
Προηγμένα Θέματα Βάσεων Δεδομένων

Διδάσκων: Άγγελος Μιχάλας

Περιεχόμενα

- Συναρτήσεις Κατακερματισμού
- Δυναμικός Κατακερματισμός
- Επεκτατός Κατακερματισμός
- **Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο**
- **Γραμμικός Κατακερματισμός**

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Επεκτατός Κατακερματισμός
 - Ανεξέλεγκτος διπλασιασμός του καταλόγου
 - Επειδή οι εγγραφές κατανέμονται ισοπίθانا στους κάδους, για αυτό οι κάδοι διασπώνται περίπου ταυτόχρονα
 - Ενώ το **κόστος της εισαγωγής** εγγραφών διατηρείται περίπου σταθερό περιοδικά **εμφανίζει εξάρσεις** που οφείλονται στις ταυτόχρονες διασπάσεις των κάδων του αρχείου

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο επιλύει και τα δύο προβλήματα του Επεκτατού Κατακερματισμού
- Προτάθηκε από το Lomet το 1983

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Όσον αφορά το πρώτο πρόβλημα
- Ο **κατάλογος** δεν μπορεί να ξεπεράσει ένα **μέγιστο μέγεθος** που καθορίζεται από τη διαθέσιμη κύρια μνήμη
- οι σελίδες του καταλόγου είναι δυνάμεις του 2 με μέγιστο 2^{\max} .
- *Από το σημείο αυτό και μετά το αρχείο μεγαλώνει αυξάνοντας όχι το πλήθος των κάδων αλλά το μέγεθός τους.*

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Όσον αφορά το δεύτερο πρόβλημα
- Ο μετασχηματισμός του κλειδιού ακολουθεί έναν αριθμό βημάτων που επιλύει το πρόβλημα των ταυτόχρονων διασπάσεων.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Τα κλειδιά μετασχηματίζονται με βάση μία συνάρτηση ($h(k)$) ώστε να προκύψει μία ομοιόμορφη κατανομή.
- Το αποτέλεσμα της συνάρτησης ανακατανέμεται εκθετικά με βάση τον τύπο:

$$\text{exhash}(k) = 2^{h(k)} - 1$$

- k : είναι το κλειδί,
- $h(k)$: το αποτέλεσμα του πρώτου μετασχηματισμού, και
- $\text{exhash}(k)$: το αποτέλεσμα του δεύτερου.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

Τελικό αποτέλεσμα

- Κατανομή που διακρίνεται για τη **συσσώρευση των κλειδιών προς το κάτω άκρο του διαστήματος των διευθύνσεων**.
- Έτσι, οι κάδοι δεν δέχονται τον ίδιο αριθμό **εγγραφών** και η επίδοση της εισαγωγής είναι περίπου σταθερή σε όλη τη διάρκεια της ζωής του αρχείου.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Για $h(k)$ 0,067
- $exhash(k)$ 0,047 αρχικά ενώ στο τέλος 0,09

$h(k)$	$exhash(k)$	$h(k)$	$exhash(k)$	$h(k)$	$exhash(k)$	$h(k)$	$exhash(k)$
0,000	0,000	0,267	0,203	0,533	0,447	0,800	0,741
0,067	0,047	0,333	0,260	0,600	0,516	0,867	0,823
0,133	0,097	0,400	0,320	0,667	0,662	0,933	0,910
0,200	0,149	0,467	0,382	0,733	0,662	1,000	1,000

