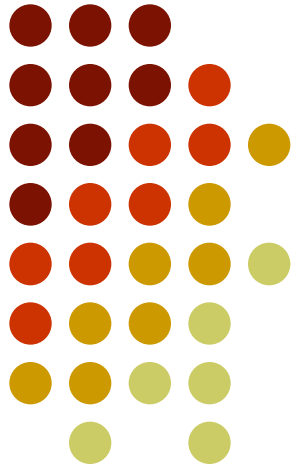
ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ Ι

Φροντιστήριο 13-1-2011

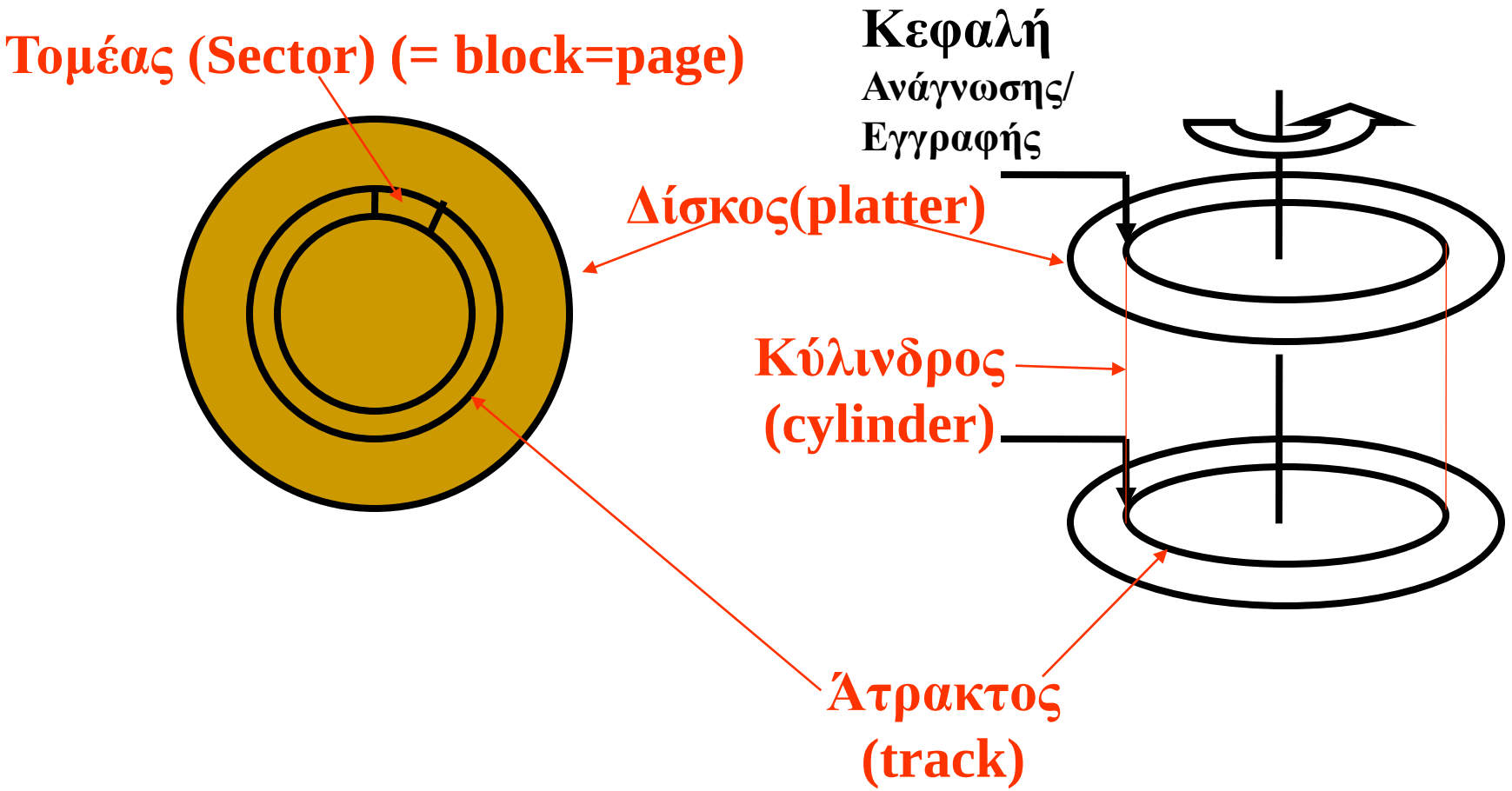
Αποθήκευση σε δίσκο, βασικές οργανώσεις αρχείων κατακερματισμός και δομές ευρετηρίων για αρχεία



Θεωρία

- **Άτρακτος/αυλάκι :** ομόκεντροι κύκλοι στον δίσκο
- **Κύλινδρος:** οι άτρακτοι με την ίδια διάμετρο σε κάποιο πακέτο δίσκων
- **Μπλοκ ή Τομείς(sectors):** μέρη της ατράκτου
- **Ισομεγέθη μπλοκ ή σελίδες :** καθορίζονται κατά την διαδικασία της μορφοποίησης από το ΛΣ
  - Χωρίζονται με διάκενα μεταξύ τους – interblock gaps
- **Συστάδα (cluster) :** συνεχόμενα μπλοκ

Δίσκος


Τομέας (Sector) (= block=page)
Κεφαλή
Ανάγνωσης/
Εγγραφής
Δίσκος(platter)
Κύλινδρος
(cylinder)
Άτρακτος
(track)

Θεωρία

- **Εγγραφή – record** : μορφή αποθήκευσης των δεδομένων στον δίσκο.
  - Αποτελείται από *τιμές (values)* που αντιστοιχούν σε συγκεκριμένα *πεδία (fields)* της εγγραφής
- **Αρχείο – File** : ακολουθία από εγγραφές
  - Εγγραφές σταθερού μήκους : όλες οι εγγραφές στο αρχείο έχουν το ίδιο μέγεθος σε Byte
  - Εγγραφές μεταβλητού μήκους

Θεωρία

Για αρχεία με εγγραφές σταθερού μήκους

- Παράγοντας ομαδοποίησης **bfr=** $\lfloor B/R \rfloor$
  - **B**=μέγεθος ενός μπλοκ σε bytes
  - **R**=μέγεθος εγγραφής σε bytes
- Αχρησιμοποίητος χώρος **B-(bfr*R)** bytes
- **Εκτεινόμενη οργάνωση (spanned)** – μία εγγραφή μπορεί να εκτείνεται σε περισσότερα από ένα μπλοκ
  - Μέσο πλήθος εγγραφών ανά μπλοκ του αρχείου→ bfr
  - # μπλοκ για ένα αρχείο r εγγραφών
    - r : το πλήθος των εγγραφών του αρχείου
    - b : το πλήθος των μπλοκ που απαιτούνται για την αποθήκευση ενός αρχείου r εγγραφών

b= $\lceil r/bfr \rceil$ μπλοκ

- **Μη εκτεινόμενη οργάνωση (unspanned) –** η εγγραφή δεν επιτρέπεται να εκτείνεται σε περισσότερα από ένα μπλοκ

