



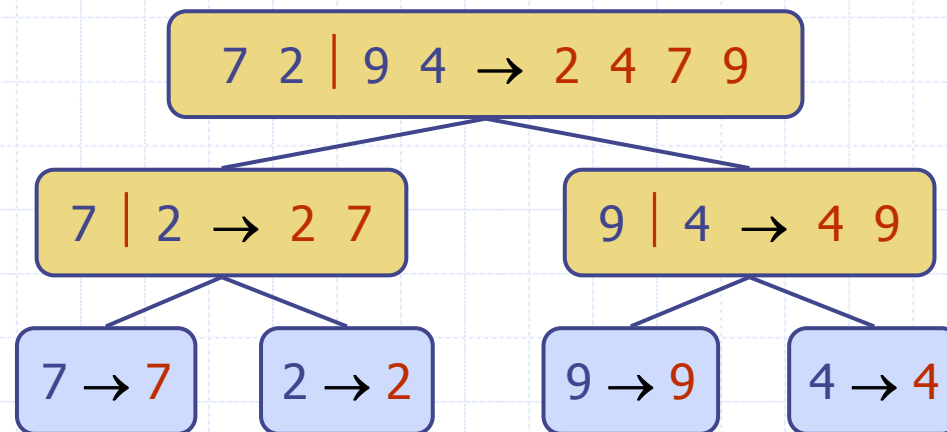
ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

Αλγόριθμοι και πολυπλοκότητα

Συγχωνευτική Ταξινόμηση

Ιωάννης Τόλλης
Τμήμα Επιστήμης Υπολογιστών

Συγχωνευτική Ταξινόμηση (Merge Sort)



Περίληψη και ανάγνωση

- ◆ Παράδειγμα διαίρει και κυρίευε
- ◆ Συγχωνευτική Ταξινόμηση (Merge-sort)
 - Αλγόριθμος
 - Συγχωνεύοντας δύο ταξινομημένες ακολουθίες
 - Δέντρο της Συγχωνευτικής Ταξινόμησης
 - Παράδειγμα εκτέλεσης
 - Ανάλυση
- ◆ Γενική συγχώνευση και σύνολο λειτουργιών
- ◆ Περίληψη αλγορίθμων ταξινόμησης

Διαίρει και κυρίευε

- ◆ Ο διαίρει και κυρίευε είναι ένα γενικό παράδειγμα σχεδίασης αλγορίθμου :
 - **Διαίρει**: διαίρεσε τα δεδομένα εισόδου S σε δύο αποσυνδεδεμένα υποσύνολα S_1 και S_2
 - **Επανάλαβε**: λύσε τα υπο-προβλήματα που σχετίζονται με τα S_1 και S_2
 - **Κυρίευε**: συνδύασε τις λύσεις για το S_1 και S_2 σε μια λύση για το S
- ◆ Η ταπεινή περίπτωση για αναδρομή είναι τα υπο-προβλήματα μεγέθους 0 ή 1
- ◆ Η ταξινόμηση συγχώνευσης είναι ένας αλγόριθμος ταξινόμησης που βασίζεται στο παράδειγμα διαίρει και κυρίευε
- ◆ Σαν ταξινομημένος σωρός
 - Χρησιμοποιεί έναν comparator
 - Έχει $O(n \log n)$ χρόνο εκτέλεσης
- ◆ Όχι σαν ταξινομημένος σωρός
 - Δεν χρησιμοποιεί βοηθητική ουρά προτεραιότητας
 - Προσπελάσει τα δεδομένα με έναν διαδοχικό τρόπο (κατάλληλο για ταξινόμηση δεδομένων σε έναν δίσκο)

Συγχωνευτική Ταξινόμηση (Merge-sort)

- ◆ Η Συγχωνευτική Ταξινόμηση σε μία ακολουθία εισόδου S με n στοιχεία αποτελείται από τρία βήματα:
 - **Διαίρει** : διαχωρισμός της S σε δύο ακολουθίες S_1 και S_2 με $n/2$ στοιχεία περίπου η κάθε μία
 - **Επανάλαβε** : επαναληπτικά ταξινομήσε τις S_1 και S_2
 - **Κυρίευε** : συγχώνευσε τις S_1 και S_2 σε μία μοναδική ταξινομημένη πρόταση

```
Αλγόριθμος mergeSort(S, C)  
  Είσοδος ακολουθία  $S$  με  $n$   
           στοιχεία, comparator  $C$   
  Έξοδος ακολουθία  $S$   
           ταξινομημένη σύμφωνα με το  $C$   
  if  $S.size() > 1$  ( $S_1, S_2$ ) ←  
    partition(S, n/2)  
    mergeSort(S_1, C)  
    mergeSort(S_2, C)  
     $S \leftarrow merge(S_1, S_2)$ 
```

Συγχωνεύοντας δύο ταξινομημένες ακολουθίες

- ◆ Το βήμα κυρίως της ταξινόμησης της συγχώνευσης αποτελείται από την συγχώνευση δύο ταξινομημένων ακολουθιών A και B σε με ταξινομημένη ακολουθία S που περιέχει την ένωση των στοιχείων της A και της B
- ◆ Συγχωνεύοντας δύο ταξινομημένες ακολουθίες, με $n/2$ στοιχεία η κάθε μία και χρησιμοποιώντας διπλά συνδεδεμένη λίστα, παίρνει χρόνο $O(n)$

Αλγόριθμος *merge*(A, B)

