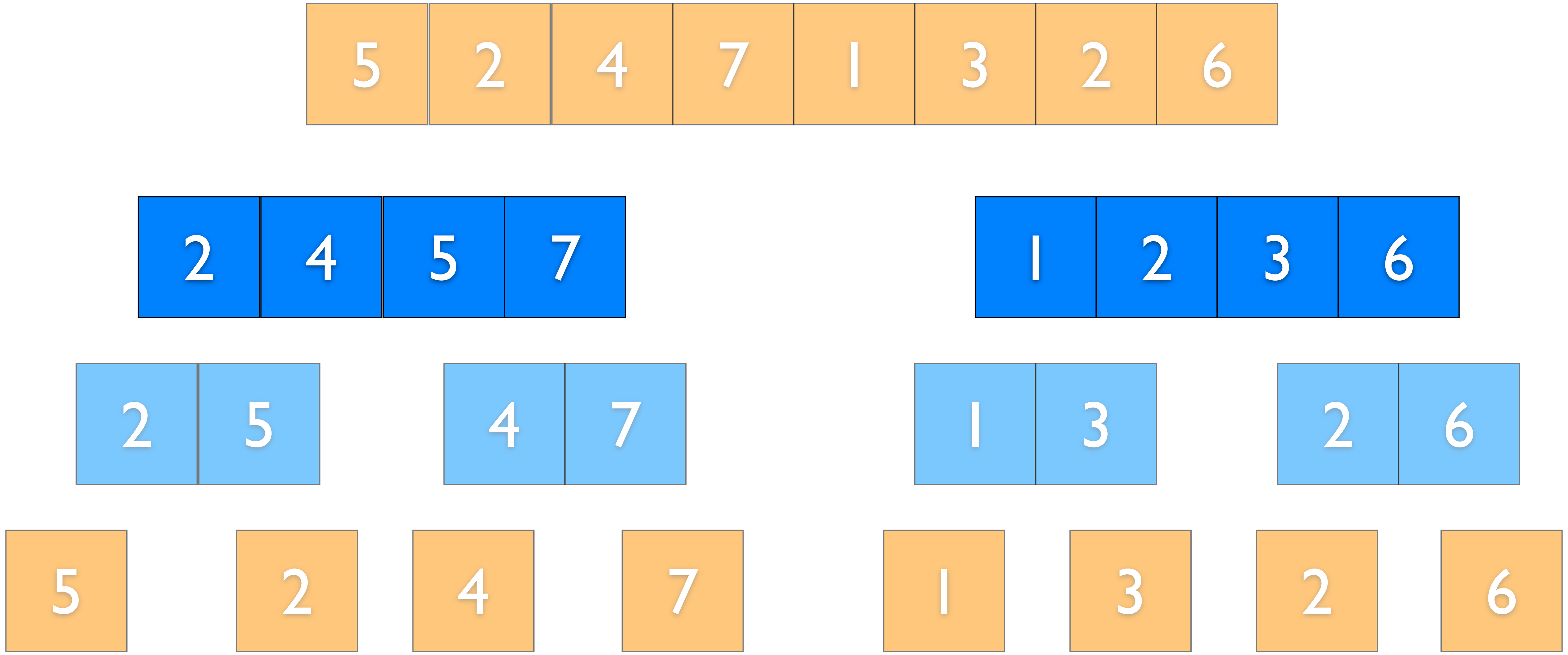
jeff erickson



merge-sort (A, p, r)
  if $p < r$
    $q \leftarrow \lfloor (p + r)/2 \rfloor$
    merge-sort (A, p, q)
    merge-sort $(A, q + 1, r)$
    merge(A, p, q, r)

```
MERGE(A[1..n], m):
  i ← 1; j ← m+1
  for k ← 1 to n
      if j > n
          B[k] ← A[i]; i ← i+1
      else if i > m
          B[k] ← A[j]; j ← j+1
      else if A[i] < A[j]
          B[k] ← A[i]; i ← i+1
      else
          B[k] ← A[j]; j ← j+1
  for k ← 1 to n
      A[k] ← B[k]
```

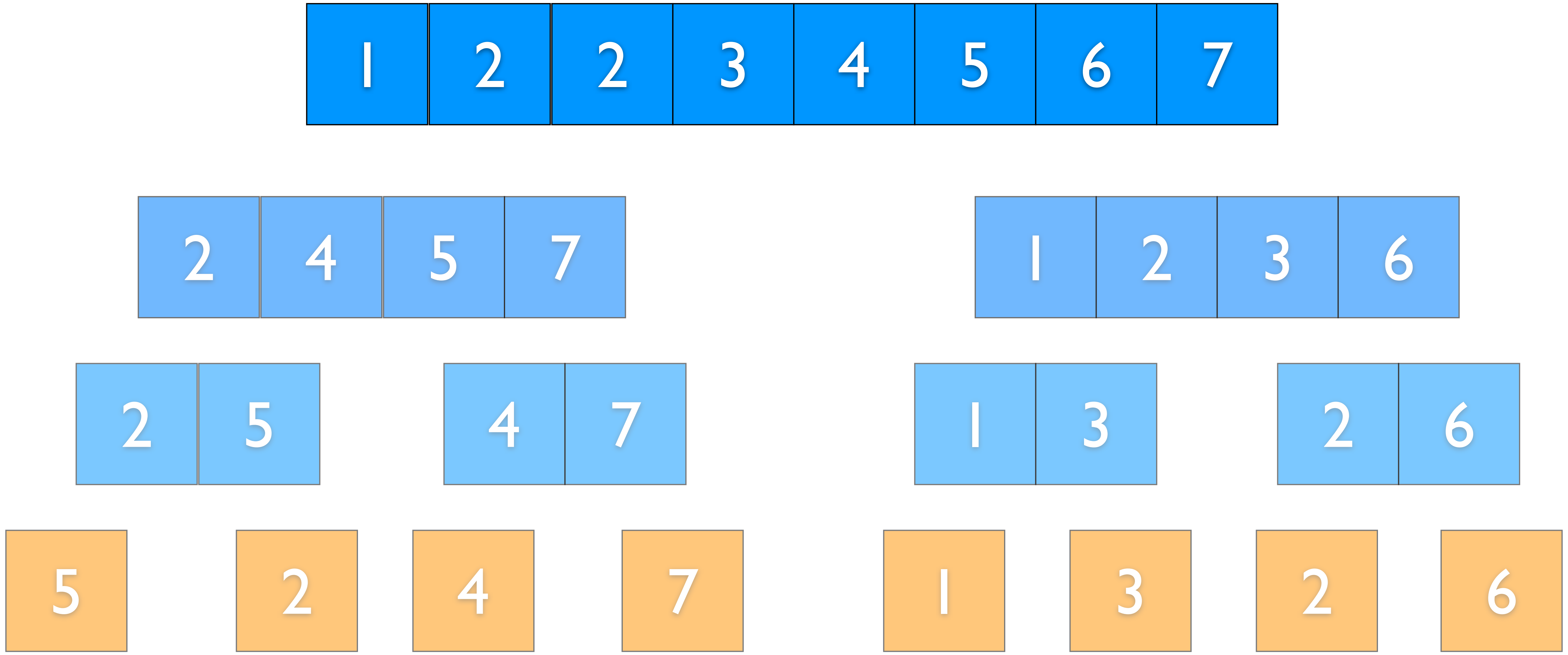
jeff erickson



merge-sort (A, p, r)
  if $p < r$
    $q \leftarrow \lfloor (p + r)/2 \rfloor$
    merge-sort (A, p, q)
    merge-sort $(A, q + 1, r)$
    merge(A, p, q, r)