Εκφώνηση

- Ένα αρχείο έχει r=20.000 εγγραφές του τύπου φοιτητής με σταθερό μήκος. Κάθε εγγραφή έχει τα ακόλουθα πεδία:
  - ΟΝΟΜΑ (30 byte)
  - ΑΤ(9 byte)
  - Διεύθυνση(40 byte)
  - Τηλέφωνο (9 byte)
  - Ημερ_Γεν(8 byte)
  - Φύλο (1 byte)
  - Κύρια_κατεύθυνση(4 byte)
  - Δευτερεύσουσα_κατεύθυνση (4 byte)
  - Έτος(4 byte)
  - Κωδ_Πτυχίου(3 byte)

ΑΣΚΗΣΗ 1 - Εκφώνηση

- Το αρχείο είναι αποθηκευμένο σε δίσκο με τα εξής χαρακτηριστικά
  - Μέγεθος μπλοκ B=512 byte
  - Πλήθος μπλοκ ανά άτρακτο=20
  - Πλήθος ατράκτων ανά επιφάνεια = 400
  - Ένα πακέτο δίσκων αποτελείται από 15 δίσκους διπλής όψης

ΑΣΚΗΣΗ 1 - Ερωτήσεις

- Ποια είναι η συνολική χωρητικότητα της ατράκτου;
- Πόσοι κύλινδροι υπάρχουν σε έναν δίσκο;
- Ποια είναι η συνολική χωρητικότητα ενός κυλίνδρου;
- Ποια είναι η συνολική χωρητικότητα ενός πακέτου δίσκων;
- Υπολογίστε το μέγεθος εγγραφής R σε byte.
- Υπολογίστε τον παράγοντα ομαδοποίησης bfr και τον αριθμό των μπλοκ b, υποθέτοντας μη εκτεινόμενη οργάνωση

ΑΣΚΗΣΗ 1 - Απαντήσεις

Ποια η συνολική χωρητικότητα της ατράκτου;

- 20 μπλοκ * 512 Byte = 10kb

Πόσοι κύλινδροι υπάρχουν σε έναν δίσκο;

- *Όσοι και οι άτρακτοι → κύλινδρος: το σύνολο των ατράκτων με την ίδια διάμετρο. Άρα 400*

ΑΣΚΗΣΗ 1 - Απαντήσεις

Ποια είναι η συνολική χωρητικότητα ενός κυλίνδρου

- 1 κύλινδρος 15 δίσκοι * δυο επιφάνειες = 30 άτρακτοι. Άρα:
  - χωρητικότητα 30 * 10=300 kb

Ποια είναι η συνολική χωρητικότητα ενός πακέτου δίσκων

- 1 πακέτο δίσκων = 400 κύλινδροι. Άρα
  - Χωρητικότητα 300 * 400=120Mb

ΑΣΚΗΣΗ 1 - Απαντήσεις

Υπολογίστε το μέγεθος εγγραφής R σε byte

- (30byte +9+40+9+8+1+4+4+4+3+**1 σημάδι διαγραφής σε κάθε εγγραφή**) →R=113

Υπολογίστε τον παράγοντα ομαδοποίησης bfr και τον αριθμό των μπλοκ b, υποθέτοντας μη εκτεινόμενη οργάνωση

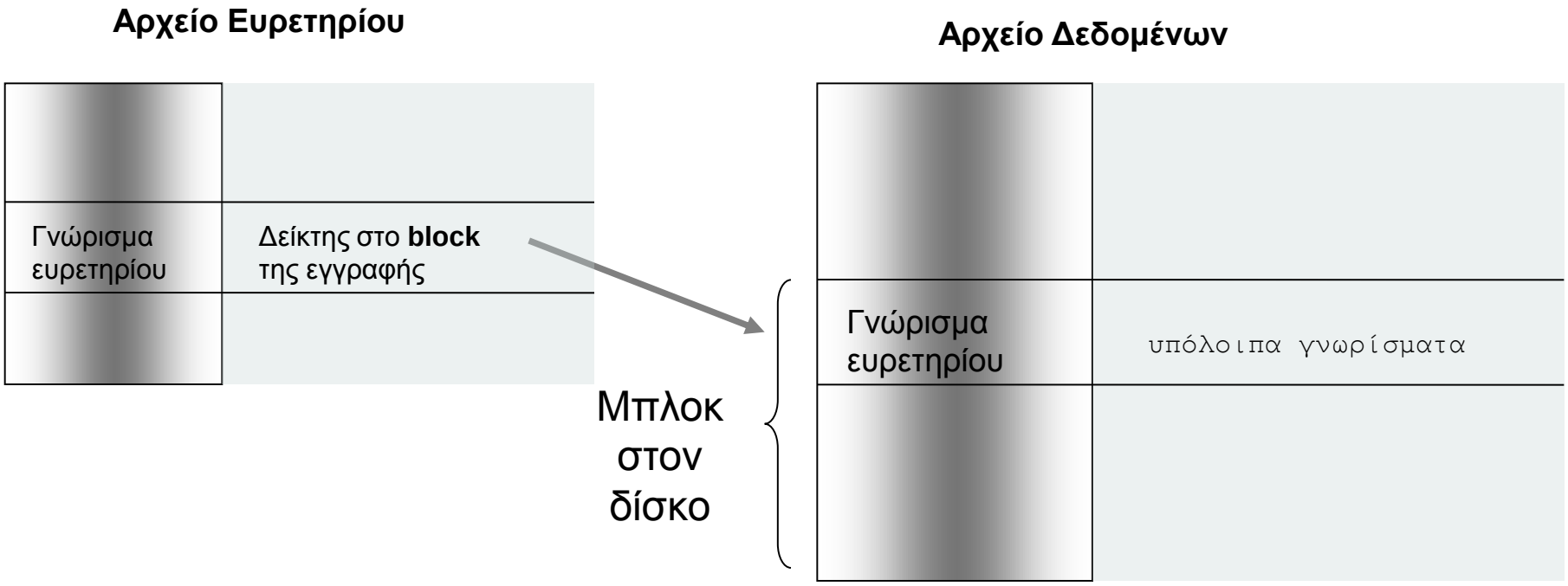
- bfr= ⌊B/R⌋ = ⌊512/113⌋ = 4
- b=⌈r/bfr⌉=20000/4=5000 μπλοκ

Θεωρία - Πρωτεύουσες μέθοδοι για την οργάνωση των εγγραφών

- Τεχνική των μη ταξινομημένων εγγραφών
  - Αρχείο σωρού – τοποθετεί τις εγγραφές στον δίσκο χωρίς καμία ιδιαίτερη διάταξη
  - Γραμμική αναζήτηση ΜΠ:**(b/2)**block
  - Εύκολη διαγραφή - εισαγωγή
- Τεχνική ταξινομημένων εγγραφών
  - Ταξινομημένο (ή σειριακό) αρχείο-διατηρεί τις εγγραφές σε διάταξη σύμφωνα με την τιμή κάποιου συγκεκριμένου πεδίου (κλειδί ταξινόμησης)
  - Δυαδική αναζήτηση ΜΠ: **$\log_2$(b)** block
  - Δύσκολη εισαγωγή – αρχείο υπερχείλισης
- Τεχνική κατακερματισμένων εγγραφών
  - Κατακερματισμένο αρχείο – χρησιμοποιεί συνάρτηση κατακερματισμού που εφαρμόζεται σε κάποιο πεδίο (κλειδί κατακερματισμού) για να προσδιορίσει την διεύθυνση των εγγραφών στον δίσκο

