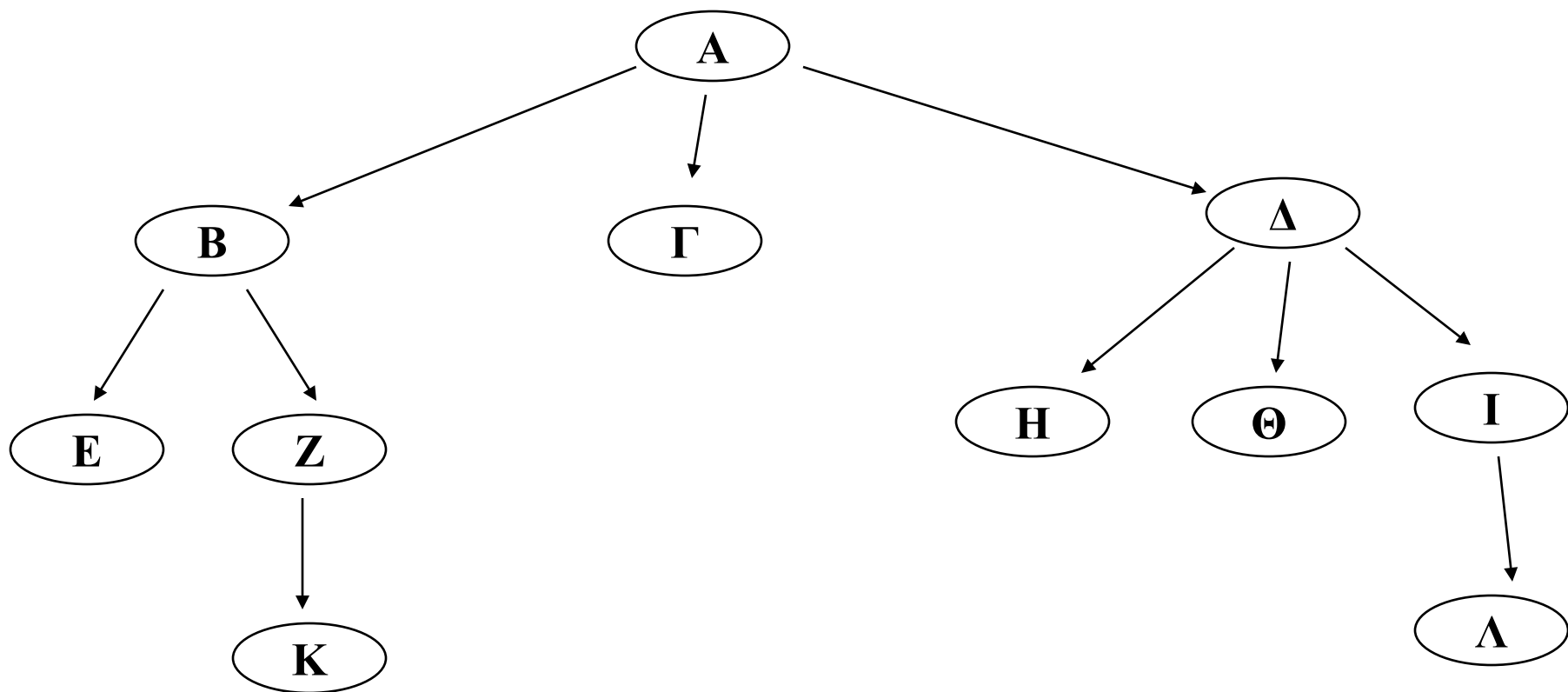
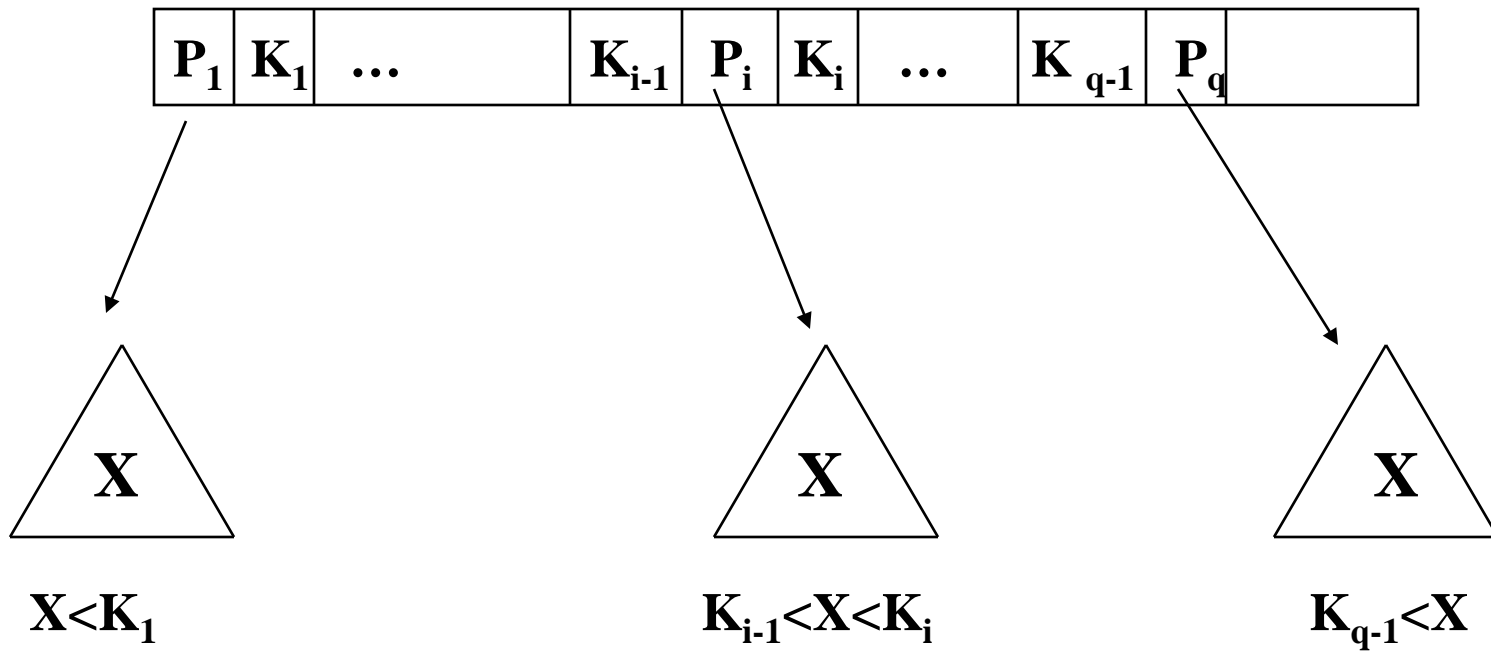


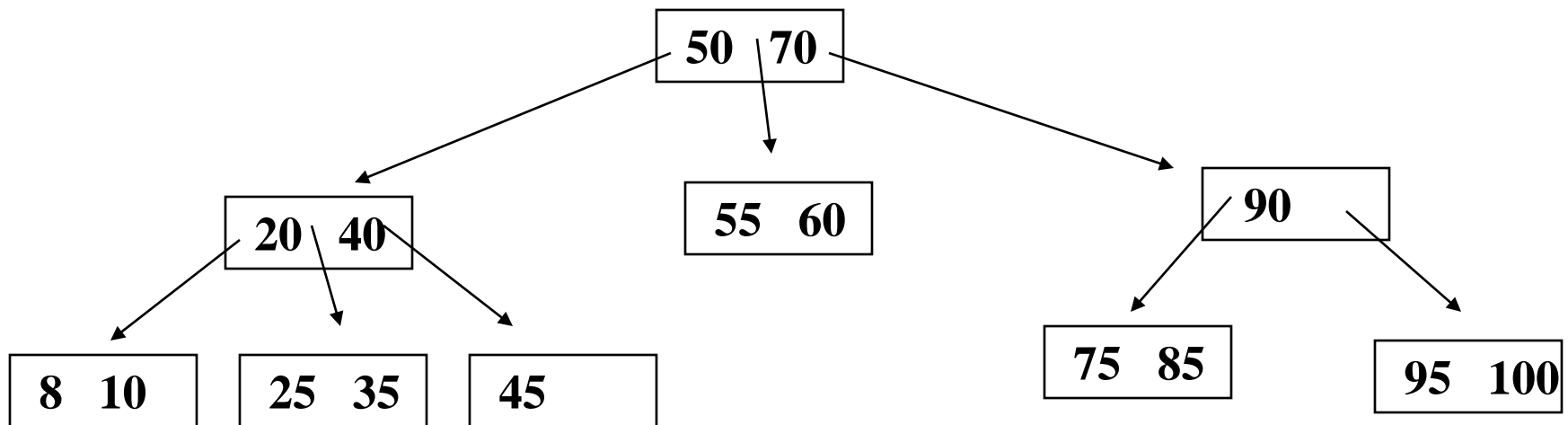
Δυναμικά Πολυεπίπεδα Ευρετήρια (B-δένδρα)



Δένδρο αναζήτησης είναι ένας ειδικός τύπος δένδρου που χρησιμοποιείται για να καθοδηγήσει την αναζήτηση μιας εγγραφής όταν δίνεται η τιμή ενός πεδίου της.

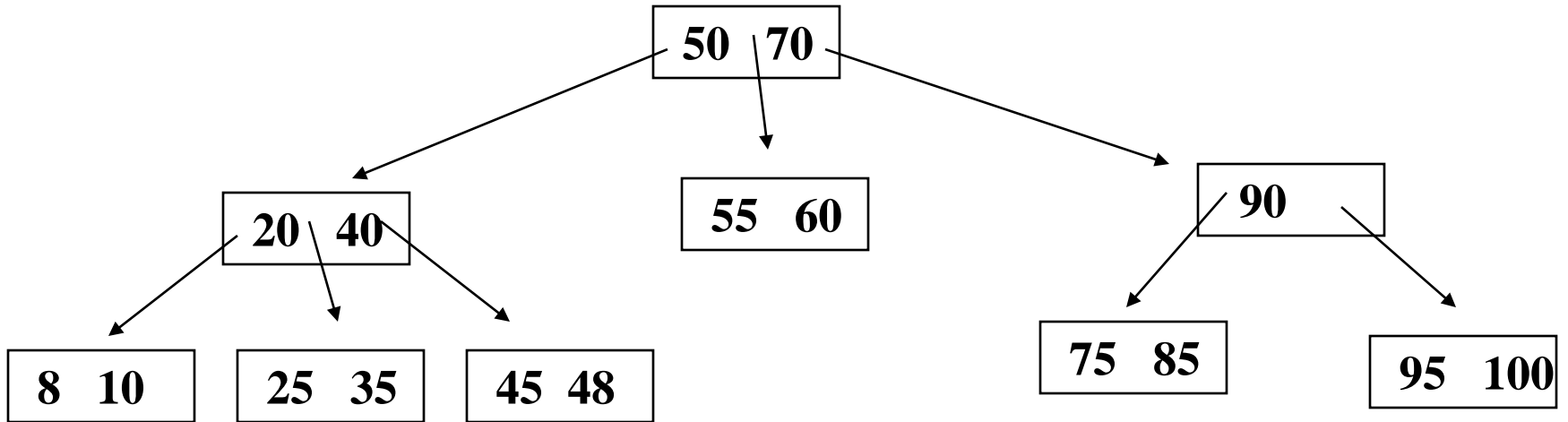
$$\mathbf{K}_1 < \mathbf{K}_2 < \dots < \mathbf{K}_{q-1}$$



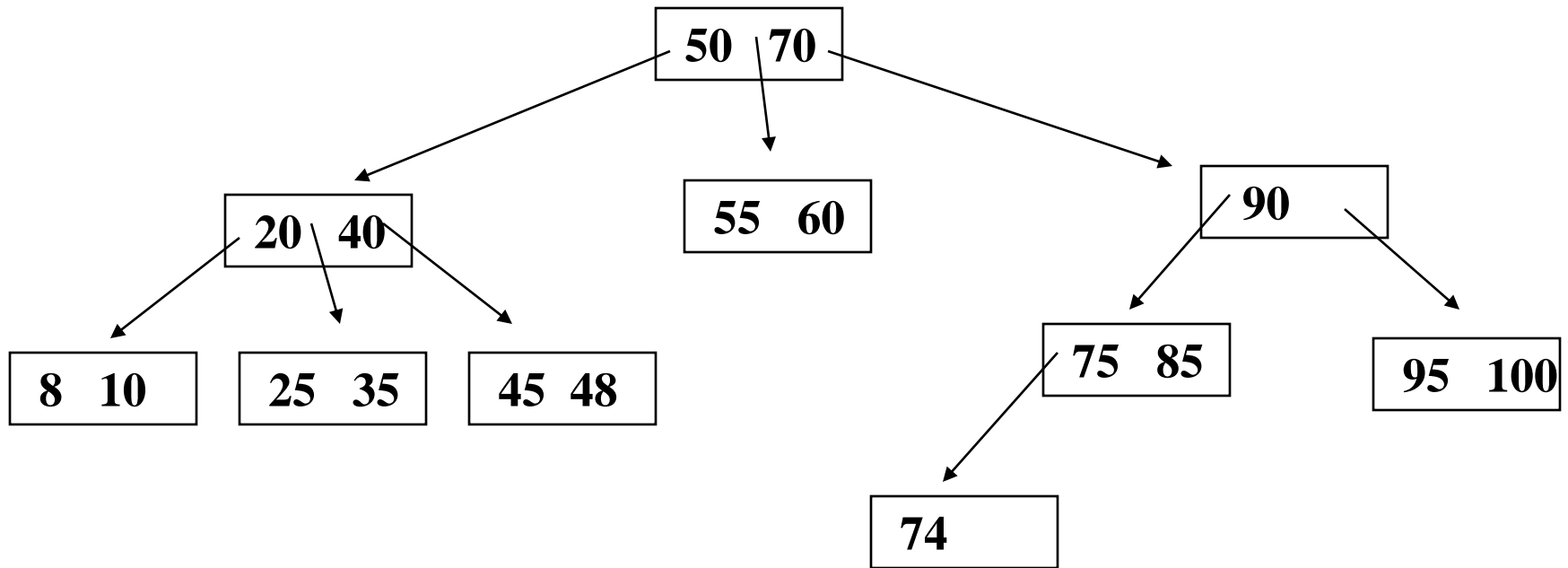


Έστω ότι ο κόμβος χωράει 2 κλειδιά

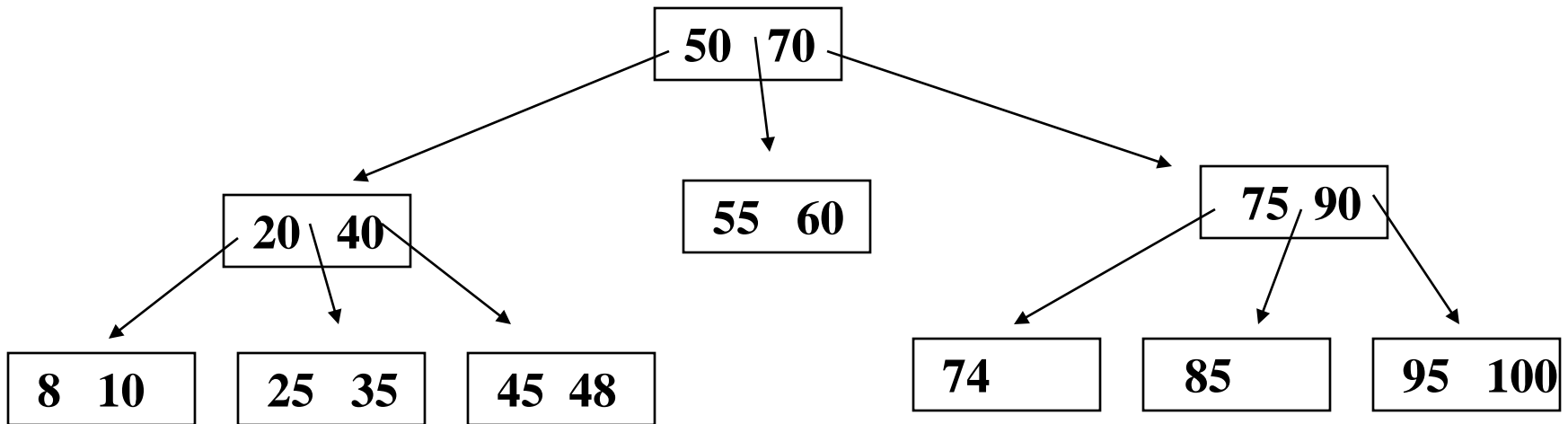
Εισαγωγή 48



Εισαγωγή 74



Όμως έτσι μπορεί να μην έχουμε καλή απόδοση ενώ θα μπορούσαμε να έχουμε



Αν το δένδρο είναι τάξεως m σε κάθε κόμβο μπορεί να περιέχει το μέγιστο $m-1$ κλειδιά. Επομένως ένα δένδρο αναζήτησης ύψους h περιέχει το μέγιστο

$$(m-1)+m(m-1)+m^2(m-1)+\dots+m^{h-1}(m-1)=(m-1)(1+m+m^2+\dots+m^{h-1})$$

$$=m^h-1$$

Επομένως για N κλειδιά το καλλίτερο δένδρο που μπορεί να δημιουργηθεί είναι $N = m^h - 1$ δηλαδή το πιο κοντό θα έχει ύψος

$$h = \log_m(N+1)$$

Όμως επειδή η χρήση είναι δυναμική δεν είναι πρακτικό με κάθε εισαγωγή και διαγραφή να γίνεται αναδιοργάνωση του δένδρου ώστε να έχω την καλλίτερη δυνατή απόδοση