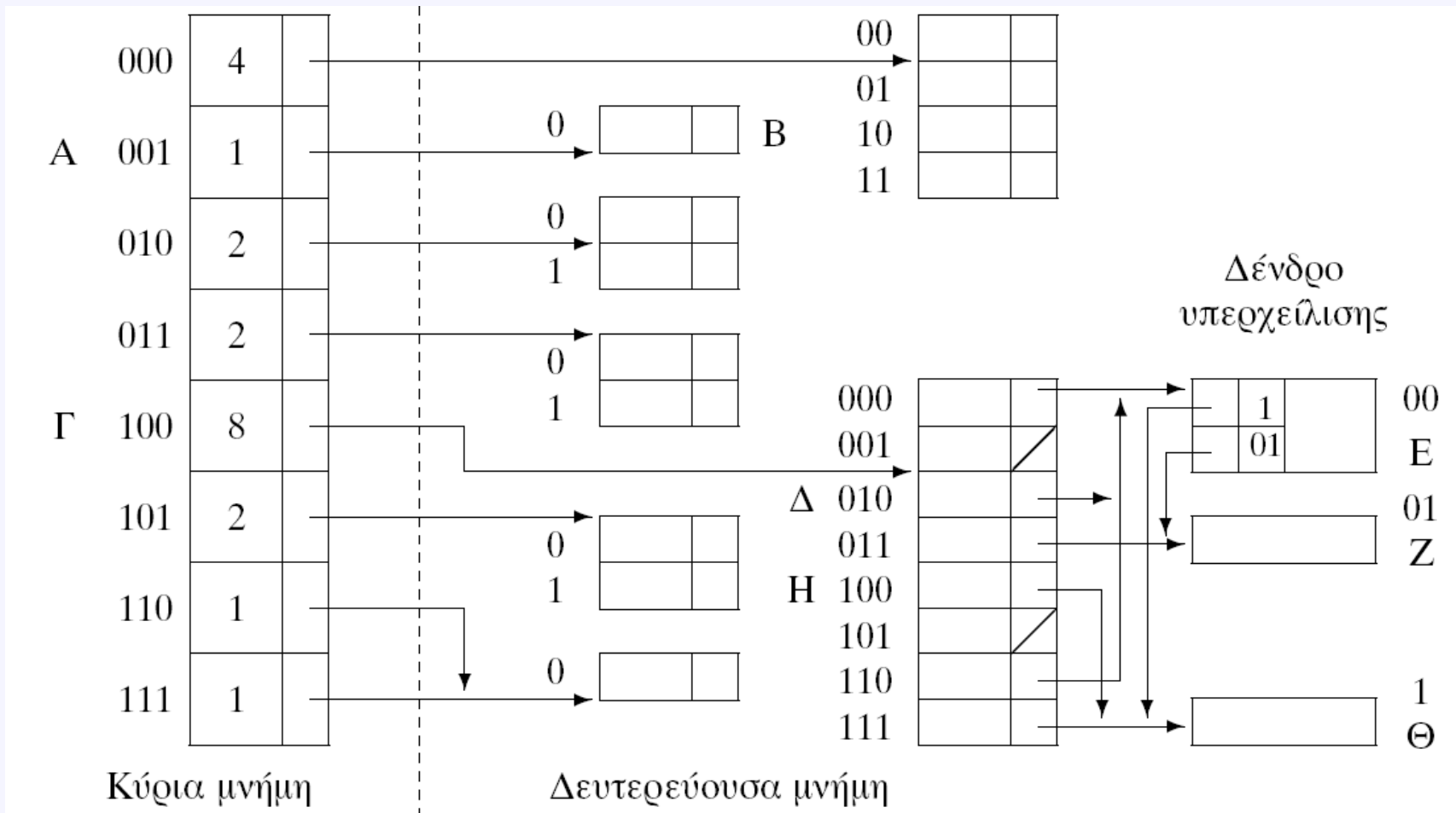
Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

1. Τα πρώτα **bits** του $\text{exhash}(k)$ καθορίζουν τη θέση (σελίδα) του καταλόγου, όπου είναι αποθηκευμένος ο δείκτης που δείχνει προς τον κατάλληλο κάδο.
2. Η επιλογή της σελίδας (από το σύνολο των σελίδων του κάδου), όπου θα πρέπει να συνεχισθεί η αναζήτηση, γίνεται με βάση μερικά από τα επόμενα **bits**.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Σε κάθε κάδο αντιστοιχεί μία περιοχή υπερχείλισης, που είναι *οργανωμένη ως δυαδικό δένδρο* που ονομάζεται *ο-δένδρο*.
 - Ο κατάλογος για αυτό το δένδρο αποθηκεύεται στην πρώτη σελίδα του αντίστοιχου κάδου.
- Κατά παρόμοιο τρόπο, η διαχείριση της περιοχής υπερχείλισης γίνεται με τα επόμενα bits του $\text{exhash}(k)$.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο - Παράδειγμα



Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο - Παράδειγμα

- Αναζητούμε το κλειδί 204.
- Ισχύει $\text{exhash}(204) = 0011001100_2$.
- Ο κατάλογος αποτελείται από 8 θέσεις (σελίδες):
 - θεωρούνται τα 3 πρώτα bits (001) για την προσπέλαση του καταλόγου,
 - η αναζήτηση κατευθύνεται στη 2η σελίδα του καταλόγου,
 - η αναζήτηση τερματίζεται στον κάδο B.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο - Παράδειγμα

- Αναζητούμε το κλειδί 641.
- Ισχύει $\text{exhash}(641) = 1000101100_2$.
- Ο κατάλογος αποτελείται από 8 σελίδες:
 - θεωρούνται τα 3 πρώτα bits (100) για την προσπέλαση του καταλόγου,
 - η αναζήτηση κατευθύνεται στη 5η σελίδα του καταλόγου
 - ο κάδος έχει 8 σελίδες,
 - λαμβάνονται τα επόμενα 3 bits (010) για διευκρίνιση της σελίδας,
 - η αναζήτηση συνεχίζεται στην σελίδα Δ,
 - Αν δεν υπάρχει, συνεχίζεται στο ο-δένδρο
 - Αρχικά κατευθύνεται στον κόμβο E όπου υπάρχει αποθηκευμένος ο κατάλογος του δένδρου.
 - επειδή το 4ο και 5ο bit =01, συνεχίζεται στη σελίδα Z
 - απαιτούνται 3 προσπελάσεις (στις σελίδες Δ, E, Z).

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- **Αν** εμφανισθεί το φαινόμενο να χρειαστούν **> 2 προσπελάσεις** τότε **οι δείκτες αλλάζουν για να συντομευθεί η διαδικασία**
- Έτσι ο δείκτης της σελίδας Δ δε θα δείχνει στη σελίδα Ε αλλά στη Ζ

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο - Παράδειγμα

- Αναζητούμε το κλειδί 670.
- Ισχύει $\text{exhash}(670) = 1001001011_2$.
- Ο κατάλογος αποτελείται από 8 σελίδες:
 - θεωρούνται τα 3 πρώτα bits (100) για την προσπέλαση του καταλόγου,
 - η αναζήτηση κατευθύνεται στη 5η σελίδα του καταλόγου,
 - ο κάδος έχει 8 σελίδες,
 - λαμβάνονται τα επόμενα 3 bits (100) για διευκρίνιση της σελίδας,
 - η αναζήτηση συνεχίζεται στην σελίδα Η,
 - αν δεν υπάρχει, συνεχίζεται στο ο-δένδρο στην σελίδα Θ.

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Όταν ένας κάδος γεμίσει (1^η τεχνική), τότε το μέγεθος του **διπλασιάζεται** και ενημερώνεται κατάλληλα η είσοδος του καταλόγου.
 - Ο κάδος επαναποθηκεύεται σε άλλο σημείο του δίσκου ώστε να μην χρειάζονται 2 προσπελάσεις
- Όταν ένας κάδος γεμίσει (2^η τεχνική), τότε **αφήνονται εγγραφές στην περιοχή υπερχείλισης** αντί να διπλασιαστεί το μέγεθος του κάδου.
 - Αυξημένο κόστος αναζήτησης αλλά καλύτερη χρήση του χώρου

Εκθετικός Κατακερματισμός με Περιορισμένο Κατάλογο

- Πλεονεκτήματα:
 - ✚ Δε χρειάζεται προσπέλαση στο δίσκο για τον κατάλογο
 - ✚ Απαιτείται μόνο μια προσπέλαση στο δίσκο αν δεν υπάρχει υπερχείλιση

Γραμμικός Κατακερματισμός

- Η μέθοδος του γραμμικού κατακερματισμού (linear hashing) **δεν χρησιμοποιεί κατάλογο**.
- **Διατηρεί λίστες υπερχείλισης**
 - Χρησιμοποιεί μεγαλύτερες αλυσίδες υπερχείλισης για να επιτευχθεί μεγαλύτερη πληρότητα των κάδων.