Θεωρία – ευρετήρια

- Ένα ευρετήριο (index) είναι μια βοηθητική δομή αρχείου που κάνει πιο αποδοτική την αναζήτηση μιας εγγραφής σε ένα αρχείο
- Το ευρετήριο καθορίζεται (συνήθως) σε ένα γνώρισμα του αρχείου που καλείται πεδίο ευρετηριοποίησης (indexing field)
- Το ευρετήριο αρχείου είναι ένα διατεταγμένο αρχείο με σταθερού μήκους εγγραφές
- Για ένα αρχείο μπορούμε να ορίσουμε περισσότερα από ένα ευρετήρια



Πρωτεύον Ευρετήριο

- Ευρετήριο ορισμένο επί του πεδίου κλειδιού διάταξης ενός διατεταγμένου αρχείου εγγραφών
- Το πρωτεύον ευρετήριο είναι ένα *μη πυκνό* ευρετήριο
  - **Πυκνό ευρετήριο**: μια καταχώρηση για κάθε τιμή αναζήτησης (εγγραφή του αρχείου δεδομένων)
  - **Αραιό ευρετήριο**: καταχωρήσεις μόνο για μερικές από τις τιμές αναζήτησης
- Το κλειδί αναζήτησης του ευρετηρίου είναι συνήθως το πρωτεύον κλειδί των εγγραφών.

ΑΣΚΗΣΗ 2 - Αναζήτηση χωρίς ευρετήριο σε διατεταγμένο αρχείο

- Διατεταγμένο αρχείο
- r = 30000 εγγραφές
- Μέγεθος μπλοκ B=1024 byte
- Εγγραφές σταθερού μεγέθους – μη εκτεινόμενες
- Μήκος εγγραφής R=100 byte

Ποιος ο απαιτούμενος αριθμός προσπελάσεων μπλοκ στον δίσκο για να βρεθεί η ζητούμενη εγγραφή στην ΜΠ;

1. Παράγοντας σελιδοποίησης
   - $bfr = \lfloor (B/R) \rfloor = 10$ εγγραφές ανά μπλοκ
2. Απαιτούμενος αριθμός μπλοκ
   - $b = \lceil (r/bfr) \rceil = 3000$ μπλοκ
3. Δυαδική αναζήτηση
   - $\lceil \log_2 b \rceil = \lceil \log_2 3000 \rceil =$ 12 προσπελάσεις μπλοκ

ΑΣΚΗΣΗ 2 – βελτίωση στην αναζήτηση με χρήση ευρετηρίου

- Υποθέστε για το ίδιο αρχείο ότι
  - Μήκος πεδίου κλειδιού διάταξης αρχείου V=9byte
  - Μήκος δείκτη block P=6 byte

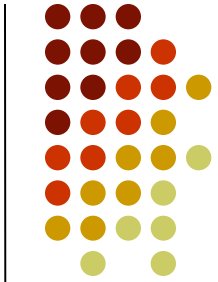
Ποιος ο απαιτούμενος αριθμός προσπελάσεων μπλοκ στον δίσκο για να βρεθεί η ζητούμενη εγγραφή στην ΜΠ;

- Μέγεθος κάθε καταχώρησης στο αρχείο ευρετηρίου
  - Ri=9+6=15byte
- Παράγοντας ομαδοποίησης στο ευρετήριο $bfri=\lfloor(B/R_i)\rfloor$
  - bfri=1024/15=68 καταχωρήσεις ανά μπλοκ
- Ολικός αριθμός καταχωρήσεων του ευρετηρίου είναι r_i
  - r_i=#των μπλοκ του αρχείου δεδομένων =3000 (από πριν)
- αριθμός των μπλοκ που απαιτούνται για το ευρετήριο είναι $b_i=\lceil(r_i/bfr_i)\rceil=\lceil(3000/68)\rceil=45$ block
- Για μια δυαδική αναζήτηση θα απαιτούνταν $\lceil(\log_2 b_i)\rceil=\lceil(\log_2 45)\rceil=6$ προσπελάσεις block
- Για την αναζήτηση μιας εγγραφής χρειαζόμαστε μία επιπλέον προσπέλαση στο αρχείο δεδομένων. Άρα → 7 προσπελάσεις block

Θεωρία - Δευτερεύον Ευρετήριο

- Οι εγγραφές του αρχείου δεν είναι διατεταγμένες ως προς το πεδίο ευρετηριοποίησης
- το πεδίο ευρετηριοποίησης μπορεί είτε να είναι υποψήφιο κλειδί είτε όχι
- **Πυκνό ευρετήριο**: μία καταχώρηση για κάθε εγγραφή του αρχείου δεδομένων
- Μήκος εγγραφής σταθερό ή μεταβλητό
- Μία εγγραφή ευρετηρίου για κάθε τιμή του πεδίου ευρετηριοποίησης + ένα ενδιάμεσο επίπεδο για την διαχείριση των πολλαπλών δεικτών

ΑΣΚΗΣΗ 3 – Γραμμική αναζήτηση



- Διατεταγμένο αρχείο
- r = 30000 εγγραφές
- Μέγεθος μπλοκ B=1024 byte
- Εγγραφές σταθερού μεγέθους – μη εκτεινόμενες
- Μήκος εγγραφής R=100 byte

Ποιος ο απαιτούμενος αριθμός προσπελάσεων μπλοκ στον δίσκο για να βρεθεί η ζητούμενη εγγραφή στην ΜΠ με χρήση γραμμικής αναζήτησης;

- Παράγοντας σελιδοποίησης
  - bfr= $\lfloor (B/R) \rfloor$ =10 εγγραφές ανά μπλοκ
- Απαιτούμενος αριθμός μπλοκ
  - b= $\lceil (r/bfr) \rceil$ =3000 μπλοκ
- Για γραμμική αναζήτηση στο αρχείο στο αρχείο θα χρειαζόμασταν
  - b/2=3000/2=1500 προσπελάσεις μπλοκ στην ΜΠ.