MERGE$(A[1\,..\,n], m)$:
  $i \leftarrow 1$; $j \leftarrow m + 1$
  for $k \leftarrow 1$ to n
    if $j > n$
      $B[k] \leftarrow A[i]$; $i \leftarrow i + 1$
    else if $i > m$
      $B[k] \leftarrow A[j]$; $j \leftarrow j + 1$
    else if $A[i] < A[j]$
      $B[k] \leftarrow A[i]$; $i \leftarrow i + 1$
    else
      $B[k] \leftarrow A[j]$; $j \leftarrow j + 1$
  for $k \leftarrow 1$ to n
    $A[k] \leftarrow B[k]$

jeff erickson



merge-sort (A, p, r)
    if $p < r$
        $q \leftarrow \lfloor (p + r)/2 \rfloor$
        merge-sort (A, p, q)
        merge-sort $(A, q + 1, r)$ → $2T(n/2)$
        merge(A, p, q, r) → $\Theta(n)$

$$T(n) = 2T(n/2) + O(n)$$

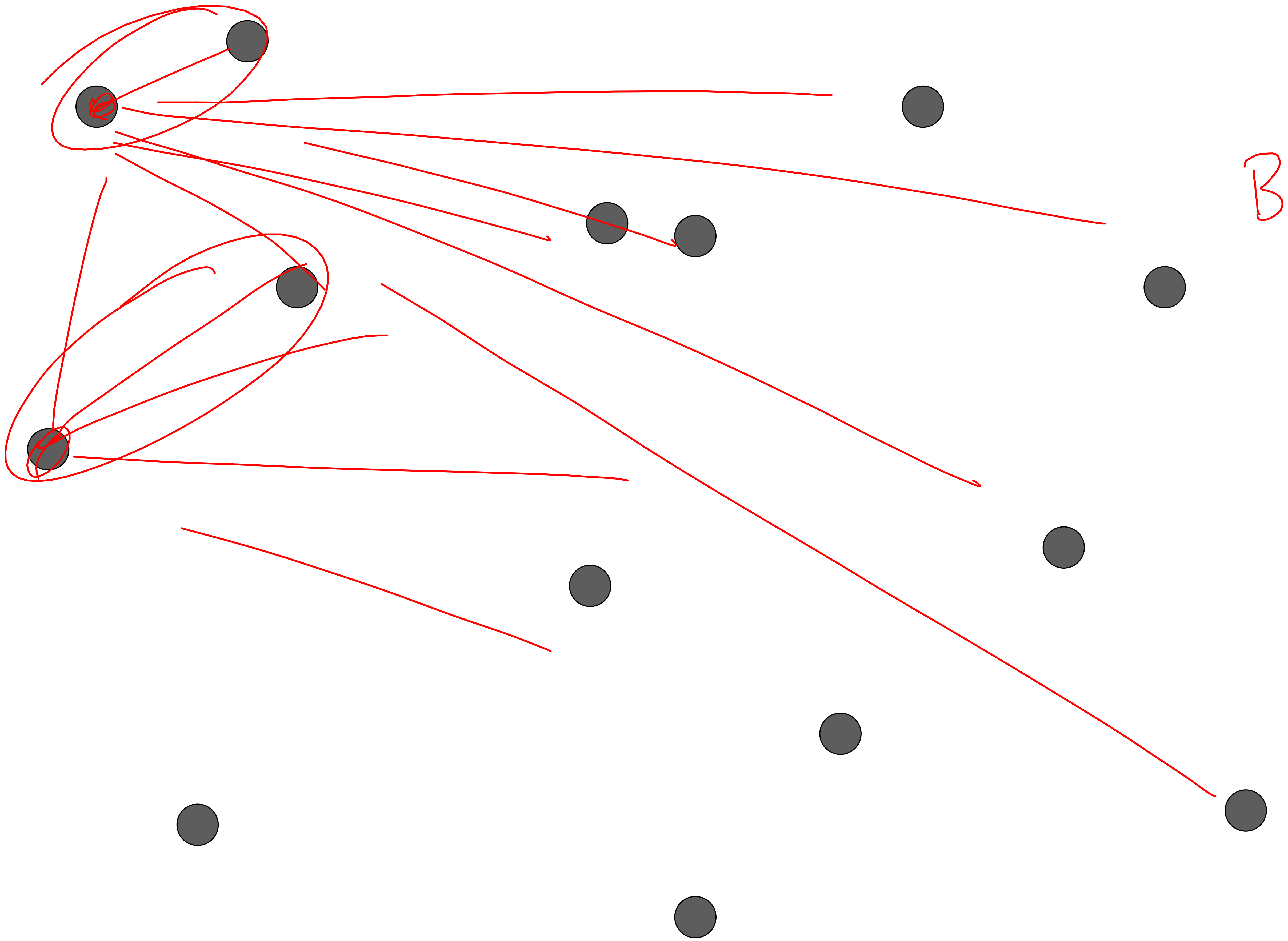
$$= \Theta(n \log n)$$

closest pair

of points

Input: n points in the plane (2-d)

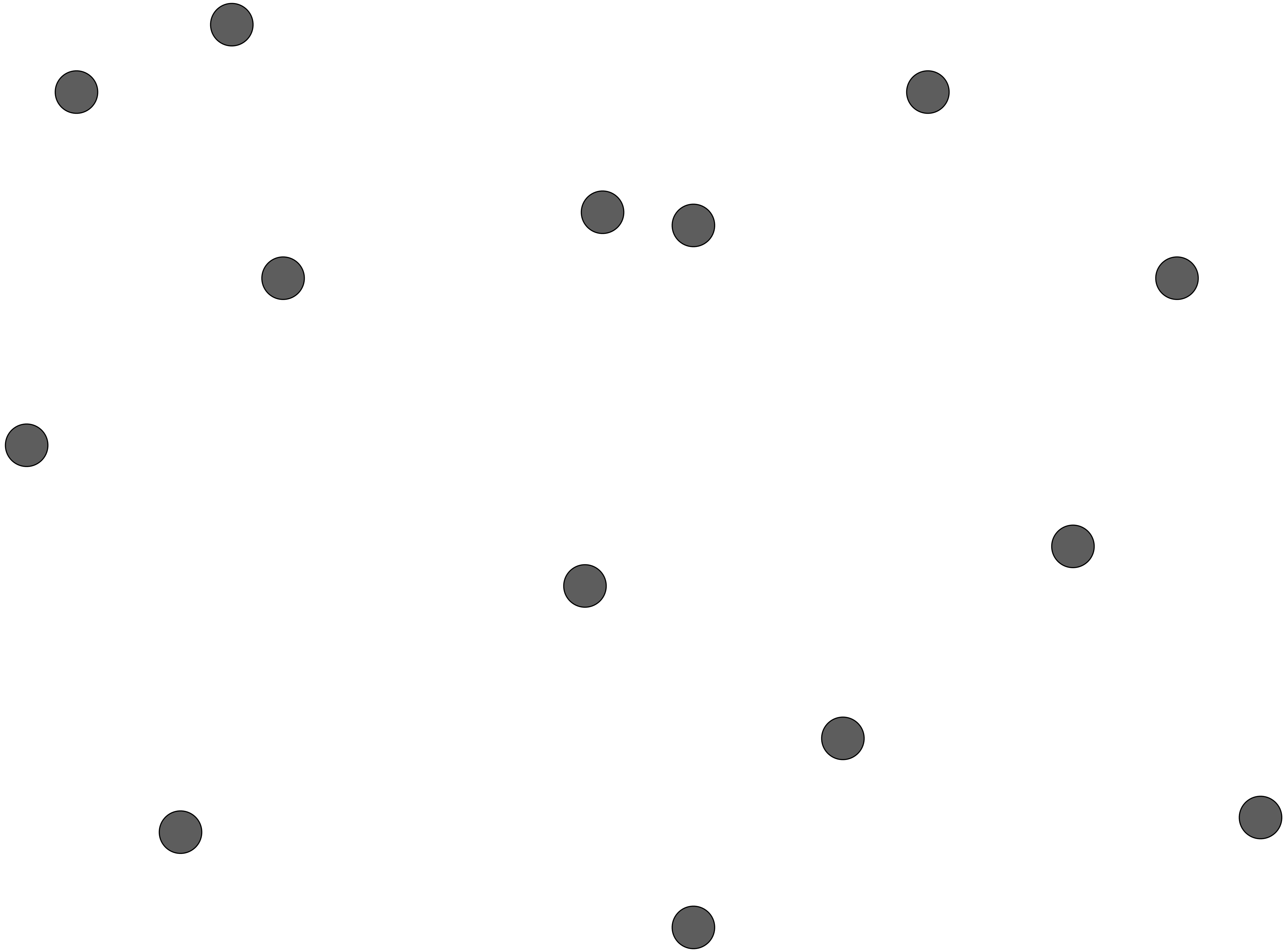
Output: pair of points closest to one another.