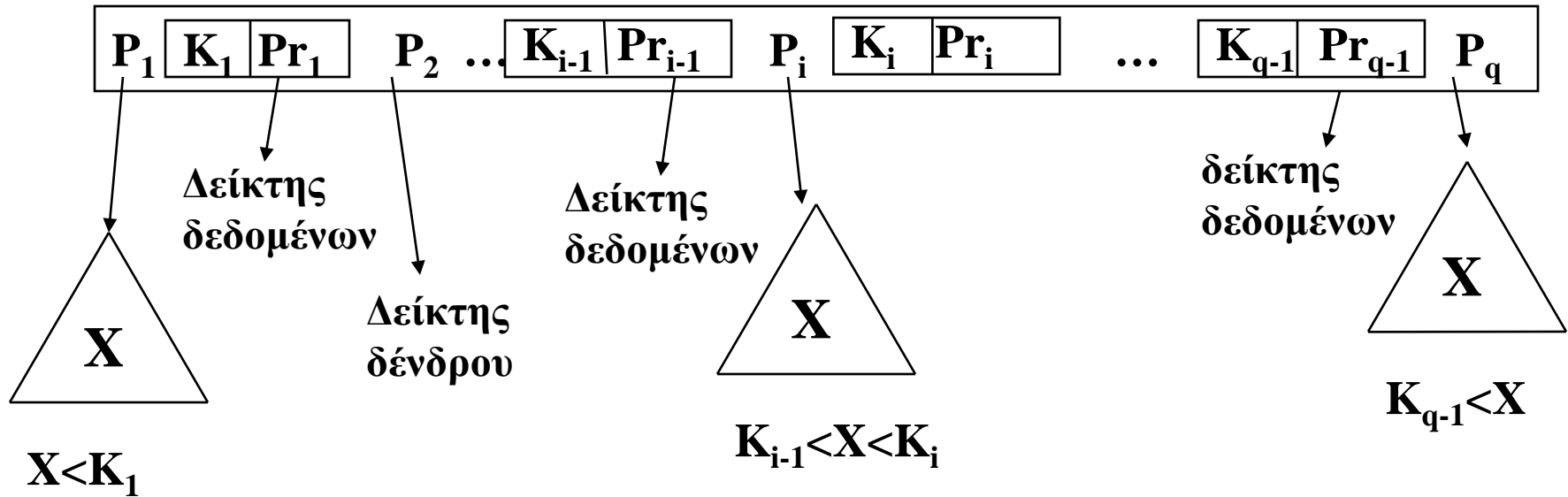
Θέλουμε να μπορεί να γίνεται εύκολα η εισαγωγή αλλά ταυτόχρονα το δένδρο που δημιουργείται να είναι όσο το δυνατόν ισοζυγισμένο με την έννοια οι κόμβοι φύλα να είναι στο ίδιο επίπεδο. Το να γέρνει προς μια πλευρά το δένδρο οδηγεί σε απόδοση που πλησιάζει αυτήν του ταξινομημένου αρχείου. Επίσης θέλουμε να γίνεται η καλλίτερη δυνατή χρήση του χώρου. Οι διαγραφές δημιουργούν κενά στους κόμβους που μπορεί να αφήσουν το δένδρο με σχεδόν κενούς κόμβους.

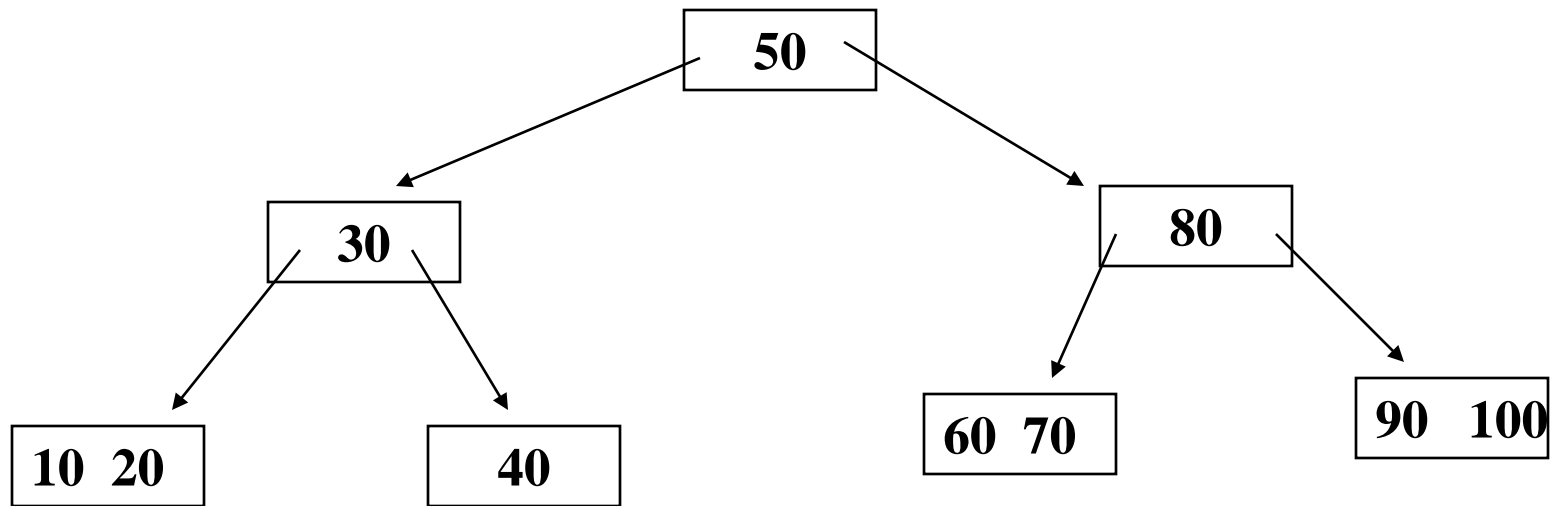
Το B-δένδρο (και οι παραλλαγές του) αντιμετωπίζει με επιτυχία αυτά τα δύο προβλήματα

B-δένδρο τάξεως p

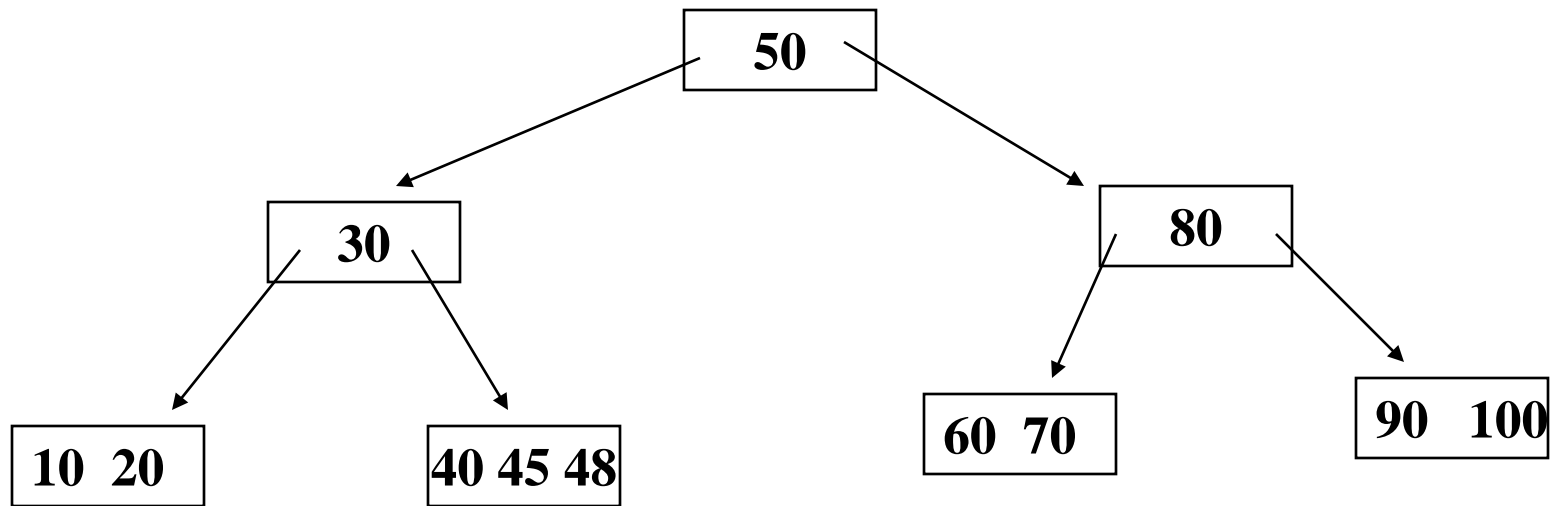
1. Κάθε εσωτερικός κόμβος είναι της μορφής $\langle P_1, \langle k_1, Pr_1 \rangle, P_2, \langle K_2, Pr_2 \rangle, \dots, \langle k_{q-1}, Pr_{q-1} \rangle, P_q \rangle$ $q \leq p$. Τα P_i είναι δείκτες δένδρου ενώ τα Pr_i δείκτες δεδομένων.
2. Σε κάθε κόμβο ισχύει: $K_1 < K_2 < \dots < K_{q-1}$.
3. Για όλα τα κλειδιά X στο υποδένδρο που δείχνει το P_i ισχύει $K_{i-1} < X < K_i$ $1 < i < q$, $X < K_i$ για $i=1$ και $K_{i-1} < X$ για $i=q$
4. Κάθε κόμβος έχει το πολύ p δείκτες δένδρου.
5. Κάθε κόμβος εκτός της ρίζας και των φύλων έχει τουλάχιστον $\lceil p/2 \rceil$ δείκτες δένδρου. Ο κόμβος της ρίζας έχει τουλάχιστον δύο δείκτες δένδρου εκτός αν είναι ο μοναδικός κόμβος του δένδρου.
6. Ένας κόμβος με q δείκτες δένδρου περιέχει $q-1$ κλειδιά.
7. Όλοι οι κόμβοι φύλα είναι στο ίδιο επίπεδο. Οι κόμβοι φύλα έχουν την ίδια δομή με τους εσωτερικούς κόμβους μόνο που οι δείκτες δένδρου έχουν τιμή null.

Κόμβος του B-δένδρου

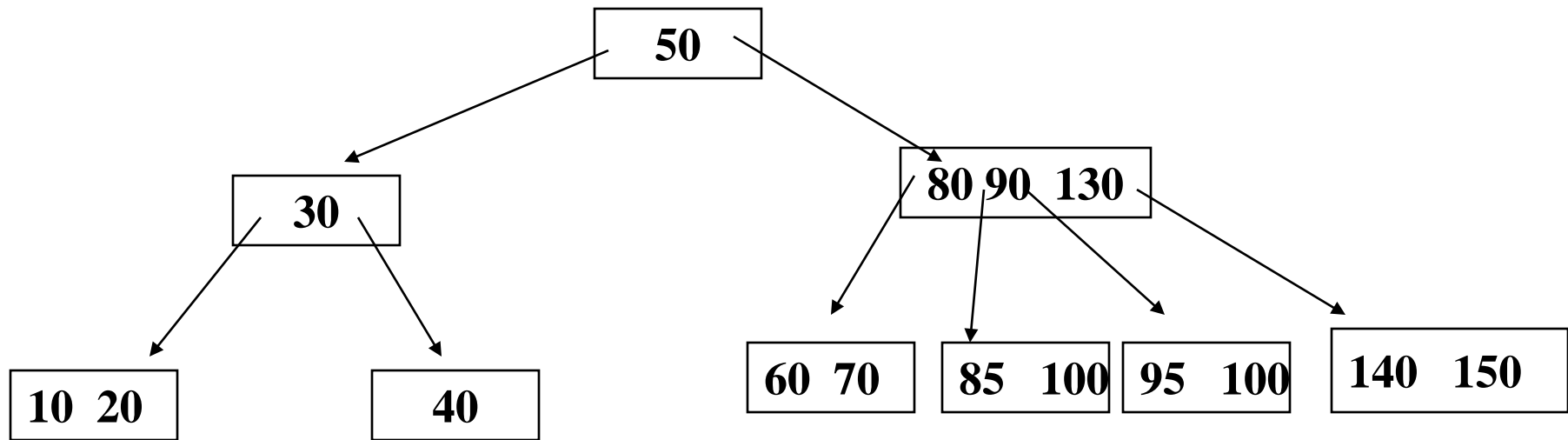




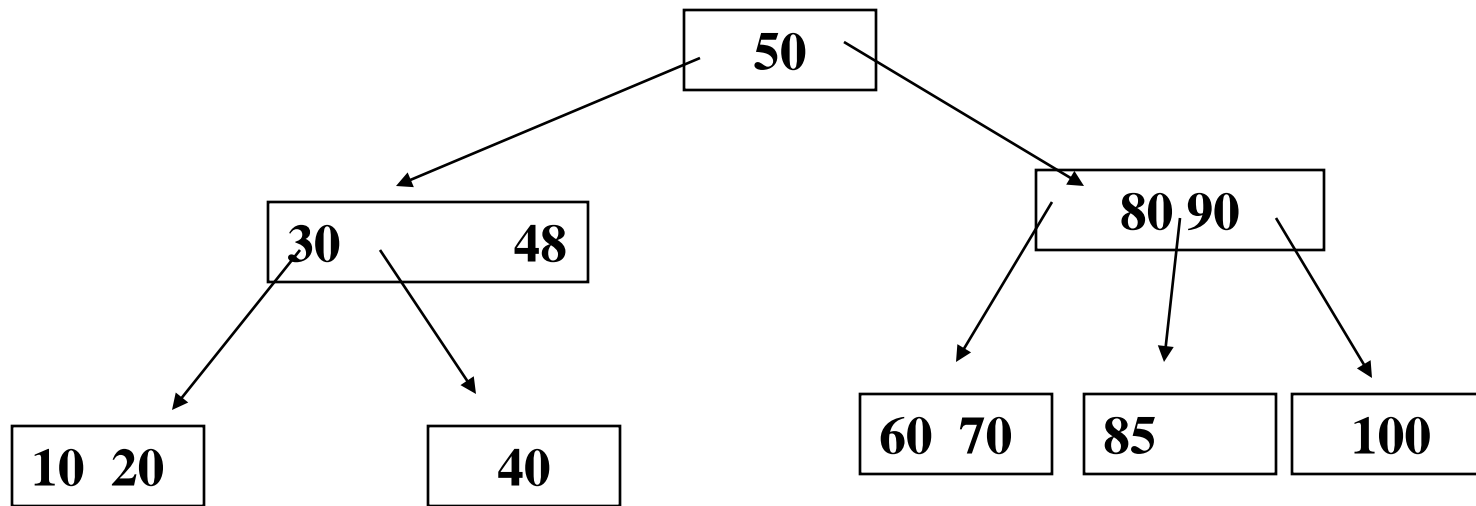
Το δένδρο τάξεως 3 που σημαίνει ότι κάθε κόμβος έχει 1 ή 2 κλειδιά και 2 ή 3 παιδιά



