



# ΣΧΕΔΙΑΣΗ ΚΑΙ ΑΝΑΛΥΣΗ ΑΛΓΟΡΙΘΜΩΝ

---

## ΔΙΔΑΚΤΙΚΗ ΕΝΟΤΗΤΑ 8

ΘΕΜΑ: ΧΡΗΣΗ ΔΟΜΩΝ ΔΕΝΤΡΟΥ ΚΑΙ  
ΣΩΡΟΥ ΓΙΑ ΕΠΙΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ  
ΤΑΞΙΝΟΜΗΣΗΣ – ΑΛΓΟΡΙΘΜΟΣ HEAPSORT



# ΠΕΡΙΕΧΟΜΕΝΟ

---

- Βασικές Έννοιες – Δομή Σωρού, Σωρός
- Αλγόριθμος Κατασκευής Σωρού
- Αλγόριθμος Συμπλήρωσης Σωρού
- Παράδειγμα
- Αλγόριθμος Ταξινόμησης Σωρού - HeapSort
- Παράδειγμα



# ΔΟΜΗ ΣΩΡΟΥ

---

## Ορισμός

Δέντρο που προκύπτει από πλήρες δυαδικό δέντρο με ρίζα αν του αφαιρεθούν μερικοί κόμβοι (ή κανένας) από τα δεξιά του τελευταίου επιπέδου.

Ειδικότερα δομή σωρού έχω σε ένα δέντρο αν:

- όλοι οι εσωτερικοί κόμβοι έχουν δυο παιδιά, με μόνη εξαίρεση τον πιο δεξιά εσωτερικό κόμβο του επιπέδου  $d-1$  που επιτρέπεται να έχει βαθμό 1, οπότε θα έχει μόνο αριστερό παιδί, και
- στο επίπεδο  $d-1$  τα φύλλα είναι όλα στα δεξιά των εσωτερικών κόμβων.



# ΑΝΤΙΣΤΟΙΧΙΣΗ ΛΙΣΤΑΣ ΣΕ ΔΟΜΗ ΣΩΡΟΥ

---

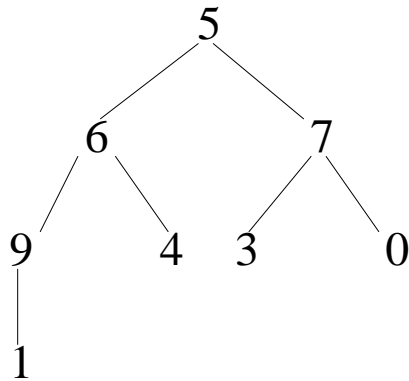
Αν έχω μια λίστα  $\mathbf{L}$  με  $n$  αριθμούς τους αντιστοιχίζω σε μια δομή σωρού βάθους  $d = \lceil \log_2 n \rceil$ , έτσι ώστε στο επίπεδο  $j$  της δομής σωρού ο πρώτος κόμβος να αντιστοιχεί στον αριθμό που βρίσκεται στη θέση  $2^j$  στη λίστα  $\mathbf{L}$  και ο τελευταίος στον αριθμό που βρίσκεται στη θέση  $2^{j+1}-1$  στη λίστα  $\mathbf{L}$  με εξαίρεση το τελευταίο επίπεδο όπου δεν υπάρχουν αρκετοί όροι.



# ΠΑΡΑΔΕΙΓΜΑ

---

Αν  $L=[5, 6, 7, 9, 4, 3, 0, 1]$  τότε η δομή σωρού είναι:





# ΣΩΡΟΣ - HEAP

---

## Ορισμός

Σωρός ονομάζεται ένα δέντρο στους κόμβους του οποίου αντιστοιχούν αριθμοί από μια λίστα  $L$  και είναι δομή σωρού με την επιπλέον ιδιότητα ο αριθμός που αντιστοιχεί σε έναν κόμβο να μην είναι μικρότερος από τους αριθμούς που αντιστοιχούν στα παιδιά του, αν ο κόμβος έχει παιδιά.