Είσοδος ακολουθίες A και B με $n/2$ στοιχεία η κάθε μία

Έξοδος ταξινομημένη ακολουθία $A \cup B$

$S \leftarrow$ κενή ακολουθία

while $\neg A.isEmpty() \wedge \neg B.isEmpty()$

if $A.first().element() < B.first().element()$

$S.insertLast(A.remove(A.first()))$

else

$S.insertLast(B.remove(B.first()))$

while $\neg A.isEmpty()$

$S.insertLast(A.remove(A.first()))$

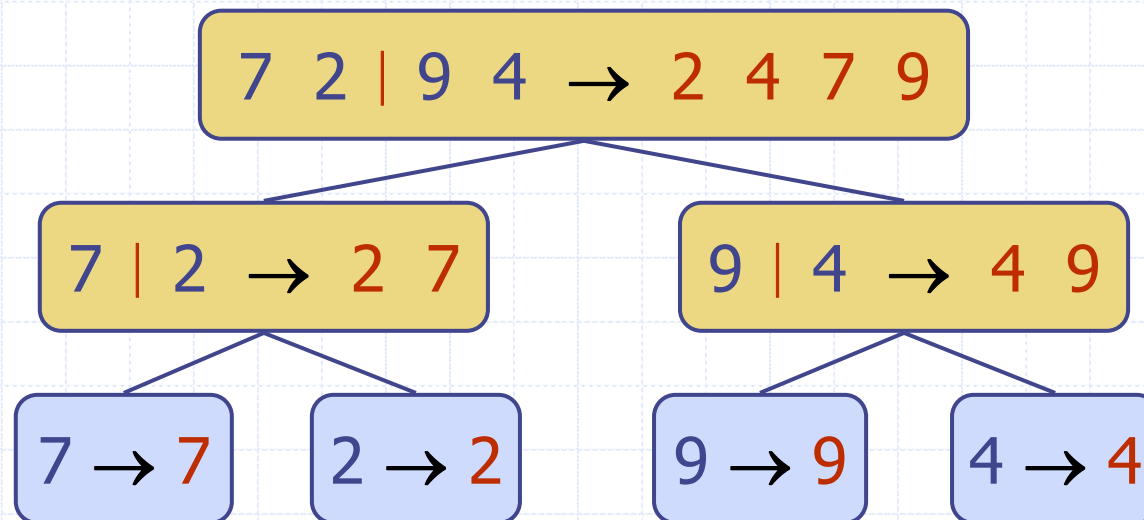
while $\neg B.isEmpty()$

$S.insertLast(B.remove(B.first()))$

return S

Δέντρο Συγχωνευτικής Ταξινόμησης (Merge-Sort Tree)

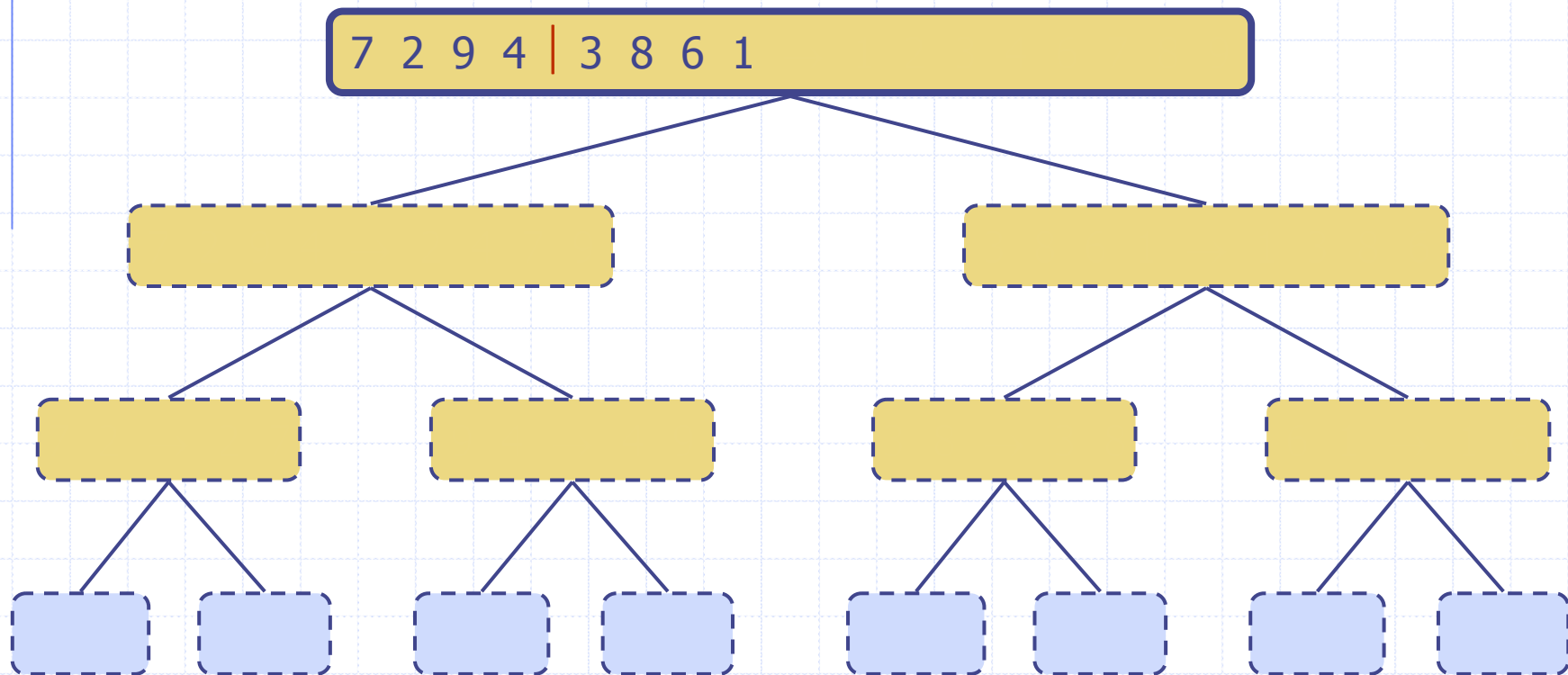
- ◆ Μία εκτέλεση της ταξινόμησης συγχώνευσης απεικονίζεται από ένα δυαδικό δέντρο
 - Κάθε κόμβος αναπαριστά μία αναδρομική κλήση της ταξινόμησης συγχώνευσης και κρατά
 - ◆ Μη ταξινομημένες ακολουθίες πριν την εκτέλεση και τον διαχωρισμό του
 - ◆ Ταξινομημένες ακολουθίες στο τέλος της εκτέλεσης
 - Η ρίζα είναι η αρχική κλήση
 - Τα φύλλα είναι κλήσεις υπο-ακολουθιών μεγέθους 0 ή 1