Brute force: try all pairs, pick the smallest

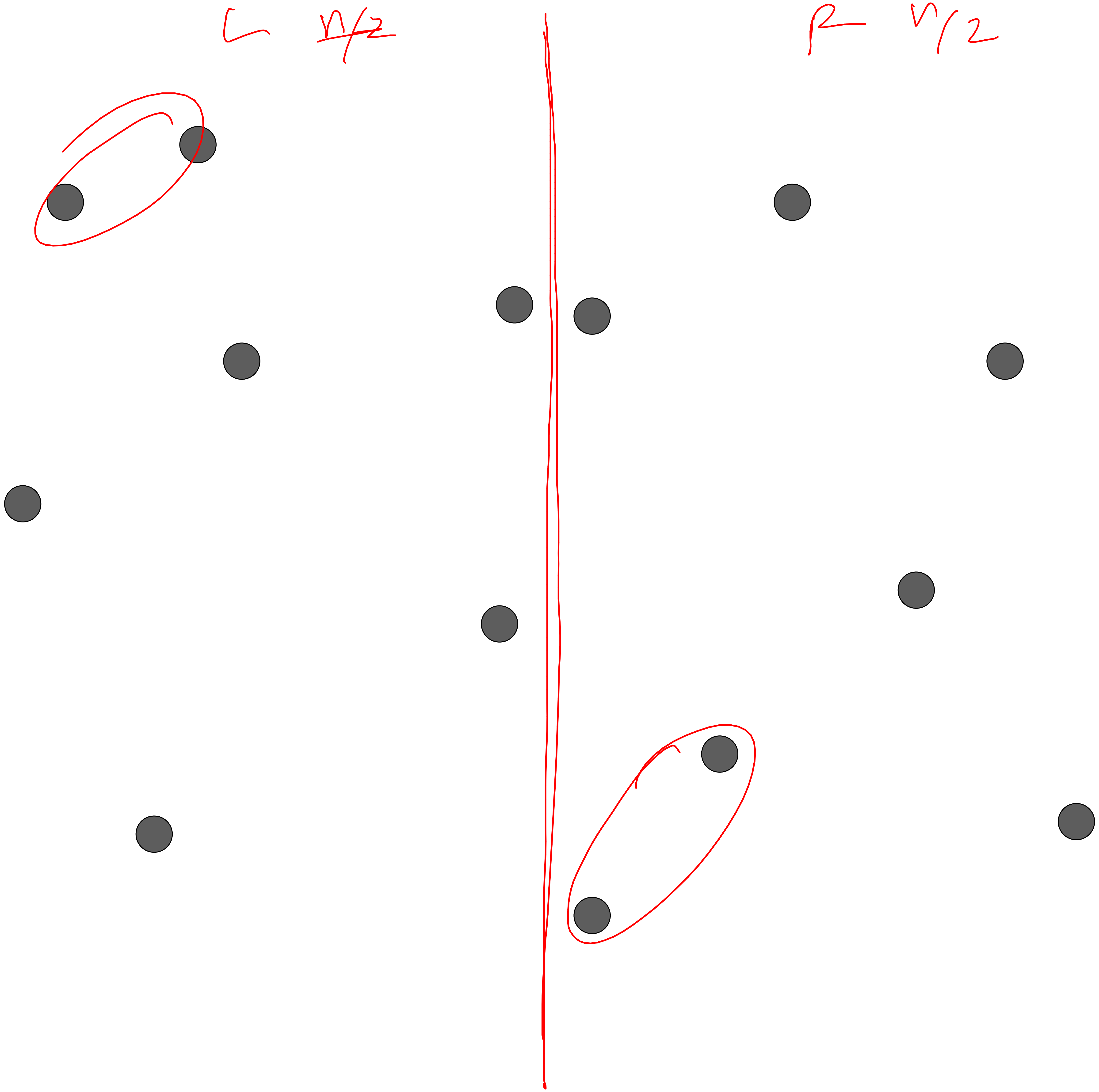
$\Theta(n^2)$

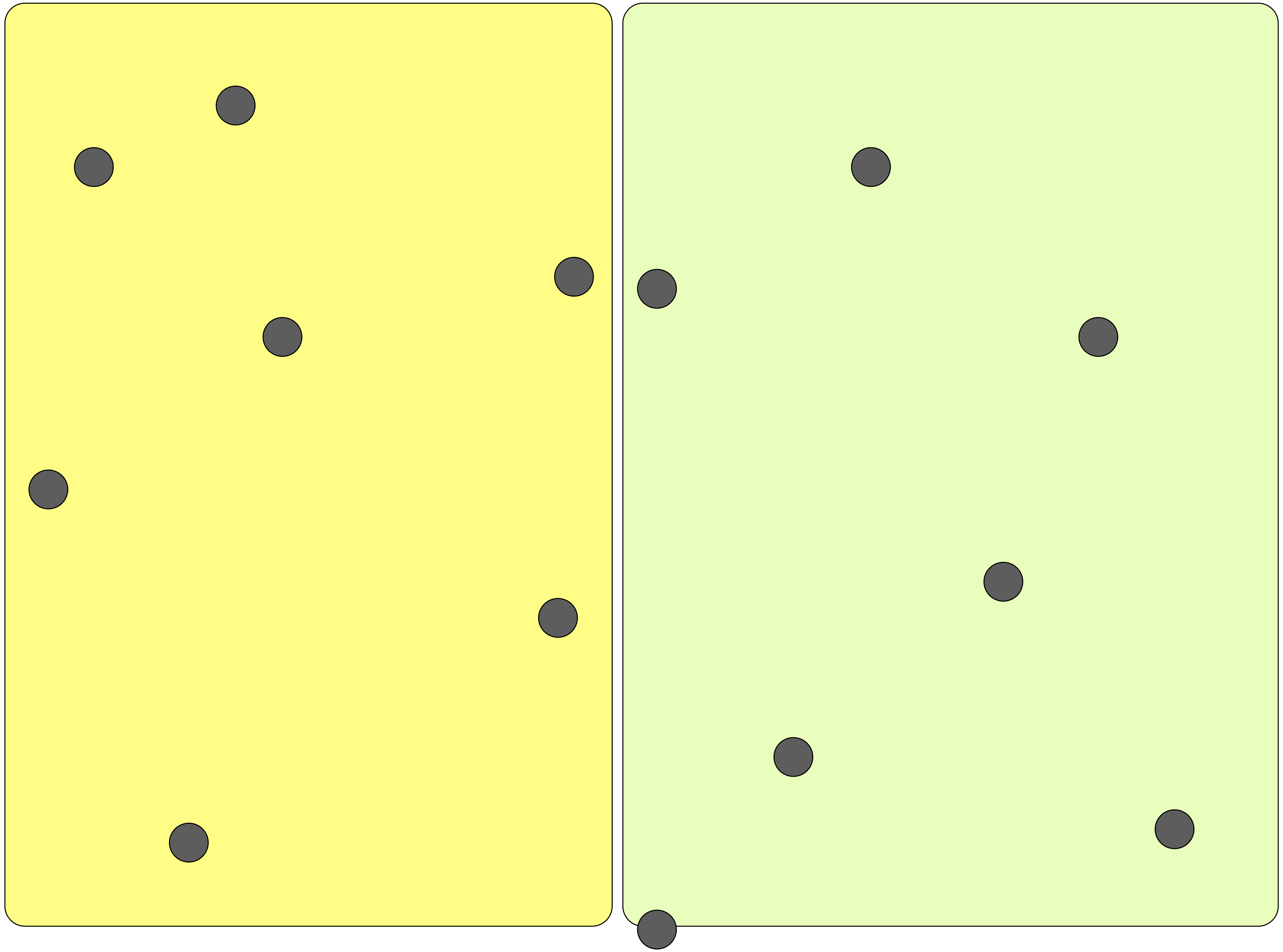
simple solution: brute force: $\Theta(n^2)$