# ΣΧΕΔΙΑΣΜΟΣ ΑΛΓΟΡΙΘΜΟΥ ΚΑΤΑΣΚΕΥΗΣ ΣΩΡΟΥ

---

- Ο αλγόριθμος κατασκευής σωρού παίρνει σαν είσοδο μια δομή σωρού **L** και δίνει σαν έξοδο ένα σωρό.
- Ο αλγόριθμος κατασκευής σωρού χρησιμοποιεί ως υποαλγόριθμο τον αλγόριθμο συμπλήρωσης σωρού που παίρνει σαν είσοδο μια δομή σωρού όπου τα παιδιά της ρίζας είναι σωροί και δίνει σαν έξοδο ένα σωρό.



## ΒΗΜΑΤΑ

---

- Ο αλγόριθμος κατασκευής σωρού επισκέπτεται τους εσωτερικούς κόμβους του δέντρου εισόδου από επίπεδο σε επίπεδο και σε κάθε επίπεδο από αριστερά προς τα δεξιά. Η επίσκεψη ξεκινά από το προτελευταίο επίπεδο προς τα υψηλότερα επίπεδα.
- Σε κάθε εσωτερικό κόμβο ενεργοποιείται ο αλγόριθμος συμπλήρωσης σωρού για το δέντρο που έχει ρίζα τον κόμβο αυτό.





# ΑΛΓΟΡΙΘΜΟΣ ΚΑΤΑΣΚΕΥΗΣ ΣΩΡΟΥ-Κ.Σ

---

**Είσοδος:** Δομή σωρού **L** με **n** κόμβους.

**Έξοδος:** Σωρός

**Ψευδοκώδικας:**

```
 $d = \lfloor \log_2 n \rfloor$   
 $m_1 = 2^d$   
for  $m=d-1$  downto  $0$  do  
     $k=m$   
    while  $2k \leq n$  and  $k < m_1$  do  
        κλήση αλγόριθμου  $\sigma.\sigma(r,k,n)$   
         $k=k+1$   
    end while  
     $m_1=2^m$   
end for
```



# ΑΛΓΟΡΙΘΜΟΣ ΣΥΜΠΛΗΡΩΣΗΣ ΣΩΡΟΥ- Σ.Σ(r,k,n)

**Είσοδος:** Δομή σωρού με δείκτη ρίζας  $r$  και  $n$  στοιχεία όπου καθένα από τα υποδέντρα της ρίζας είναι σωρός.

**Έξοδος:** Σωρός

**Ψευδοκώδικας:**

```
x=L(r)
rleft=2r
while rleft <= n do
    max=rleft
    rright=rleft+1
    if rright<=n and L(rright)>=L(rleft) then max=rright
    if x<L(max) then do
        L(r)=L(max)
        r=max
        rleft=2max
    end if
end while
L(r)=x
```