ΑΣΚΗΣΗ 3 – Χρήση δευτερεύοντος ευρετηρίου

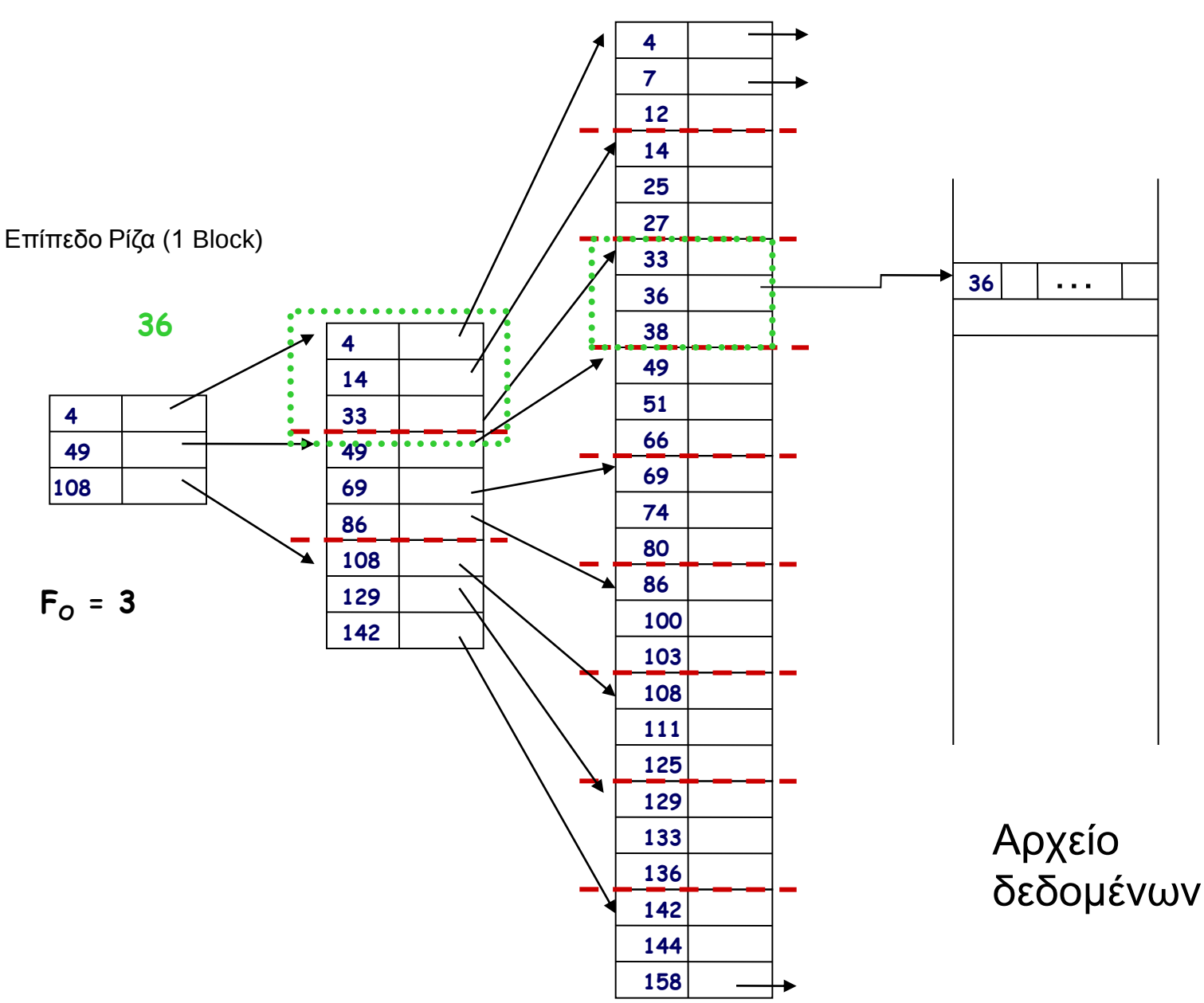
- Κατασκευάζουμε δευτερεύον ευρετήριο
- Υποθέστε ότι
  - Μήκος πεδίου-κλειδί του αρχείου V=9byte
  - Μήκος δείκτη block P=6 byte
  - Δεδομένα ίδια με την προηγούμενη άσκηση

Ποιος ο απαιτούμενος αριθμός προσπελάσεων μπλοκ στον δίσκο για να βρεθεί η ζητούμενη εγγραφή στην ΜΠ με χρήση δευτερεύοντος ευρετηρίου

- Μέγεθος κάθε καταχώρησης στο ευρετήριο R_i=9+6=15byte
- Παράγοντας ομαδοποίησης στο ευρετήριο bfri= $\lfloor(B/R_i)\rfloor$ =1024/15=68 καταχωρήσεις ανά μπλοκ
- Σε ένα πυκνό δευτερεύον ευρετήριο. Ο ολικός αριθμός καταχωρήσεων r_i είναι ίσος με το πλήθος εγγραφών του αρχείου δεδομένων που είναι 30000.
- αριθμός των μπλοκ που απαιτούνται για το ευρετήριο είναι b_i= $\lceil(r_i/bfr_i)\rceil$ = $\lceil(30000/68)\rceil$ =442 block
- Για μια δυαδική αναζήτηση θα απαιτούνταν $\lceil(\log_2 b_i)\rceil$ = $\lceil(\log_2 442)\rceil$ =9 προσπελάσεις block
- Για την αναζήτηση μιας εγγραφής χρειαζόμαστε μία επιπλέον προσπέλαση άρα →10 προσπελάσεις block

Πολυεπίπεδα ευρετήρια

- Στόχος : να μειώσουμε το ευρετήριο που διεξάγουμε την αναζήτηση κατά έναν παράγοντα bfr_i =fo (παράγοντας διακλάδωσης (fan-out))
  - Αν fo>2 τοτε $\log_{fo} b_i < \log_2 b_i$ προσπελάσεις μπλοκ
  - fo σταθερό για κάθε επίπεδο
- Το πρώτο επίπεδο χρειάζεται $\lceil (r_1/f_o) \rceil$μπλοκ =r_2 = # καταχωρήσεων για το 2^o επιπέδο κτλ.
- Κάθε επίπεδο ελαττώνει τον αριθμό των καταχωρήσεων του προηγούμενου κατά έναν παράγοντα fo→ Ισχύει : $1 \leq (r_1/((fo)^t)$
  - → Ένα πολυεπίπεδο ευρετήριο με r_1 καταχωρήσεις στο πρώτο επίπεδο θα έχει t= $\lceil (\log_{fo}(r_1)) \rceil$
  - t = το πλήθος των επιπέδων που θα έχει το ευρετήριο


Επίπεδο Ρίζα (1 Block)
36
4
49
108
F_O = 3
4
14
33
49
69
86
108
129
142
4
7
12
14
25
27
33
36
38
49
51
66
69
74
80
86
100
103
108
111
125
129
133
136
142
144
158
36
. . .
Αρχείο δεδομένων

Άσκηση 4 – Χρήση πολυεπίπεδων ευρετηρίων

- Υποθέστε ότι
  - Μήκος πεδίου κλειδιού διάταξης αρχείου V=9byte
  - Μήκος δείκτη block P=6 byte
  - Λοιπά στοιχεία όμοια με τις προηγούμενες ασκήσεις

Ποιος ο απαιτούμενος αριθμός προσπελάσεων μπλοκ στον δίσκο για να βρεθεί η ζητούμενη εγγραφή στην ΜΠ με χρήση πολυεπίπεδου ευρετηρίου

  - Μέγεθος κάθε καταχώρησης στο ευρετήριο
    - R_i=9+6=15byte
  - Παράγοντας ομαδοποίησης στο ευρετήριο
    - bfri= $\llcorner(B/R_i)\lrcorner$ =1024/15=68 καταχωρήσεις ανά μπλοκ
  - Παράγοντας διακλάδωσης fo = bfri=68
    - αριθμός των μπλοκ πρώτου επιπέδου b_1= $\lceil(r_1/fo)\rceil$ = 442 block
    - αριθμός των μπλοκ δεύτερου επιπέδου b_2= $\lceil(b_1/fo)\rceil$ = 7 block
    - αριθμός των μπλοκ τρίτου επιπέδου b_3= $\lceil(b2/fo)\rceil$ = 1 block
  - Άρα t=3 και το τρίτο επίπεδο είναι το κορυφαίο
  - Για την αναζήτηση μιας εγγραφής χρειαζόμαστε μία προσπέλαση για κάθε μπλοκ κάθε επιπέδου συν ένα μπλοκ από το αρχείο δεδομένων άρα →3+1=4 προσπελάσεις block