Συγχωνευτική ταξινόμηση

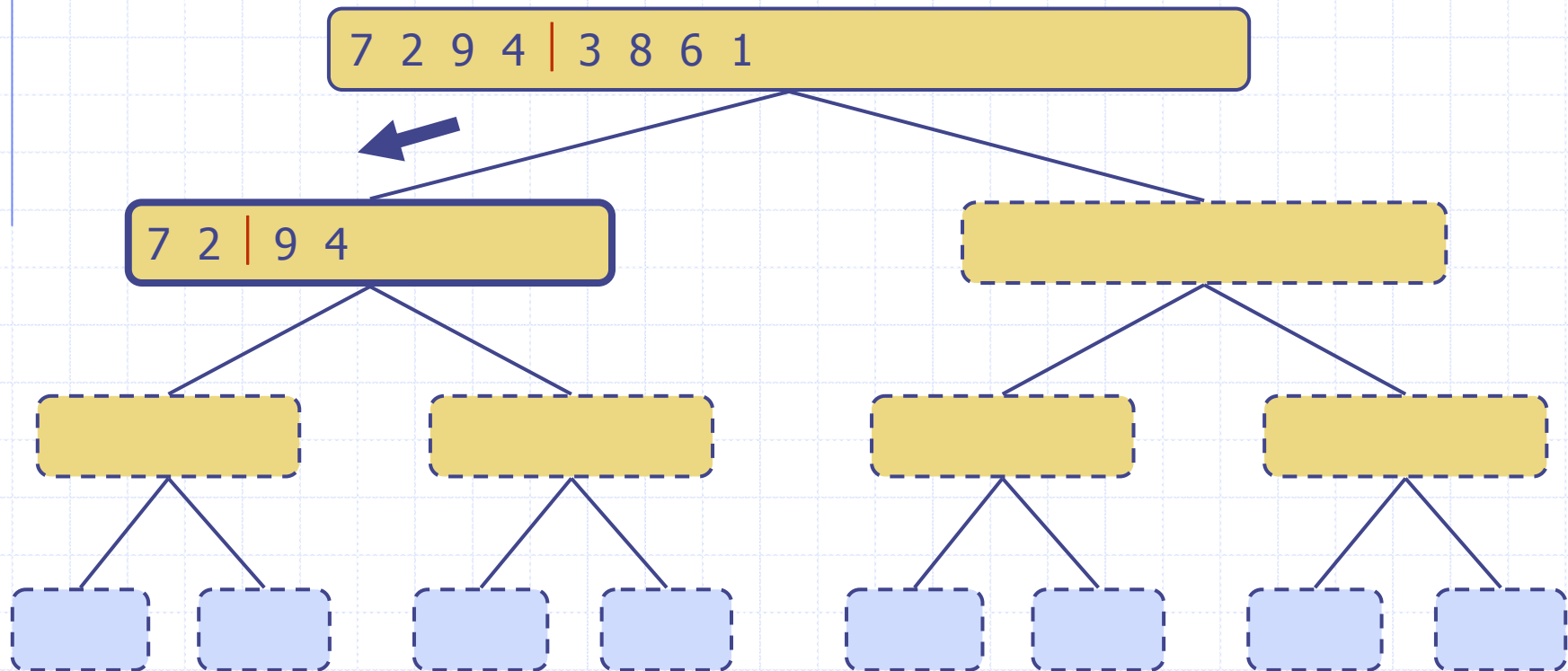
Παράδειγμα εκτέλεσης

◆ Διαχωρισμός



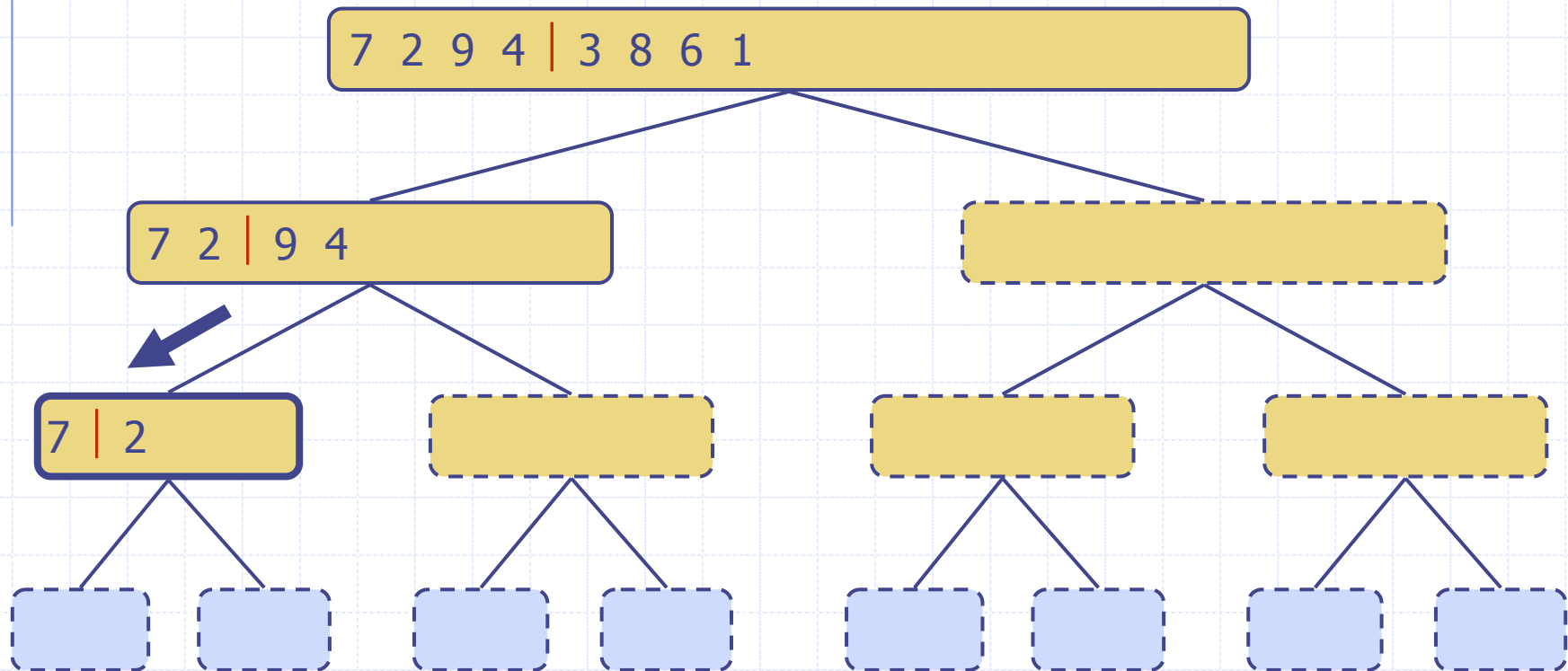
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Αναδρομική κλήση, διαχωρισμός



