Προηγμένα Θέματα Βάσεων Δεδομένων

Διδάσκων: Άγγελος Μιχάλας

Περιεχόμενα

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

Εισαγωγή

- Τυχαία ή άμεσα αρχεία (random, direct) είναι πολύ χρήσιμα για την φυσική οργάνωση των δεδομένων (μνήμη και δίσκο).
- Σε ένα τυχαίο/άμεσο αρχείο τα δεδομένα δεν αποθηκεύονται ακολουθιακά (sequentially). **Οι εγγραφές κατανέμονται στο αρχείο τυχαία**.

Εισαγωγή

- Τυχαία ή άμεσα αρχεία (random, direct) είναι πολύ χρήσιμα για την φυσική οργάνωση των δεδομένων (μνήμη και δίσκο).
  - εντοπίζεται η ζητούμενη εγγραφή με λίγες προσπελάσεις (ίσως και μόνο μία),
  - χρησιμοποιούνται για on-line εφαρμογές (OLTP),
  - Οι εγγραφές δεν χρειάζεται να ταξινομηθούν,
  - Η προσπέλαση κάθε εγγραφής είναι πολύ γρήγορη (ομοίως διαγραφή, ενημέρωση) καθώς η διέθυνση του block (page/track) στο δίσκο είναι γνωστή από τη συνάρτηση κατακεραμισμού (hash function).

  Παράδειγμα: Εάν ο πίνακας STUDENT γίνεται hashed στο πεδίο Name τότε η ανάκτηση της εγγραφής με Name ίσο με “Nikos Dimokas” είναι αποδοτική

Εισαγωγή

- Τυχαία ή άμεσα αρχεία (random, direct) είναι πολύ χρήσιμα για την φυσική οργάνωση των δεδομένων (μνήμη και δίσκο)
  - δεν προσφέρονται για **σειριακή ανάκτηση ταξινομημένων εγγραφών ή** για **ερωτήσεις διαστήματος** (range queries).
    - Π.χ., ανάκτηση όλων των φοιτητών που το όνομα ξεκινά με "R".
    - Π.χ., Εάν η σχέση STUDENT έχει hashed πεδίο το Roll Number και ζητείται η ανάκτηση όλων των φοιτητών με roll numbers μεταξύ 3000-5000.
  - δεν προσφέρονται για **ανάκτηση εγγραφών βασισμένη σε πεδίο διαφορετικό από το πεδίο κατακερματισμού** (hash field)
    - Π.χ., Εάν η σχέση STUDENT έχει hashed πεδίο το Roll Number, τότε ο κατακερματισμός δεν μπορεί να χρησιμοποιηθεί για αναζήτηση εγγραφής με βάση το πεδίο Class.

Εισαγωγή

- Τυχαία ή άμεσα αρχεία (random, direct) είναι πολύ χρήσιμα για την φυσική οργάνωση των δεδομένων (μνήμη και δίσκο)
  - δεν προσφέρονται **για ανάκτηση εγγραφών με βάση ένα μέρος του πεδίου κατακερματισμού** (hash field).
    - Π.χ., Εάν η σχέση STUDENT έχει hashed πεδίο το Roll Number και το Class, τότε ο κατακερματισμός δεν μπορεί να χρησιμοποιηθεί για αναζήτηση εγγραφής μόνο με βάση το πεδίο Class.
  - δεν προσφέρονται **όταν το πεδίο κατακερματισμού (hash field) ενημερώνεται τακτικά**. Το DBMS πρέπει να διαγράψει την εγγραφή και να επανατοποθετήση (πιθανόν) σε νέα θέση.

Εισαγωγή

- Βασική ιδέα: ***Με αλγεβρικό μετασχηματισμό στην τιμή του κλειδιού προκύπτει η απόλυτη διεύθυνση της εγγραφής.***
- Η απόλυτη διεύθυνση αποτελείται από την τριάδα:
  - αριθμός κυλίνδρου,
  - αριθμός ατράκτου και
  - αριθμός εγγραφής.

Εισαγωγή

- **Πυκνά** (dense) ονομάζονται τα κλειδιά με συνεχόμενες τιμές:
  - πχ. για κλειδιά από 1 έως 1000, η i-οστή εγγραφή θα αποθηκεύεται στην (i-1)-οστή σελίδα του αρχείου (αν η χωρητικότητα μιας σελίδας είναι μια εγγραφή),
  - πχ. για κλειδιά από 10.001 έως 15.000, αφαιρείται μια σταθερή τιμή (10000) από την τιμή του κλειδιού (θα μπορούσε οι τιμές των κλειδιών να μην αρχίζουν από το 1 αλλά από το 10000) και βρίσκεται η σελίδα του αρχείου.
- Τα αρχεία αυτού του είδους λέγονται αυτοδεικτοδοτούμενα και υλοποιούνται εύκολα σε όλες τις γλώσσες προγραμματισμού (self-indexed).

Εισαγωγή

- Όταν τα κλειδιά δεν είναι πυκνά το μεγαλύτερο μέρος του αρχείου είναι κενό.
- Στην περίπτωση αυτή εφαρμόζεται μετασχηματισμός των κλειδιών έτσι ώστε το διάστημα των κλειδιών να αντιστοιχεί σε ένα πολύ μικρότερο διάστημα τιμών διευθύνσεων.
- Η μέθοδος αυτή ονομάζεται **τεχνική διασκορπισμού αποθήκευσης** (scatter storage technique) ή **μετασχηματισμός του κλειδιού σε διεύθυνση** (key-to-address- transformation) ή αλλιώς **κατακερματισμός** (hashing) και για αυτό τα αρχεία λέγονται και αρχεία κατακερματισμού.