Γραμμικός Κατακερματισμός

- Το κλειδί μετασχηματίζεται με μια συνάρτηση κατακερματισμού ($h(k)$).
- Λαμβάνονται **τα k τελευταία δυαδικά ψηφία** του αποτελέσματος του μετασχηματισμού.

Γραμμικός Κατακερματισμός

- Αρχείο 8 κάδων χωρητικότητας 3 εγγραφών σε κάθε κάδο

000	56		
001	113	193	
010	10	146	
011	19		
100	4		
101	5		
110			
111			

$$4 = 0000\ 0100$$

$$5 = 0000\ 0101$$

$$10 = 0000\ 1010$$

$$19 = 0001\ 0011$$

$$56 = 0011\ 1000$$

$$113 = 0111\ 0001$$

$$146 = 1001\ 0010$$

$$193 = 1100\ 0001$$

Γραμμικός Κατακερματισμός

- Το αρχείο μεγαλώνει με διαδοχικές διασπάσεις κάδων:
 - οι εγγραφές μοιράζονται στους δύο κάδους ανάλογα με τις τιμές των $k+1$ τελευταίων bits.
- **Οριακή τιμή** (boundary value) ονομάζεται η τιμή πέρα από την οποία οι κάδοι διακρίνονται από τα τελευταία k bits.
- Στην επικεφαλίδα κάθε αρχείου γραμμικού κατακερματισμού αποθηκεύεται η οριακή τιμή και η αντίστοιχη τιμή του k .
- Η διαχείριση των εισαγωγών γίνεται με βάση την οριακή τιμή και την τιμή του k .

Γραμμικός Κατακερματισμός

- Η υπερχείλιση δεν προκαλεί απαραίτητα διάσπαση, ούτε η διάσπαση μειώνει απαραίτητα την υπερχείλιση.
- Η διάσπαση των κάδων είναι γραμμική από την αρχή προς το τέλος άσχετα από το ποιος κάδος υπερχειλίζει.

Γραμμικός Κατακερματισμός

- Εισαγωγή πέντε νέων εγγραφών

Πριν την εισαγωγή

000	56			4 = 0000 0100
001	113	193		5 = 0000 0101
010	10	146		10 = 0000 1010
011	19			19 = 0001 0011
100	4			56 = 0011 1000
101	5			113 = 0111 0001
110				146 = 1001 0010
111				193 = 1100 0001

Μετά την εισαγωγή

Διάσπαση των 3 πρώτων κάδων

0000				
0001	113	193		
0010	90	146		
* 011	19			
100	4	12	52	→ 100
101	5			
110				12 = 0000 1100
111	71			52 = 0011 0100
1000	56			71 = 0100 0111
1001				90 = 0101 1010
1010	10			100 = 0110 0100

