

&

(Binary search)

“ : z [0,1,õ ,n-1]

“ : k

“ : k z[n/2]
k

z[0,õ ,n/2-1],
z[n/2,õ ,n-1]

“

$$T(n) = aT\left(\left\lfloor \frac{n}{b} \right\rfloor\right) + O(n^d) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + O(1) \quad (a=1, b=2, d=0)$$

“

,

O(log n)

(Binary search)

//

:

n

.

.

.

//

:

(Binary search)

“

»

3

«

:

.

.

.

”

.

,

.

,

,

:

.

(Binary search)

“

:

```
Αλγόριθμος Δυαδική_αναζήτηση
Δεδομένα // names, phones, onoma, arhi, telos //
meso ← [arhi + telos]/2
Αν onoma = names[meso] τότε
    Tel ← phones[meso]
αλλιώς
    Αν onoma < names[meso] τότε
        Δυαδική_αναζήτηση(names, phones, onoma, arhi, meso-1)
    αλλιώς
        Δυαδική_αναζήτηση(names, phones, onoma, meso+1, telos)
Τέλος_αν
Τέλος_αν
Αποτελέσματα // Tel //
Τέλος Δυαδική_αναζήτηση
```

(Binary search)

:

:

: 2101234567
: 2310252627
: 2610454647
: 2641013145
: 2261055869

0



1



2



3

(Binary search)

:

;

:

;

()

\Rightarrow

2

\tilde{O}

.

$$n/2^k=1 \Rightarrow n=2^k \Rightarrow k=\log_2 n \Rightarrow \log n = (\log n) .$$

\tilde{O}

“ : n

” ZHTOYMENO:

“ :
. : 5 1 6 9 4 8
. : 1 4 5 6 8 9

”

:

- .
- .
- .
- .
- .
- .
- .
- .
- .
- .
- .
- .
- .
- .
- .

MP3

blog

PageRank

Google

-

(. . , mailing list)

Mergesort:

" : 2 ,
,

" (**John von Neumann 1945**):

```
function mergesort ( $a[1..n]$ )
```

```
Input: An array of numbers  $a[1..n]$ 
```

```
Output: A sorted version of this array
```

```
if  $n > 1$ :
```

```
    return merge (mergesort ( $a[1.. \lfloor n/2 \rfloor]$ ) , mergesort ( $a[\lfloor n/2 \rfloor + 1..n]$ ))
```