Αυτό όμως δεν είναι



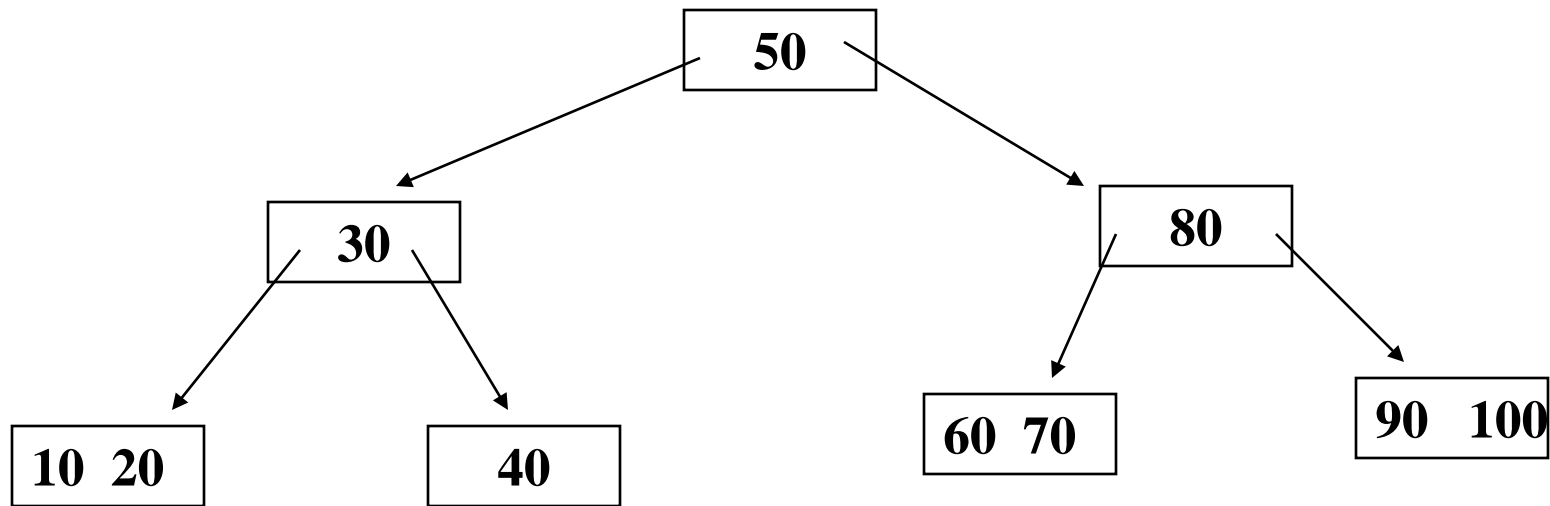
Ούτε αυτό είναι



Ούτε αυτό είναι B

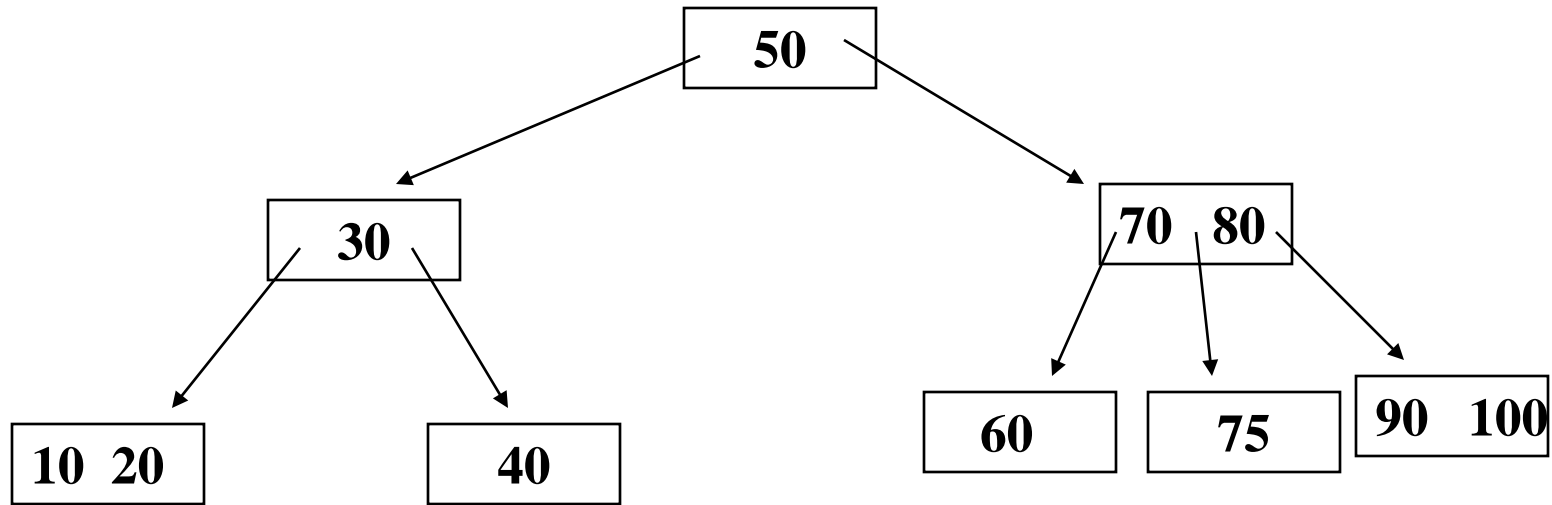
Εισαγωγή του κλειδιού X στο B-δένδρο

- Σαρώνουμε το B-δένδρο μέχρι που ή θα βρούμε την τιμή X ή θα φθάσουμε σε ένα φύλο που θα έπρεπε να βρίσκεται η τιμή X.
- Αν ο κόμβος αυτός έχει λιγότερα από $p-1$ κλειδιά τότε προσθέτουμε το κλειδί στον κόμβο αυτό και τελειώσε η εισαγωγή.
- Αν ο κόμβος έχει ήδη p κλειδιά τότε δεν μπορεί να προστεθεί στον κόμβο αυτό. Στην περίπτωση αυτή χωρίζουμε τα p κλειδιά $(K_1 < K_2 < K_3 < \dots < K_q)$ σε δύο κόμβους όπου ο πρώτος έχει τα πρώτα $\lceil q/2 \rceil$ ο δεύτερος τα τελευταία $\lceil q/2 \rceil$ και το μεσαίο στοιχείο ανεβαίνει στον κόμβο γονέα σαν διαχωριστικό.
- Αν ο κόμβος γονέας έχει χώρο τότε τελειώσαμε. Αν δεν έχει τότε χωρίζεται με τον ίδιο τρόπο και αυτός και προχωράμε προς τη ρίζα. Στην χειρότερη περίπτωση θα διασπασθεί και η ρίζα και θα ανέβουμε ένα επίπεδο (θα ψηλώσει το δένδρο).

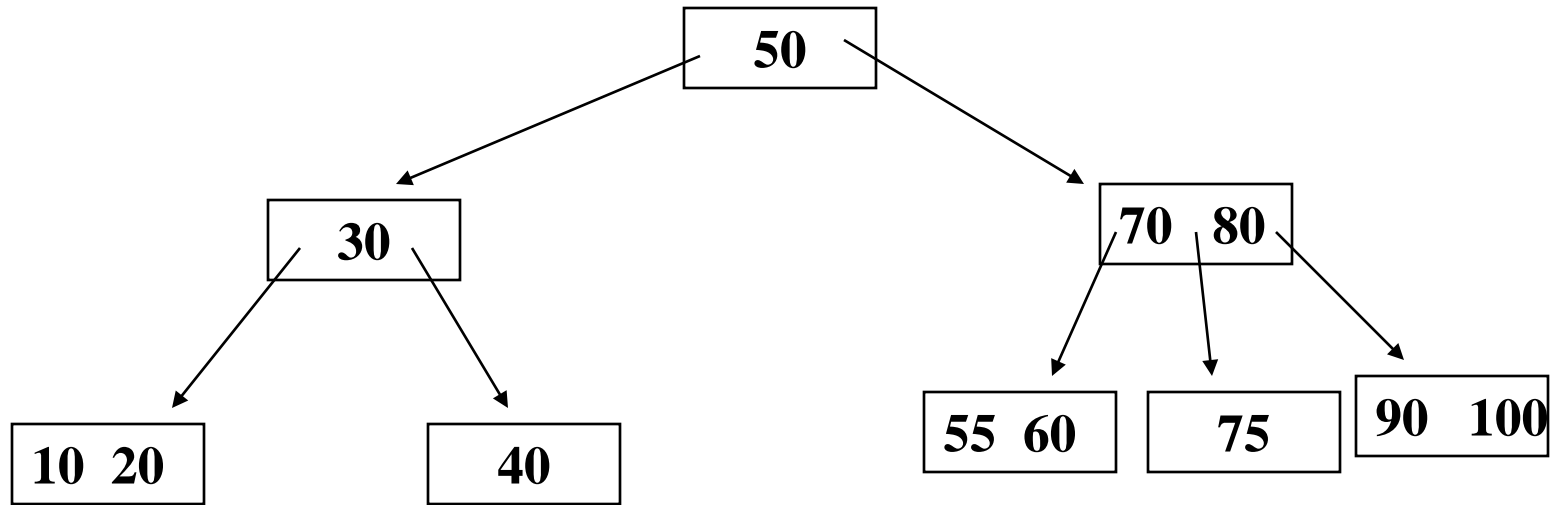


Το δένδρο τάξεως 3 που σημαίνει ότι κάθε κόμβος έχει 1 ή 2 κλειδιά και 2 ή 3 παιδιά

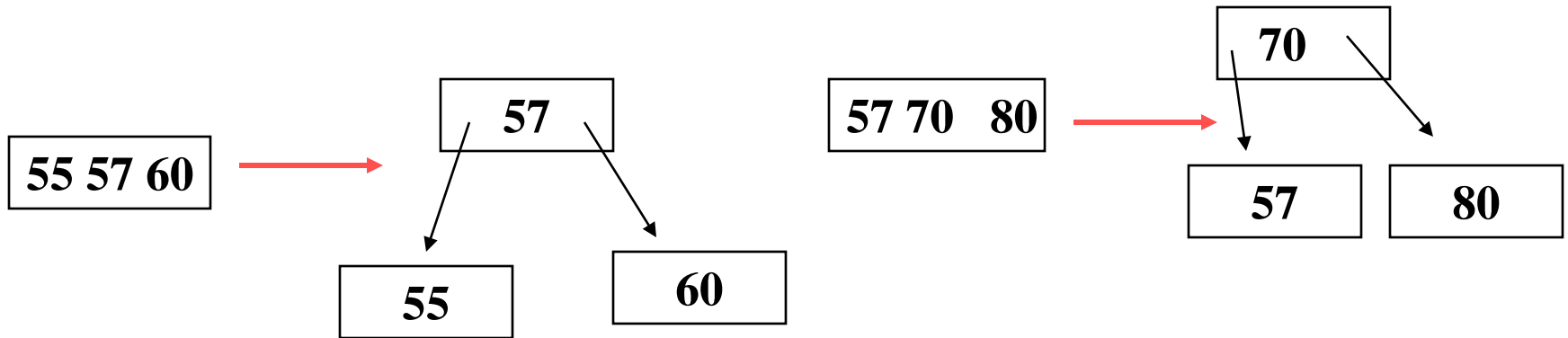
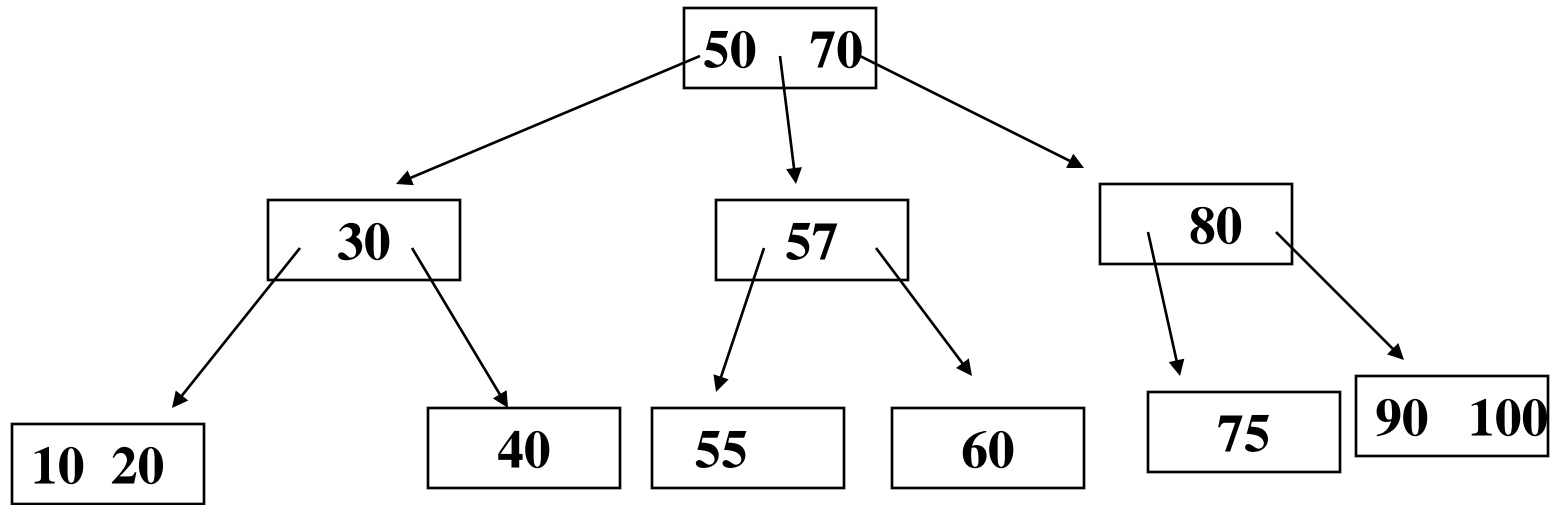
Εισαγωγή του 75



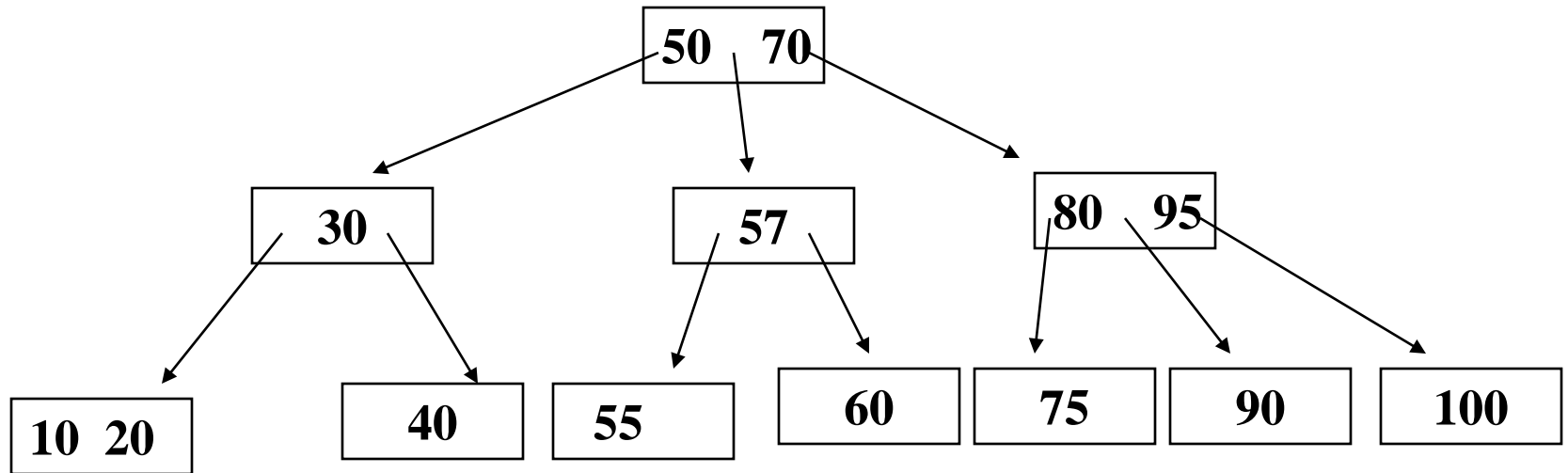
Εισαγωγή του 55



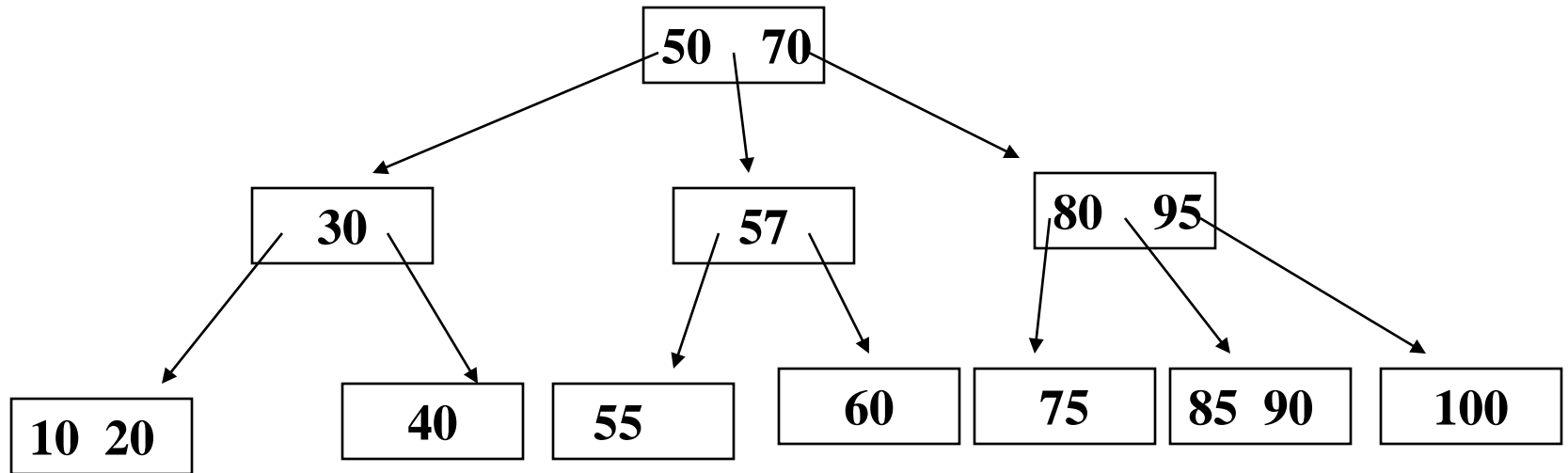
Εισαγωγή του 57



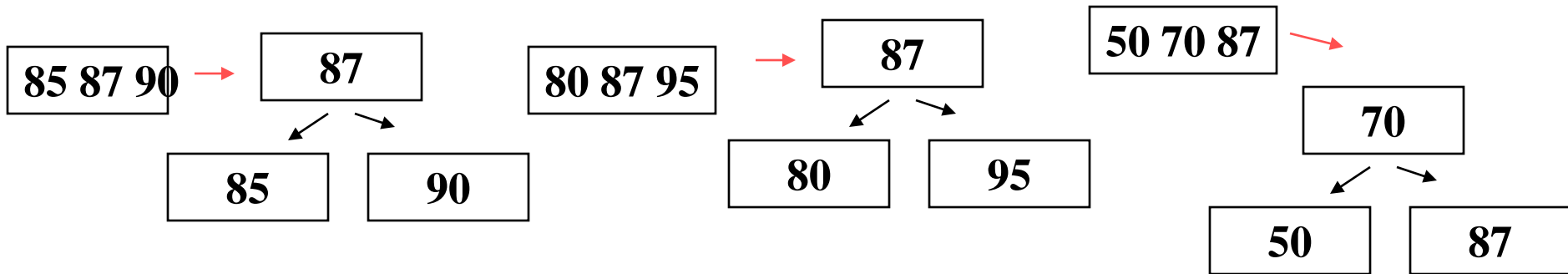
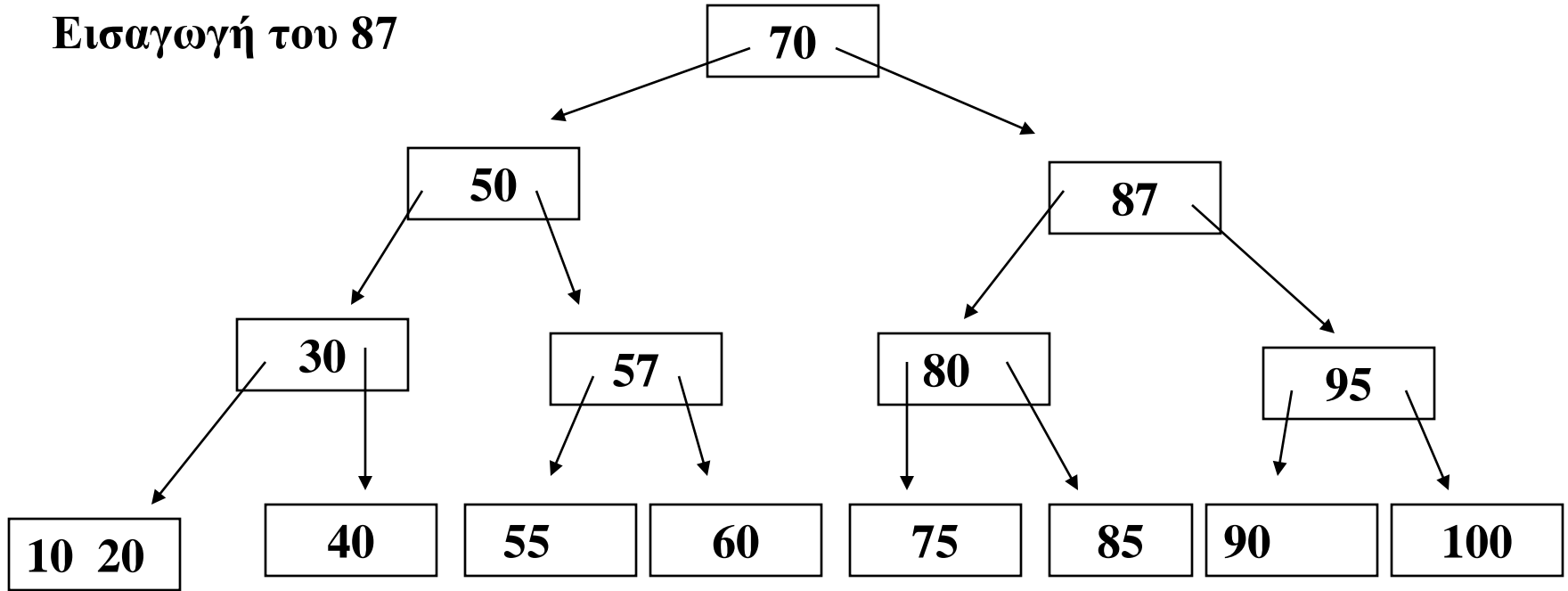
Εισαγωγή του 95