Θεωρία Β-δέντρα (B-trees)

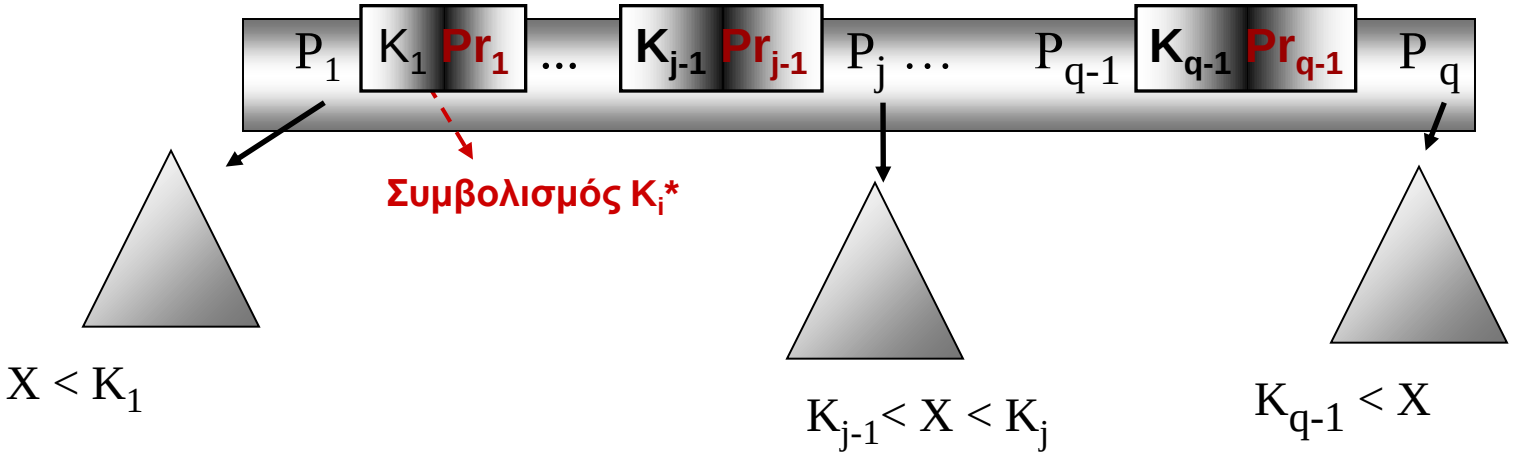
- Κάθε κόμβος του δέντρου είναι ένα block στο δίσκο
- Ισοζυγισμένο: όλοι οι κόμβοι-φύλλα στο ίδιο επίπεδο

Ένα **Β-δέντρο τάξεως (order) p** ορίζεται ως εξής:

1. Κάθε **εσωτερικός κόμβος** είναι της μορφής

$$<P_1, <K_1, Pr_1>, P_2, <K_2, Pr_2>, \ldots <K_{q-1}, Pr_{q-1}>, P_q>,$$

- $q <= p$,
- όπου P_i δείκτης δέντρου,
- K_i τιμή αναζήτησης,
- **Pr_i** δείκτης δεδομένων



B-δέντρα

1. Σε κάθε κόμβο K1 < K2 < … K q -1
2. Για όλες τις τιμές X στο υποδέντρο που δείχνει το Pj ισχύει Kj-1 < X < Kj για 1 < j < q, X < Kj για j =1, και Kj -1 < X για j = q
3. Κάθε κόμβος έχει το **πολύ** p δείκτες δέντρου
4. Κάθε κόμβος *εκτός της ρίζα* και των *φύλλων* έχει **τουλάχιστον** ⌈(p/2)⌉ δείκτες δέντρου. Η ρίζα έχει τουλάχιστον 2 εκτός αν είναι ο μόνος κόμβος του δέντρου.
5. Ένας κόμβος με q δείκτες δέντρου περιέχει q - 1 τιμές πεδίου αναζήτησης (και άρα και q - 1 δείκτες δεδομένων)
6. Όλα τα φύλλα βρίσκονται στο ίδιο επίπεδο. Τα φύλλα έχουν την ίδια δομή εκτός του ότι οι δείκτες δέντρου είναι null.

Εισαγωγή σε Β-δέντρο

- Εισαγωγή σε φύλλο;
  - σε περίπτωση υπερχείλισης προώθησε την μέση τιμή στο επάνω επίπεδο (περιοδικά)
  - Διέσπασε (split) ώστε να διατηρούνται οι ιδιότητες ενός B - tree
- Εάν υπάρχουν μέσες τιμές; (Π.χ., τάξη 4)
  - Επιλέγουμε ποιον κόμβο θα προωθήσουμε προς τα επάνω
- Το ύψος αυξάνεται όταν υπάρξει υπερχείλιση και διασπαστεί η ρίζα
  - Αυτόματη αύξηση και αναδιοργάνωση (συγκριτικά με ISAM!)

Διαγραφή

Ο αλγόριθμος συνοπτικά :

- Διέγραψε κλειδί
- Σε περίπτωση υποχείλισης μπορεί να προκληθεί συγχώνευση

Στην πράξη κάποιοι σχεδιαστές απλά αφήνουν να συμβεί υποχείλιση ...

Διαγραφή Β-δέντρα Περιπτώσεις

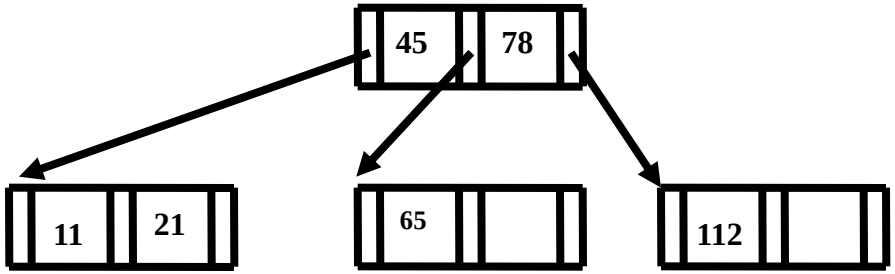
- **1η Περίπτωση:** διαγραφή κλειδιού από κόμβο φύλο - χωρίς υποχείλιση
- **2η Περίπτωση:** διαγραφή κλειδιού από εσωτερικό κόμβο - χωρίς υποχείλιση
- **3η Περίπτωση:** διαγραφή κλειδιού από εσωτερικό κόμβο - υποχείλιση και “πλούσιος” γειτονικός κόμβος
- **4η Περίπτωση:** διαγραφή κλειδιού από εσωτερικό κόμβο - υποχείλιση και “φτωχός” γειτονικός κόμβος