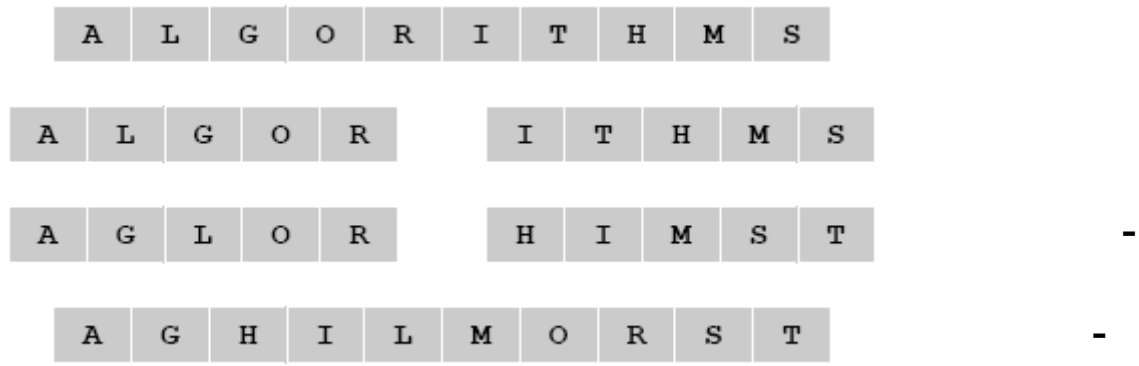
```
else:
```

```
    return  $a$ 
```

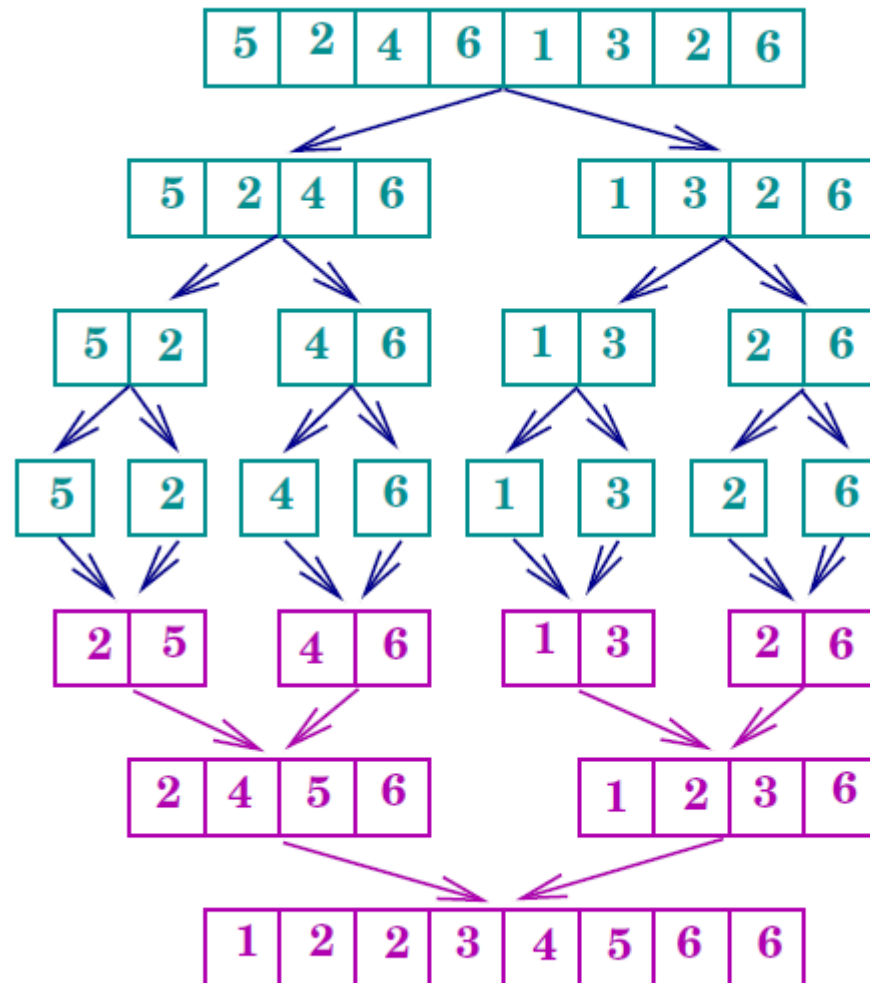


Mergesort:

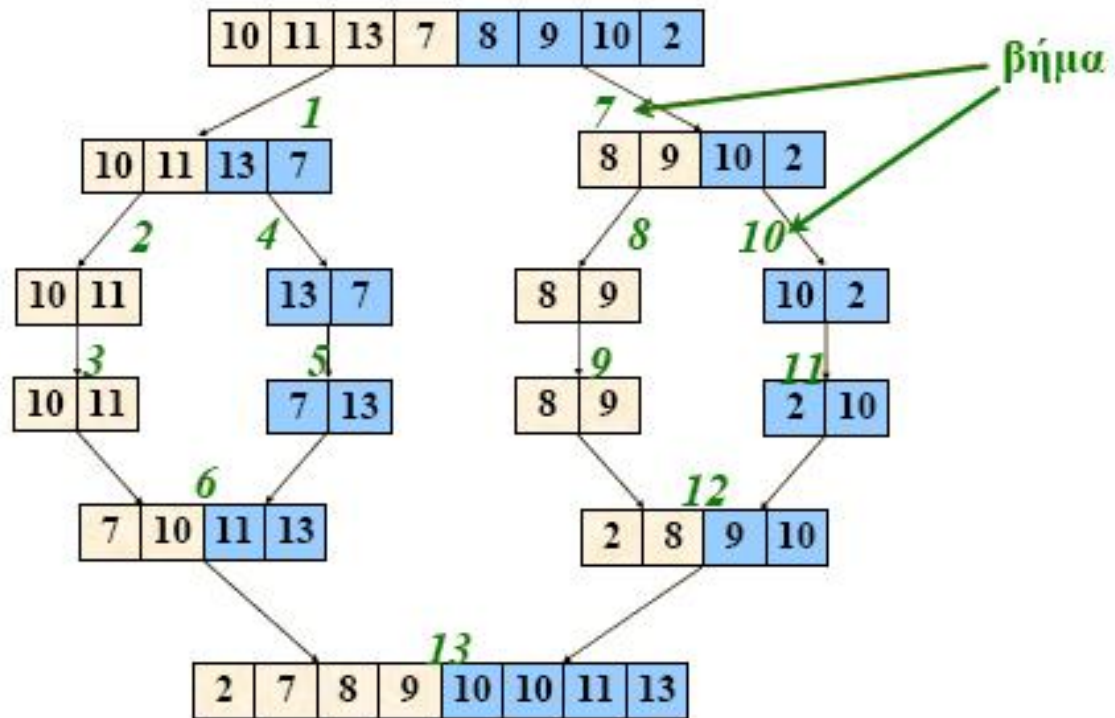
“ : $N (N / 2)$
“ : 2
“ : MergeSort



Mergesort:



Mergesort:



Mergesort:

“

;

.

\tilde{O} : MergeSort²

:

.

“

n

$O(n)$

Mergesort:

" $T(n)$

"

. $O(n)$

. $T(n/2)$

. $O(n)$

()

n

:

$n/2$

"

$T(n)$

Mergesort

(recurrence relation):