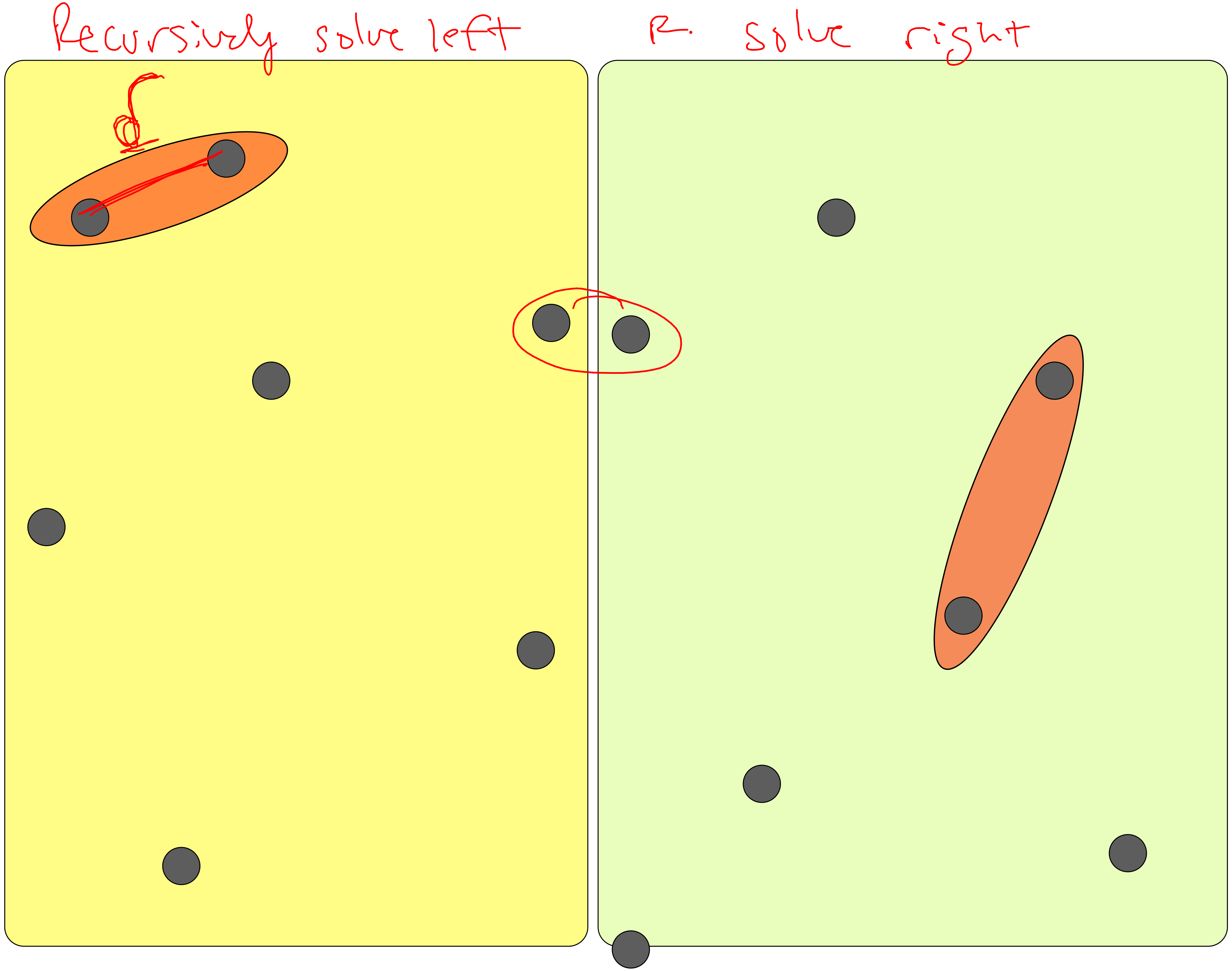


solve the large problem by

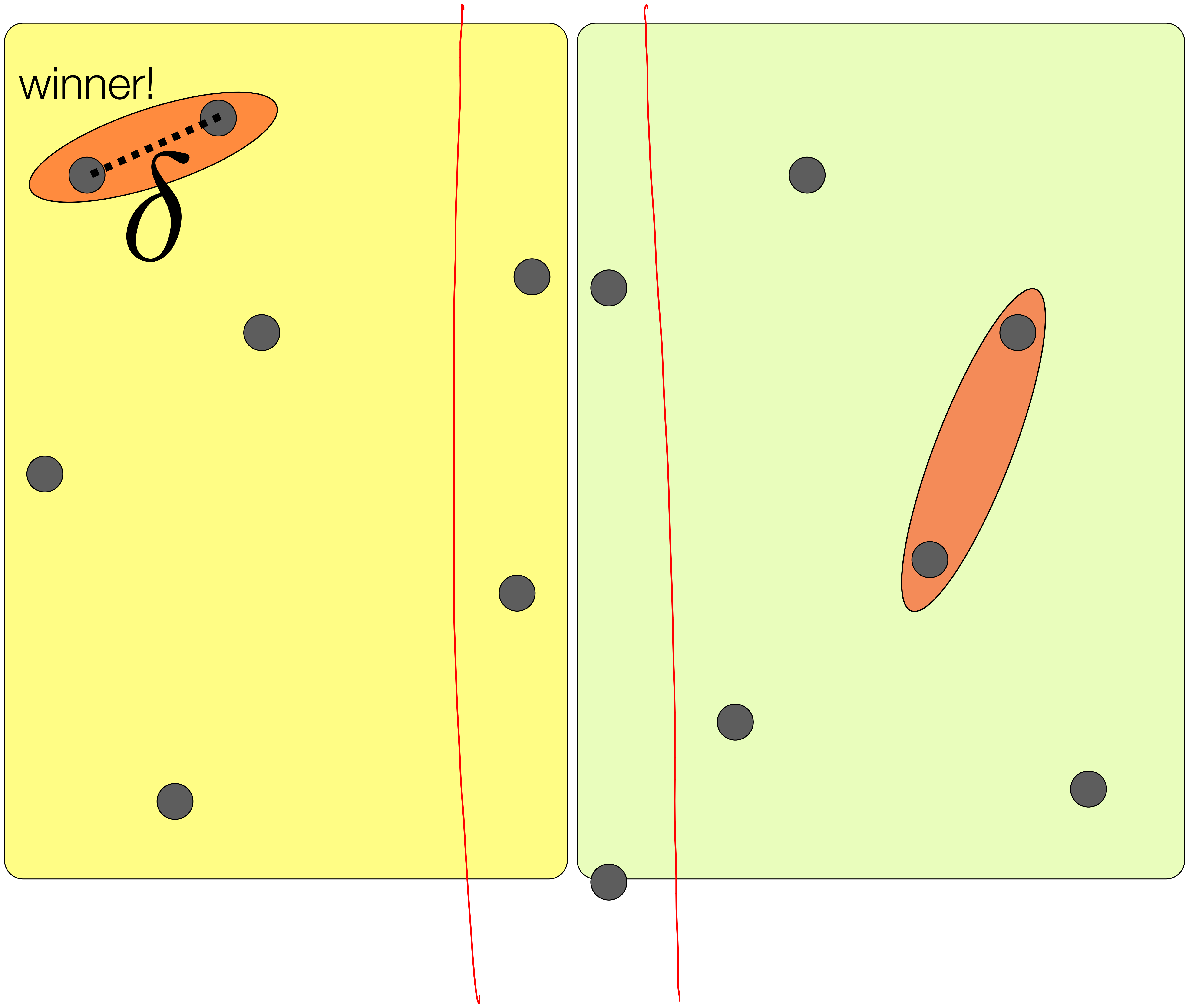
solving smaller problems
and combining solutions

