



**ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**  
**Τ Μ Η Μ Α Π Λ Η Ρ Ο Φ Ο Ρ Ι Κ Η Σ**

12.02.2013

**Δομές Δεδομένων**

**Εξεταστική Ιανουαρίου 2013**

**Διδάσκων : Ευάγγελος Μαρκάκης**

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:	
ΑΜ:	
ΥΠΟΓΡΑΦΗ ΕΠΟΠΤΗ:	

**Διάρκεια εξέτασης : 2 ώρες και 30 λεπτά**

**Απαντήστε σε όλα τα θέματα**

Θέμα	Βαθμός
1	
2	
3	
4	
5	
6	
Σύνολο:	

### ΘΕΜΑ 1 (12 μονάδες)

Απαντήστε στις παρακάτω ερωτήσεις. Δεν χρειάζεται να δικαιολογήσετε την απάντησή σας. Απλώς κυκλώστε τις σωστές απαντήσεις (ενδέχεται να υπάρχει παραπάνω από μία σωστή απάντηση ή να μην υπάρχει καμία σωστή απάντηση).

1.  $4n^4 + n^3 - 50n^2 =$ 
  - a.  $O(n^2)$
  - b.  $O(n^3)$
  - c.  $O(n^4)$
  - d.  $O(n!)$
2. Σε ένα δέντρο δυαδικής αναζήτησης με  $n$  κόμβους, ο χρόνος αναζήτησης στη χειρότερη περίπτωση είναι
  - a.  $O(\log n)$
  - b.  $O(n)$
  - c.  $O(\log \log n)$
  - d.  $O(n^2)$
3. Σε ένα B-δέντρο με  $n$  αντικείμενα, ο χρόνος αναζήτησης στη χειρότερη περίπτωση είναι
  - a.  $O(\log n)$
  - b.  $O((\log n)^5)$
  - c.  $O(\log \log n)$
  - d.  $O(n^2)$
4. Οι μέθοδοι ταξινόμησης Mergesort και Insertionsort έχουν
  - a. τις ίδιες απαιτήσεις μνήμης
  - b. την ίδια πολυπλοκότητα χειρότερης περίπτωσης
  - c. την ίδια πολυπλοκότητα μέσης περίπτωσης
5.  $f(n) = O(n \log n)$ , όπου  $f(n) = 2f(n/2) + n$ , για  $n \geq 2$ , και  $f(1) = 1$ 
  - a. Σωστό
  - b. Λάθος
6. Ένας γράφος που έχει  $n$  κορυφές,  $n-1$  πλευρές και κανένα κύκλο είναι δέντρο.
  - a. Σωστό
  - b. Λάθος

## ΘΕΜΑ 2 (10 μονάδες)

Δίνεται ο ακόλουθος ορισμός των κόμβων μίας απλά συνδεδεμένης λίστας.

```
class Node{  
    int key;  
    Node next;  
    Node(int x){ key = x; next = null; } }
```

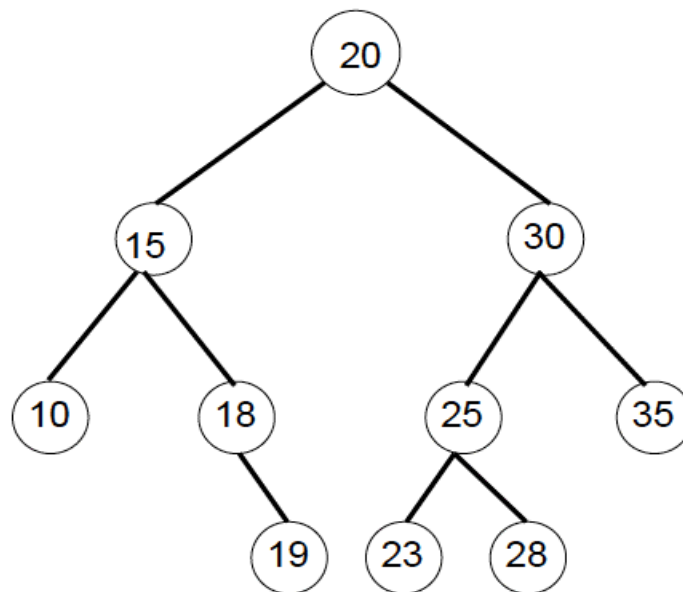
Να γράψετε σε Java τη μέθοδο `Node findmax(Node h)` η οποία κάνει αναζήτηση σε μια *αταξινόμητη* λίστα με κεφαλή `h` για να βρει τον κόμβο με το μέγιστο κλειδί `key`. Όταν βρει τον επιθυμητό κόμβο, τον μεταφέρει στην κεφαλή της λίστας. Η μέθοδος επιστρέφει `null` αν η λίστα ήταν κενή, και σε κάθε άλλη περίπτωση επιστρέφει τη νέα κεφαλή της λίστας. Η λίστα δεν έχει κόμβους φρουρούς και τερματίζεται με `null`.

### ΘΕΜΑ 3 (14 μονάδες)

Δίνεται το παρακάτω ΔΔΑ.