Δρ. Δημήτριος Κ. Κουκόπουλος  
Αναπληρωτής Καθηγητής



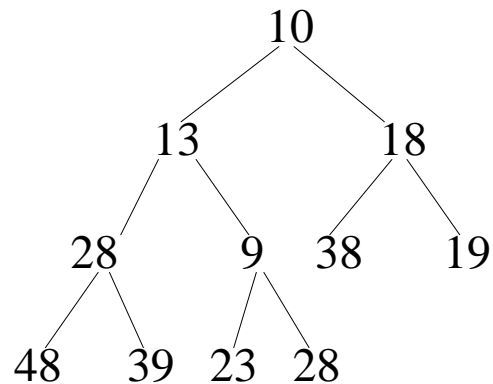
## ΠΑΡΑΔΕΙΓΜΑ 1/4

---

**Πρόβλημα:** Να εφαρμοστεί ο αλγόριθμος κατασκευής σωρού για τη λίστα  $L=[10,13,18,28,9,38,19,48,39,23,28]$ .

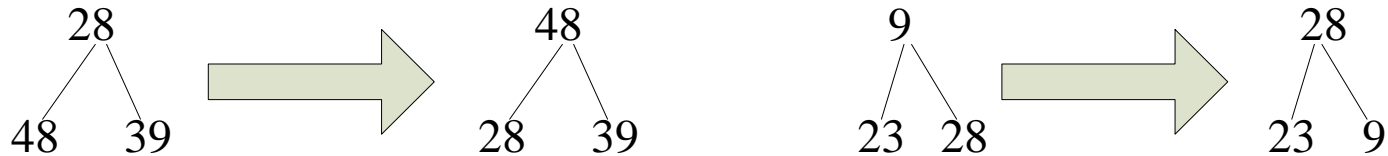
**Απάντηση:**

1. Μετατρέπουμε τη λίστα  $L$  σε δομή σωρού:

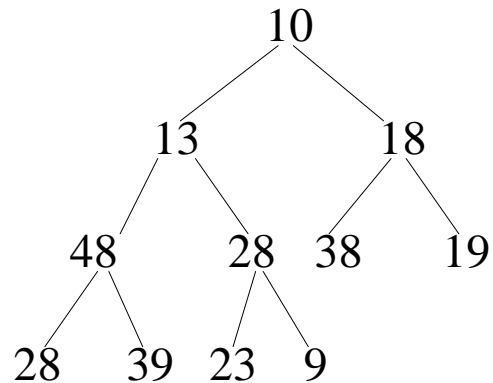


## ΠΑΡΑΔΕΙΓΜΑ 2/4

2. Ξεκινούμε από το προτελευταίο επίπεδο,  $d-1=2$ . Καλούμε τον **αλγόριθμο σ.σ** για το υποδέντρο με ρίζα το 28, και στη συνέχεια για το υποδέντρο με ρίζα το 9:

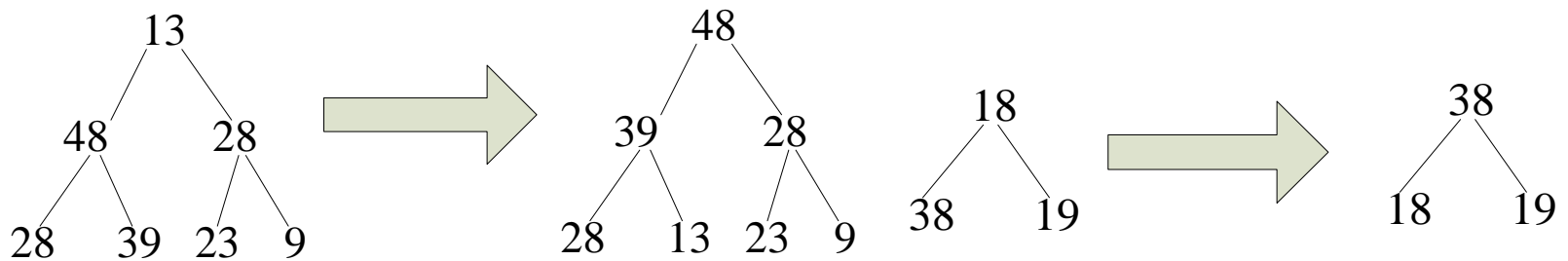


Οπότε:

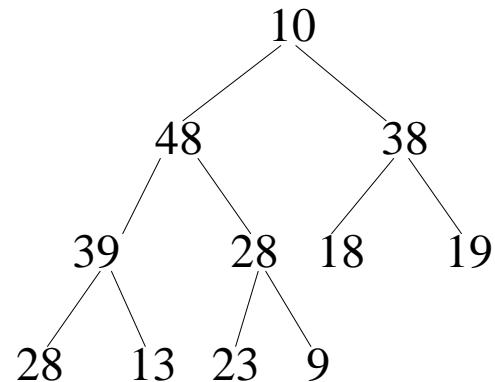


## ΠΑΡΑΔΕΙΓΜΑ 3/4

3. Πηγαίνουμε σε επίπεδο 1 και καλούμε τον **αλγόριθμο σ.σ** για το υποδέντρο με ρίζα το 13, και στη συνέχεια για το υποδέντρο με ρίζα το 18:



Οπότε:

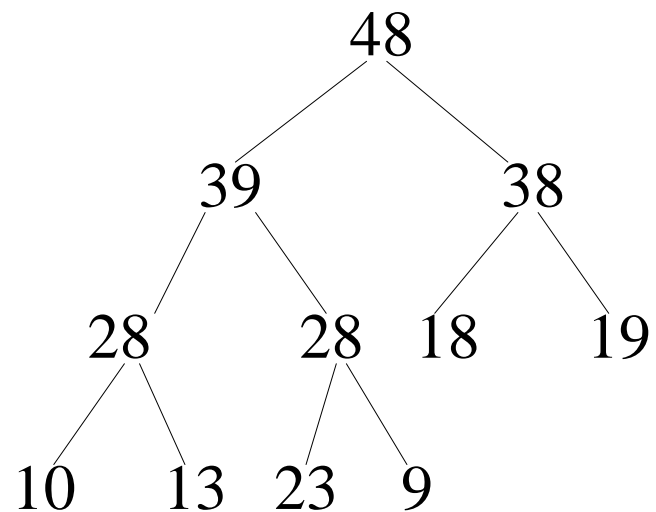




## ΠΑΡΑΔΕΙΓΜΑ 4/4

---

4. Πηγαίνουμε σε επίπεδο 0 και καλούμε τον αλγόριθμο σ.σ για τη ρίζα με τον αριθμό 10:





# ΑΛΓΟΡΙΘΜΟΣ HEAPSORT ΓΙΑ ΤΑΞΙΝΟΜΗΣΗ ΛΙΣΤΑΣ 1/2

---

## **Βήματα:**

1. Ο HeapSort εφαρμόζει πρώτα τον αλγόριθμο κατασκευής σωρού στην αρχική λίστα **L**. Τότε στη ρίζα τοποθετείται το μέγιστο στοιχείο της λίστας.
2. Το στοιχείο που βρίσκεται στη ρίζα του σωρού το παίρνουμε και το βάζουμε στο τέλος της **L**.
3. Στην κενή θέση τοποθετείται το τελευταίο φύλλο (το δεξιότερο στο τελευταίο επίπεδο) που αφαιρείται από δέντρο. Ο σωρός γίνεται δομή σωρού.



# ΑΛΓΟΡΙΘΜΟΣ HEAPSORT ΓΙΑ ΤΑΞΙΝΟΜΗΣΗ ΛΙΣΤΑΣ 2/2

---

4. Καλείται ο αλγόριθμος συμπλήρωσης σωρού, οπότε η δομή σωρού γίνεται πάλι σωρός με το μεγαλύτερο στοιχείο στη ρίζα. Το στοιχείο αυτό τοποθετείται στη θέση  $n-1$  της **L** και στη ρίζα μπαίνει πάλι τελευταίο φύλλο.
5. Επαναλαμβάνουμε διαδικασία μέχρι τέλος στοιχείων λίστας.





# ΨΕΥΔΟΚΩΔΙΚΑΣ HEAPSORT

---

κλήση αλγορίθμου κατασκευής σωρού κ.σ.