Γραμμικός Κατακερματισμός - Αναζήτηση

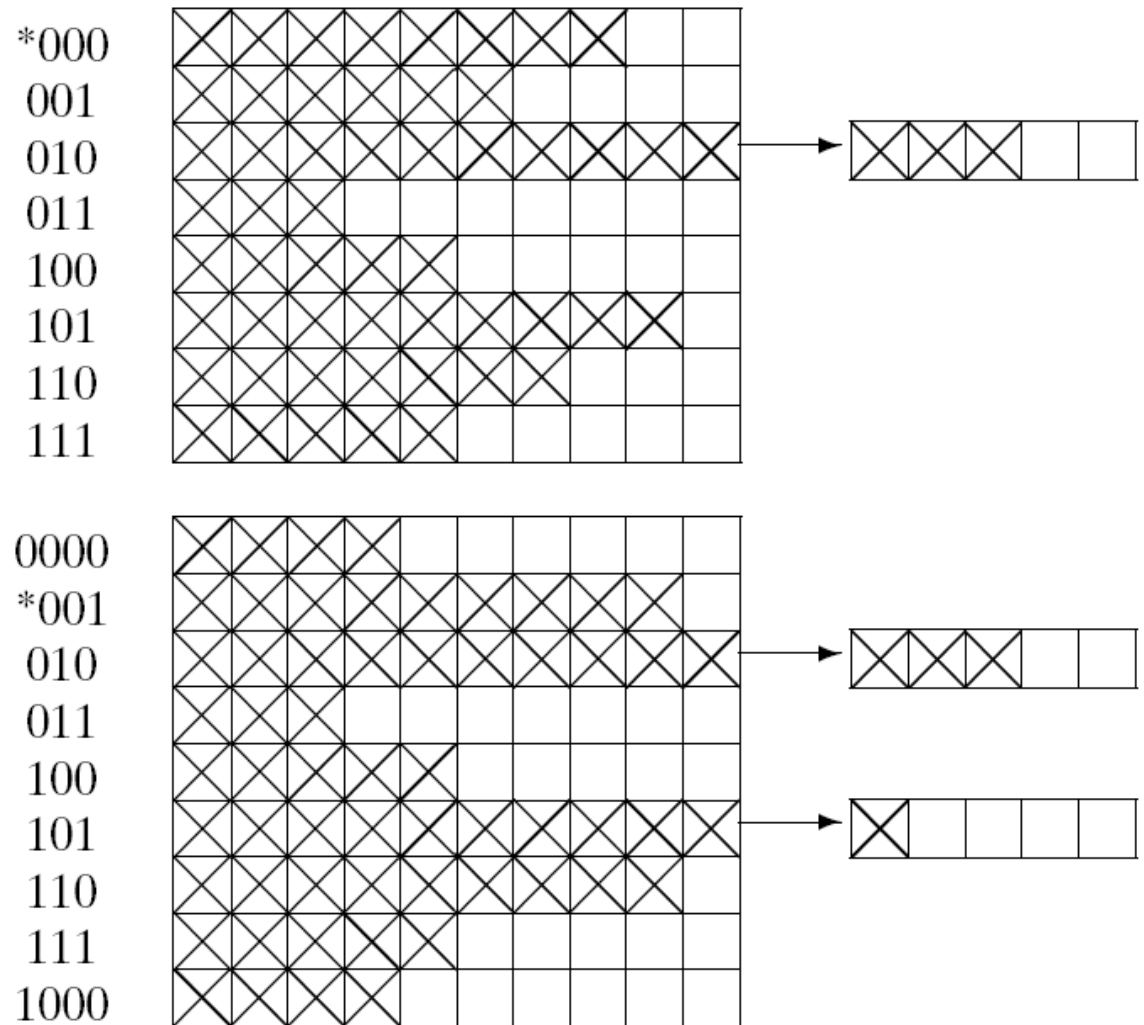
1. Απομονώνονται τα τελευταία k bits του αποτελέσματος του μετασχηματισμού.
 2. Αν η τιμή που προκύπτει είναι μικρότερη της οριακής τιμής τότε λαμβάνονται $k+1$ bits.
- Η επίδοση της αναζήτησης σε αρχείο γραμμικού κατακερματισμού είναι πολύ καλή και προσεγγίζει τη μία προσπέλαση στο δίσκο.

Γραμμικός Κατακερματισμός - Εισαγωγή

- Ένα αρχείο αποτελείται από 8 κύριους κάδους με χωρητικότητα 10 εγγραφών.
- Ο παράγοντας φόρτισης δεν πρέπει να υπερβεί το 70%.
- Εκτός από την κύρια περιοχή υπάρχει και η περιοχή υπερχείλισης.
- Γενικά όταν εισάγονται $Lf * Bkfr$ εγγραφές δημιουργείται ένας νέος κάδος.

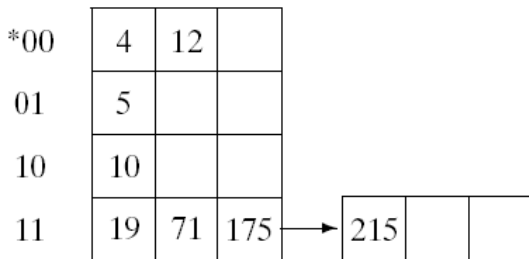
Γραμμικός Κατακερματισμός - Εισαγωγή

- Εγγραφές $n=56$.
- Παράγοντας φόρτισης $L_f=70\%$.
- Εισαγωγή 7 νέων εγγραφών.
- Διάσπαση κάδου και δέσμευση νέου κάδου ώστε να παραμείνει $L_f=70\%$.
- δε λαμβάνουμε υπόψη τις κενές θέσεις στις αλυσίδες υπερχείλισης