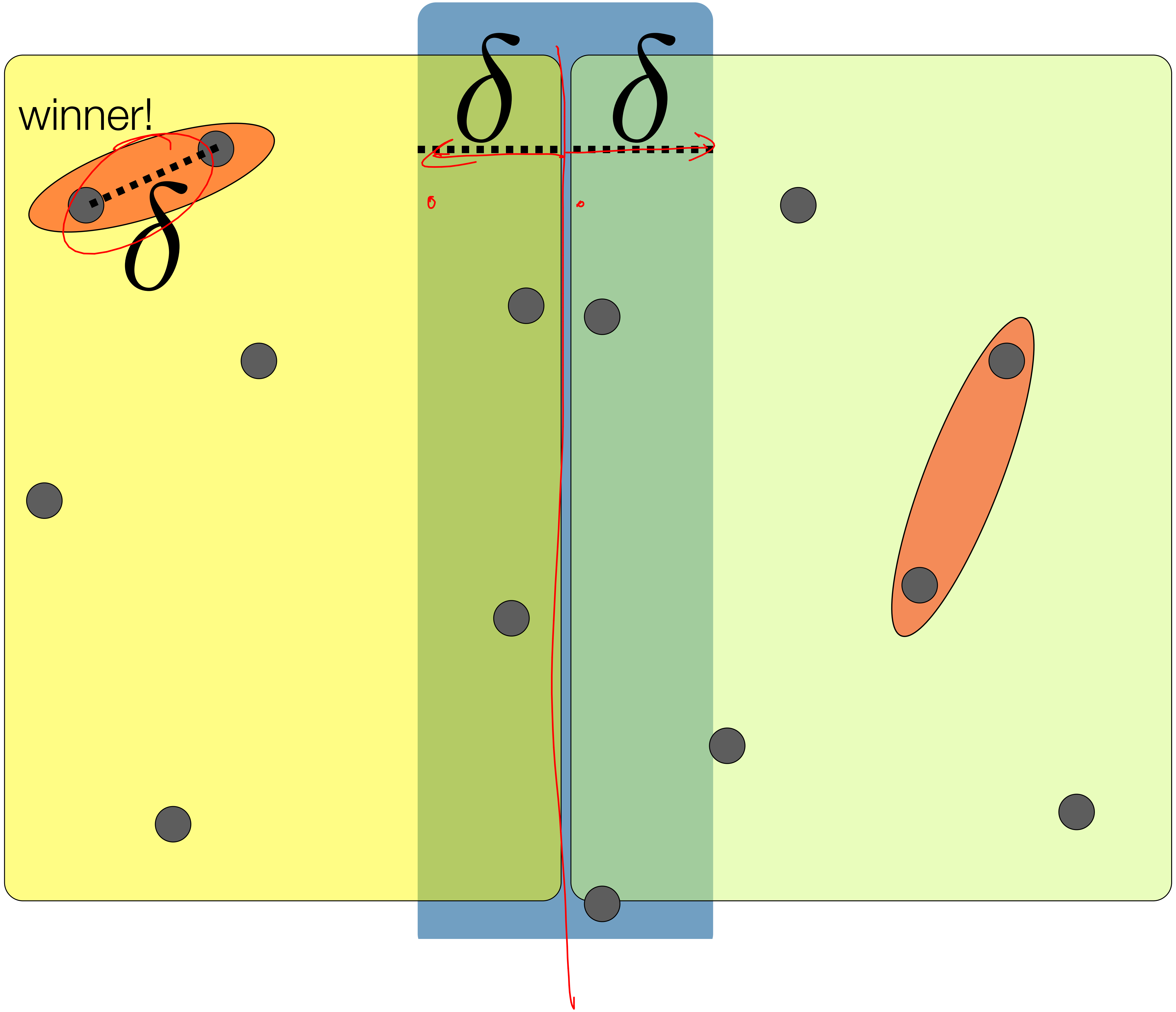

L n/2
R n/2

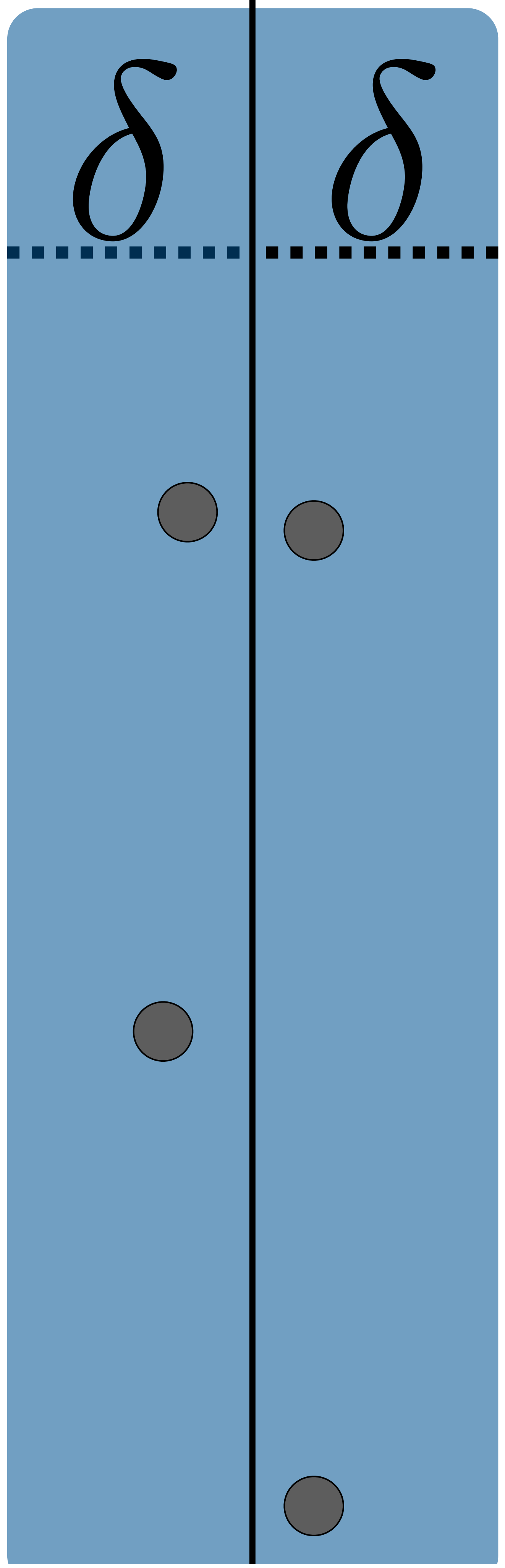


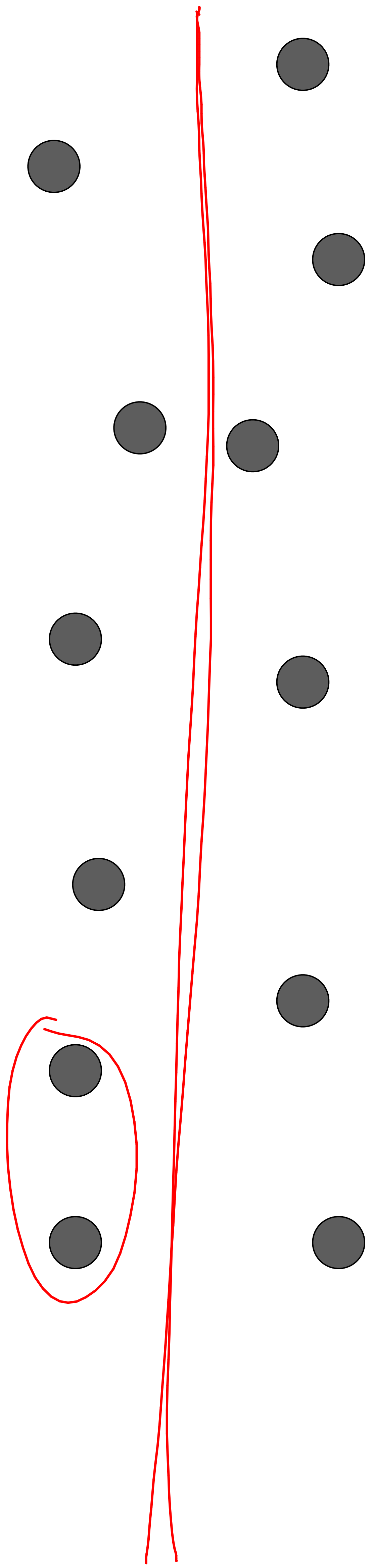


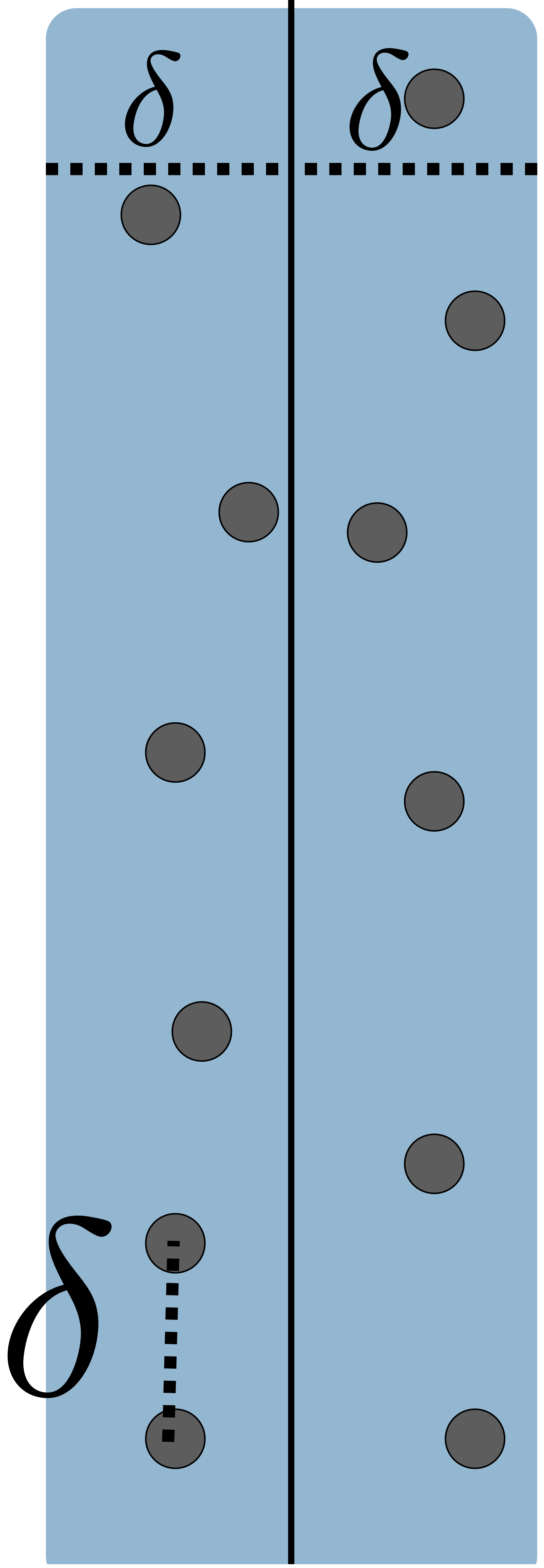
Recursively solve left
R. solve right
δ

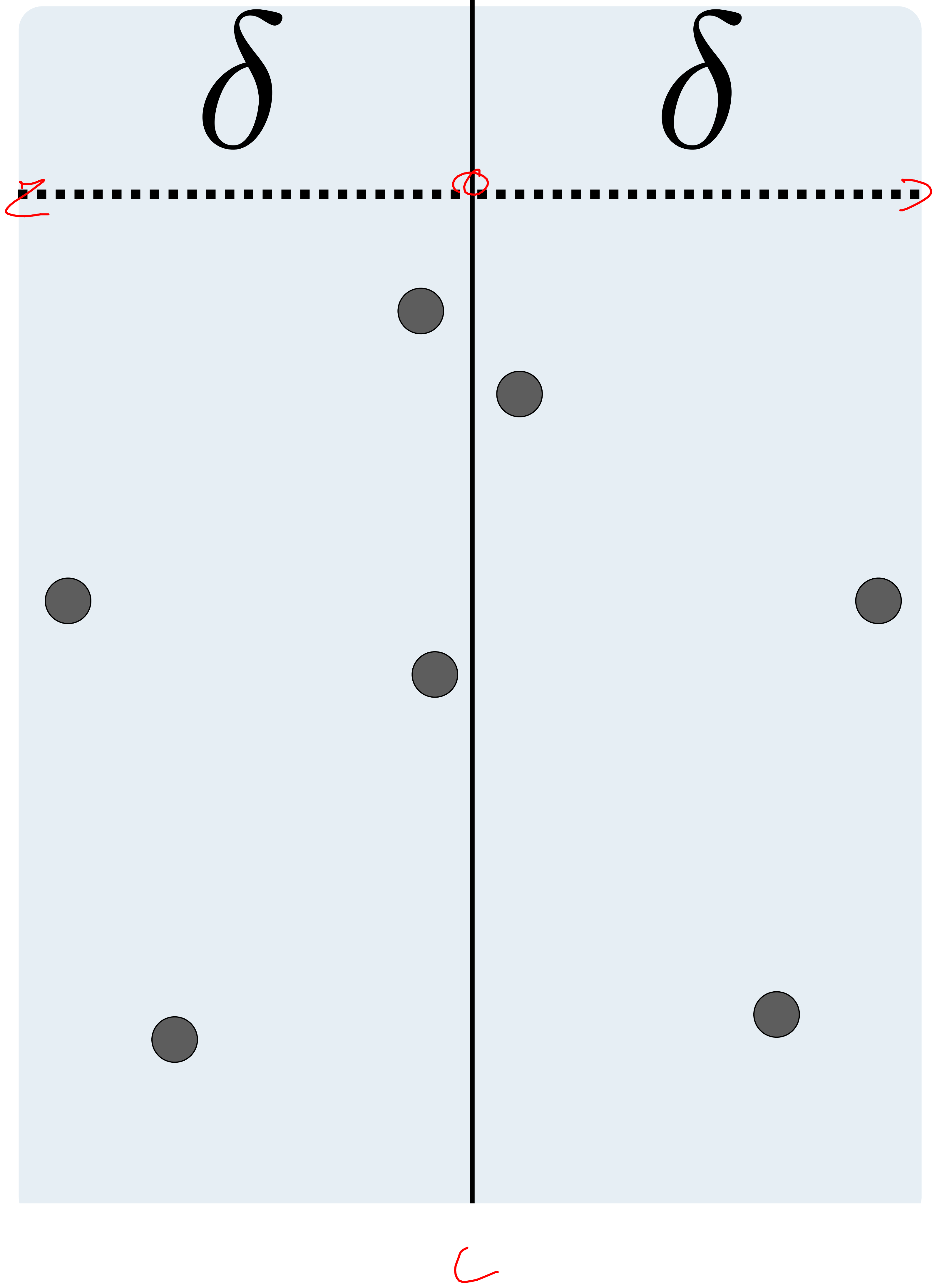

winner!
δ

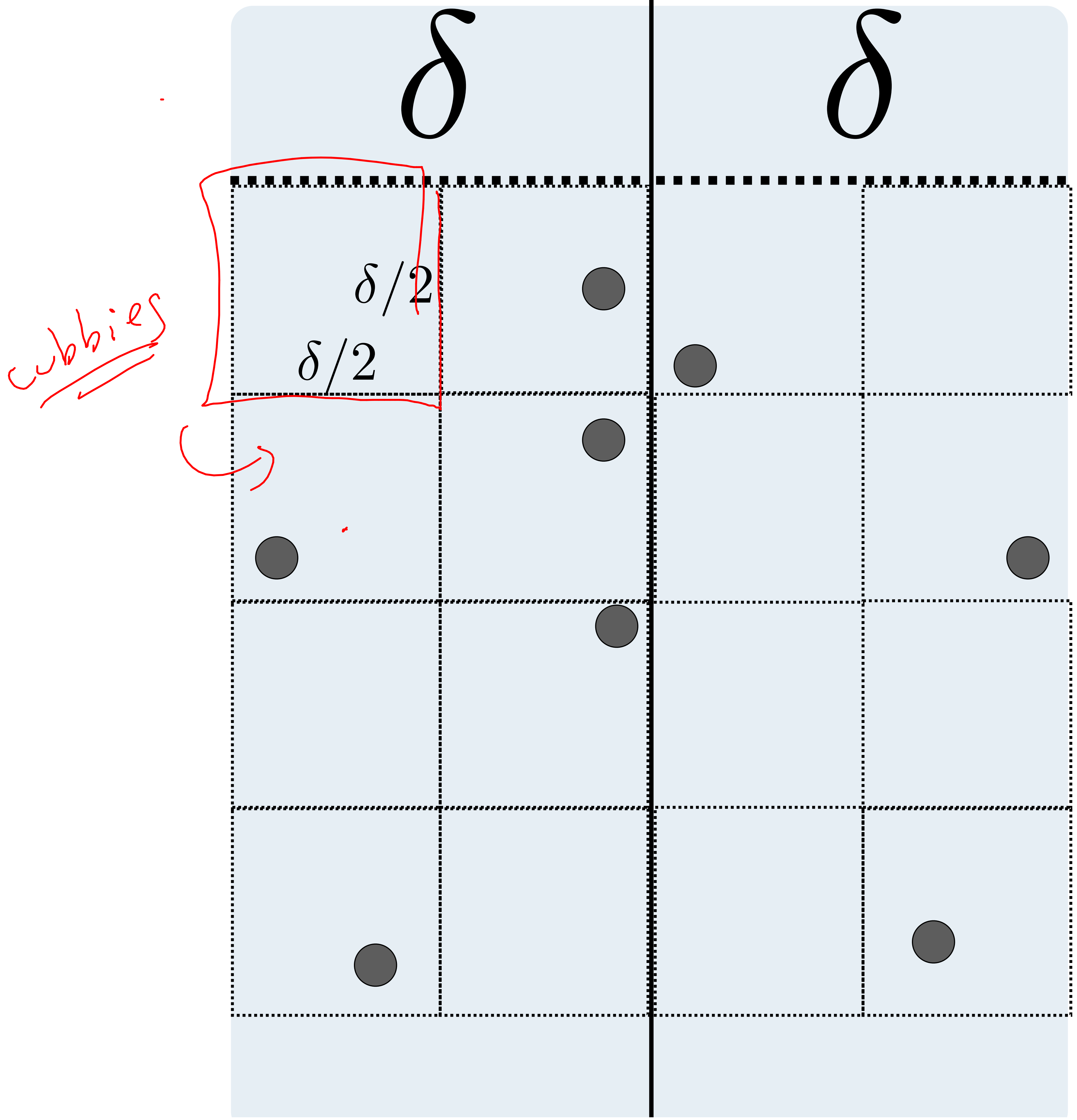