Εισαγωγή

- Αρχικά δεσμεύεται χώρος στο δίσκο για το αρχείο που ονομάζεται **κύρια περιοχή** (main area).
- Σε μερικές παραλλαγές της μεθόδου προβλέπεται και χρήση **περιοχής υπερχείλισης** (overflow area).
  - Σε άλλες παραλλαγές δε προβλέπεται
  - Η περιοχή υπερχείλισης (όταν προβλέπεται) **βρίσκεται στον ίδιο κύλινδρο** για να μειωθεί το κόστος εντοπισμού.
- **Παράγοντας διευθύνσεων** (address factor) είναι το *κλάσμα του μεγέθους της κύριας περιοχής προς το συνολικό μέγεθος του αρχείου*.
  - Αν ο χώρος που έχει κρατηθεί για το αρχείο δεν αυξομειώνεται τότε το αρχείο είναι στατικό

Εισαγωγή

- Τα δυναμικά τυχαία (dynamic random) αρχεία είναι αυτά που μεγεθύνονται ή συρρικνώνονται ανάλογα με τις εισαγωγές και διαγραφές.
- Οργανώσεις για τον χειρισμό αυτών των αρχείων είναι:
  - Δυναμικός κατακερματισμός (dynamic hashing),
  - Επεκτατός κατακερματισμός (extendible hashing),
  - Εκθετικός κατακερματισμός με περιορισμένο κατάλογο (bounded index exponential hashing),
  - Γραμμικός κατακερματισμός (linear hashing).

Εισαγωγή

- Οι τεχνικές αυτές χρησιμοποιούν σύνθετους αλγόριθμους για να απαντήσουν στις ερωτήσεις:
  1. πως και πότε διασπάται ένας κάδος,
  2. πως διανέμονται οι εγγραφές από τον παλιό κάδο στους νέους,
  3. πως και πότε συγχωνεύονται δύο κάδοι,
  4. πως οι εγγραφές από τους δύο παλιούς κάδους αποδίδονται στο νέο κάδο.

Συναρτήσεις κατακερματισμού

- Μια συνάρτηση μετασχηματίζει την τιμή του κλειδιού σε τιμή διεύθυνσης στη δευτερεύουσα μνήμη με τυχαίο τρόπο:
  - όταν το κλειδί είναι αριθμητικό, εφαρμόζεται απλή αλγεβρική έκφραση,
  - όταν το κλειδί είναι αλφαριθμητικό, εφαρμόζεται μετατροπή σε αριθμητικό χρησιμοποιώντας τους κώδικες ASCII, EBCDIC, κ.α.
- *Καλή συνάρτηση είναι εκείνη που διασπείρει τις εγγραφές σε όλη την έκταση του αρχείου.*

Συναρτήσεις κατακερματισμού

- Το πρόβλημα είναι ότι **δύο ή περισσότερες τιμές κλειδιών** είναι δυνατόν να δώσουν την **ίδια διεύθυνση**.
  - Το φαινόμενο καλείται **σύγκρουση** (collision) και οι εγγραφές **συνώνυμες** (synonyms).
- Όταν σημαντικός αριθμός εγγραφών συνωστίζονται στην ίδια περιοχή του αρχείου τότε έχουμε **πρωτεύουσα συγκέντρωση** (primary clustering).
  - Έτσι δεσμεύεται για το αρχείο περισσότερος χώρος από την αρχική πρόβλεψη.

Συναρτήσεις κατακερματισμού

- Το λεγόμενο παράδοξο των γενεθλίων (birthday paradox):
  - έστω ότι επιλέγονται τυχαία από το πλήθος 23 άτομα,
  - η πιθανότητα τουλάχιστον δύο άτομα από το σύνολο των 23 ατόμων να έχουν την ίδια μέρα γενέθλια είναι 50.83%.

Μοντελοποίηση συγκρούσεων

- Αν **n εγγραφές** πρέπει να αποθηκευθούν σε **m θέσεις**, τότε ο αριθμός των δυνατών τοποθετήσεων είναι $\mathbf{m^n}$.(**Διάταξη με επανάληψη**)
  - για παράδειγμα, αν n=4 και m=5, τότε ο αριθμός των τοποθετήσεων είναι 625.
- Οι βολικές τοποθετήσεις (1 προσπέλαση) είναι **m!/(m–n)!**. (**Διάταξη**)
  - στο παράδειγμα, m!/(m–n)! = 120.

Μοντελοποίηση συγκρούσεων

- Χειρότερη περίπτωση
  - Όλες οι εγγραφές να αντιστοιχούν σε μια θέση.
    - στο παράδειγμα, και οι 4 εγγραφές να αντιστοιχούν σε 1 θέση
  - Απαιτούμενες προσπελάσεις (n+1)/2.
    - στο παράδειγμα, 2,5 προσπελάσεις κατά μέσο όρο.
  - *Ο αριθμός των δυσμενών περιπτώσεων είναι m*.
    - στο παράδειγμα είναι 5.