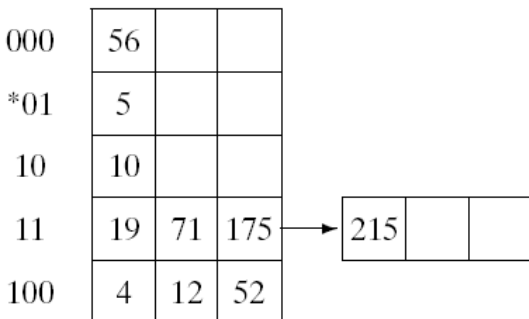


Γραμμικός Κατακερματισμός - Εισαγωγή

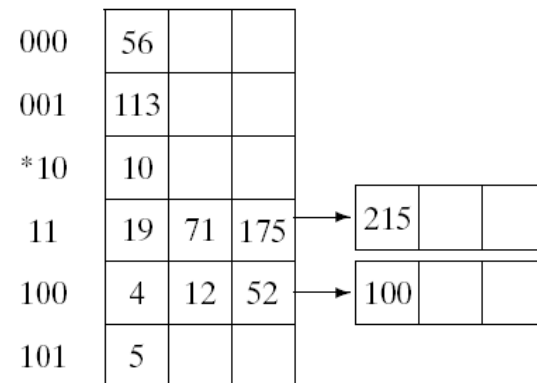
- Διαδοχικές εισαγωγές – παράγοντας φόρτισης: 67%
- 4 διαδοχικοί κάδοι με 3 εγγραφές σε κάθε κάδο (δε λαμβάνουμε υπόψη τις κενές θέσεις στις αλυσίδες υπερχειλίσης). Με την εισαγωγή 52, 56 διάσπαση του 1^{ου} κάδου. Εισαγωγή 100, 113 & διάσπαση του 2^{ου}.



(α)



(β) εισαγωγή 52, 56



(γ) εισαγωγή 100, 113

Γραμμικός Κατακερματισμός - Εισαγωγή

- Σε αρχείο με $Bkfr=50$ και $Lf=75\%$ απαιτούνται κατά μέσο όρο 2,62 προσπελάσεις στο δίσκο για την εισαγωγή μιας νέας εγγραφής:
 - απαιτείται 1 προσπέλαση για να έρθει ο κύριος κάδος στην κύρια μνήμη,
 - θα δημιουργηθεί ένας νέος κάδος → οι εγγραφές του αρχικού πρέπει να κατανεμηθούν στους 2 κάδους
 - απαιτούνται άλλες 2 προσπελάσεις στο δίσκο για την επανα-αποθήκευση των εγγραφών, και
 - αν ο αρχικός κάδος διαθέτει και αλυσίδα υπερχείλισης, τότε θα είναι απαραίτητο να προσπελασθεί και αυτή.