Άσκηση 5

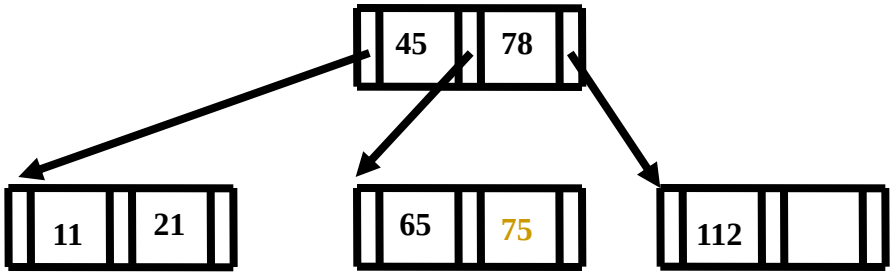
- Σε κάθε υποερώτημα της παρούσας άσκησης σας δίνεται ένα στιγμιότυπο της ανάπτυξης ενός B-tree. Εσείς καλείστε να σχεδιάσετε το δέντρο που θα προκύψει μετά από μία πράξη εισαγωγής ή διαγραφής

Άσκηση 5-1

- Εισαγωγή του στοιχείου 75

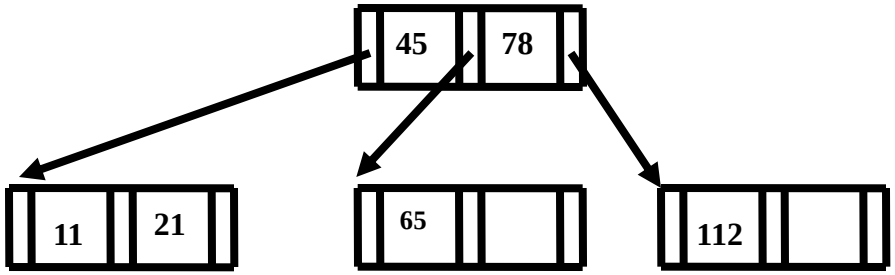


➔ *Λύση :*

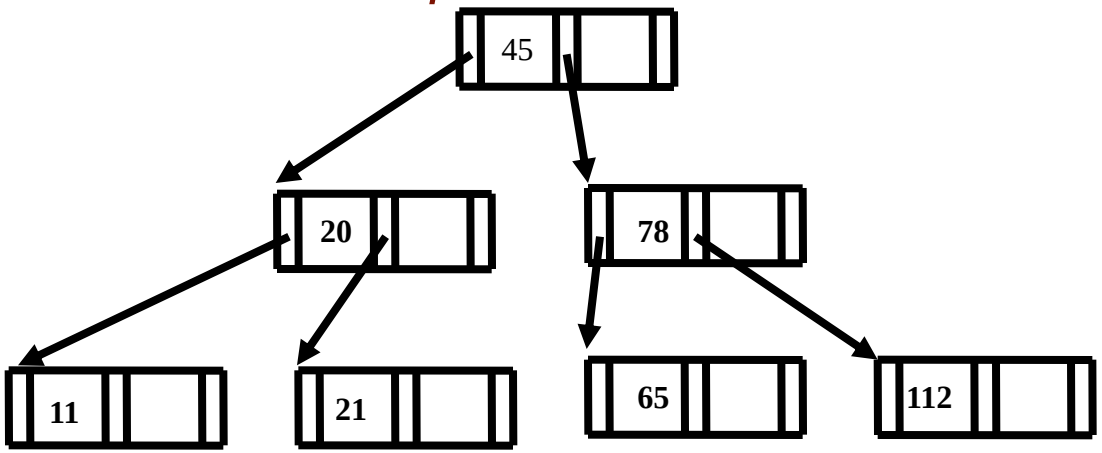


Άσκηση 5-2

- Εισαγωγή του στοιχείου 20

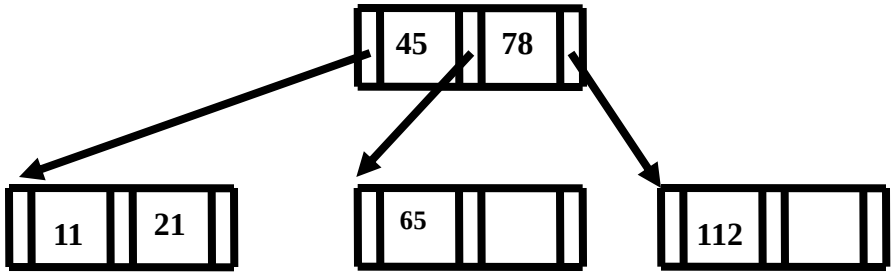


➔ *Λύση :*

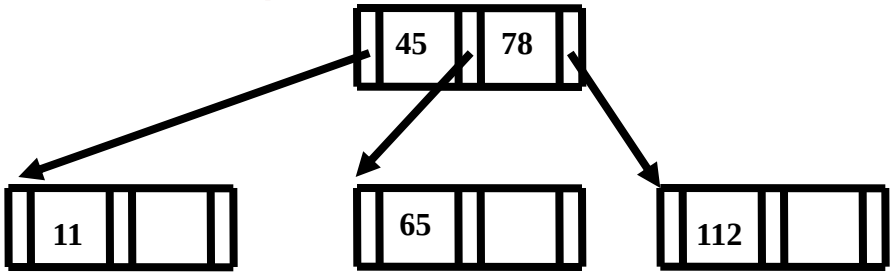


Άσκηση 5-3

- Διαγραφή του στοιχείου 21

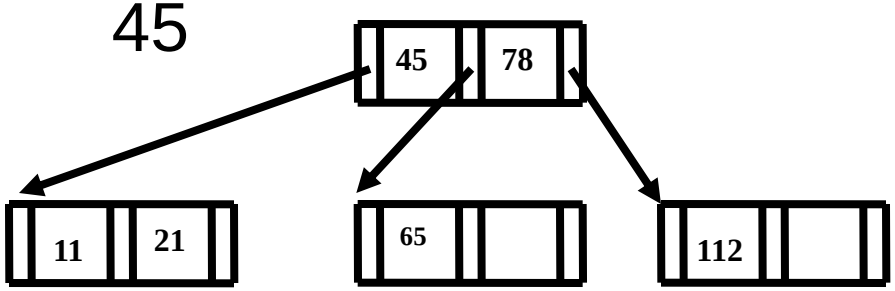


➔ *Λύση :*



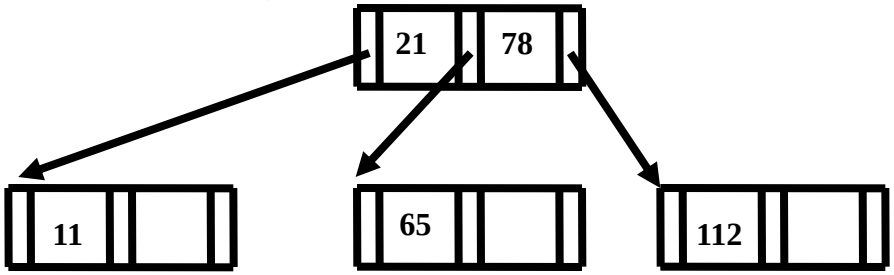
Άσκηση 5-4

- Διαγραφή του στοιχείου 45



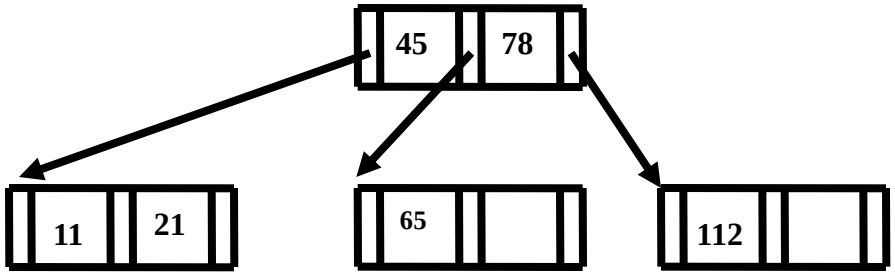
- Επέλεξε το μεγαλύτερο κλειδί από το αριστερότερο υποδέντρο (ή το μικρότερο από το δεξί υπόδεντρο)
- Κάθε διαγραφή οδηγεί τελικά σε διαγραφή κλειδιού από κάποιο κόμβο φύλλο