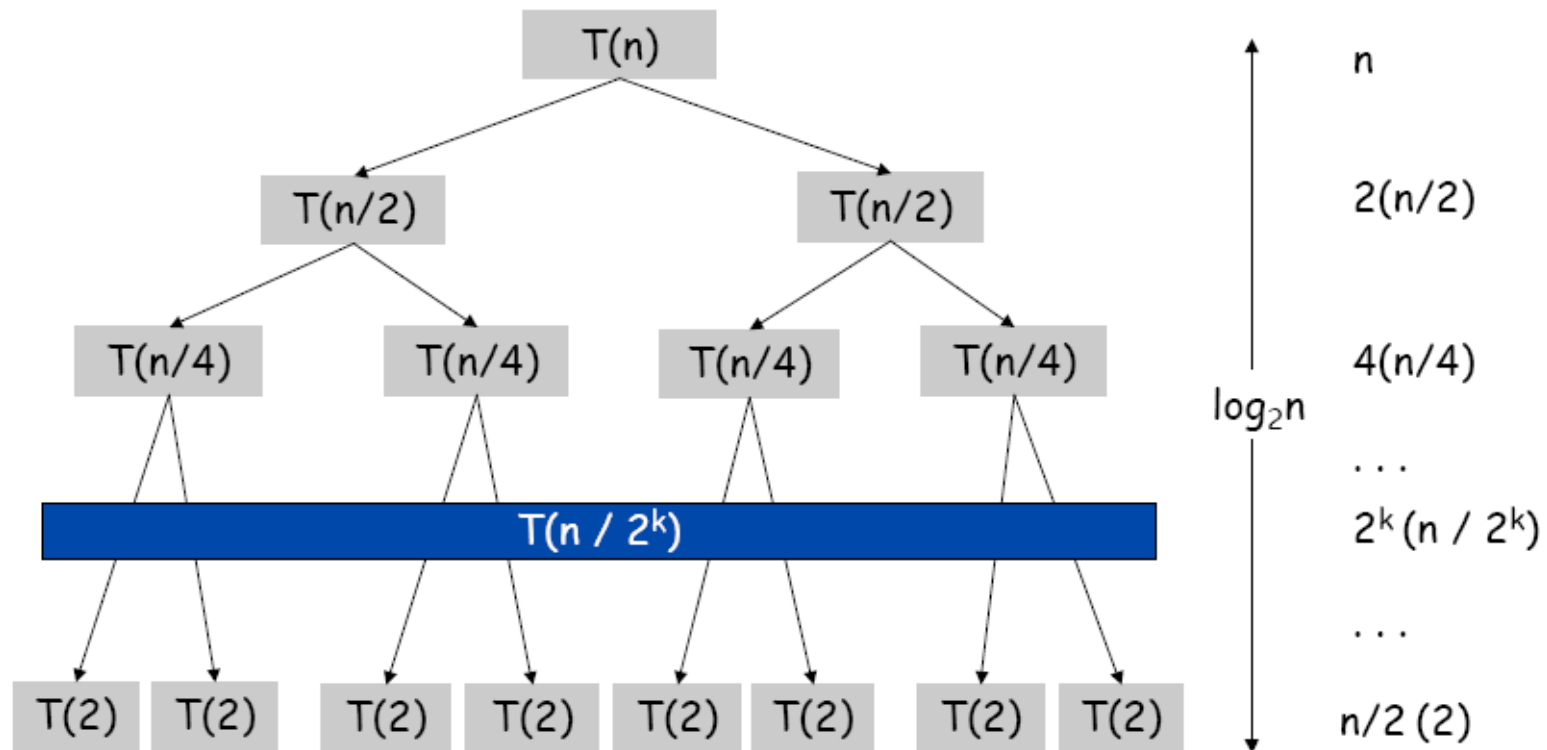
. $c, T(n) = 2T(n/2) + cn$

$n > 2 \quad T(n) = mcn$

Mergesort:

$$T(n) \leq \begin{cases} 0 & \text{if } n = 1 \\ \underbrace{T(\lfloor n/2 \rfloor)}_{\text{solve left half}} + \underbrace{T(\lfloor n/2 \rfloor)}_{\text{solve right half}} + \underbrace{n}_{\text{merging}} & \text{otherwise} \end{cases}$$

$$T(n) = \begin{cases} 0 & \text{if } n = 1 \\ \underbrace{2T(n/2)}_{\text{sorting both halves}} + \underbrace{n}_{\text{merging}} & \text{otherwise} \end{cases}$$