for  $i=n$  to 2 do

$a=L(i)$

    κλήση αλγορίθμου συμπλήρωσης σωρού  $\sigma.\sigma(1,1,i-1)$

$L(i)=a$

end for

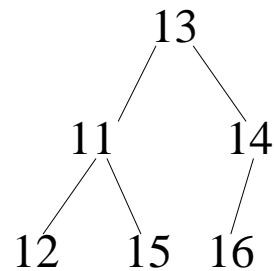


## ΠΑΡΑΔΕΙΓΜΑ 1/5

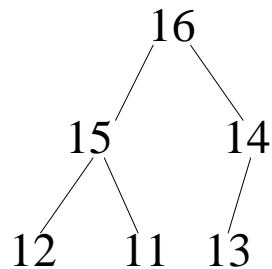
---

**Πρόβλημα:** Εφαρμόστε τον αλγόριθμο HeapSort για να ταξινομήσετε τη λίστα  $L=[13,11,14,12,15,16]$

**Απάντηση:** Η δομή σωρού στην οποία αντιστοιχεί η λίστα  $L$  είναι:



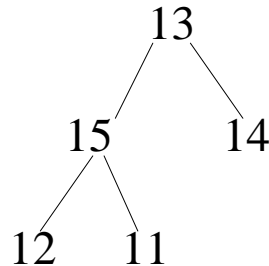
Εφαρμόζουμε τον αλγόριθμο κατασκευής σωρού και παίρνουμε:



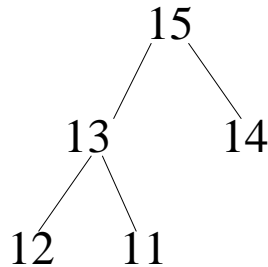
## ΠΑΡΑΔΕΙΓΜΑ 2/5

Τοποθετώ το στοιχείο στη ρίζα στη θέση L(6) και στη ρίζα βάζω το τελευταίο φύλλο:

$L=[13,15,14,12,11,16]$



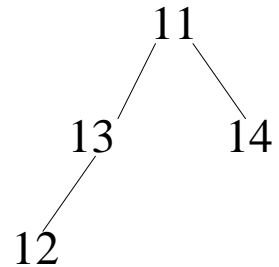
Εφαρμόζουμε τον αλγόριθμο συμπλήρωσης σωρού στο νέο δέντρο:



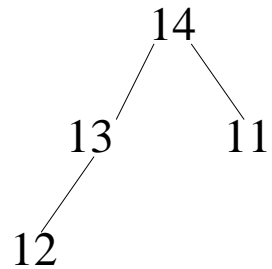
## ΠΑΡΑΔΕΙΓΜΑ 3/5

Τοποθετώ το στοιχείο στη ρίζα στη θέση L(5) και στη ρίζα βάζω το τελευταίο φύλλο:

$L=[11,13,14,12,15,16]$



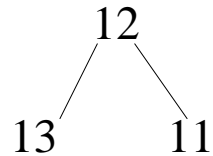
Εφαρμόζουμε τον αλγόριθμο συμπλήρωσης σωρού στο νέο δέντρο:



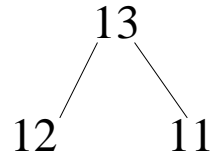
## ΠΑΡΑΔΕΙΓΜΑ 4/5

Τοποθετώ το στοιχείο στη ρίζα στη θέση L(4) και στη ρίζα βάζω το τελευταίο φύλλο:

$L=[12,13,11,14,15,16]$

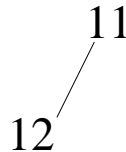


Εφαρμόζουμε τον αλγόριθμο συμπλήρωσης σωρού στο νέο δέντρο:



Τοποθετώ το στοιχείο στη ρίζα στη θέση L(3) και στη ρίζα βάζω το τελευταίο φύλλο:

$L=[12,11,13,14,15,16]$

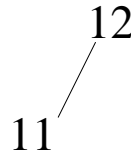




## ΠΑΡΑΔΕΙΓΜΑ 5/5

---

Εφαρμόζουμε τον αλγόριθμο συμπλήρωσης σωρού στο νέο δέντρο:



Τοποθετώ το στοιχείο στη ρίζα στη θέση L(2) και στη ρίζα βάζω το τελευταίο φύλλο:

$L=[11,12,13,14,15,16]$