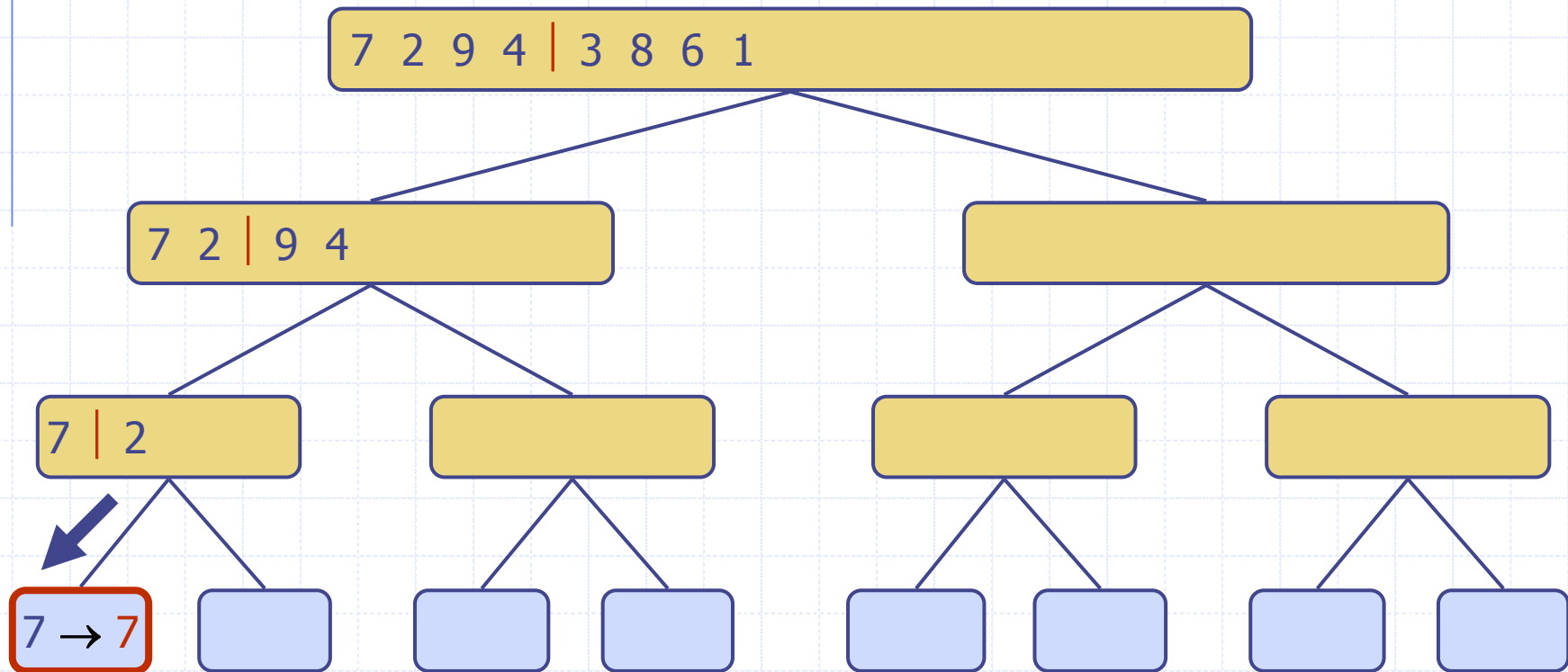
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Αναδρομική κλήση, διαχωρισμός