winner!
δ
δ
δ


δ
δ

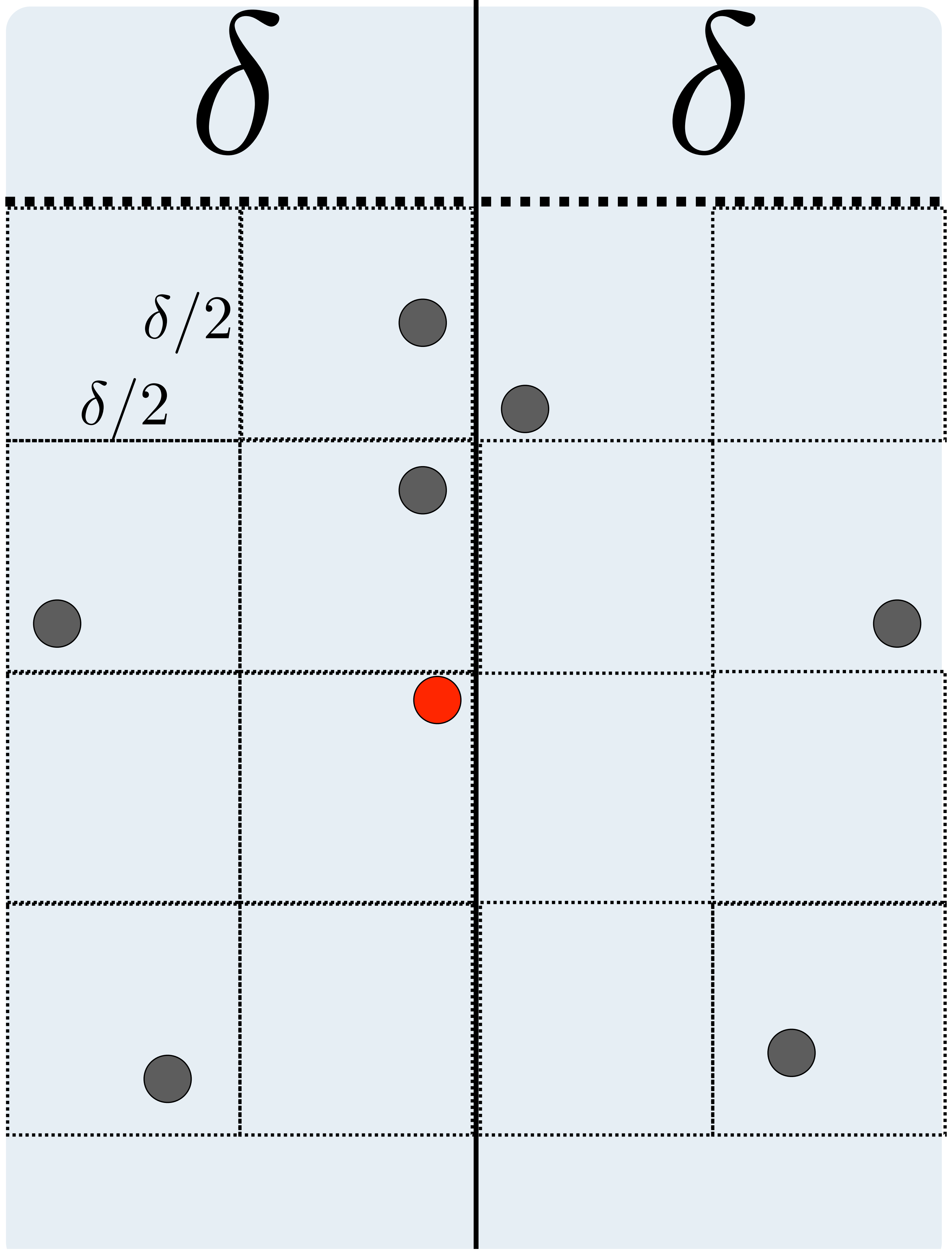



δ
δ
δ

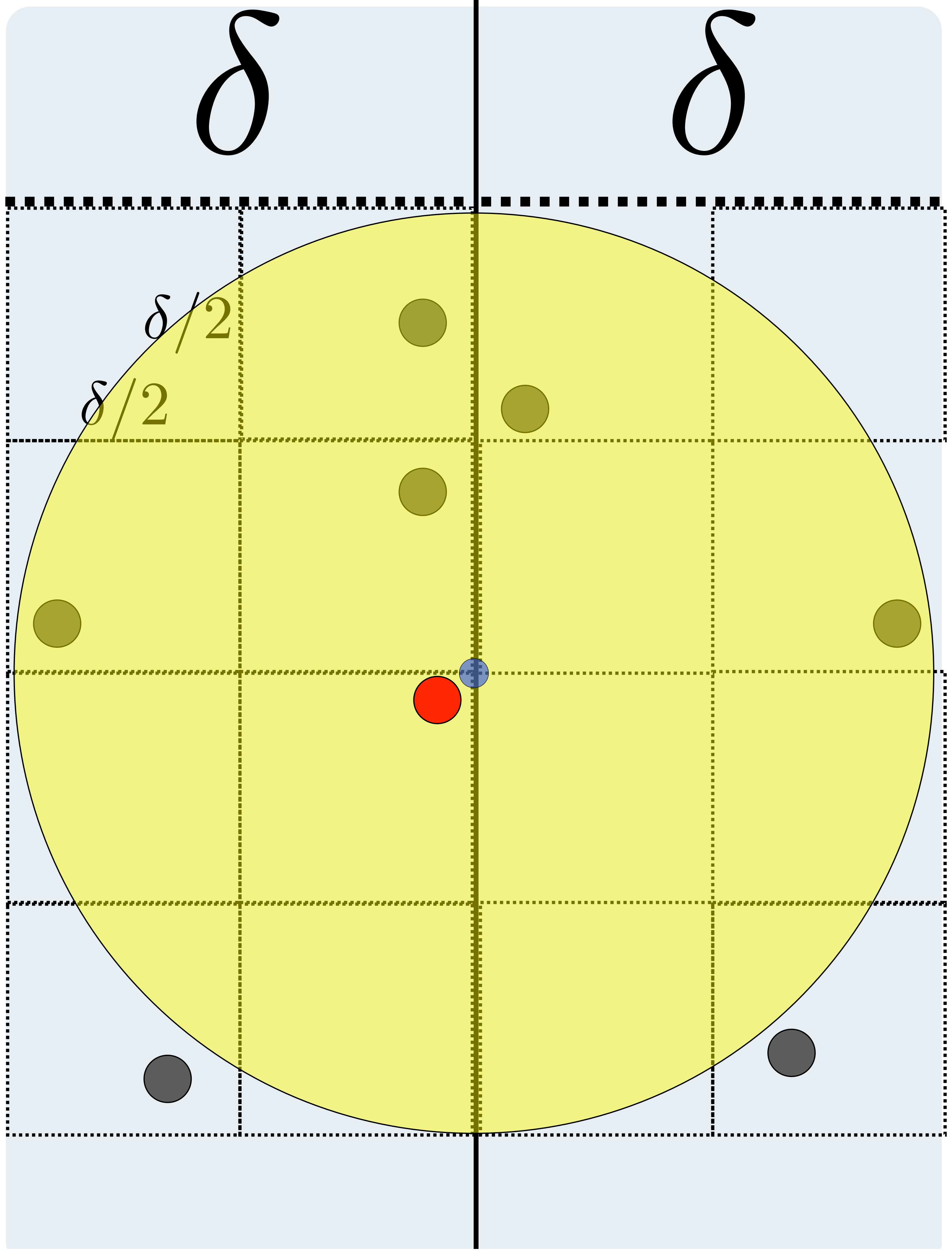

δ
δ



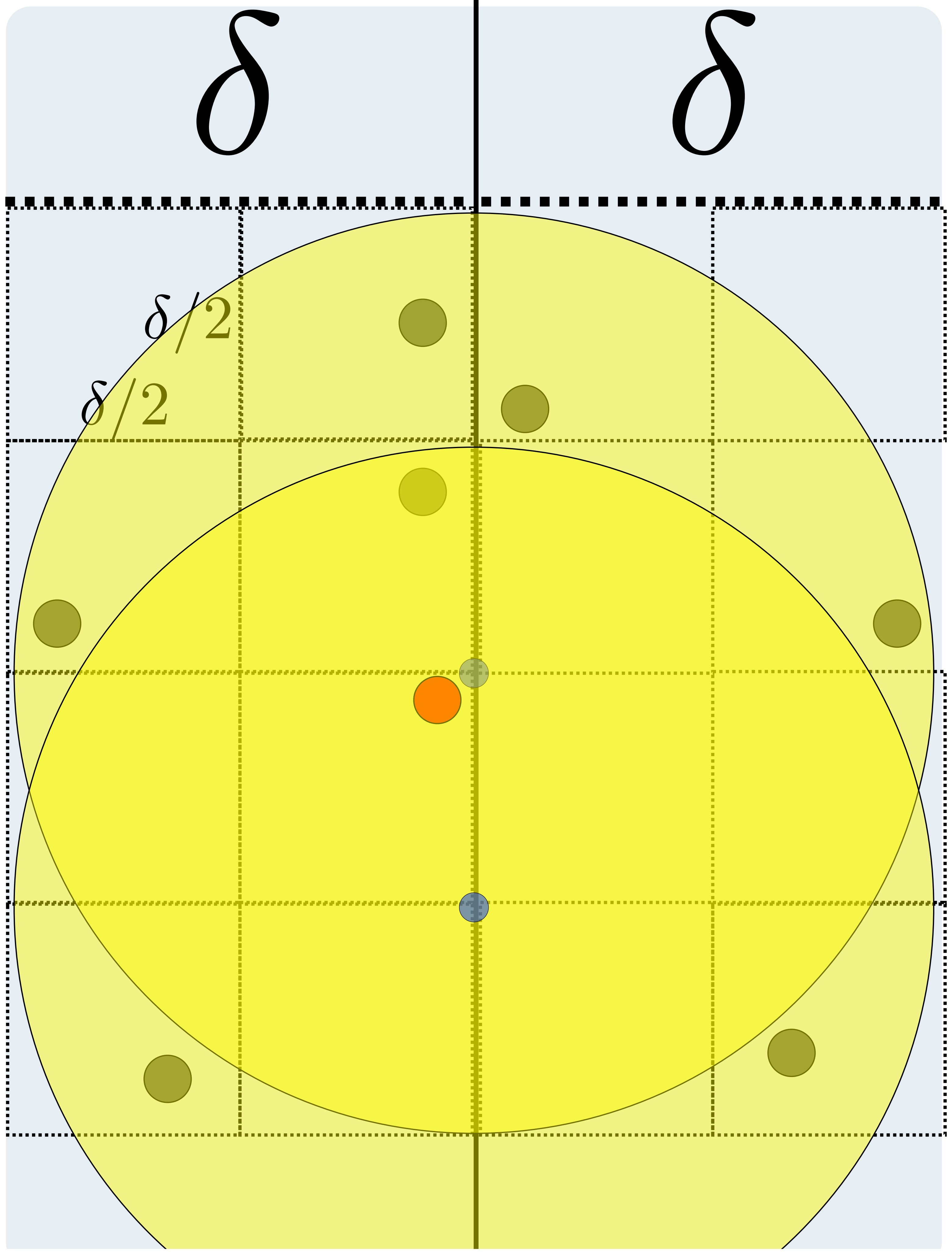
δ
δ
δ/2
δ/2
cubbies



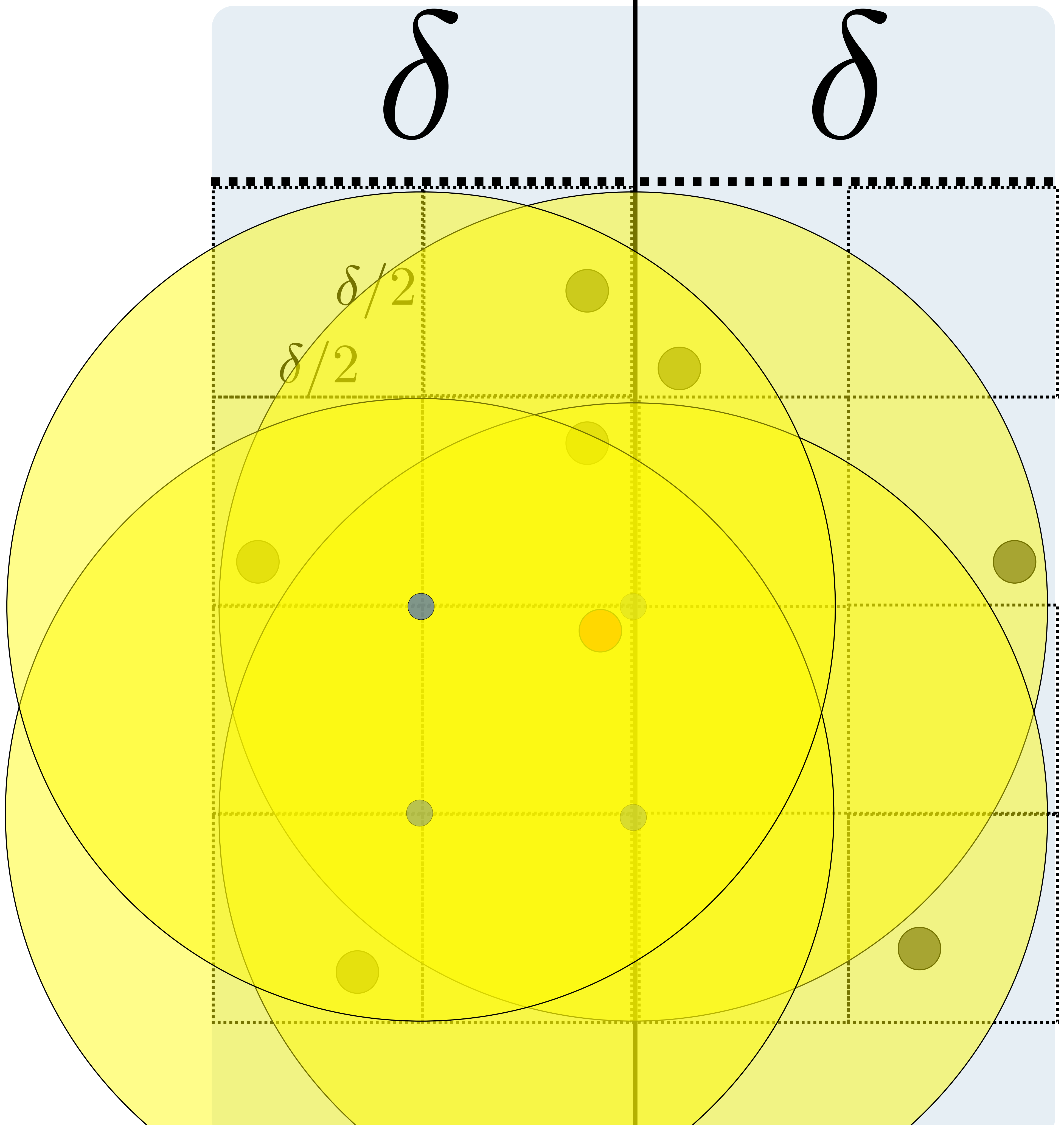
δ
δ
δ/2
δ/2



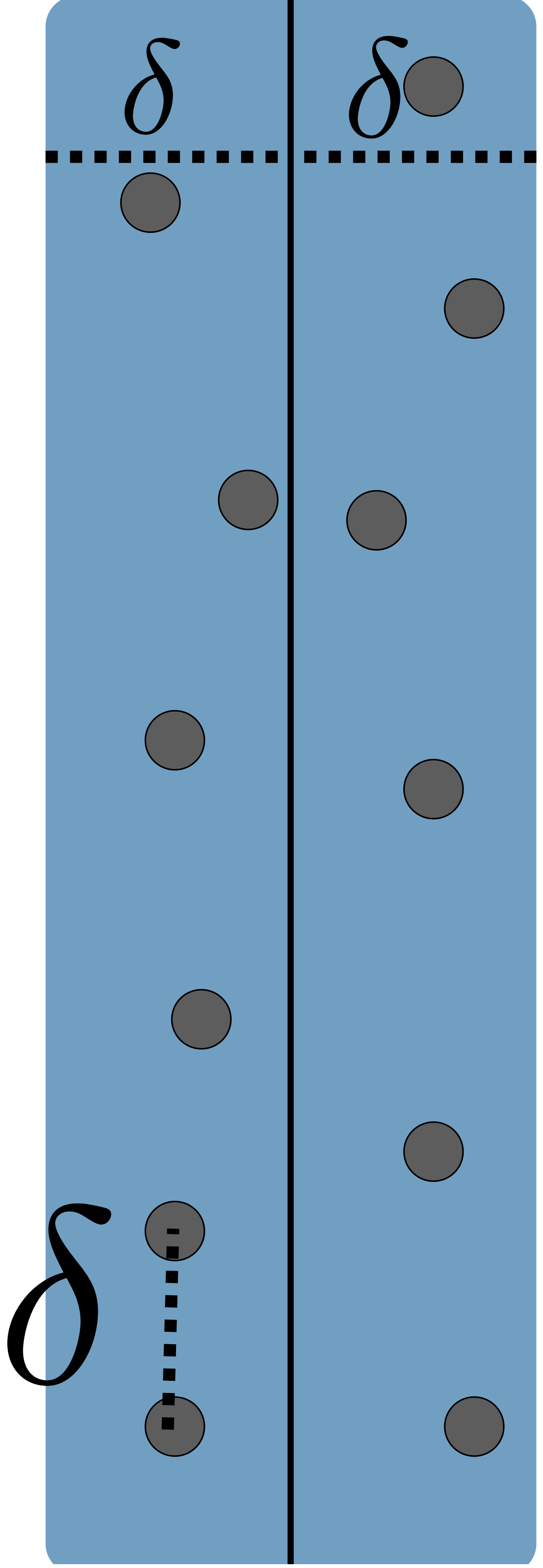
δ
δ
$\delta/2$
$\delta/2$



δ
δ
$\delta/2$
$\delta/2$



δ
δ
δ/2
δ/2



(make data structures, only once)

closest pair:

base case of <5 points

solve left half, right half

let δ be min from left/right

add points δ from middle to set S

assign points to boxes of side $\delta/2$

for each point in S,

compare w/10 neighbor boxes
find minimum in this list

return closest pair

$$T(n) =$$

$$T(n) = 2T(n/2) + \Theta(n) = \Theta(n \log n)$$

Matrix

multiplication

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \bigstar \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} =$$