Μοντελοποίηση συγκρούσεων

- Οι υπόλοιπες τοποθετήσεις διακρίνονται σε συγκρούσεις από 2 έως n–1 εγγραφών.
  - στο παράδειγμα, οι υπόλοιπες 500 τοποθετήσεις διακρίνονται σε τρία είδη:
    - μία σύγκρουση 2 εγγραφών,
    - δύο συγκρούσεις 2 εγγραφών, και
    - μία σύγκρουση 3 εγγραφών.

Μοντελοποίηση συγκρούσεων

- Η πιθανότητα μιας βολικής τοποθέτησης είναι:

$$p_0 = \frac{\frac{m!}{(m-n)!}}{m^n}$$

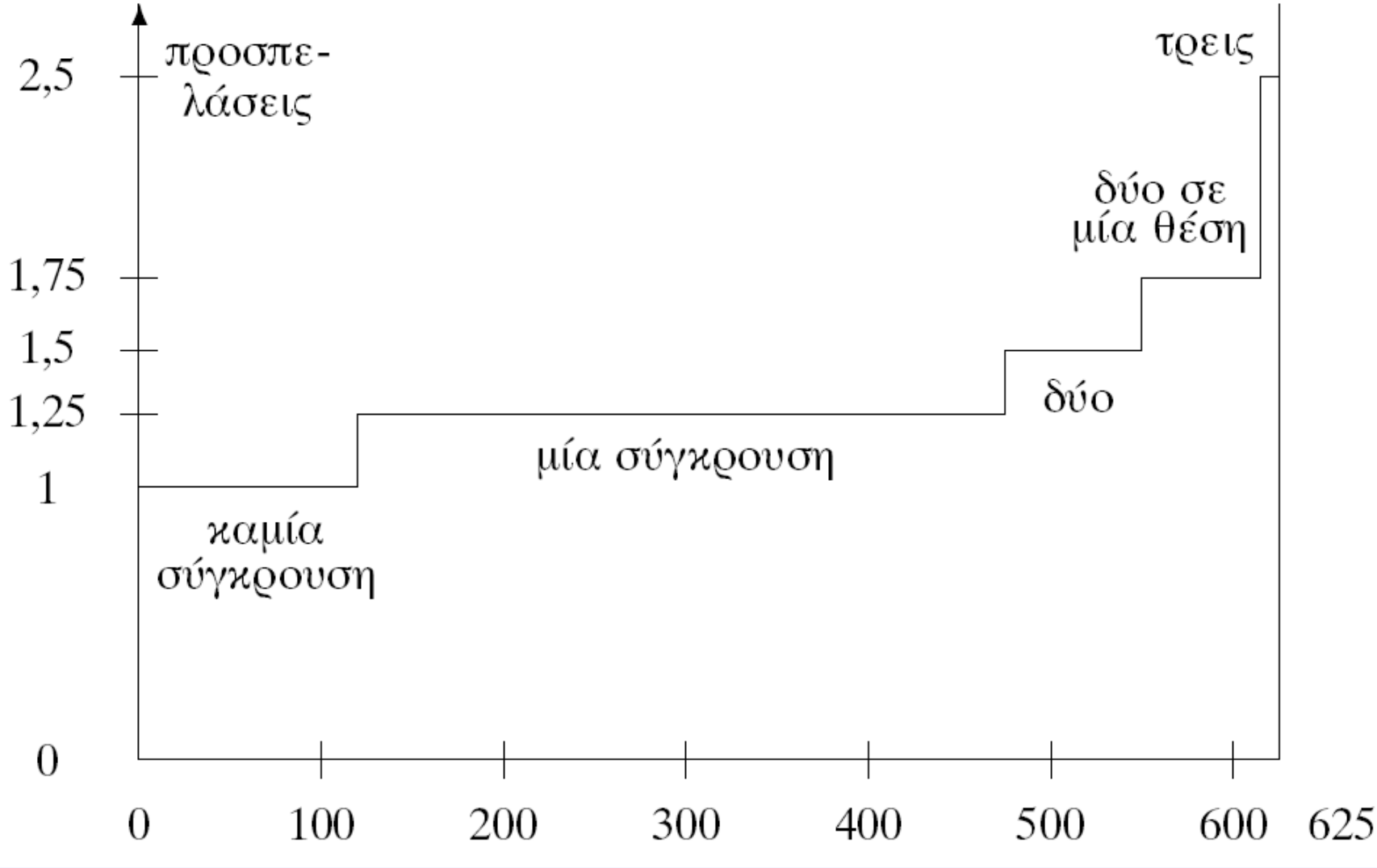
  - στο παράδειγμα, για n=4 και m=5, p0=0.192.

- Η πιθανότητα της πιο δυσμενούς τοποθέτηση είναι:

$$p_{max} = \frac{m}{m^n}$$

  - στο παράδειγμα, για n=4 και m=5, pmax=0.008.

Μοντελοποίηση συγκρούσεων



Χειρισμός συγκρούσεων

- Σχήματα χειρισμού συγκρούσεων:
  - **Open addressing** (ανοιχτή διευθυνσιοδότηση): Ξεκινώντας από την κατειλημμένη θέση που προσδιορίζεται από τη διεύθυνση κατακερματισμού, το πρόγραμμα ελέγχει τις διαδοχικές θέσεις στη σειρά μέχρι να βρεθεί μια αχρησιμοποίητη (κενή) θέση.