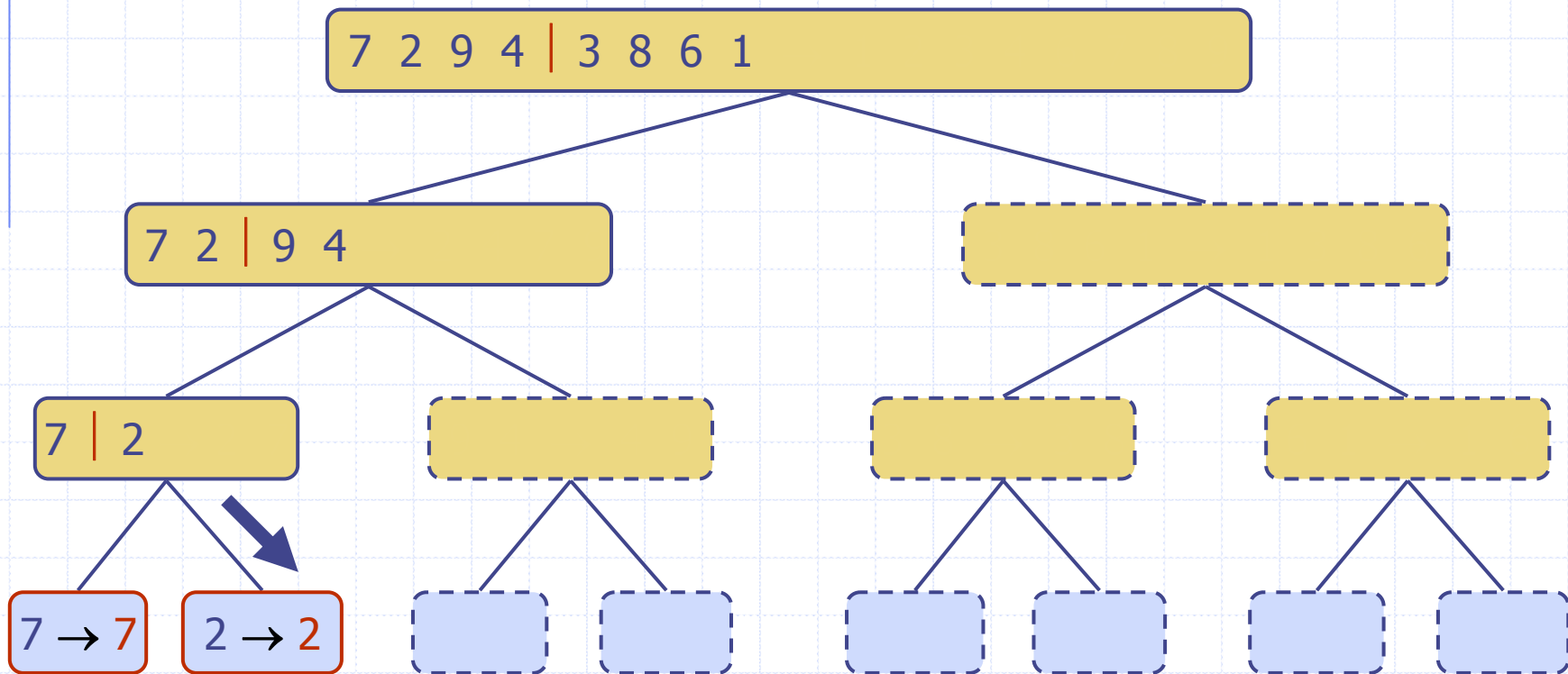
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Αναδρομική κλήση, ταπεινή περίπτωση



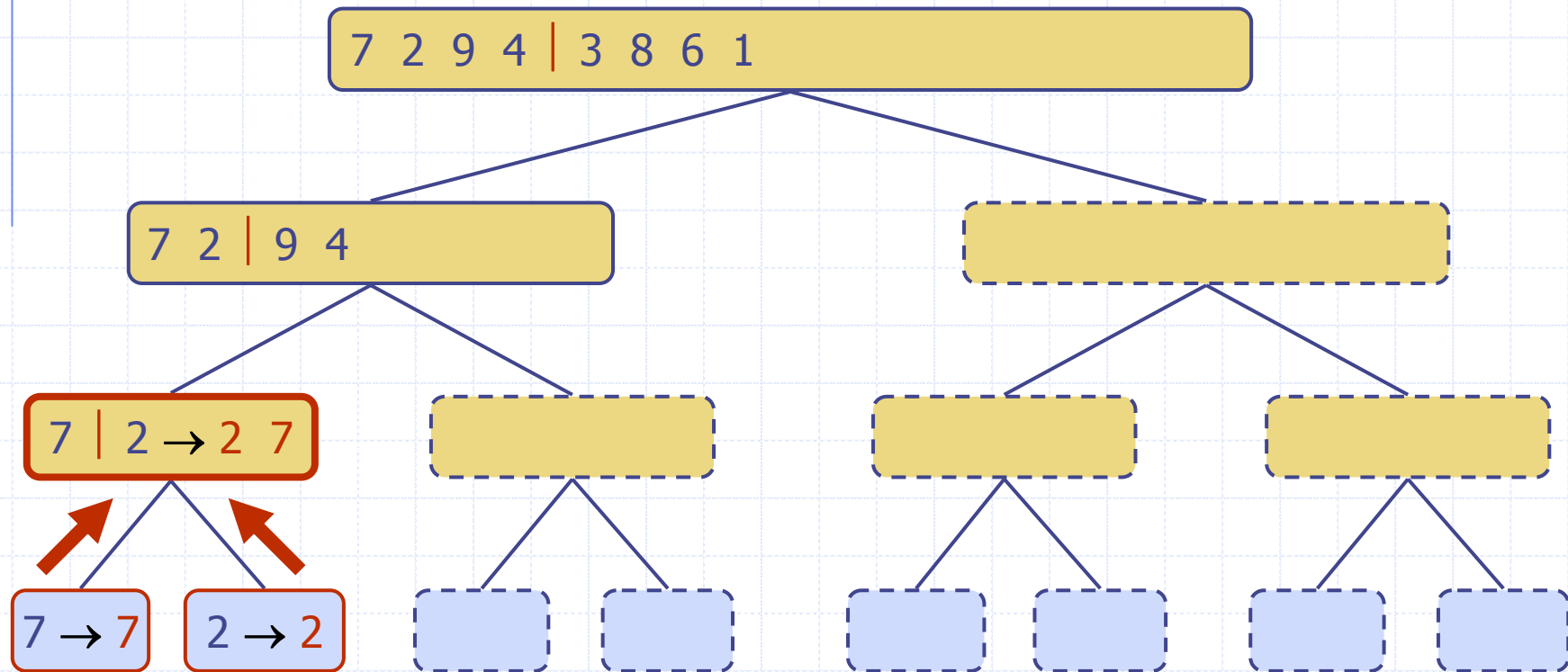
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Αναδρομική κλήση, ταπεινή περίπτωση