Εισαγωγή του 85



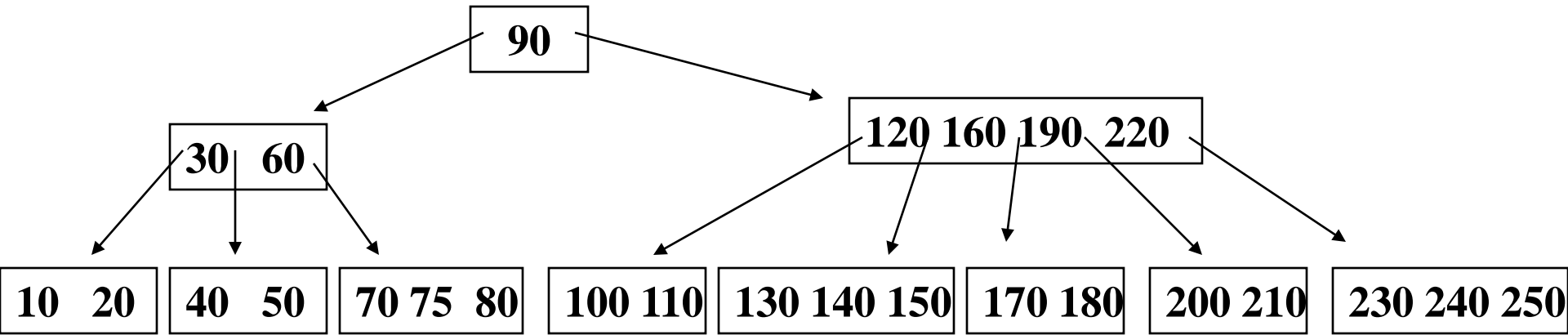
Εισαγωγή του 87



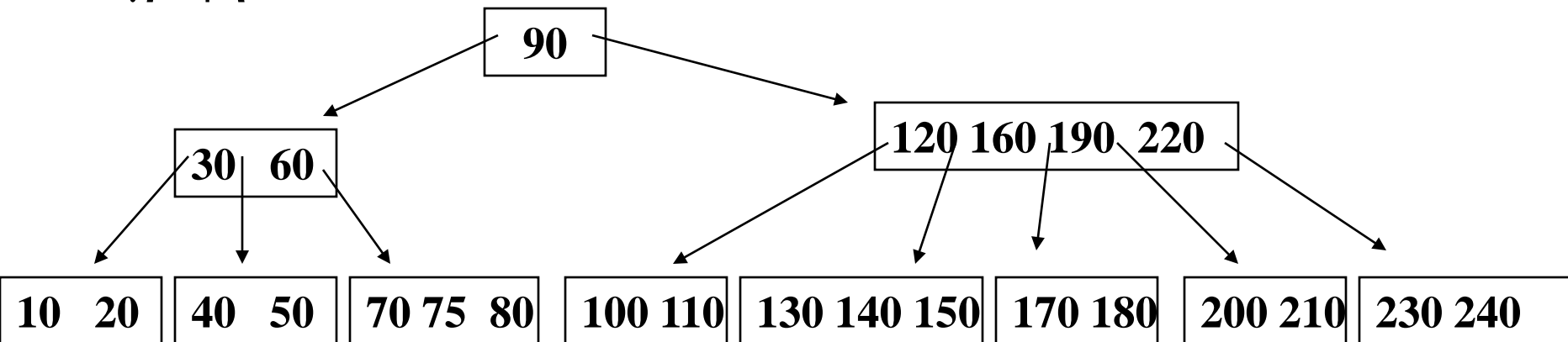
Διαγραφή ενός κλειδιού από ένα τερματικό κόμβο

- **Εντοπίζεται ο κόμβος. Αν με την διαγραφή το πλήθος των κλειδιών του κόμβου παραμένει επιτρεπτό τότε έχουμε τελειώσει.**
- **Αν ο κόμβος πέφτει κάτω από το επιτρεπόμενο τότε ανατρέχουμε στους κόμβους άμεσα αδέρφια ώστε να βρεθεί κόμβος να δανείσει ένα στοιχείο και μεταφέρουμε το στοιχείο από τον αδελφό στο γονέα και από τον γονέα στον κόμβο που έχει πρόβλημα και τελειώνουμε.**
- **Αν δεν μπορεί να δανείσει και ο αδελφός τότε οι δύο κόμβοι είναι οριακοί και μπορούμε να τους συμπτύξουμε σε έναν. Στην περίπτωση αυτή αν ο γονέας δεν έχει λιγότερα από τα επιτρεπόμενα στοιχεία τελειώσαμε. Αν έχει λιγότερα τότε επαναλαμβάνουμε την διαδικασία προς τη ρίζα. Στην περίπτωση αυτή μπορεί να φτάσουμε μέχρι τη ρίζα και μπορεί να κοντύνει το δένδρο κατά ένα επίπεδο.**

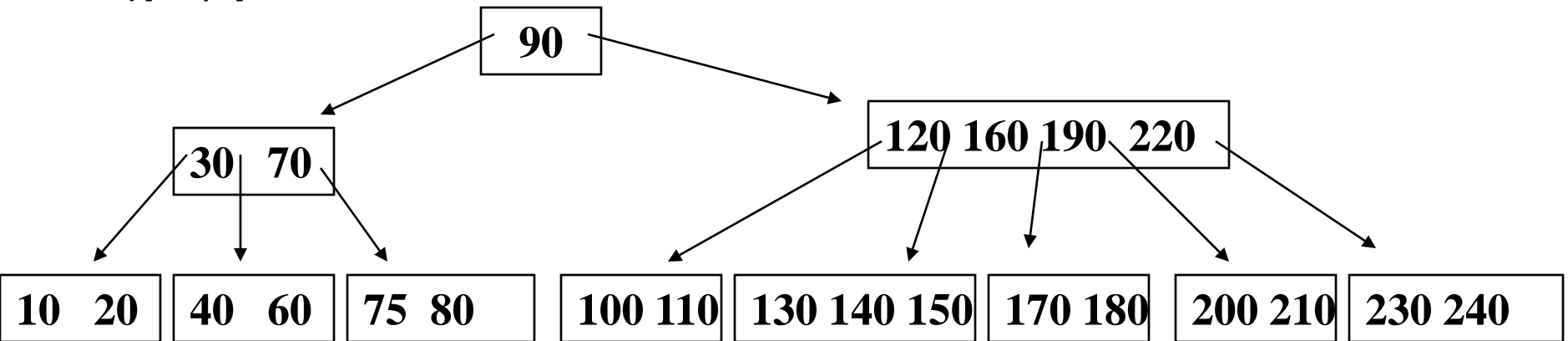
Έστω το B-δένδρο τάξεως 5 (δηλαδή από 2 μέχρι 4 κλειδιά σε κάθε κόμβο εκτός της ρίζας)



Διαγραφή του 250

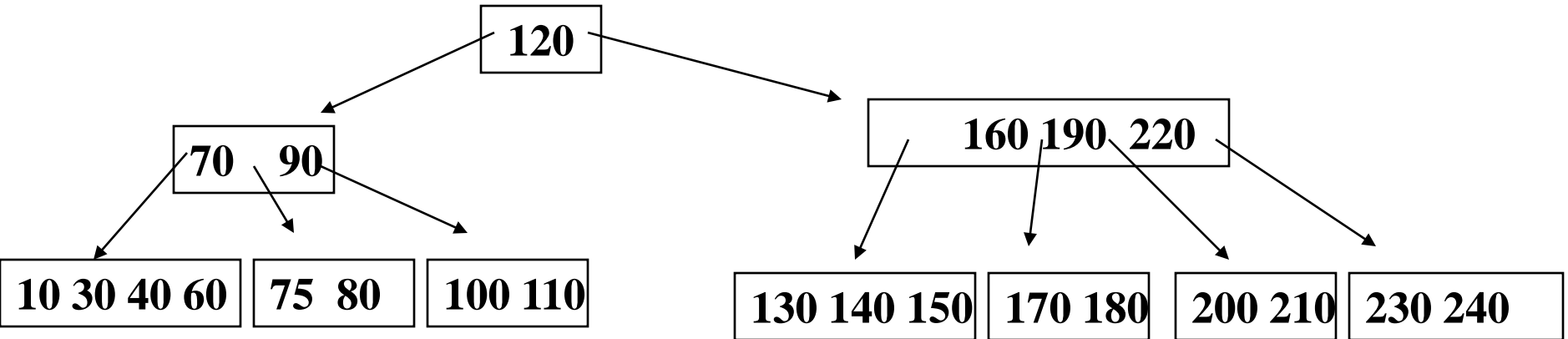


Διαγραφή του 50

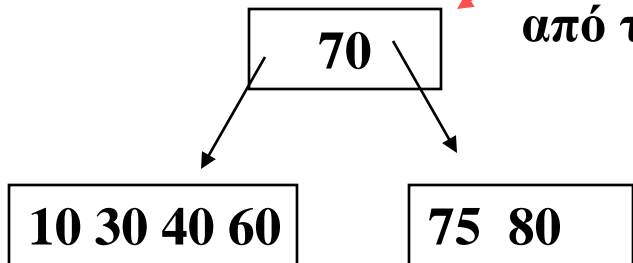


Έχει να δανείσει ο διπλανός επομένως πάει το 60 στη θέση του 50 και το 70 αναβαίνει στον γονέα.

Διαγραφή του 20



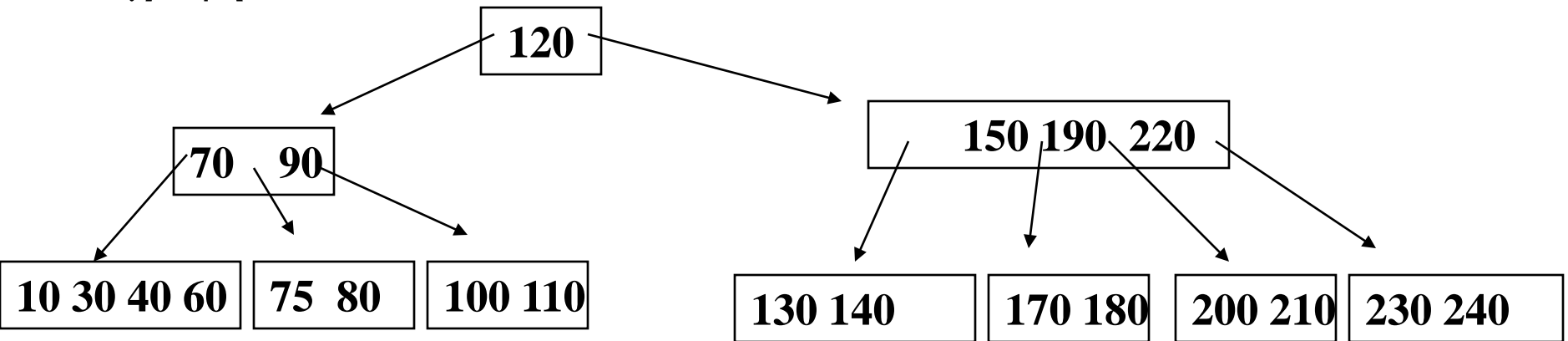
Τώρα όμως αυτός έχει λιγότερα και δανείζεται από τον αδελφό του



Διαγραφή από εσωτερικό κόμβο

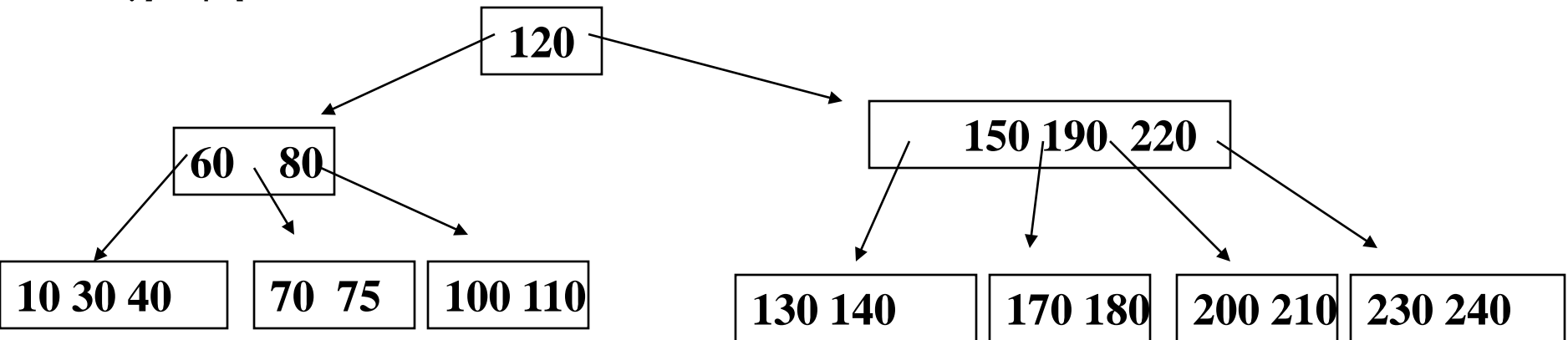
Στην περίπτωση αυτή πρέπει να βρεθεί το αμέσως επόμενο(ή προηγούμενο) από το στοιχείο που θέλουμε να διαγράψουμε. Το αμέσως επόμενο (ή προηγούμενο βρίσκεται πάντα σε φύλο). Στην περίπτωση αυτή το αμέσως προηγούμενο (ή επόμενο) καταλαμβάνει την θέση του προς διαγραφή στοιχείου και η διαγραφή ανάγεται στην προηγούμενη περίπτωση διαγραφής από φύλο.

Διαγραφή του 160



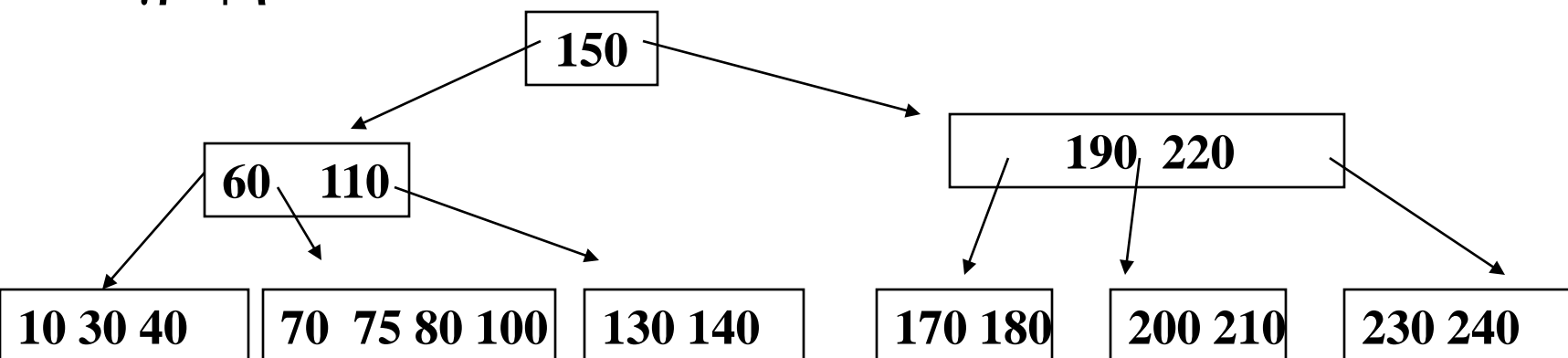
Στην περίπτωση αυτή πάει στη θέση του το αμέσως προηγούμενο δηλαδή το 150.