➔ *Λύση :*



Άσκηση 5-5

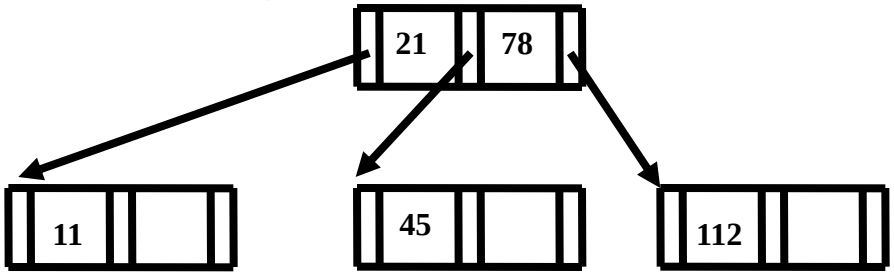
- Διαγραφή του στοιχείου 65



'Πλούσιος γείτονας' = μπορεί να δώσει ένα κλειδί χωρίς υπερχείλιση

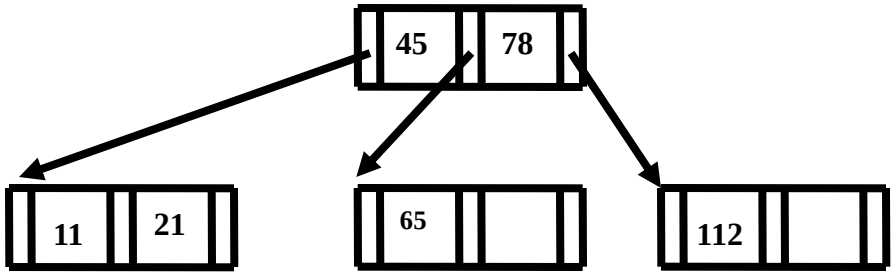
'Δανείζει' ένα κλειδί : πάντα ΔΙΑΜΕΣΟΥ του ΜΗΤΡΙΚΟΥ ΚΟΜΒΟΥ!