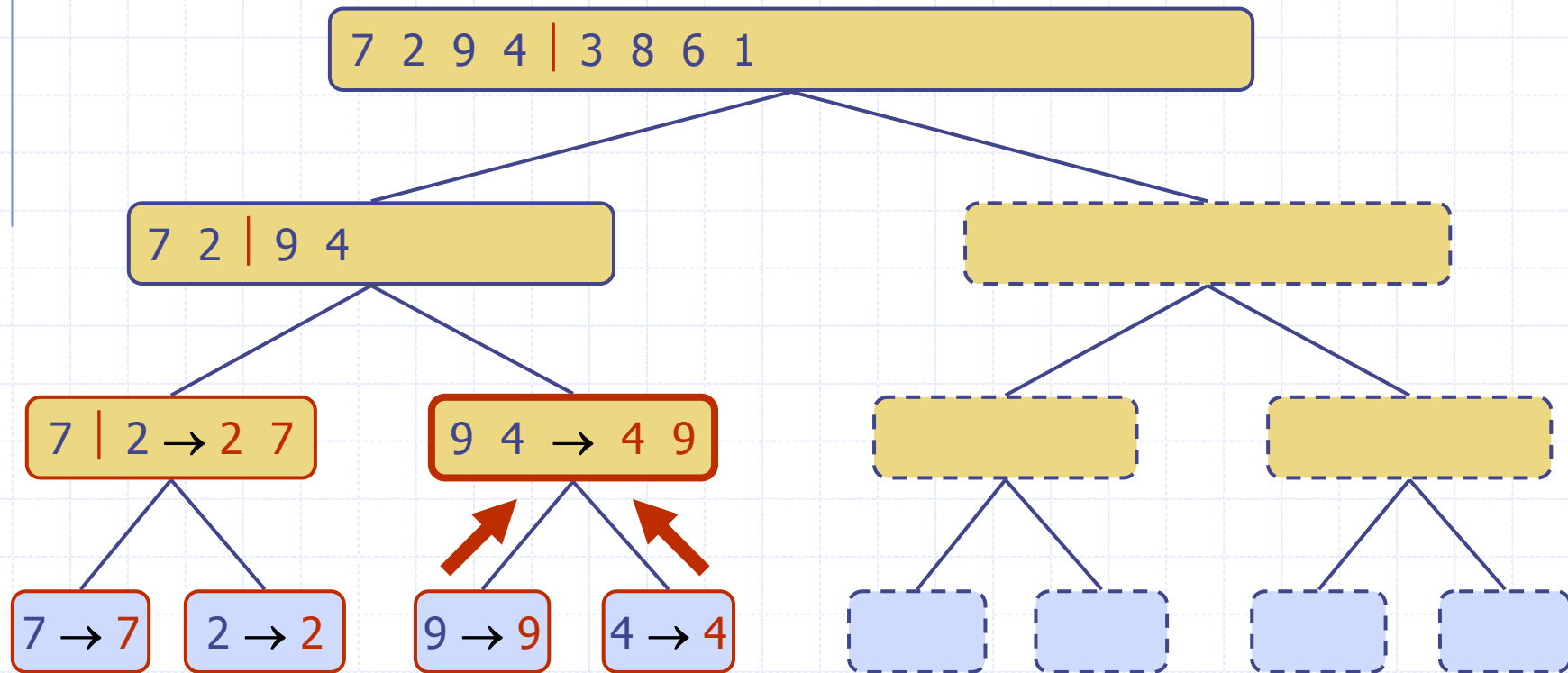
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Συγχώνευση (Merge)



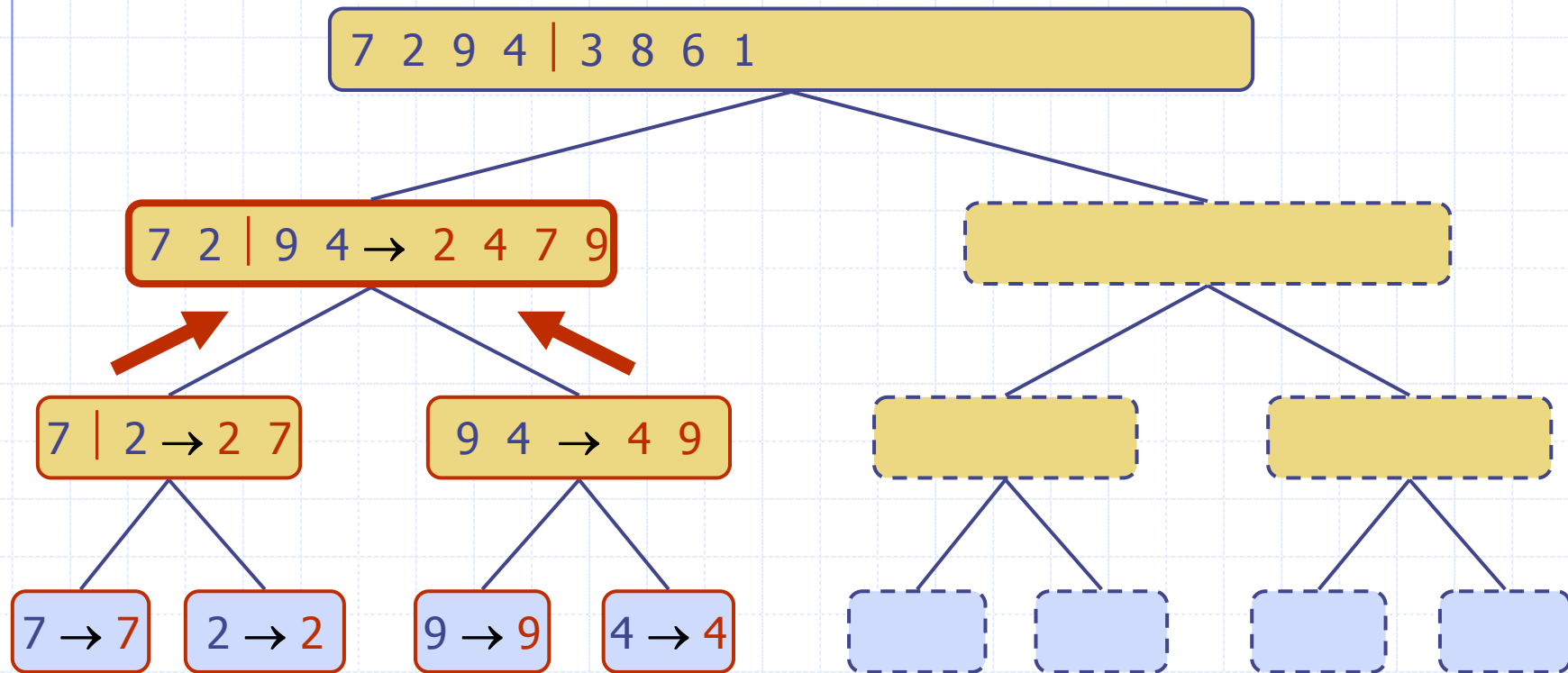
Παράδειγμα εκτέλεσης (Συνέχεια)