Χειρισμός συγκρούσεων

- Σχήματα χειρισμού συγκρούσεων:
  - **Chaining** (Αλυσιδωτή σύνδεση): Διατηρούμε περιοχές θέσεων υπερχείλισης, συνήθως επεκτείνοντας τον πίνακα με έναν αριθμό θέσεων υπερχείλισης.
    - Σε κάθε θέση εγγραφής προστίθεται ένα πεδίο δείκτη
    - Μία σύγκρουση επιλύεται τοποθετώντας τη νέα εγγραφή σε μια αχρησιμοποίητη θέση υπερχείλισης και θέτοντας ως τιμή του δείκτη της κατειλημμένης θέσης τη διεύθυνση αυτής της θέση υπερχείλισης

Χειρισμός συγκρούσεων

- Σχήματα χειρισμού συγκρούσεων:
  - **Πολλαπλός κατακερματισμός**: Το πρόγραμμα εφαρμόζει μια δεύτερη συνάρτηση κατακερματισμού σε περίπτωση σύγκρουσης.
    - Αν υπάρχει πάλι σύγκρουση τότε ανοιχτή διευθυνσιοδότηση ή εφαρμογή τρίτης συνάρτησης κατακερματισμού (και στη συνέχεια ανοιχτή διευθυνσιοδότηση αν είναι απαραίτητο)

Συναρτήσεις κατακερματισμού

- **Τέλειες** (perfect) συναρτήσεις κατακερματισμού ονομάζονται αυτές που δεν παράγουν συνώνυμα.
- Χρησιμοποιούνται μόνο σε μικρούς πίνακες (επειδή στη πράξη προϋποθέτουν την εκ των προτέρων γνώση των κλειδιών) της κύριας μνήμης για ειδικές εφαρμογές όπως:
  - σε μεταφραστές για αποθήκευση δεσμευμένων λέξεων
  - σε επεξεργασία φυσικής γλώσσας για φιλτράρισμα λέξεων υψηλής συχνότητας

Συναρτήσεις κατακερματισμού

- **Ελάχιστη** (minimal) ονομάζεται η τέλεια συνάρτηση που δεσμεύει τον ελάχιστο δυνατό χώρο.
- Για περιπτώσεις ογκωδών στατικών ή δυναμικών αρχείων (πχ. αρχεία μηχανών αναζήτησης) έχει προταθεί μια κλάση μεθόδων που λέγεται εξωτερικός (external) τέλειος κατακερματισμός.
  - Δε θα ασχοληθούμε.
  - Εξαιρετικά πολύπλοκη.

Παράγοντας φόρτισης

- **Ο λόγος των κατειλημμένων θέσεων προς το σύνολο των θέσεων** του αρχείου ονομάζεται παράγοντας φόρτωσης (load factor) και ισούται:

$$Lf = \frac{n}{bk_m \times Bkfr}$$

  - bk_m είναι ο αριθμός των κάδων στην κύρια περιοχή, και
  - *Bkfr* είναι ο παράγοντας καδοποίησης (bucket factor), δηλαδή ο αριθμός των εγγραφών ανά κάδο.

Παράγοντας φόρτωσης

- Όσο μικρότερος είναι ο παράγοντας φόρτωσης, τόσο μικρότερη είναι η πιθανότητα σύγκρουσης και το αντίστροφο.
- Ο σχεδιαστής αρχείων είναι υπεύθυνος για να βρει την ισορροπία μεταξύ των συγκρούσεων και του ποσοστού αχρησιμοποίητου χώρου του αρχείου.

Υπερχείλιση

- **Υπερχείλιση** (overflow) συμβαίνει όταν μία εγγραφή πρέπει να αποθηκευθεί σε ένα πλήρη κάδο. Τότε η εγγραφή κατευθύνεται για αποθήκευση σε άλλον κάδο.
- Όσο μεγαλύτερος είναι ο κάδος, τόσο μικρότερη είναι η πιθανότητα να υπάρχει υπερχείλιση.
- Όσο μεγαλύτερο είναι το μέγεθος του κάδου, τόσο πιο χρονοβόρα είναι η προσπέλαση στο δίσκο.

Συναρτήσεις κατακερματισμού

- Μια **συνάρτηση κατακερματισμού** (hash function) h **απεικονίζει κλειδιά** ενός δοσμένου τύπου **σε ακεραίους ενός σταθερού διαστήματος** [0, N − 1] όπου N το πλήθος των κάδων
- Παράδειγμα:
  - h(x) = x mod N

    είναι μια συνάρτηση κατακερματισμού για ακέραια κλειδιά
- Ο ακέραιος h(x) ονομάζεται τιμή κατακερματισμού (hash value) του κλειδιού x
- Ο σκοπός μιας συνάρτησης κατακερματισμού είναι η **ομοιόμορφη διασπορά κλειδιών στο πεδίο [0, N – 1]**

Συναρτήσεις κατακερματισμού

- **Διαίρεση με πρώτο αριθμό** (prime number division). Η τιμή της διεύθυνσης, όπου θα αποθηκευθεί η εγγραφή, ισούται με το **υπόλοιπο της διαίρεσης της τιμής του κλειδιού δια του μεγέθους του αρχείου**.
- Οι συγκρούσεις ελαχιστοποιούνται αν διαιρέτης είναι ο μεγαλύτερος πρώτος αριθμός που είναι μικρότερος από το μέγεθος του αρχείου.
- Έστω ότι δίνεται μία εγγραφή με κλειδί 172.148 για να αποθηκευθεί σε αρχείο με 7000 κάδους. Ισχύει:

  172148 mod 6997 = 4220

Συναρτήσεις κατακερματισμού

- **Μετατροπή ρίζας** (radix conversion). Θεωρείται ότι η τιμή του κλειδιού δεν είναι αριθμός του δεκαδικού συστήματος και επομένως πρέπει να μετατραπεί σε αριθμό του συστήματος αυτού.
- Έτσι αν υποτεθεί ότι ο συγκεκριμένος αριθμός έχει ως βάση το 11, τότε με τη μετατροπή προκύπτει:

  $1x11^5 + 7x11^4 + 2x11^3 + 1x11^2 + 4x11 + 8 = 266373$

  - Το αποτέλεσμα **κανονικοποιείται** διαιρώντας με 6997.
  - Η διεύθυνση του κάδου του αρχείου ισούται με 487.

Συναρτήσεις κατακερματισμού

- **Μέση του τετραγώνου** (mid square). Λαμβάνονται τα μεσαία ψηφία του τετραγώνου της τιμής του κλειδιού.
- Τετραγωνίζει την τιμή του κλειδιού, και μετά παίρνει τα **r μεσαία bits** του αποτελέσματος, δίνοντας μια τιμή μεταξύ 0 μέχρι 2r-1
- Κανονικοποίηση.
  - π.χ. (Επιλογή r μεσαίων ψηφίων από (172148* 172148)) mod 6997

Συναρτήσεις κατακερματισμού

- **Μετακίνηση** (move) ή **Δίπλωση** (fold). Η τιμή του κλειδιού χωρίζεται σε δύο τμήματα που προστίθενται.
  - δίνεται ο αριθμός 17207359 που χωρίζεται σε δύο τετραψήφιους αριθμούς, τους 1720 και 7359.
- Σύμφωνα με την πρώτη μέθοδο (**μετακίνηση**):
  - γίνεται **πρόσθεση** και προκύπτει 9079,
  - ακολουθεί κανονικοποίηση ως προς το μέγεθος του αρχείου.
- Σύμφωνα με τη δεύτερη μέθοδο (**δίπλωση**):
  - **προστίθενται οι αριθμοί, αφού πρώτα η σειρά των ψηφίων του δεύτερου αριθμού αντιστραφεί**: 1720+9537=11257,
  - ακολουθεί κανονικοποίηση.

Δυναμικός κατακερματισμός

- Ο δυναμικός κατακερματισμός ήταν η **πρώτη** χρονικά **δομή δυναμικών τυχαίων αρχείων** που εμφανίσθηκε στη βιβλιογραφία.
  - Ο ερευνητής που την παρουσίασε **[1978]**, ο **Larson** θεωρείται από τους θεμελιωτές της περιοχής αυτής.
- Ο δυναμικός κατακερματισμός **αποτελείται από** δύο φυσικά ανεξάρτητες δομές: **έναν κατάλογο** και **ένα κύριο αρχείο**.

Δυναμικός κατακερματισμός

- Ο κατάλογος είναι ένα **δάσος δυαδικών δένδρων** που υλοποιούνται ως συνδεδεμένες δομές στην κύρια μνήμη.
- Στο τελευταίο επίπεδο των δυαδικών δένδρων περιέχονται δείκτες προς τις σελίδες του κύριου αρχείου που είναι αποθηκευμένο στη δευτερεύουσα μνήμη.
- **Ο αριθμός των δυαδικών δένδρων** που αποτελούν το δάσος ισούται με τον αριθμό των κάδων, **bk**.
- **Μία συνάρτηση κατακερματισμού**:

  $$h(key) = key \bmod bk$$

  δίνει τον κάδο (και το δυαδικό δένδρο επομένως), που περιέχει την εγγραφή.

Δυναμικός κατακερματισμός

- *Κάθε κάδος έχει χωρητικότητα Bkfr εγγραφές*.
- Κατά την εισαγωγή της (*Bkfr*+1)-οστής εγγραφής γίνεται **διάσπαση κάδου** για να την δεχθεί.
- Το σύστημα παραχωρεί ένα νέο κάδο και **οι (*Bkfr*+1) εγγραφές αναδιανέμονται μεταξύ των δύο κάδων**.

Δυναμικός κατακερματισμός

- Για την αναδιανομή μεταξύ των δύο κάδων:
  - Χρησιμοποιείται μία **δεύτερη συνάρτηση h2(key)** που με σπόρο το κλειδί **παράγει μία ψευδοτυχαία δυαδική συμβολοσειρά** οποιουδήποτε μήκους, όπου τα **0 και 1 εμφανίζονται ισοπίθανα**.
  - Αν το πρώτο bit είναι 0 (αντίστοιχα, 1), τότε η εγγραφή κατευθύνεται στον παλαιό (αντίστοιχα, στο νέο) κάδο.
  - Γενικά, χρησιμοποιούνται τόσα bits της δυαδικής συμβολοσειράς όσα είναι απαραίτητα
    - Αρχικά, θέλουμε λίγα μόνο bits αλλά με τις διαδοχικές διασπάσεις των κάδων απαιτούνται όλο και περισσότερα.