Διαγραφή του 90



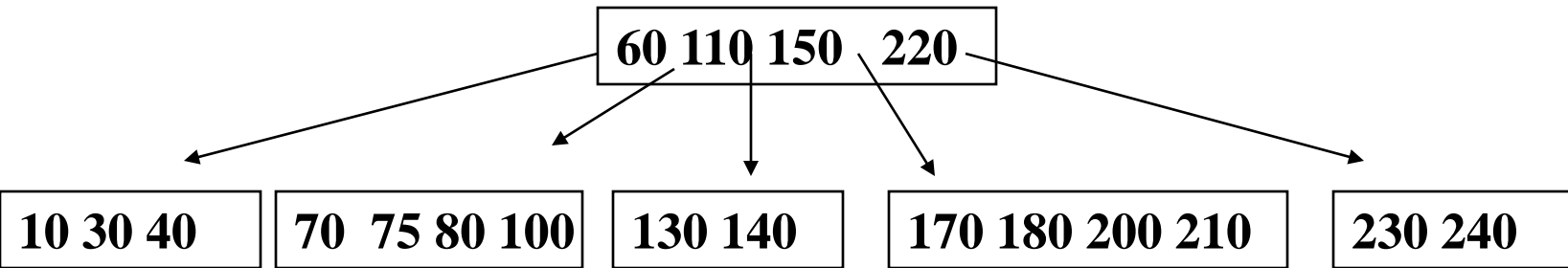
Στην περίπτωση αυτή πάει στη θέση του το αμέσως προηγούμενο δηλαδή το 80 αλλά ο κόμβος που είχε το 80 θα πέσει κάτω από το επιτρεπτό και θα ακολουθηθεί ότι και στην περίπτωση διαγραφής από φύλο

Διαγραφή του 120

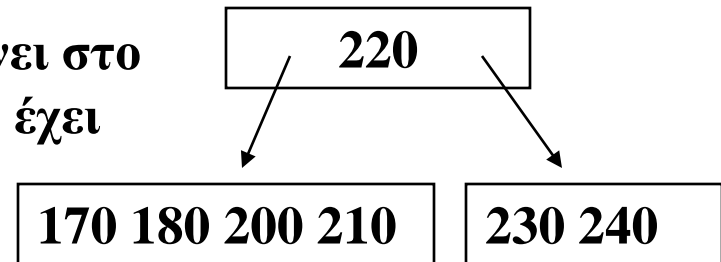


Στην περίπτωση αυτή είναι σαν να διαγράψουμε το 110 και ακολουθούμε την διαδικασία διαγραφής από φύλο

Διαγραφή του 190



Στην περίπτωση αυτή το 180 αναβαίνει στο γονέα. Ο κόμβος με μόνο το φύλο 170 έχει πρόβλημα. Θα ενωθεί με τον αδελφό



Ο κόμβος με το 220 έχει πρόβλημα και θα συνενωθεί με τον αδελφό και τη ρίζα μειώνοντας το ύψος κατά 1

Θα προσπαθήσουμε να βρούμε το μέγιστο ύψος ενός B-δένδρου ύψος ενός δένδρου με τάξεως m για N στοιχεία.

$$1 + 2(\lceil m/2 \rceil - 1) + 2(\lceil m/2 \rceil - 1)(\lceil m/2 \rceil) + \dots + 2(\lceil m/2 \rceil - 1)(\lceil m/2 \rceil)^{h-2} \\ = 2(\lceil m/2 \rceil)^{h-1} - 1$$

$$h-1 = \log_{\lceil m/2 \rceil} ((N+1)/2)$$

$$\text{Επομένως } h \leq \log_{\lceil m/2 \rceil} ((N+1)/2)$$

Όμως έχουμε $h \geq \log_m(N+1)$

Άρα $\log_m(N+1) \leq h \leq \log_{\lceil m/2 \rceil}((N+1)/2)$

Μήκος κλειδιού $V=9$ byte

Μέγεθος block $B=512$ byte

Μήκος δείκτη υποδένδρου $P=6$ byte

Μήκος δείκτη εγγραφής $P_r=7$ byte

Για την τάξη p του δένδρου πρέπει να ισχύει

$$(p * P) + ((p - 1) * (P_r + V)) \leq B$$

Δηλαδή στην περίπτωση μας

$$(p * 6) + ((p - 1) * (7 + 9)) \leq 512$$

$$22 * p \leq 521$$

$$p = 23$$

**Με αυτά τα δεδομένα για ένα αρχείο με 1000000 εγγραφές
το ύψος του δένδρου είναι 2 ή 3.**

**Ή ακόμη 94143178827
 $3 \leq h \leq 8$**

B+δένδρα

Τα B+δένδρα και οι παραλλαγές τους είναι η κυρίως δομή που χρησιμοποιούν τα εμπορικά συστήματα διαχείρισης βάσεων δεδομένων. Σε αντίθεση με τα B-δένδρα όπου κάθε τιμή του πεδίου αναζήτησης μια φορά σε κάποιο επίπεδο του δένδρου στα B+δένδρα οι δείκτες δεδομένων αποθηκεύονται μόνο στα φύλλα. Επομένως οι κόμβοι φύλλα διαφέρουν από τους εσωτερικούς κόμβους.

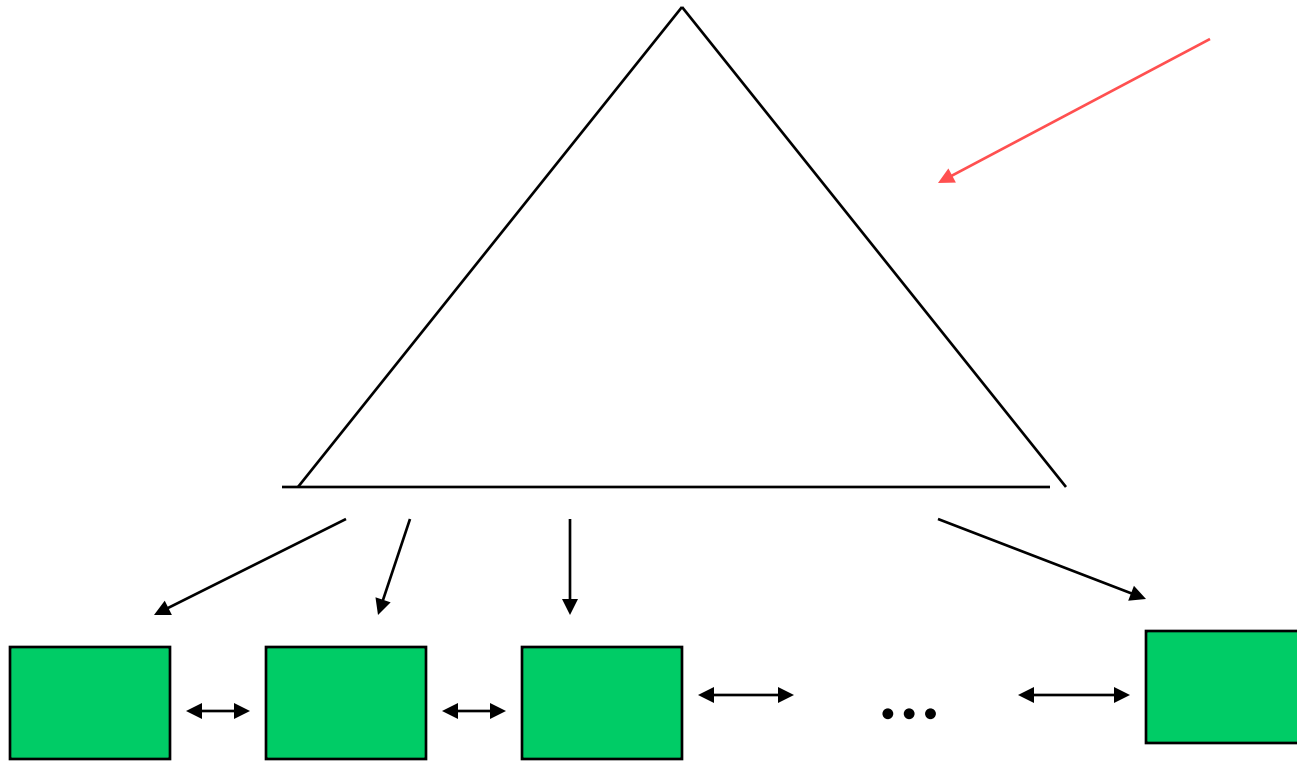
Ορισμός B+δένδρου (Εσωτερικοί κόμβοι)

1. Κάθε εσωτερικός κόμβος είναι της μορφής $\langle P_1, K_1, P_2, K_2, \dots, K_{q-1}, P_q \rangle$ $q \leq p$ και κάθε P_i είναι δείκτης δένδρου
2. Σε κάθε εσωτερικό κόμβο $K_1 < K_2 < \dots < K_{q-1}$.
3. Κάθε εσωτερικός κόμβος έχει το πολύ p δείκτες δένδρου.
4. Κάθε εσωτερικός κόμβος εκτός από τη ρίζα έχει το λιγότερο $\lceil p/2 \rceil$ δείκτες δένδρου. Ο κόμβος της ρίζας έχει τουλάχιστον δυο δείκτες δένδρου.
5. Ένας εσωτερικός κόμβος με q δείκτες, $q \leq p$ έχει $q-1$ τιμές πεδίου αναζήτησης

Ορισμός B+δένδρου (Κόμβοι φύλα)

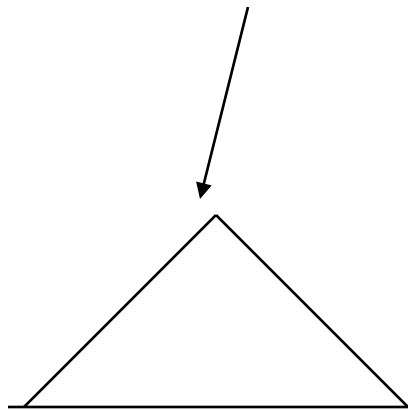
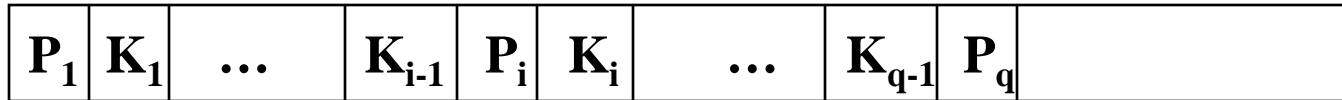
1. Κάθε κόμβος φύλο είναι της μορφής:
 $\langle \langle K_1, Pr_1 \rangle, \langle K_2, Pr_2 \rangle, \dots, \langle K_{q-1}, Pr_{q-1} \rangle, P_{next} \rangle$ $q \leq p$ κάθε Pr_i είναι δείκτης δεδομένων και το P_{next} δείχνει τον επόμενο κόμβο φύλο του B+δένδρου.
2. Σε κάθε κόμβο $K_1 < K_2 < \dots < K_{q-1}$, $q \leq p$.
3. Κάθε Pr_i είναι δείκτης δεδομένων που δείχνει την εγγραφή της οποίας η τιμή του πεδίου του ευρετηρίου είναι K_i .
4. Κάθε κόμβος φύλο έχει το λιγότερο $\lceil p/2 \rceil$ τιμές.
5. Όλοι οι κόμβοι φύλα είναι στο ίδιο επίπεδο.

Καταχωρήσεις ευρετηρίου

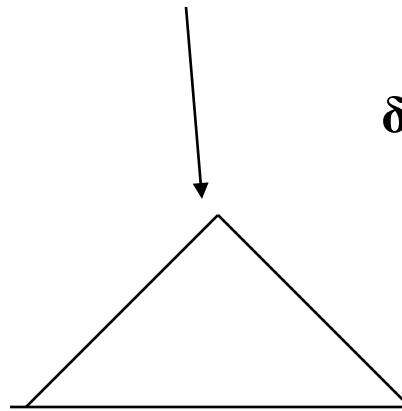


Δεδομένα ή δείκτες στα δεδομένα

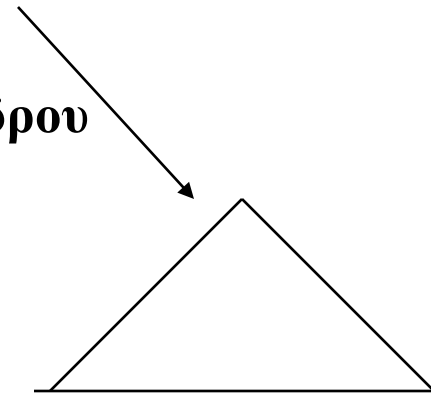
Δομή εσωτερικού κόμβου



$$X \leq K_1$$



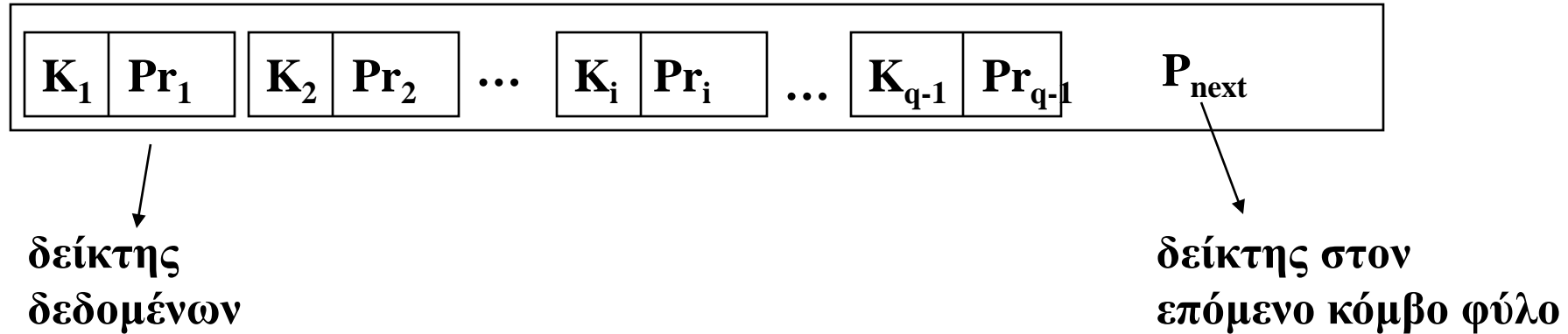
$$K_{i-1} \leq X \leq K_i$$

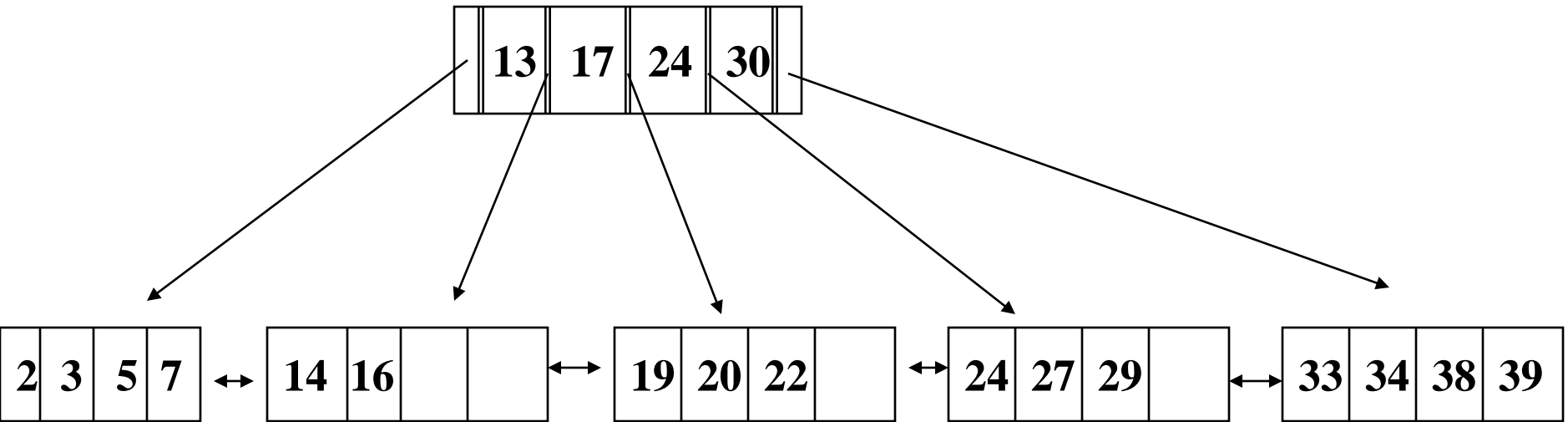


$$K_{q-1} < X$$

δείκτης δένδρου

Δομή κόμβου φύλου





Υπολογισμός της τάξης p του B+δένδρου

Μήκος κλειδιού $V=9$ byte

Μέγεθος block $B=512$ byte

Μήκος δείκτη υποδένδρου $P=6$ byte

Μήκος δείκτη εγγραφής $P_r=7$ byte

Για την τάξη p του δένδρου πρέπει να ισχύει

$$(p * P) + ((p - 1) * V) \leq B$$

Δηλαδή στην περίπτωση μας

$$(p * 6) + ((p - 1) * 9) \leq 512$$

$$15 * p \leq 521$$

$$P = 34$$

Ο κόμβος φύλο μπορεί να είναι άλλης τάξης

Θα πρέπει να ισχύει:

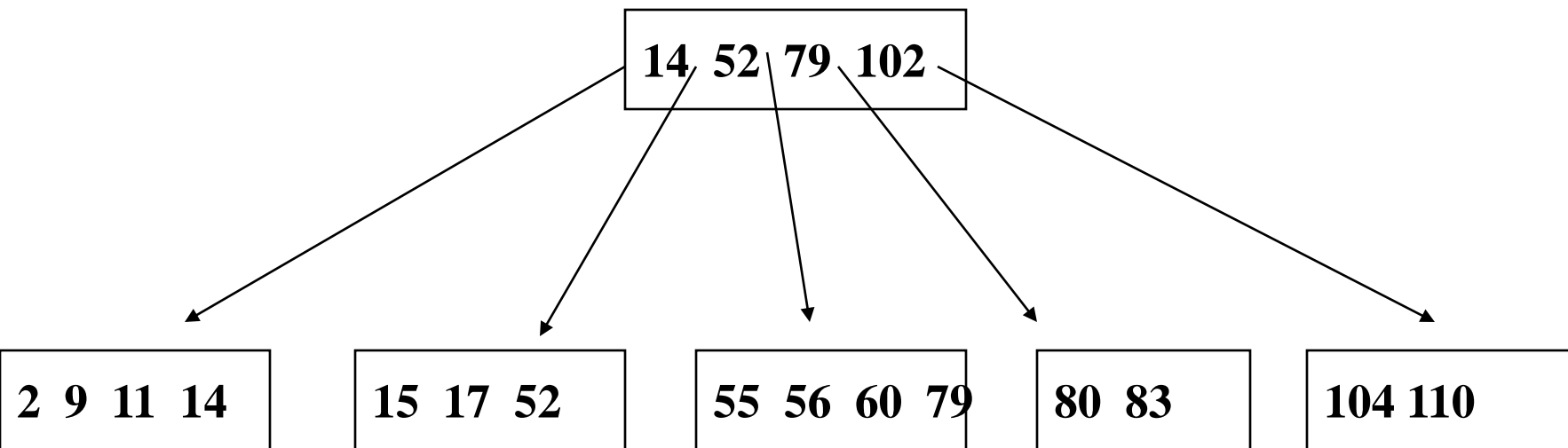
$$(p_{\text{leaf}} * (P_r + V)) + P \leq B$$

$$(p_{\text{leaf}} * (7 + 9)) + 6 \leq 512$$

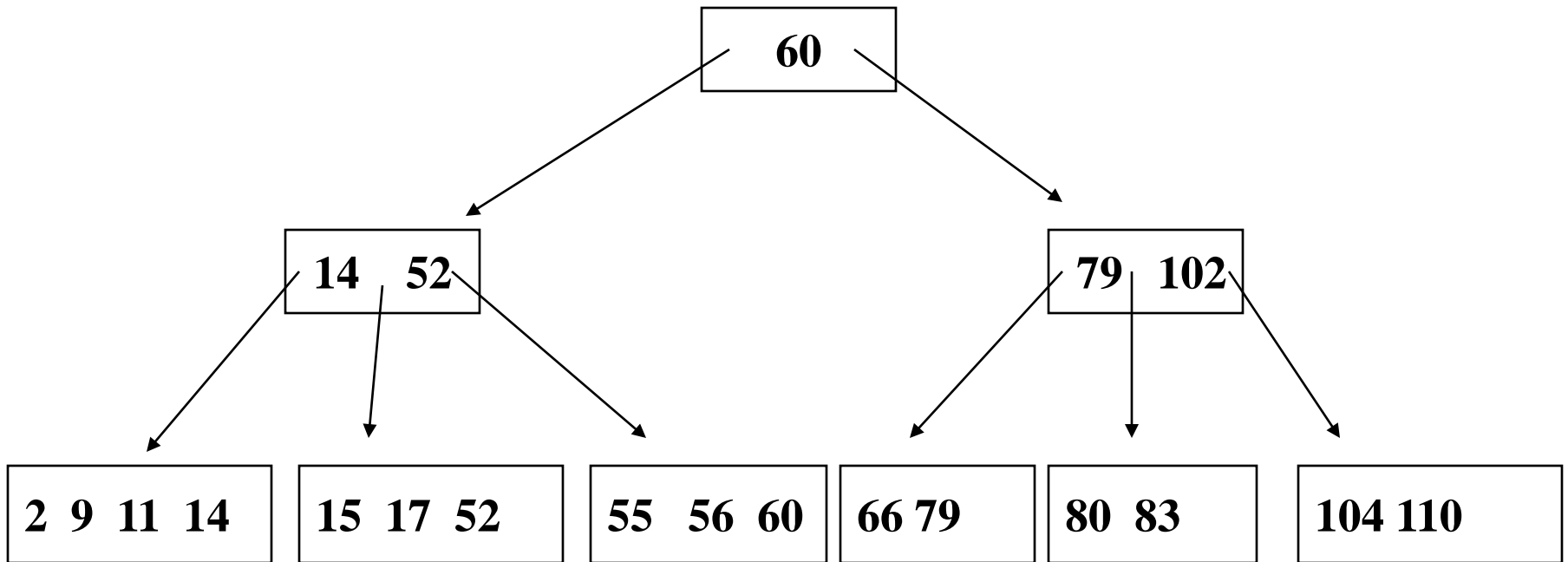
$$(16p_{\text{leaf}}) \leq 506$$

Δηλαδή $p_{\text{leaf}} = 31$

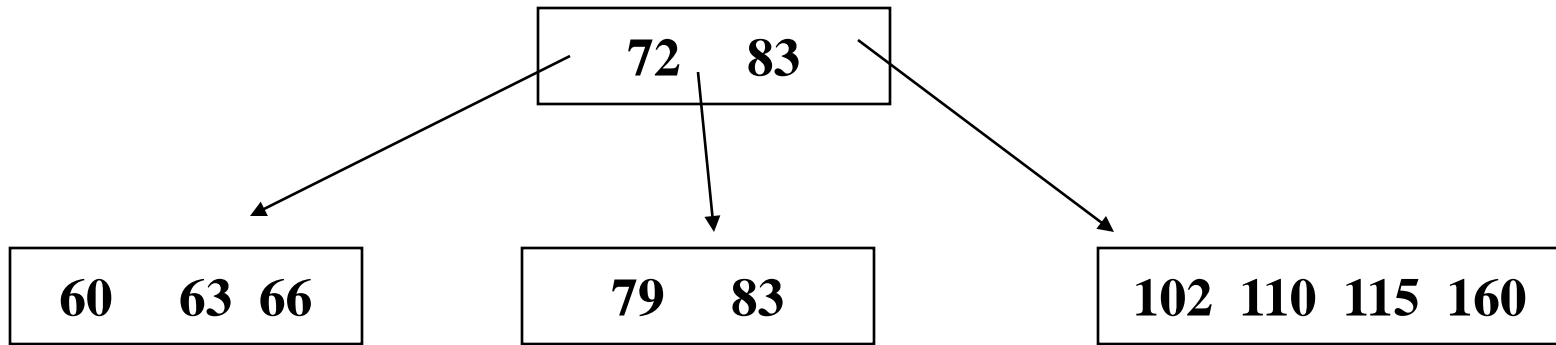
B+δένδρο βαθμού 5 έστω ότι και ο κόμβος φύλο είναι τάξεως 5



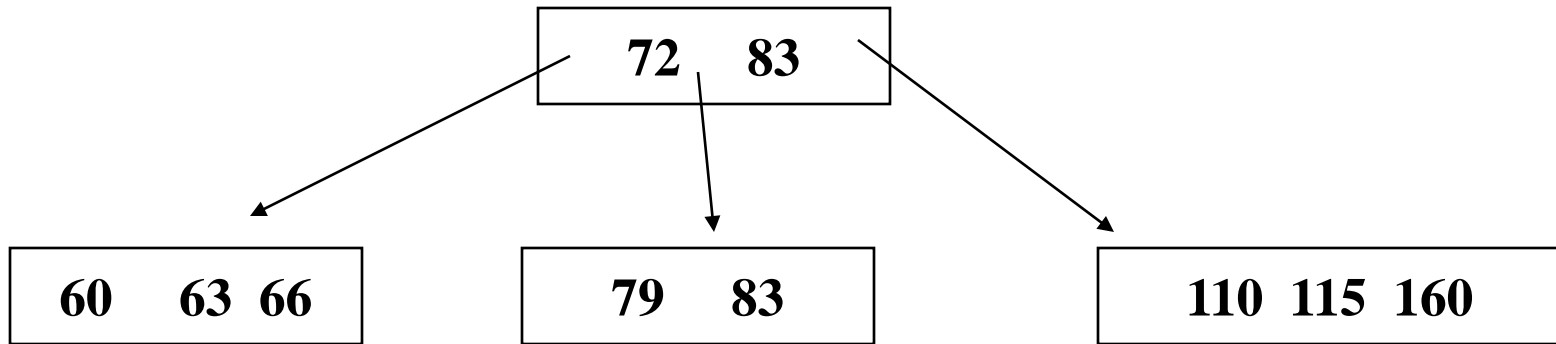
Εισαγωγή του 66



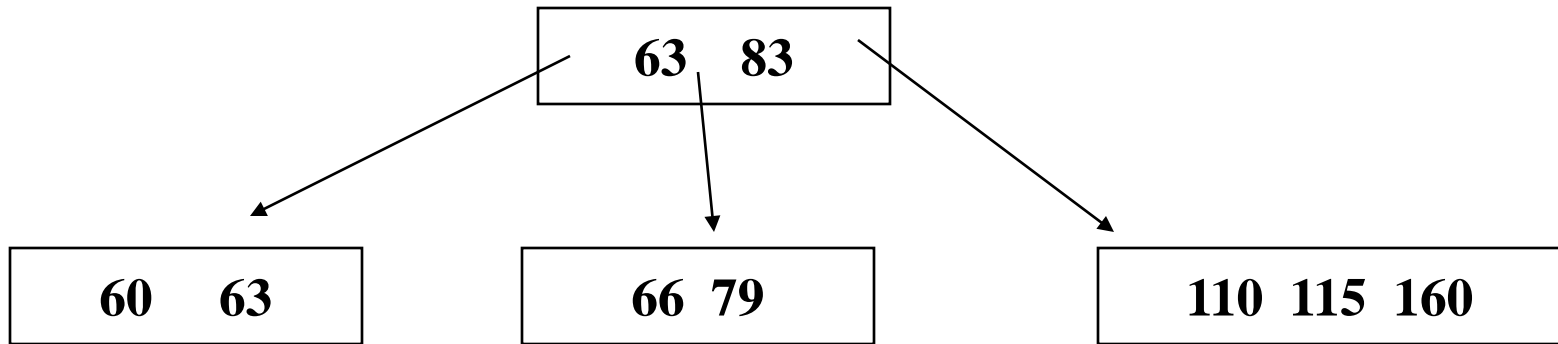
Η διαγραφή από ένα B+δένδρο είναι αρκετά πολύπλοκη γιατί έχουμε τους περιορισμούς τόσο στα φύλλα όσο και στους εσωτερικούς κόμβους να μην πέσουμε κάτω από την επιτρεπόμενη χρήση του κόμβου.



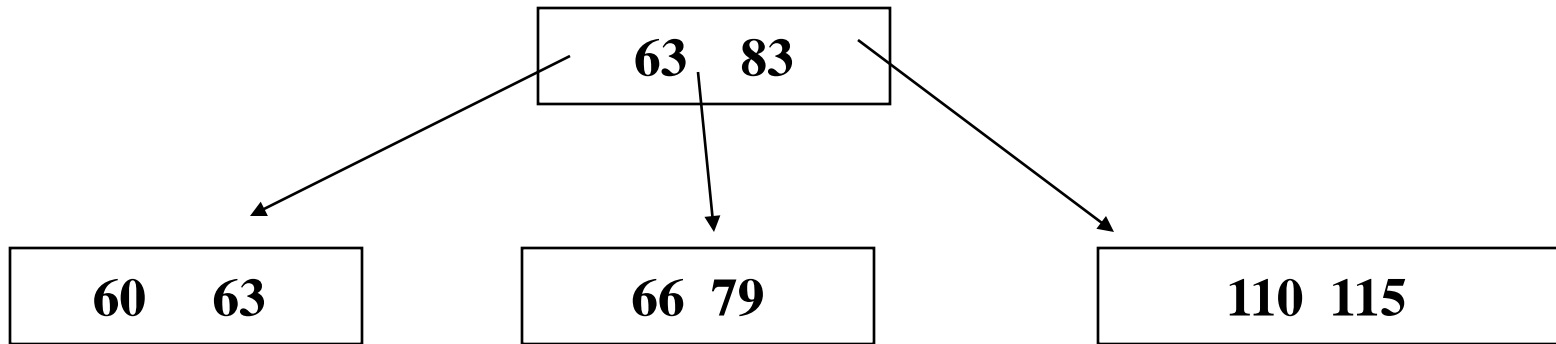
Διαγραφή 102



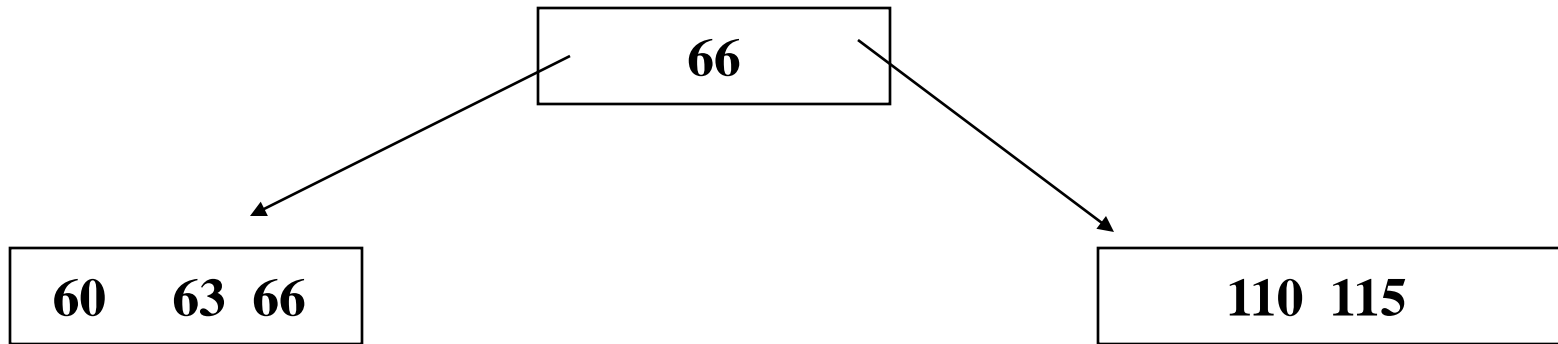
Διαγραφή 83



Διαγραφή 160



Διαγραφή 79



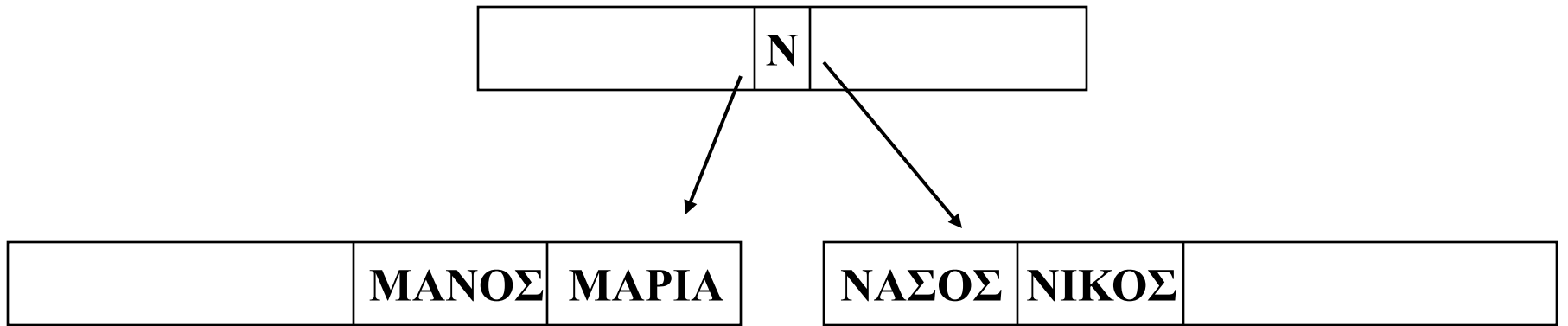
Τα Β-δένδρα μπορούν να χρησιμοποιηθούν και σαν δευτερεύοντα ευρετήρια (σε πεδία που δεν είναι κλειδιά). Στην περίπτωση αυτή μπορούμε να χρησιμοποιήσουμε μια τεχνική όμοια με αυτή της υπερχείλισης. Κάθε τιμή στο φύλο δείχνει σε μια λίστα με τις διευθύνσεις των εγγραφών με αυτή την τιμή.

Συνήθως όμως γίνεται μια διαφορετική υλοποίηση. Τις διπλές τιμές τις αντιμετωπίζουμε σαν απλές τιμές και για την ανάκτηση μιας τιμής ξεκινάμε από την πιο αριστερή τιμή και ακολουθούμε τα φύλα και πιθανόν και τους δείκτες επόμενου.

Επίσης μπορεί να χρησιμοποιηθεί συνδυασμός κλειδιού διεύθυνσης. Η λύση αυτή οδηγεί σε ευρετήρια μοναδικών τιμών.

Προθεματικά B-δένδρα

Έχουν παρόμοια δομή με τα B+δένδρα αλλά διαφέρει η υλοποίηση τους. Η βασική ιδέα είναι ότι αν το κλειδί είναι συμβολοσειρά, τότε δεν είναι σκόπιμο να αποθηκεύεται ολόκληρο στους εσωτερικούς κόμβους του καταλόγου αλλά μόνο εκείνο το πρόθεμα του που είναι απαραίτητο για την διαδικασία αναζήτησης. Έτσι ο κατάλογος περιέχει τους διαχωριστές (separators) που είναι μεταβλητού μήκους. Το μήκος των διαχωριστών ποικίλει από ένα χαρακτήρα μέχρι το μήκος ενός κλειδιού.