- ◆ Αναδρομική κλήση, ..., ταπεινή περίπτωση, συγχώνευση



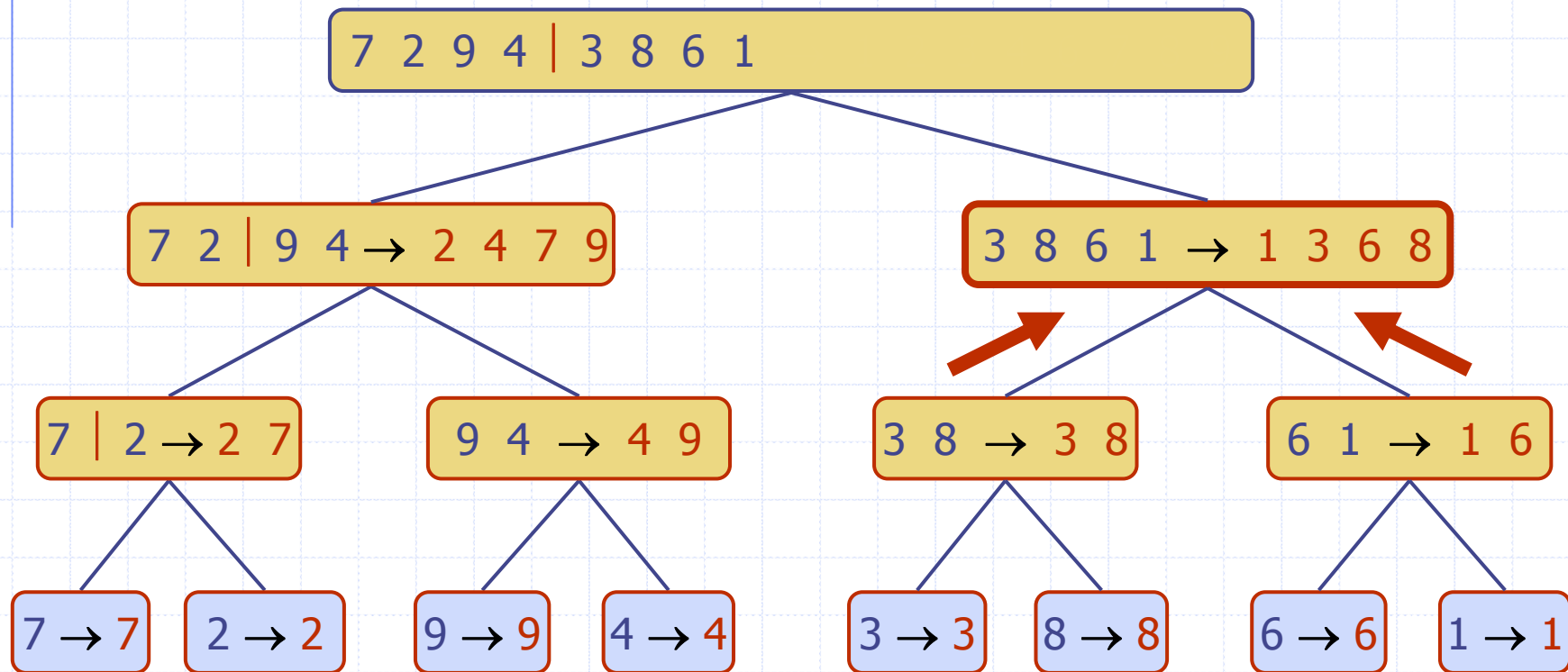
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Συγχώνευση (Merge)