Δυναμικός κατακερματισμός

- **Αρχικά κάθε ένα** από τα **bk δυαδικά δένδρα** του δάσους **αποτελείται μόνο από μία ρίζα**, οπότε ο κατάλογος έχει μόνο ένα επίπεδο.
- Όταν γίνει κάποια **διάσπαση κάδου**, τότε πρέπει το αντίστοιχο δένδρο να **επεκταθεί** κατά ένα επίπεδο.
- Από τη ρίζα του αντίστοιχου δένδρου δημιουργούνται δύο απόγονοι εξωτερικοί κόμβοι με τους αντίστοιχους δείκτες προς τους δύο κάδους.
- ***Ο αριστερός δείκτης δείχνει στον υπάρχοντα κάδο, ενώ ο δεξιός δείκτης δείχνει στο νέο κάδο***.

Δυναμικός κατακερματισμός

- Αν η συνάρτηση h_1 διασπείρει τυχαία τις εγγραφές στα δυαδικά δένδρα τότε:
  - τα **δένδρα** έχουν το **ίδιο** περίπου **μέγεθος** και
  - τα δένδρα είναι **ισοζυγισμένα** επειδή η ψευδοτυχαία συνάρτηση h_2 παράγει τα 0 και 1 ισοπίθανα.

Δυναμικός κατακερματισμός

Οι τύποι των εσωτερικών-εξωτερικών κόμβων είναι διαφορετικοί.

- Οι εσωτερικοί κόμβοι αποτελούνται από τα εξής πεδία:
  - σημαία (τιμή 0),
  - δενδρικός δείκτης προς τον πατέρα κόμβο,
  - δενδρικός δείκτης προς τον αριστερό απόγονο, και
  - δενδρικός δείκτης προς τον δεξιό απόγονο.
- Οι εξωτερικοί κόμβοι αποτελούνται από τα εξής πεδία:
  - σημαία (τιμή 1),
  - δενδρικός δείκτης προς τον πατέρα κόμβο,
  - δείκτης προς τον κάδο του αρχείου, και
  - μετρητής των αποθηκευμένων εγγραφών στον κάδο.

Δυναμικός κατακερματισμός

- Έστω ότι σε κενό αρχείο δυναμικού κατακερματισμού με κάδους μεγέθους δύο εγγραφών πρόκειται να εισαχθούν διαδοχικά εγγραφές με τιμές κλειδιών
  - 4, 5, 10, 12, 19, 52, 56, 72 και 90.
- Αρχικά το αρχείο σχεδιάζεται ώστε να έχει 3 κάδους.
- Ως **πρώτη συνάρτηση** κατακερματισμού χρησιμοποιείται η συνάρτηση ***h1(key) = key mod 3***, που χωρίζει το δάσος του καταλόγου σε 3 διακριτά δυαδικά δένδρα.
- Ως **δεύτερη συνάρτηση** επιλέγεται η συνάρτηση ***key mod 10***, ώστε από το κλειδί να προκύψει υπόλοιπο από 0 ως 9.

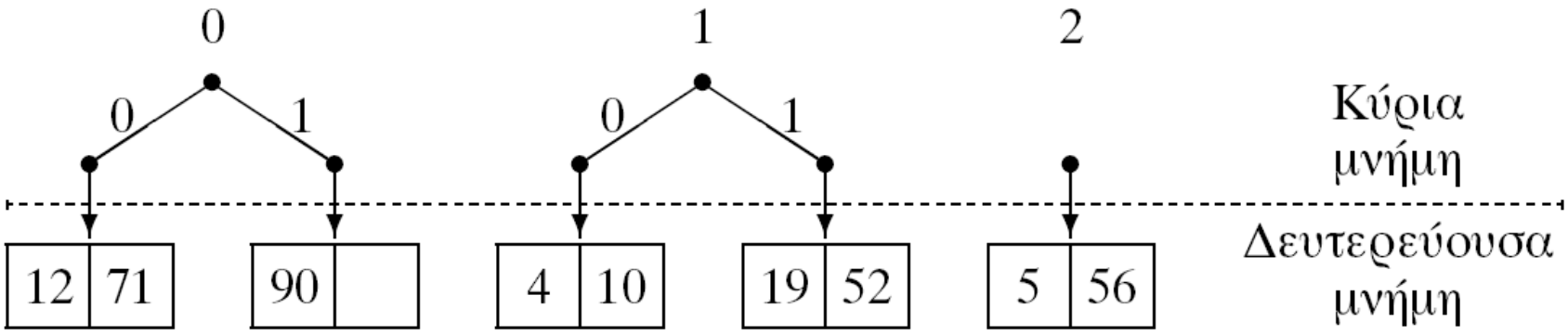
Δυναμικός κατακερματισμός

- Κάθε τιμή του υπολοίπου θεωρείται ως σπόρος σε μία ψευδοτυχαία γεννήτρια, R(), που παράγει δυαδικά ψηφία 0 και 1 με την ίδια πιθανότητα (δηλαδή 50%).
- Το αποτέλεσμα μίας τέτοιας γεννήτριας για κάθε σπόρο φαίνεται στον πίνακα:

R(0)	R(1)	R(2)	R(3)	R(4)	R(5)	R(6)	R(7)	R(8)	R(9)
1011	0000	0100	0110	1111	0101	0001	1110	1001	0011

Δυναμικός κατακερματισμός

- Το τελικό αποτέλεσμα της εισαγωγής των 9 εγγραφών με τη δημιουργία του καταλόγου των 3 δένδρων και του αρχείου των 5 κάδων είναι το εξής.