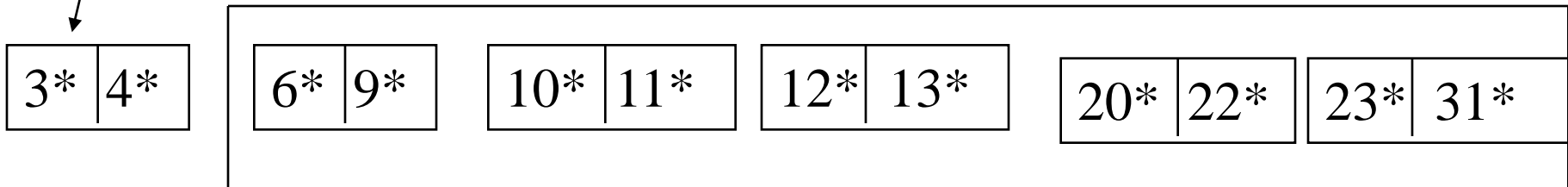
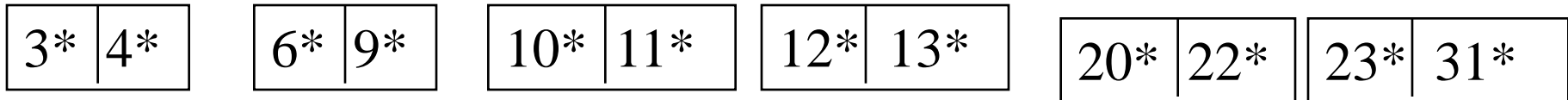
Ομαδική Φόρτωση σε B+-δένδρα

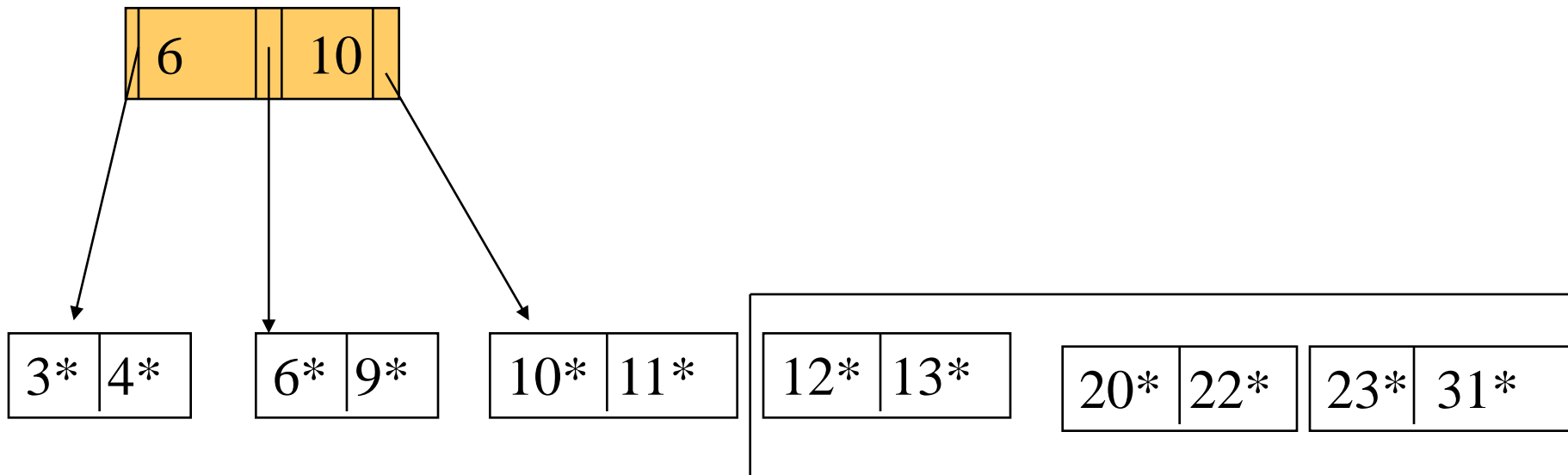
Ο αλγόριθμος εισαγωγής που περιγράψαμε εισάγει μια-μια τις καταχωρήσεις σε ένα B+-δένδρο. Αν έχουμε μια αρχική συλλογή χωρίς ευρετήριο μπορούμε να χρησιμοποιήσουμε αυτόν τον αλγόριθμο ο οποίος όμως δεν θα είναι αποτελεσματικός, γιατί κάθε φορά θα διασχίζουμε το δένδρο από τη ρίζα.

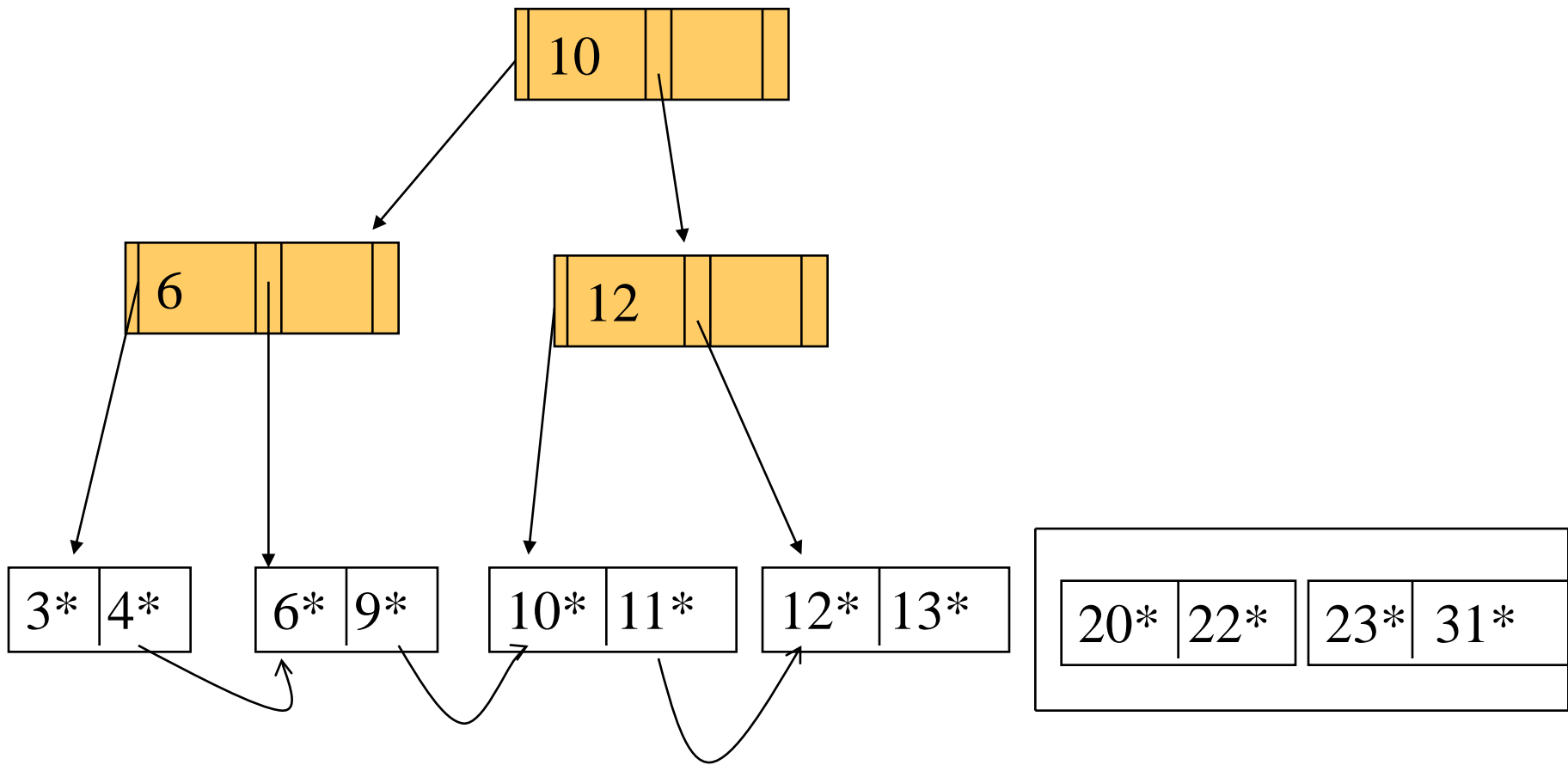
Πολλά συστήματα έχουν ένα βοηθητικό πρόγραμμα ομαδικής φόρτωσης για την δημιουργία B+δένδρου ευρετηρίου σε μια υπάρχουσα συλλογή εγγραφών

Βήματα ομαδικής φόρτωσης

- Ταξινόμηση των εγγραφών που θα εισαχθούν με βάση το κλειδί αναζήτησης k (εφόσον είναι ζεύγος κλειδί, δείκτης ταξινόμηση δεν σημαίνει ταξινόμηση του αρχείου)
- Ξεκινάμε με μια κενή σελίδα σαν ρίζα και εισάγουμε ένα δείκτη στην πρώτη σελίδα των ταξινομημένων καταχωρήσεων. Η καταχώρηση είναι ελάχιστη τιμή κλειδιού στη σελίδα και δείκτης σελίδας.
- Συνεχίζουμε μέχρι να γεμίσει η ρίζα.
- Στη συνέχεια έχουμε διάσπαση της ρίζας κοκ.







VSAM (Virtual Storage Access Method) αρχεία

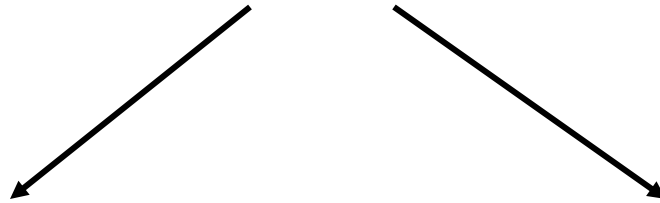
Αποτελούν μια υλοποίηση της IBM που μοιάζει πολύ με τα B+ δένδρα με την διαφορά ότι ο παράγοντας ελάχιστης φόρτωσης του κόμβου αφήνεται πιο ελεύθερος και μπορεί να καθορισθεί από τον χρήστη (σε αντίθεση με τα B+δένδρα που είναι το μισό της μέγιστης χωρητικότητας).

Δομή VSAM αρχείου

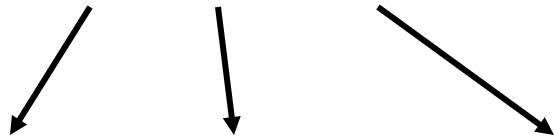
Απαρτίζεται από 4 περιοχές

- **Διαστήματα Ελέγχου (control intervals) που περιέχουν τις εγγραφές του αρχείου.**
- **Περιοχές ελέγχου (control areas) που περιέχουν πολλά διαστήματα ελέγχου**
- **Ακολουθιακές Ομάδες (sequence sets) που είναι κατάλογοι για κάθε περιοχή ελέγχου**
- **Ομάδα καταλόγων (index set) που είναι μια δενδρική δομή με το πολύ τρία επίπεδα. Οι σελίδες του κατώτερου επιπέδου δεικτοδοτούν προς το επίπεδο των ακολουθιακών ομάδων**

Ομάδα καταλόγων



Ακολουθιακή ομάδα

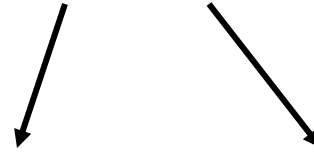


**Διάστημα
Ελέγχου**

**Διάστημα
Ελέγχου**

**Διάστημα
Ελέγχου**

Ακολουθιακή ομάδα



**Διάστημα
Ελέγχου**

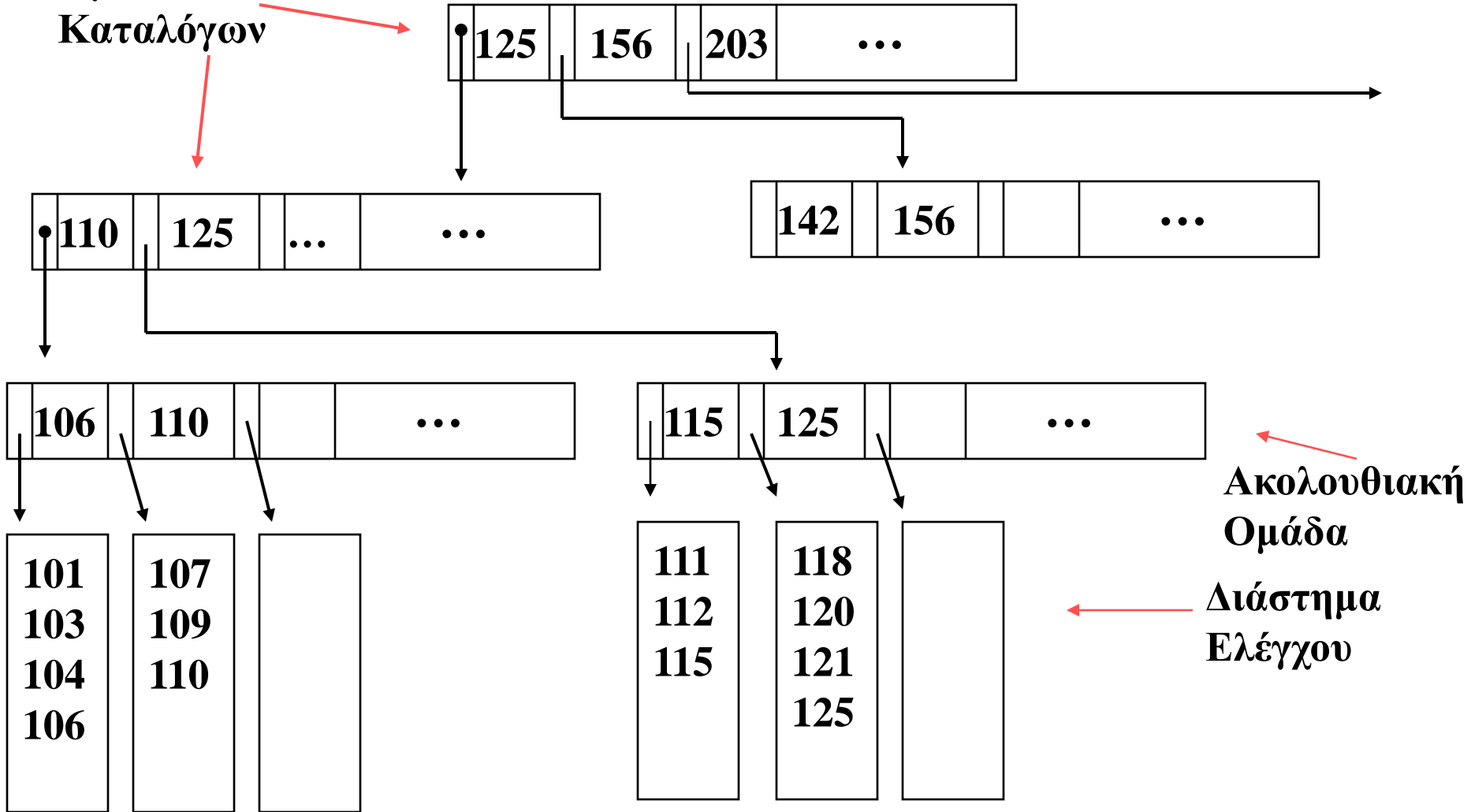
**Διάστημα
Ελέγχου**

Εγγραφή 1	Εγγραφή 2	Εγγραφή 3	Εγγραφή 4	Εγγραφή 5	
Εγγραφή 6	Ελεύθερος χώρος				
Ελεύθερος χώρος					
			RDF3	RDF2	RDF1

RDF:Record Definition Field
Πεδίο Ορισμού εγγραφής

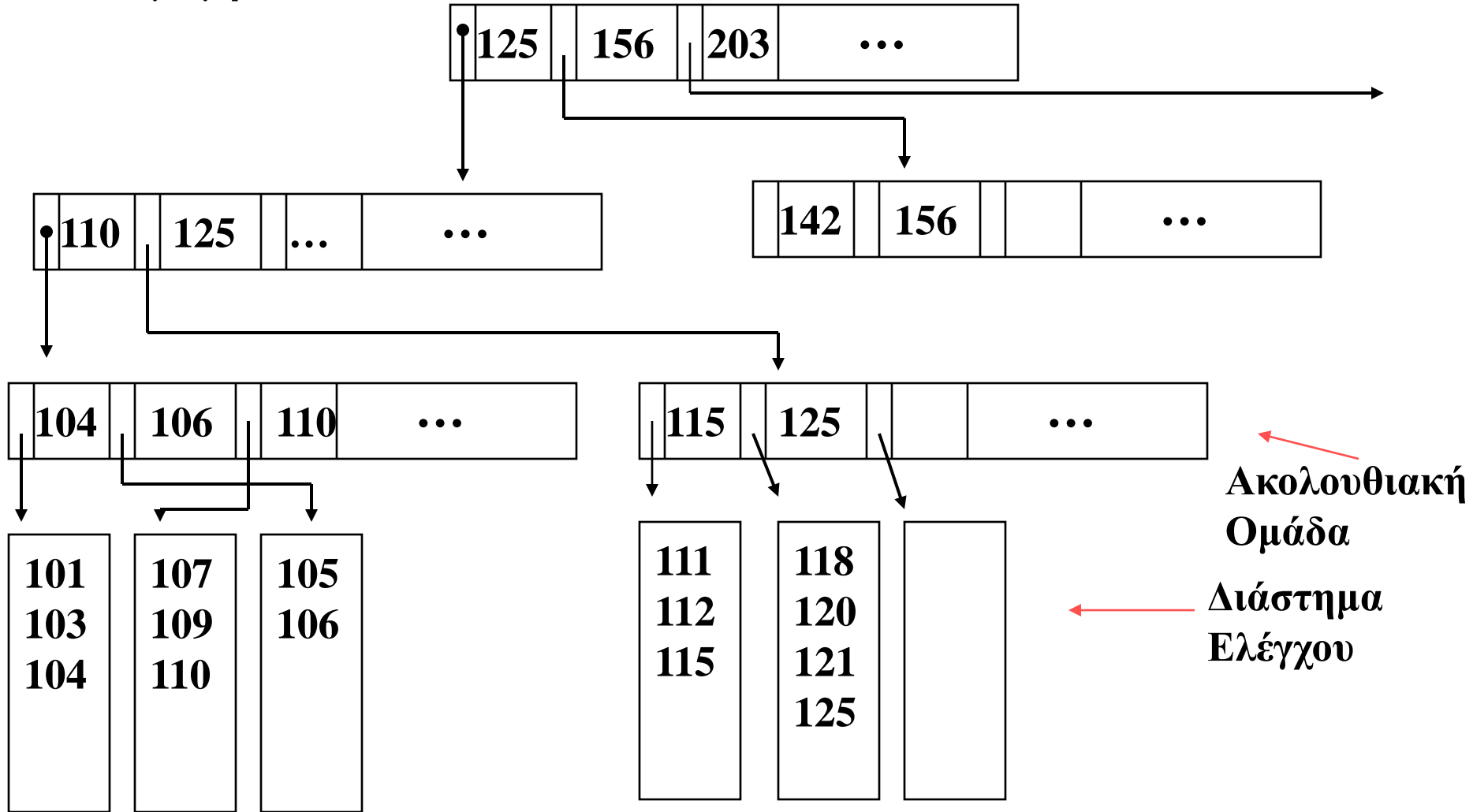
Υποθέστε ότι η περιοχή ελέγχου χωράει 4 εγγραφές