(α) Έστω ότι θέλουμε να κάνουμε μετα-διατεταγμένη διάσχιση στο δέντρο. Γράψτε με ποια σειρά θα επισκεφτεί τους κόμβους η μέθοδος αυτή.

(β) Έστω ότι θέλουμε να κάνουμε το κλειδί 28 ρίζα του δέντρου. Σχεδιάστε το δέντρο που προκύπτει μετά από κάθε απαιτούμενη περιστροφή, μέχρι να έρθει το 28 στη ρίζα.



**ΘΕΜΑ 4 (12 μονάδες)**

(α) Έστω ένα σύνολο 8 αντικειμένων με τα εξής κλειδιά: {12, 51, 62, 24, 40, 46, 100, 13}. Θέλουμε να τα αποθηκεύσουμε διαδοχικά ξεκινώντας από το 12 σε έναν πίνακα κατακερματισμού με χρήση γραμμικής διερεύνησης, και χρησιμοποιώντας ως συνάρτηση κατακερματισμού την  $h(k) = k \bmod 11$ . Σχεδιάστε τα περιεχόμενα του πίνακα μετά την ολοκλήρωση κάθε εισαγωγής.

(β) Έστω ότι έχετε έναν πίνακα κατακερματισμού με χωριστή αλυσίδωση. Ο πίνακας έχει διάσταση  $M$  και κατά μέσο όρο σε κάθε θέση του πίνακα η λίστα που ξεκινά από εκεί έχει 6 στοιχεία (άρα συνολικά έχουμε  $6M$  στοιχεία). Έστω ότι θέλετε να υλοποιήσετε ένα σύστημα με γραμμική διερεύνηση που να έχει το ίδιο μέσο πλήθος διερευνήσεων για τις επιτυχείς αναζητήσεις. Αν  $M=2000$ , ποιο πρέπει να είναι το μέγεθος του πίνακα που θα χρειαστείτε στη γραμμική διερεύνηση;

Το μέσο πλήθος διερευνήσεων για επιτυχείς αναζητήσεις είναι  $(1 + 1/(1-\alpha))/2$ , στη γραμμική διερεύνηση, όπου  $\alpha$  ο συντελεστής πλήρωσης.

**ΘΕΜΑ 5 (22 μονάδες)**

Απαντήστε στις παρακάτω ερωτήσεις δικαιολογώντας την απάντησή σας.

- 1. [4]** Έχετε να διαλέξετε μεταξύ InsertionSort και SelectionSort. Τα αντικείμενα είναι μεγάλα σε μέγεθος και τα κλειδιά είναι ακέραιοι. Ποια μέθοδο θα επιλέξετε?
- 2. [6]** Ένας φίλος σας θέλει να χρησιμοποιήσει έναν αλγόριθμο ταξινόμησης που στη χειρότερη περίπτωση να έχει πολυπλοκότητα  $O(N \log N)$  για την ταξινόμηση  $N$  στοιχείων. Ποιες από τις παρακάτω μεθόδους θα του προτείνετε;

  - i)** Insertionsort
  - ii)** Heapsort
  - iii)** Quicksort
  - iv)** Mergesort
  - v)** Selectionsort
  - vi)** Bubblesort

3. [6] Θέλετε να κάνετε αναζήτηση και εισαγωγή κλειδιών σε μία δομή και θέλετε στη χειρότερη περίπτωση ο χρόνος για την εισαγωγή να είναι  $O(\log N)$ . Κυκλώστε όσες από τις παρακάτω δομές είναι κατάλληλες και εξηγήστε συνοπτικά την απάντησή σας.

- i) Δέντρα δυαδικής αναζήτησης
- ii) Τυχαιοποιημένα δέντρα δυαδικής αναζήτησης
- iii) Αταξινόμητος πίνακας
- iv) Ταξινομημένη λίστα
- v) Δέντρα κόκκινου-μαύρου
- vi) Κατακερματισμός με χωριστή αλυσίδωση και  $\alpha = 3$ .
- vii) Λίστες παράλειψης

4. [6] Έχετε στη διάθεσή σας μία ουρά προτεραιότητας, υλοποιημένη με χρήση σωρού. Έστω ότι περιέχει  $N$  αντικείμενα με ακέραια κλειδιά και θέλετε να βρείτε το  $k$ -οστό μεγαλύτερο από αυτά. Με τι χρονική πολυπλοκότητα χειρότερης περίπτωσης μπορείτε να λύσετε το πρόβλημα αυτό? Αν τα αντικείμενα αυτά ήταν αποθηκευμένα σε ένα ΔΔΑ, τότε ποια η πολυπλοκότητα επίλυσης στη μέση και στη χειρότερη περίπτωση?

**ΘΕΜΑ 6 (10 μονάδες)**

Ξεκινώντας από ένα κενό δέντρο 2-3-4, να εισάγετε είτε με ανοδική είτε με καθοδική εισαγωγή τα κλειδιά 30, 35, 40, 18, 23, 27, 45. Δείξτε πώς μεταβάλλεται το δέντρο όταν ολοκληρώνεται κάθε εισαγωγή. Για το τελικό δέντρο που θα προκύψει, σχεδιάστε το ισοδύναμο δέντρο κόκκινου-μαύρου