- Σημείωση: 72 και όχι 71
- Σημείωση: 19, 52 είναι πεδιά στο δείκτη 0 και 4, 10 στο δείκτη 1.

Δυναμικός κατακερματισμός

- Στο προηγούμενο παράδειγμα απαιτείται 1 μόνο προσπέλαση στο δίσκο για επιτυχή αναζήτηση.
- Εξαιτίας των διαδοχικών εισαγωγών ο κατάλογος αυξάνει και τμήμα του θα πρέπει να αποθηκευτεί στη δευτερεύουσα μνήμη, οπότε απαιτούνται 2 προσπελάσεις στην επιτυχή αναζήτηση.
- **Η μέση τιμή του παράγοντα χρησιμοποίησης χώρου είναι 69% (όπως και στα B-δένδρα)**.

Δυναμικός κατακερματισμός - Παραλλαγή

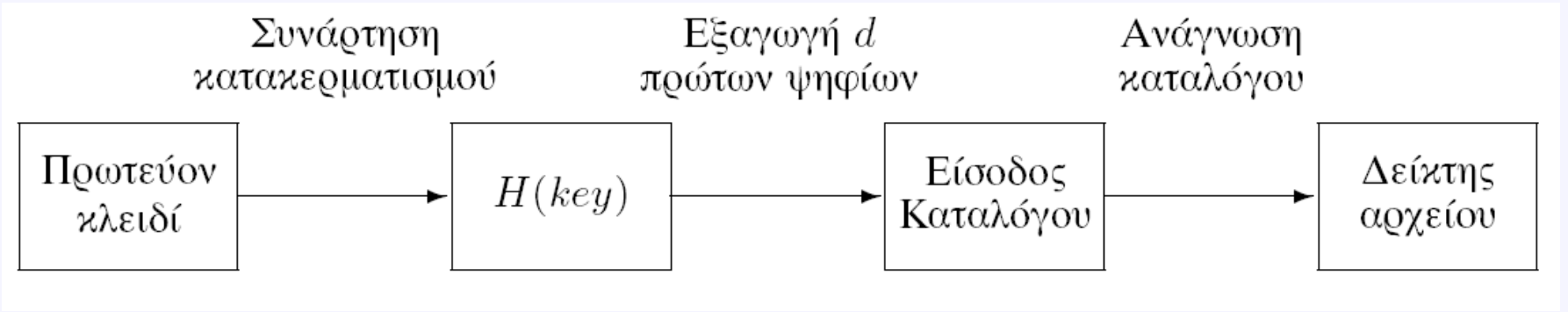
- **Τεχνική αναβολής διάσπασης** (deferred splitting).
- **Δεν γίνεται διάσπαση του κάδου** όταν σ'αυτόν κατευθυνθεί η (*Bkfr*+1)-οστή εγγραφή, αλλά **δημιουργείται αλυσίδα υπερχείλισης με ένα δεύτερο κάδο**.
- Το ζεύγος αυτό των κάδων θα διασπασθεί και ο κατάλογος θα ενημερωθεί όταν θα έχουν εισαχθεί ***β * Bkfr*** εγγραφές, όπου 1<=*β*<=2.
- Το πλεονέκτημα είναι ότι ο κατάλογος θα είναι μικρότερος κατά *β* φορές και θα χωρά στην κύρια μνήμη.

Δυναμικός κατακερματισμός - Παραλλαγή

- Αν το αρχείο μεγαλώσει υπερβολικά, τότε μπορεί δίνοντας στο β μεγαλύτερες τιμές να επιτραπούν αλυσίδες υπερχείλισης μεγαλύτερου μήκους, ώστε ο κατάλογος να χωρά οπωσδήποτε στην κύρια μνήμη.
- Στη χειρότερη περίπτωση θα γίνουν β προσπελάσεις στο δίσκο.

Επεκτατός κατακερματισμός

- Η δομή αυτή προτάθηκε από τους Fagin et al. [1979] και ήταν η χρονικά **δεύτερη υλοποίηση δυναμικού τυχαίου αρχείου**.
- Η αλληλουχία των μετασχηματισμών του κλειδιού που εκτελούνται στον επεκτατό κατακερματισμό έχει ως εξής:



Επεκτατός κατακερματισμός

- Ο πρώτος μετασχηματισμός είναι μία συνάρτηση κατακερματισμού που **απεικονίζει** κατά τυχαίο τρόπο **ένα διάστημα κλειδιών** σε ένα **σταθερό διάστημα διευθύνσεων**:
  - ο μόνος περιορισμός είναι ότι ***το διάστημα διευθύνσεων πρέπει να είναι δύναμη του δύο***,
  - προτιμάται να είναι πρώτος αριθμός, γι' αυτό επιλέγεται ο ***μεγαλύτερος ακέραιος που είναι αμέσως μικρότερος από τη δυαδική δύναμη***.
    - για παράδειγμα, ο μεγαλύτερος πρώτος αριθμός που είναι μικρότερος από το 2^{16} είναι ο 65521.