➔ *Λύση :*

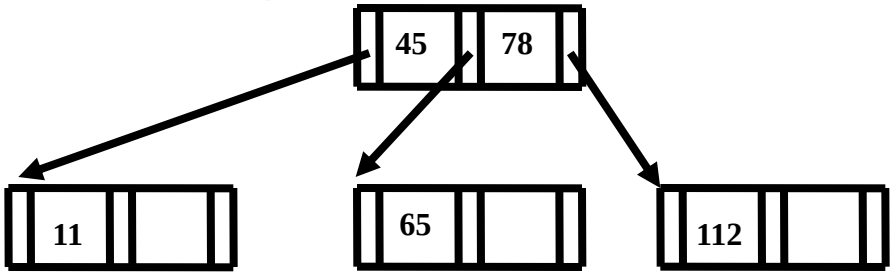


Άσκηση 5-6

- Διαγραφή του στοιχείου 21

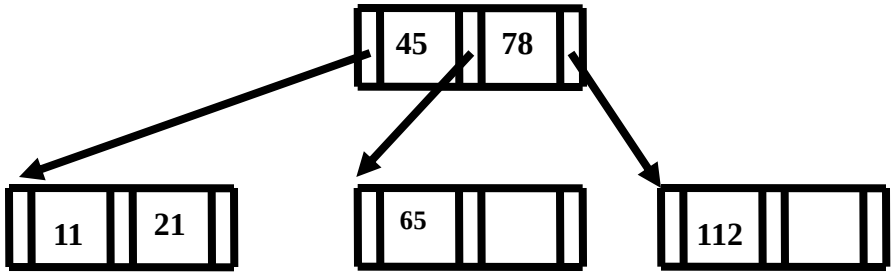


➔ *Λύση :*



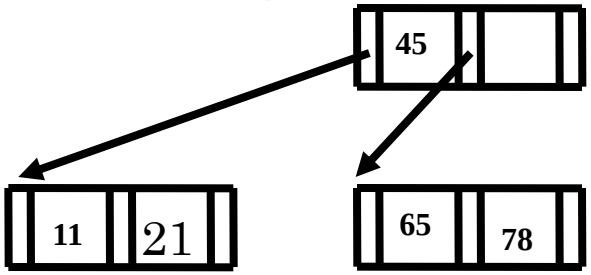
Άσκηση 5-7

- Διαγραφή του στοιχείου 112



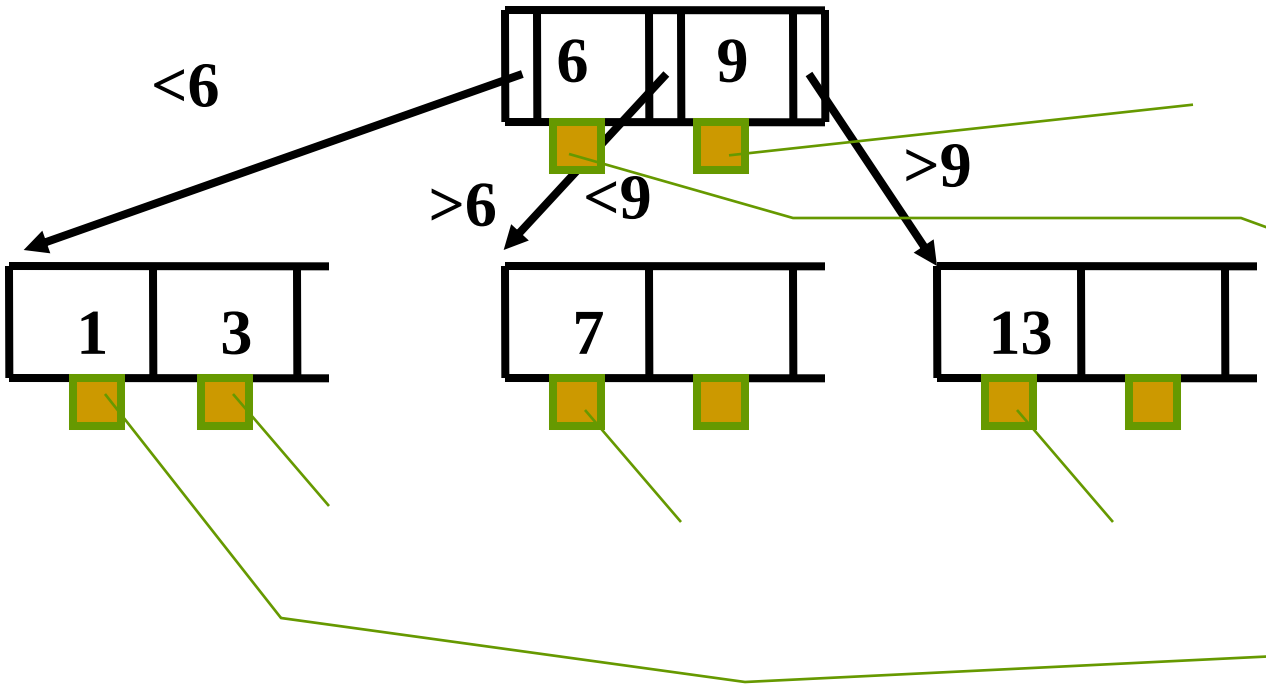
➔ *Λύση :*

- Συγχώνευσε, αποσπώντας ένα κλειδί από τον **πατέρα**
- Ακριβώς το αντίθετο από την εισαγωγή: ‘διέσπασε και προώθησε προς τα επάνω’, vs. ‘Συγχώνευσε και προώθησε προς τα κάτω’



Β-δέντρα πρακτικά:

Πρακτικά:



ΑΦΜ	.....	
3		
7		
6		
9		
1		

B-Trees vs B+-Trees

- Πλεονεκτήματα ευρετηρίων που υλοποιούνται με B-Trees:
  - Μπορεί να χρησιμοποιηθούν λιγότεροι κόμβοι από ένα αντίστοιχο B^+-Tree.
  - Μερικές φορές είναι δυνατό να βρούμε την τιμή του κλειδιού αναζήτησης πριν προσπελάσουμε τον κόμβο φύλλο.
- Μειονεκτήματα ευρετηρίων που υλοποιούνται με B-Trees:
  - Μόνο ένα μικρό κλάσμα των κλειδιών αναζήτησης μπορεί να βρεθούν νωρίς
  - Οι εσωτερικοί κόμβοι είναι συνήθως μεγαλύτεροι, έτσι ο βαθμός διακλάδωσης(fan-out)**fo** μειώνεται. Έτσι, τα B-Trees έχουν συνήθως μεγαλύτερο βάθος από τα αντίστοιχα B^+-Trees.
  - Η εισαγωγή και διαγραφή περισσότερο πολύπλοκη απ ότι στα B^+-Trees
  - Υλοποίηση δυσκολότερη από ότι για B^+-Trees.
- Τυπικά τα πλεονεκτήματα των B-Trees δεν καταργούν τα μειονεκτήματα

Θεωρία - Τεχνική κατακερματισμένων εγγραφών

- Εσωτερικός κατακερματισμός
  - Έχουμε Μ θέσεις κατακερματισμού
- **Αναδίπλωση** : εφαρμογή μιας αριθμητικής πράξης σε τμήμα/τα της τιμής του πεδίου κατακερματισμού για τον υπολογισμό της διεύθυνσης κατακερματισμού
- **Σύγκρουση**: η τιμή του πεδίου κατακερματισμού αντιστοιχίζεται σε διεύθυνση που ήδη χρησιμοποιείται
  - Ανοιχτή διευθυνσιοδότηση
    - Σε διαδοχική θέση
  - Αλυσιδωτή σύνδεση
    - Διατηρούμε περιοχές θέσεων υπερχείλισης
  - Πολλαπλός κατακερματισμός
    - Εφαρμογή δεύτερης συνάρτησης κατακερματισμού
    - Ανοιχτή διεύθυνση

Θεωρία - Τεχνική κατακερματισμένων εγγραφών

- Εξωτερικός κατακερματισμός (για αρχεία δίσκου)
  - Κάδος= block ή συστάδα από block
  - Η συνάρτηση h(K) απεικονίζει κλειδί σε αριθμό κάδου
- Στατικός κατακερματισμός
  - (χώρος διευθύνσεων σταθερός)
- Δυναμικός κατακερματισμός
  - Επεκτατός κατακερματισμός (Extendible hashing)
    - Διατηρείται πίνακας με 2^d διευθύνσεις κάδων d=ολικό βάθος
    - 2 προσπελάσεις block →1στον κατάλογο 1 στον αντίστοιχο κάδο
  - Γραμμικός κατακερματισμός (Linear hashing)
    - Επιτρέπουμε σε ένα αρχείο κατακερματισμού να επεκτείνει ή να συρρικνώνει τους κάδους του δυναμικά
    - Ύπαρξη ξεχωριστής αλυσίδας υπερχείλισης για κάθε κάδο

Θεωρία - Γραμμικός κατακερματισμός

- Αρχικά υπάρχουν M κάδοι 0,1 ...M-1
- Συνάρτηση κατακερματισμού h(K)=K mod M
- Υπερχείλιση στον κάδο 0
  - Δημιουργία νέου κάδου M
  - Οι εγγραφές του κάδου 0 κατανέμονται στους δύο κάδους (h_{i+1}(K)=K mod 2M
  - n = N→N+1 όταν συμβεί διάσπαση
  - Αν n=M όλοι οι κάδοι έχουν διασπαστεί
- Ανάκτηση μιας εγγραφής με τιμή κατακερματισμού K

1. Εφαρμόζουμε h_i στο K
2. Αν h_i(K) <n τότε h_{i+1}(K) διότι ο κάδος έχει διαπαστεί
3. Αν n =M → n=0 (όλοι οι αρχικοί κάδοι έχουν διασπαστεί)

Θεωρία - Γραμμικός κατακερματισμός

- Η διάσπαση μπορεί να ελέγχεται από τον παράγοντα φόρτωσης αντί να γίνεται μετά από κάθε υπερχείλιση
- Παράγοντας φόρτωσης αρχείου : μας δίνει τον αριθμό των κάδων που χρησιμοποιούνται

$$l = r / (bfr * N)$$

- r= το τρέχουν πλήθος εγγραφών.
- bfr=το μέγιστο πλήθος εγγραφών που μπορούν να χωρέσουν σε έναν κάδο.
- N=το τρέχον πλήθος κάδων του αρχείου.

Άσκηση 6

- Υποθέστε ότι θέλουμε να δημιουργήσουμε ένα γραμμικό αρχείο κατακερματισμού με παράγοντα φόρτωσης 0.7 και παράγοντα ομαδοποίησης 20 εγγραφές ανά κάδο που αρχικά θα περιέχει 120.000 εγγραφές.
  - Πόσους κάδους πρέπει να τοποθετήσουμε στην πρωτεύουσα περιοχή;
  - Πόσα μπιτ πρέπει να χρησιμοποιηθούν για τις διευθύνσεις κάδων

Δεδομένα

- l = 0.7
- Bfr=20
- r=120.000

Λύση:

- #Κάδων → N=r/(bfr*l)=120.000/(20*0.7) = 8572….
- #bit για τις διευθύνσεις κάδων:
  - $2^X>8572 \rightarrow 2^X=2^{14}>8572>2^{13}$
  - →X=14

Τέλος

- Ευχαριστώ!!!