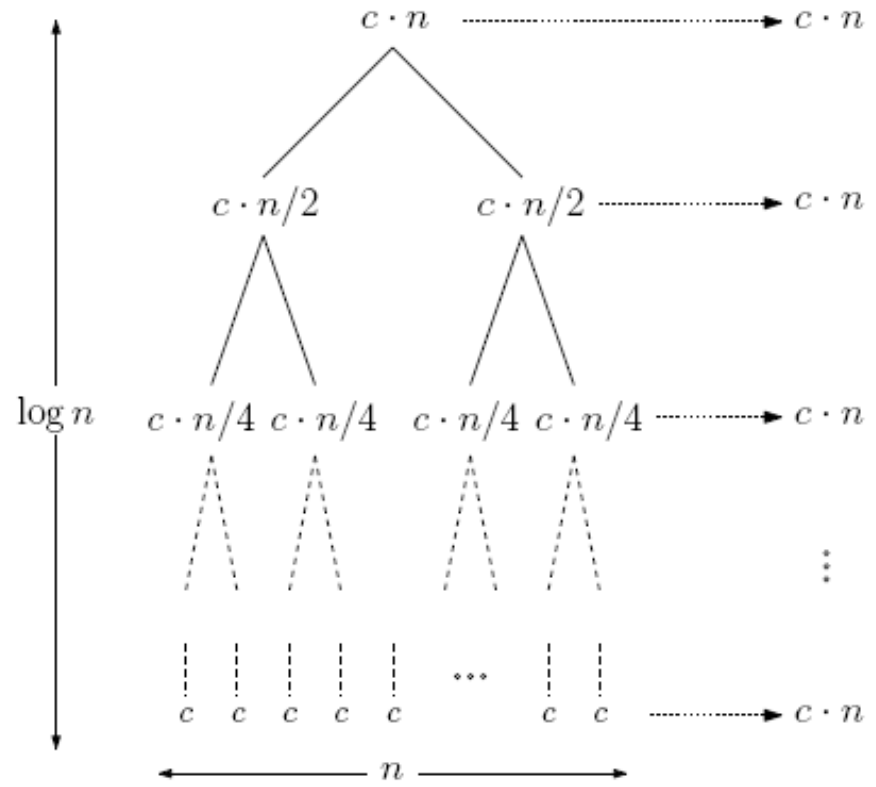


$$n/2^k=2 \Rightarrow n=2^{k+1} \Rightarrow \log n=k+1$$

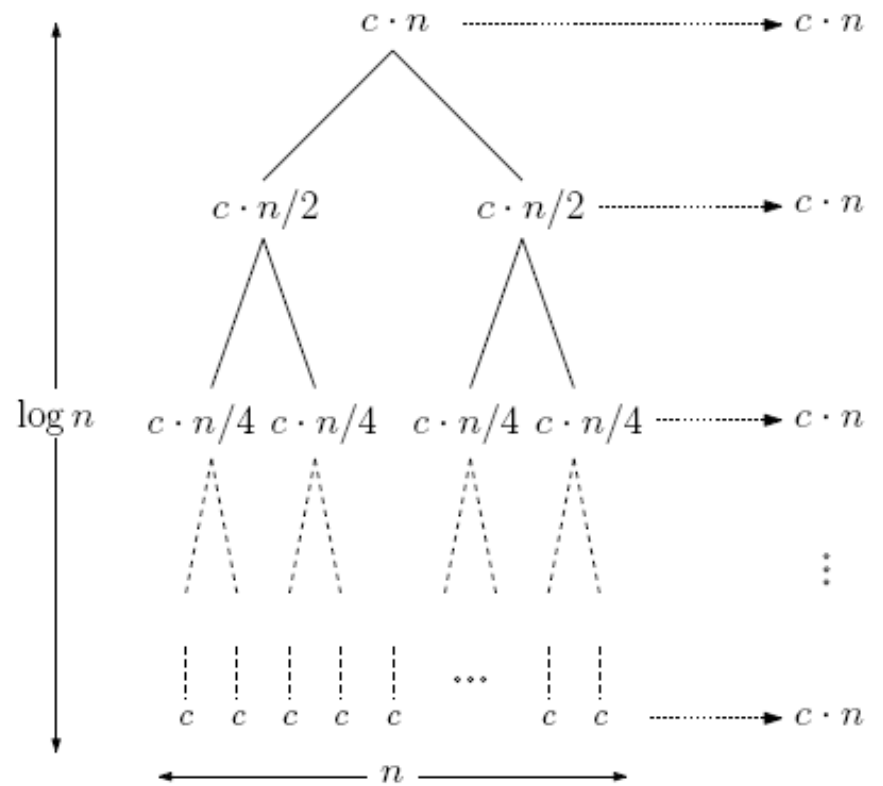
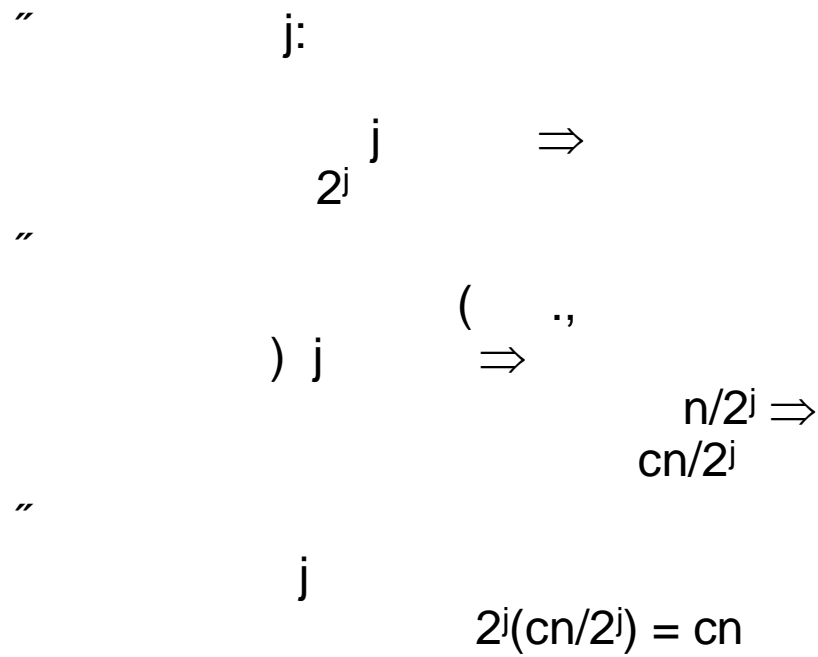
$$n \log_2 n$$

Mergesort:

" 1 : 1 ($=2^0$) $n \Rightarrow$
 $cn +$
 " 2 : 2 ($=2^1$) $n/2$
 $cn/2$
 \Rightarrow : $cn +$
 " 3 : 4 ($=2^2$) $n/4$
 $cn/4$
 \Rightarrow : $cn +$



Mergesort:



Mergesort:

"

$$n/2^k = 2 \Rightarrow n = 2^{k+1} \Rightarrow \log_2 n = k+1$$

"

$$cn \log n$$

$$O(n \log n)$$