Ομάδα
Καταλόγων



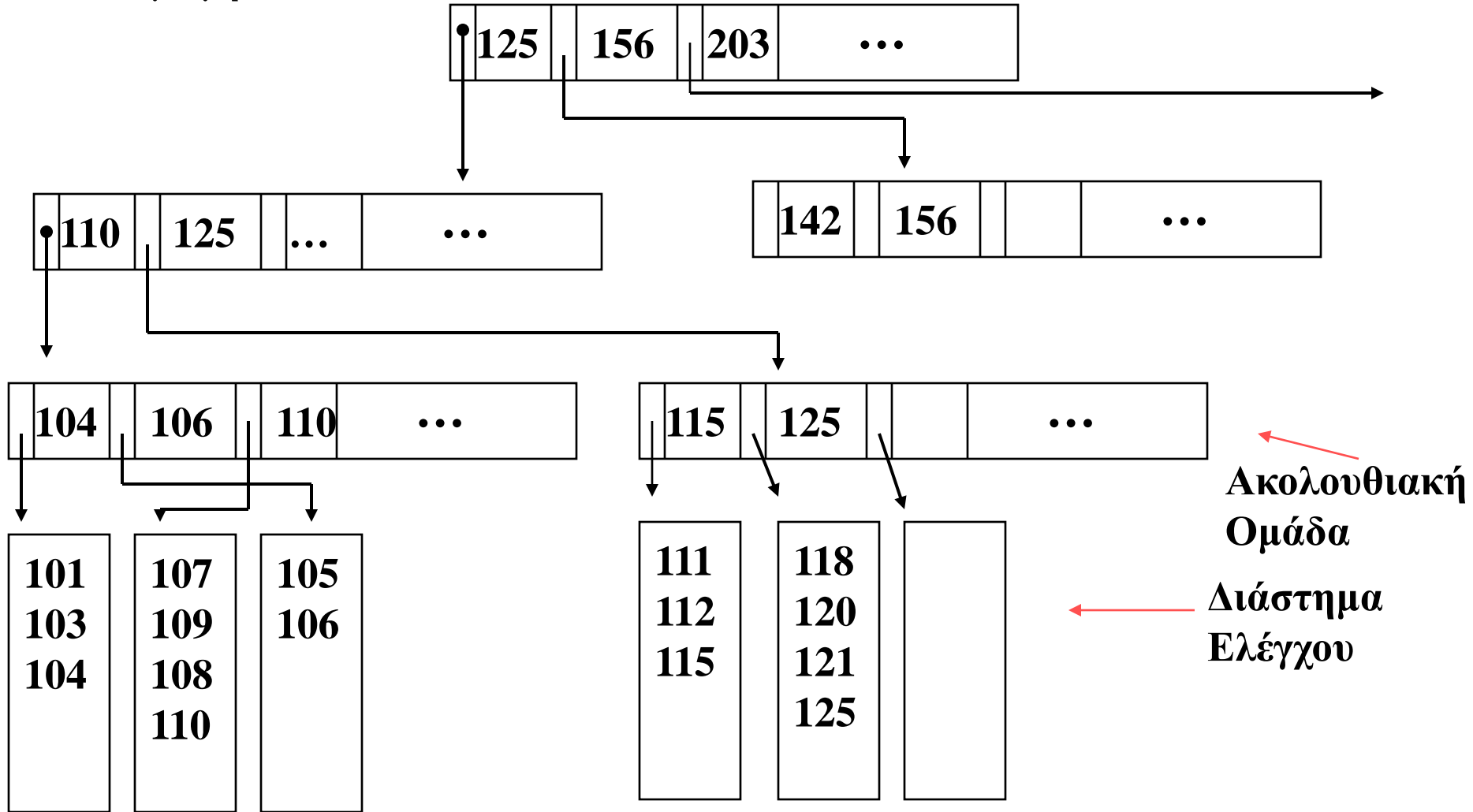
Περιοχές Ελέγχου

Εισαγωγή 105



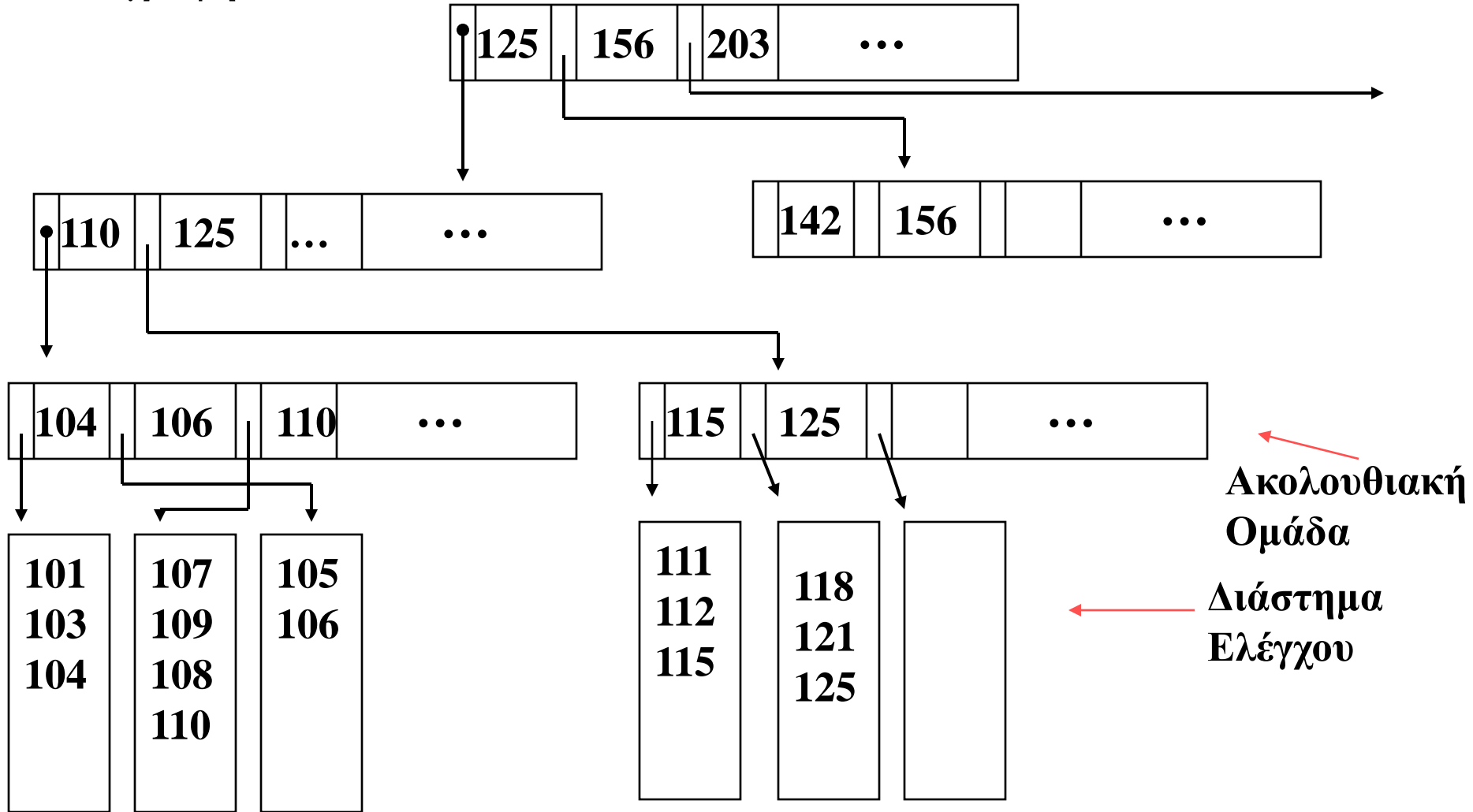
Περιοχές Ελέγχου

Εισαγωγή 108



Περιοχές Ελέγχου

Διαγραφή 120



Περιοχές Ελέγχου

Τα εμπορικά συστήματα

SYBASE: χρησιμοποιεί ευρετήρια συστάδες.
Οι σελίδες αποτελούν διπλά συνδεδεμένη λίστα.
Με τις εισαγωγές γράφονται δεδομένα στην
υπερχείλιση.

ORACLE, DB2, SQL Server: Η αντιμετώπιση
των διπλών γίνεται με την προσθήκη της
διεύθυνσης της εγγραφής. Με τον τρόπο αυτό
δεν υπάρχουν διπλά.

Εμπορικά Συστήματα και B-δένδρα

Τα DB2, Informix, SQL Server, Oracle, και SYBASE υποστηρίζουν απλά και συστάδες ευρετήρια B+δένδρα με κάποιες διαφορές στην αντιμετώπιση των διαγραφών και των δευτερευόντων πεδίων. Στην SYBASE ανάλογα με το σχήμα συγχρονισμένης προσπέλασης που χρησιμοποιείται, η εγγραφή διαγράφεται (με πιθανή συγχώνευση σελίδων) ή απλά σημαδεύεται (με κάποιο σχήμα καθαρισμού του χώρου-garbage collection).

Στην ORACLE οι διαγραφές σημαδεύονται σαν διαγραμμένες. Για ανάκληση του ελεύθερου χώρου μπορούμε on-line να χτίσουμε το ευρετήριο. Η INFORMIX σημαδεύει τις εγγραφές που έχουν διαγραφεί. Το DB2 και ο SQL Server διαγράφουν εγγραφές και συγχωνεύουν σελίδες όταν πέσουν κάτω από κάποιο όριο.

Η ORACLE σε συστάδες εγγραφές από πολλές σχέσεις. Αυτό μπορεί να στηριχθεί σε B+δένδρα.

Ευρετήρια σε πολλαπλά κλειδιά

Πολλές φορές οι ερωτήσεις αφορούν περισσότερα από ένα πεδία. Μας ενδιαφέρει η περίπτωση λογικών συζεύξεων. Δηλαδή ερωτήσεις του τύπου:

(συνθήκη1) AND (συνθήκη2)

Όπου η συνθήκη είναι της μορφής:

$$A \delta c$$

Όπου $\delta \in \{=, <, >, \leq, \geq\}$

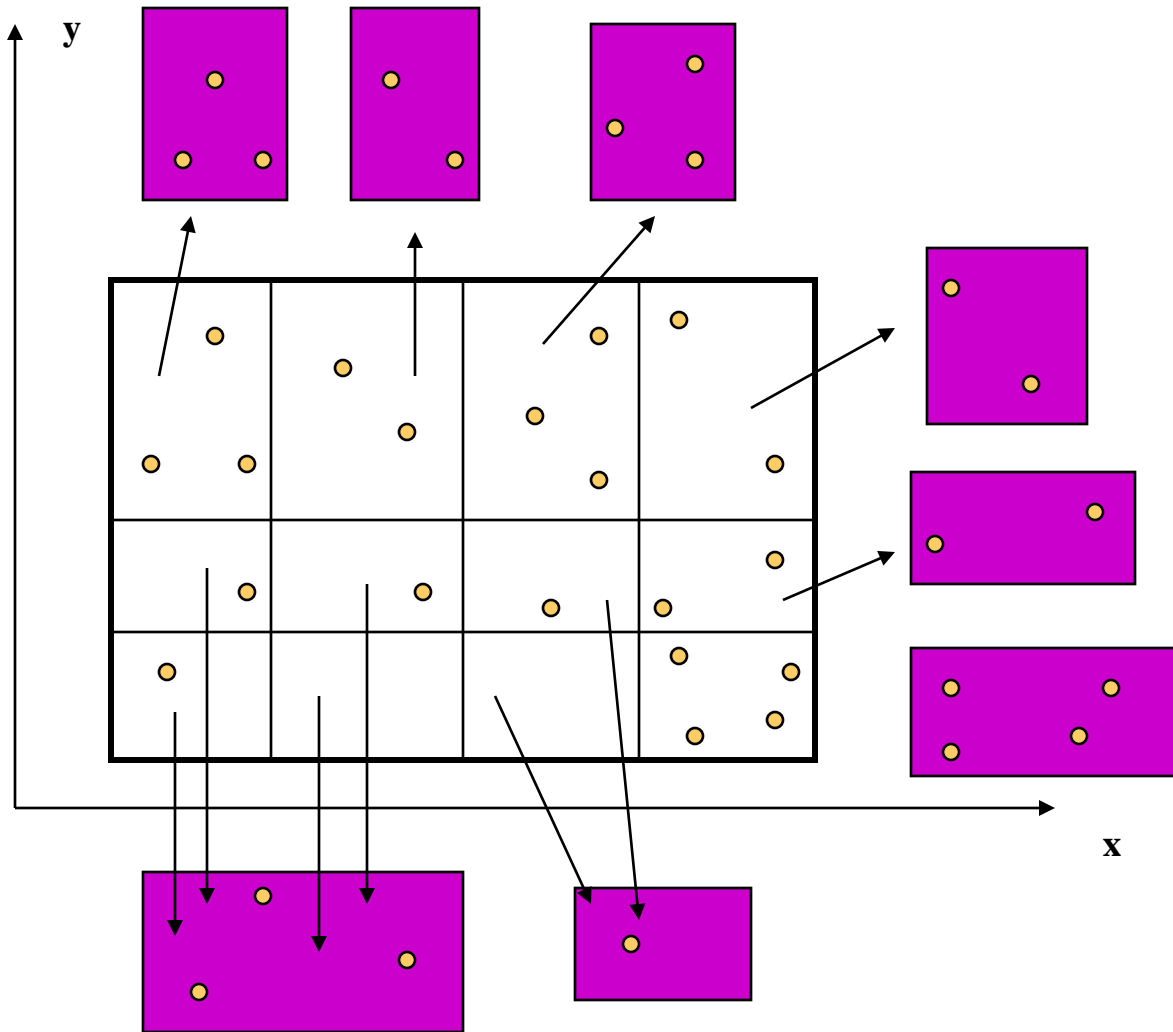
και το c σταθερά

Τι μπορούμε να κάνουμε

- **Αν δεν υπάρχει ευρετήριο θα πρέπει να γίνει εξαντλητική σάρωση και έλεγχος για την εύρεση των εγγραφών που πληρούν την σύνθετη συνθήκη.**
- **Αν δεν υπάρχει ευρετήριο αλλά το αρχείο είναι ταξινομημένο ως προς κάποιο πεδίο της συνθήκης μπορεί να επιταχυνθεί η αναζήτηση.**
- **Αν υπάρχει ευρετήριο ως προς το ένα πεδίο τότε εντοπίζονται οι εγγραφές που πληρούν την μια συνθήκη και αφού μεταφερθούν στην μνήμη εξετάζεται και η άλλη συνθήκη.**

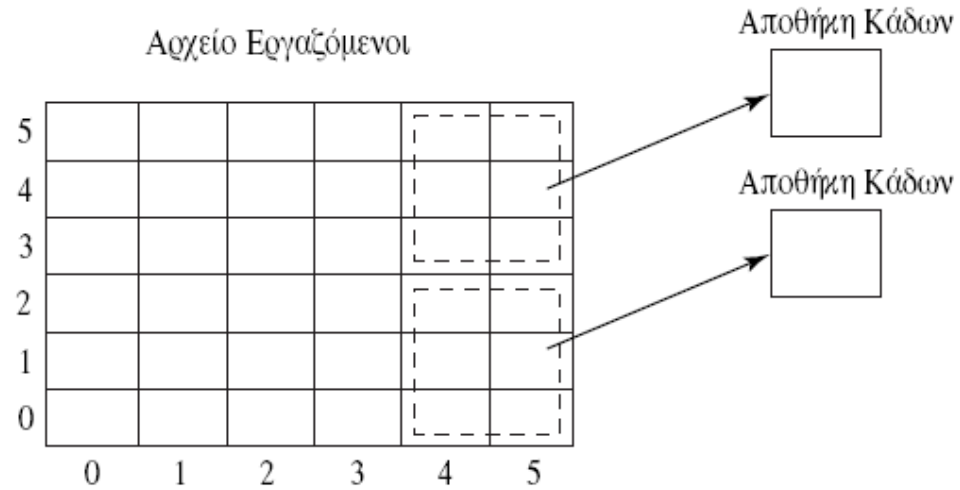
- **Αν υπάρχει ευρετήριο και στα δύο πεδία τότε μπορούμε να χρησιμοποιήσουμε και τα δύο ευρετήρια και η τομή τους είναι οι εγγραφές που αναζητούμε. Εναλλακτικά μπορούμε να χρησιμοποιήσουμε το ευρετήριο που θα δώσει το μικρότερο σύνολο εγγραφών και στη συνέχεια να εξετάσουμε τις εγγραφές αυτές αν πληρούν και την άλλη συνθήκη.**
- **Τέλος μπορούμε να συντηρούμε ευρετήρια σε συνδυασμό κλειδιών (σύνθετα κλειδιά).**

Δικτυωτό Αρχείο



ΑΡΙΘ_Τ	
0	1,2
1	3,4
2	5
3	6,7
4	8
5	9,10

Γραμμική κλίμακα για το ΑΡΙΘ_Τ



Γραμμική Κλίμακα για το Ηλικία

0	1	2	3	4	5
<20	21-25	26-30	31-40	41-50	>50

Λογικά έναντι φυσικών ευρετηρίων

- Θεωρήσαμε ευρετήρια με καταχωρήσεις του τύπου $\langle k, Pr \rangle$ ή $\langle k, P \rangle$ όπου το P ή το Pr ήταν φυσική διεύθυνση στο δίσκο. Ένα τέτοιο ευρετήριο ονομάζεται φυσικό ευρετήριο.
- Σε αντίθεση ένα λογικό ευρετήριο έχει καταχωρήσεις της μορφής $\langle k, K_p \rangle$ όπου κάθε καταχώρηση έχει μια τιμή για το δευτερεύον πεδίο ευρετηριοποίησης συνδεδεμένη με την τιμή K_p του πεδίου που χρησιμοποιείται για την πρωτεύουσα οργάνωση του αρχείου.