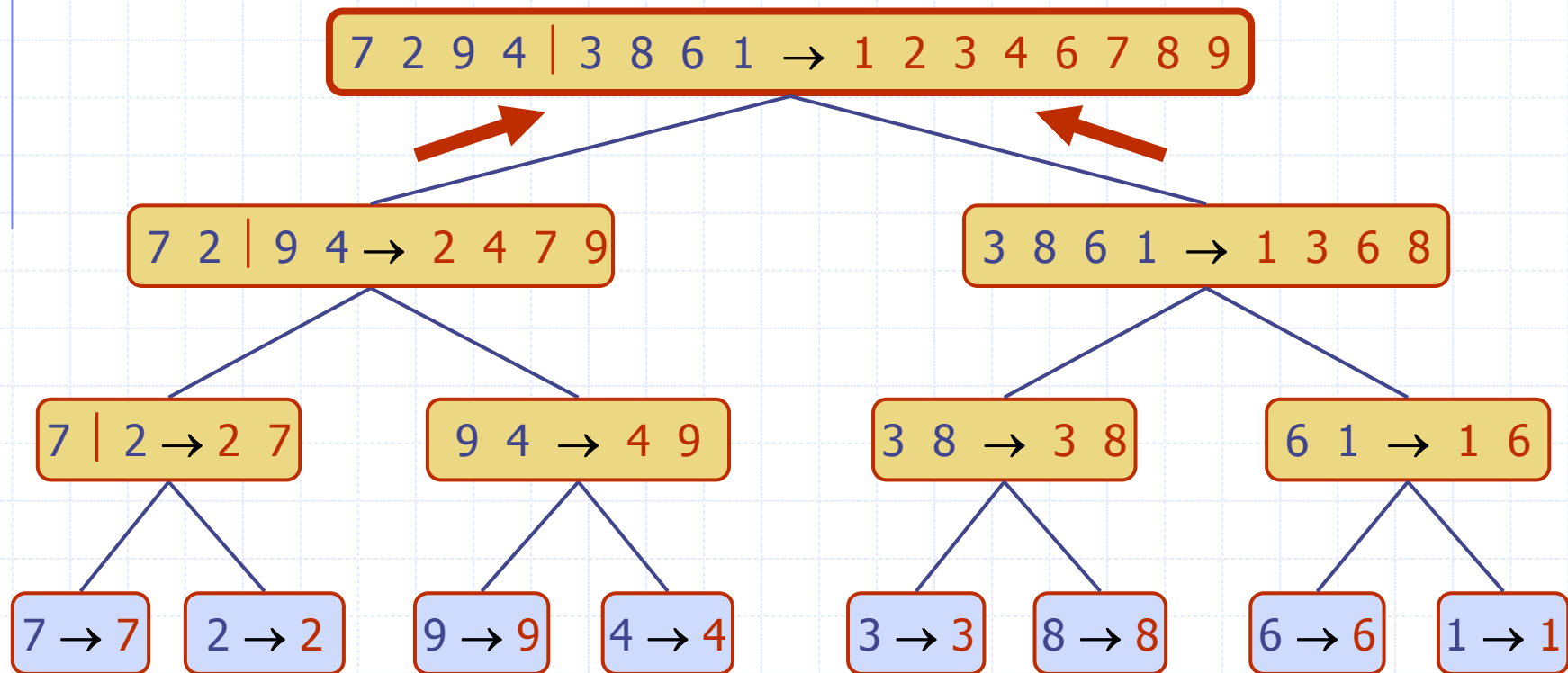
Παράδειγμα εκτέλεσης (Συνέχεια)

◆ Αναδρομική κλήση,, ..., συγχώνευση, συγχώνευση



Παράδειγμα εκτέλεσης (Συνέχεια)

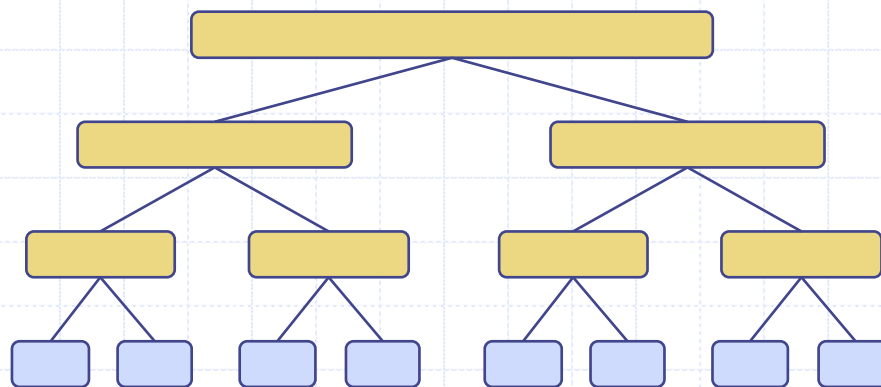
◆ Συγχώνευση



Ανάλυση της Συγχωνευτικής Ταξινόμησης (Merge-Sort).

- ◆ Το ύψος h του δέντρου της συγχωνευτικής ταξινόμησης είναι $O(\log n)$
 - Σε κάθε αναδρομική κλήση διαιρούμε στη μέση την ακολουθία,
- ◆ Το συνολικό ποσό ή δουλειά που γίνεται στους κόμβους βάθους i είναι $O(n)$
 - Διαχωρίζουμε και συγχωνεύουμε 2^i ακολουθίες μεγέθους $n/2^i$
 - κάνουμε 2^{i+1} αναδρομικές κλήσεις
- ◆ Έτσι, ο συνολικός χρόνος εκτέλεσης της συγχωνευτικής ταξινόμησης είναι $O(n \log n)$

βάθος	#ακολουθίας	μέγεθος
0	1	n
1	2	$n/2$
i	2^i	$n/2^i$
...



Συγχωνευτική ταξινόμηση

Summary of Sorting Algorithms

Αλγόριθμος	Χρόνος	Παρατηρήσεις
Επιλεκτική ταξινόμηση	$O(n^2)$	<ul style="list-style-type: none">♦ αργή♦ in-place♦ για μικρά σύνολα δεδομένων (< 1K)
Ενθετική ταξινόμηση	$O(n^2)$	<ul style="list-style-type: none">♦ αργή♦ in-place♦ για μικρά σύνολα δεδομένων (< 1K)
Ταξινόμηση σωρού	$O(n \log n)$	<ul style="list-style-type: none">♦ γρήγορη♦ in-place♦ για μεγάλα σύνολα δεδομένων (1K — 1M)
Συγχωνευτική ταξινόμηση	$O(n \log n)$	<ul style="list-style-type: none">♦ γρήγορη♦ διαδοχική πρόσβαση δεδομένων♦ για τεράστια σύνολα δεδομένων (> 1M)

Τέλος Ενότητας



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Κρήτης» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Σημειώματα

Σημείωμα αδειοδότησης

- Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση, Όχι Παράγωγο Έργο 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-nd/4.0/>

- Ως Μη Εμπορική ορίζεται η χρήση:
 - που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
 - που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
 - που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο
- Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Κρήτης, Ιωάννης Τόλλης 2015. «Αλγόριθμοι και πολυπλοκότητα. Συγχωνευτική Ταξινόμηση». Έκδοση: 1.0. Ηράκλειο 2015. Διαθέσιμο από τη δικτυακή διεύθυνση:
<https://opencourses.uoc.gr/courses/course/view.php?id=368>

Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.