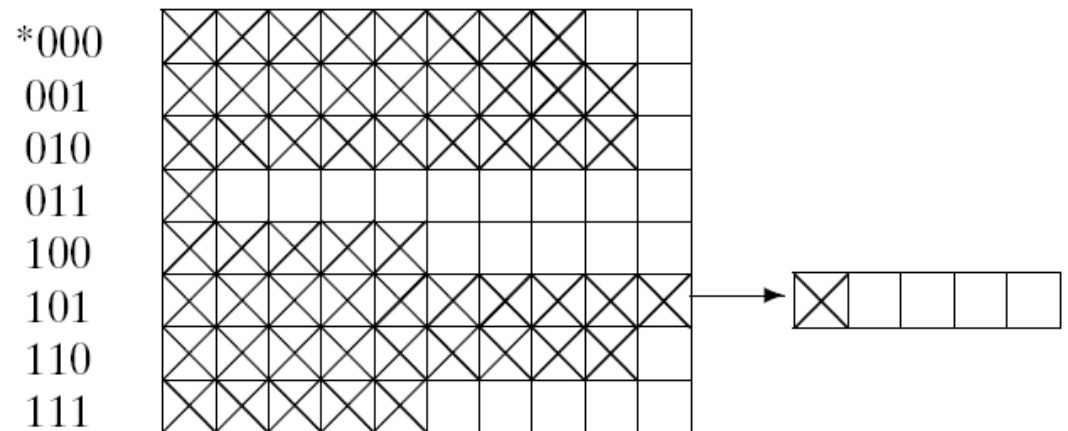
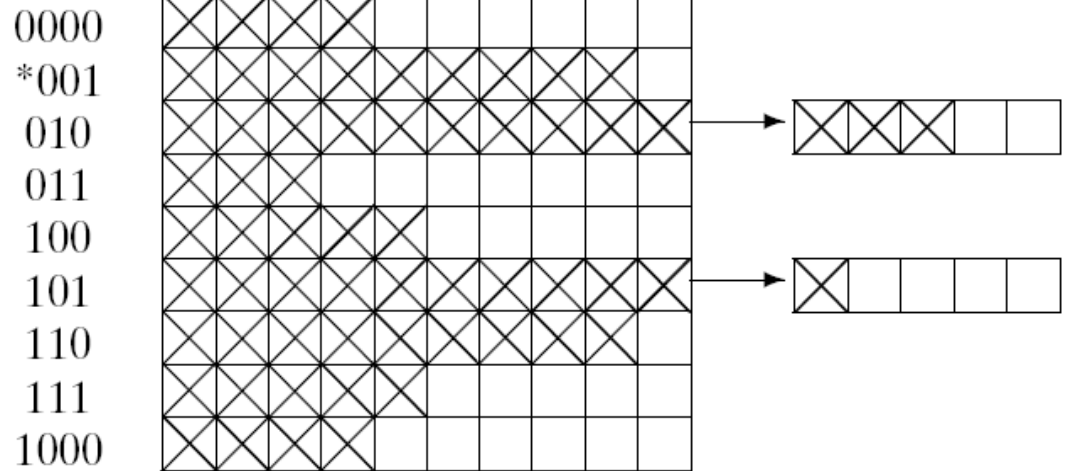
Γραμμικός Κατακερματισμός - Διαγραφή

Αλγόριθμος διαγραφής

1. Έρχονται στην κύρια μνήμη όλες οι εγγραφές του κάδου και της αλυσίδας υπερχείλισης και *μεταφέρεται η τελευταία εγγραφή της αλυσίδας στη θέση αυτής που διαγράφεται.*
2. Αν διαγράφεται η εγγραφή που είναι τελευταία στην αλυσίδα, τότε απλώς *ελευθερώνεται ο χώρος.*
3. Αν ο αριθμός των εγγραφών του αρχείου υπολείπεται κατά $Bkfr * Lf$ εγγραφές, τότε *ο τελευταίος κάδος συγχωνεύεται με τον κάδο που έχει κοινά τα k τελευταία bits.*

Γραμμικός Κατακερματισμός - Διαγραφή

- 4 εγγραφών από τον κάδο 010,
- 2 εγγραφών από τον κάδο 011,
- 1 εγγραφής από τον κάδο 1000.



Γραμμικός Κατακερματισμός - Παραλλαγές

- Γραμμικός κατακερματισμός με μερικές επεκτάσεις (linear hashing with partial expansions).
 - *Όταν ο παράγοντας χρησιμοποίησης χώρου υπερβεί την προκαθορισμένη τιμή, οι εγγραφές 2 κάδων αναδιανέμονται μεταξύ 3 κάδων.*
- +
- +
- αυξημένο κόστος εισαγωγής σε σχέση με την αρχική μέθοδο.

Γραμμικός Κατακερματισμός - Παραλλαγές

- Γραμμικός κατακερματισμός με προτεραιότητα διάσπασης (linear hashing with priority splitting).
- *Αν γίνει υπέρβαση της καθορισμένης τιμής του παράγοντα χρησιμοποίησης χώρου, τότε δίνεται προτεραιότητα στη διάσπαση του κάδου με τη μεγαλύτερη αλυσίδα υπερχείλισης.*
- Για την υλοποίηση αυτής της μεθόδου απαιτούνται επιπλέον δομές που όμως είναι σχετικά μικρές και αποθηκεύονται στην κύρια μνήμη.