Επεκτατός κατακερματισμός

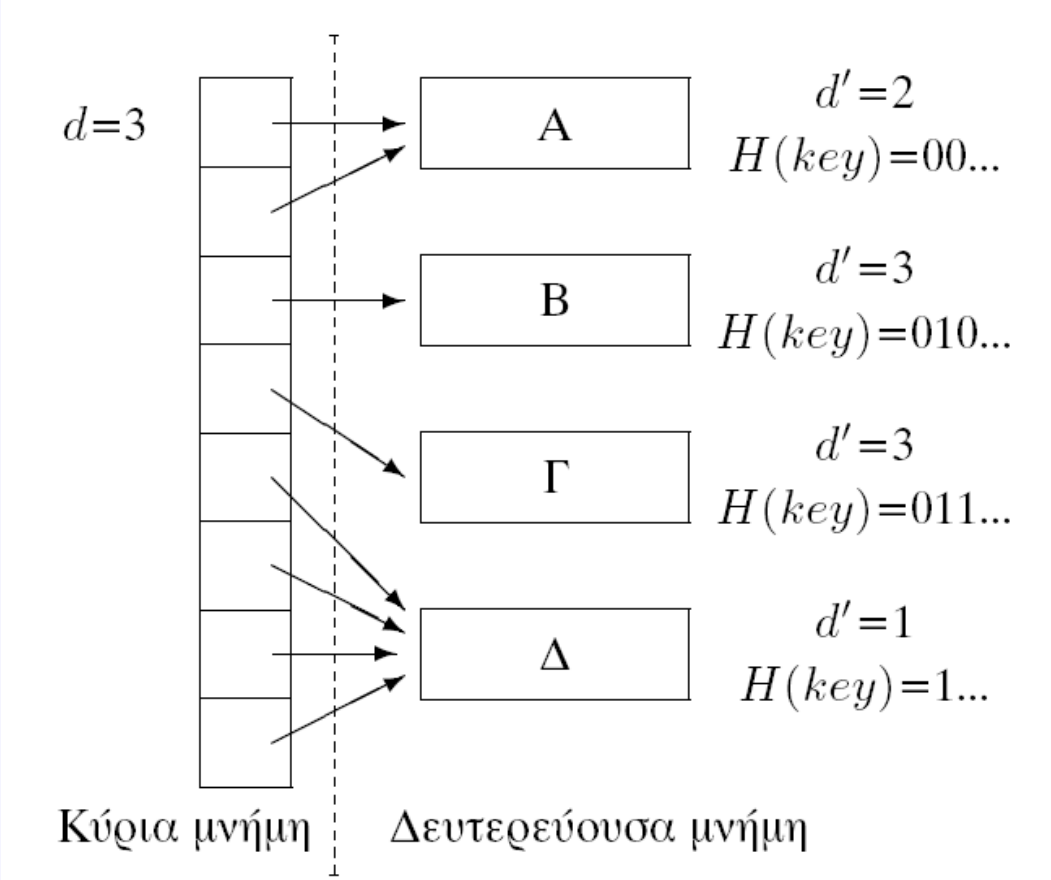
- Στη συνέχεια **η τιμή του κλειδιού μετατρέπεται στον ισοδύναμο δυαδικό αριθμό** και λαμβάνονται τα πρώτα **d ψηφία του**. Είναι δυνατόν να μη ληφθούν τα πρώτα bits αλλά τα τελευταία ή κάποια μεσαία.
- Τα bits αυτά λαμβάνονται ως είσοδος σε κατάλογο, που περιέχει δείκτες προς τους κάδους του αρχείου.

Επεκτατός κατακερματισμός

- Ο κατάλογος είναι ένας **μονοδιάστατος πίνακας με 2^d στοιχεία**, όπου **d** είναι ο αριθμός των επιλεγόμενων ψηφίων, και λέγεται **βάθος** (depth) ή **επίπεδο** (level) **του καταλόγου**.
- Ο αριθμός των ψηφίων που εξάγονται από το αποτέλεσμα της συνάρτησης κατακερματισμού μεταβάλλεται χρονικά ανάλογα με τη μεταβολή του μεγέθους του αρχείου.

Επεκτατός κατακερματισμός

- Χρησιμοποιούνται τα πρώτα τρία ψηφία του μετασχηματισμού.
- Ο πίνακας αποτελείται από 8 δείκτες που αναφέρονται σε 8 το πολύ κάδους.



Επεκτατός κατακερματισμός

- Έστω ότι η δυαδική μορφή του μετασχηματισμού ενός κλειδιού είναι 011010010110010 1. Από αυτόν τον αριθμό απομονώνονται τα 3 πρώτα bits (011) που ισοδυναμούν με το δεκαδικό αριθμό 3.
- Με προσπέλαση στην υπ' αριθμό 3 είσοδο του πίνακα βρίσκεται ο κατάλληλος δείκτης που αναφέρεται στον κάδο Γ.

Επεκτατός κατακερματισμός

- Κάθε κάδος συνοδεύεται από μία παράμετρο ***d'***, που λέγεται **βάθος** (depth) ή **επίπεδο** (level) του **κάδου**.
- ***Το βάθος δηλώνει τον αριθμό των bits που είναι κοινός για τα κλειδιά όλων των εγγραφών του κάδου και είναι d'<=d.***

Επεκτατός κατακερματισμός

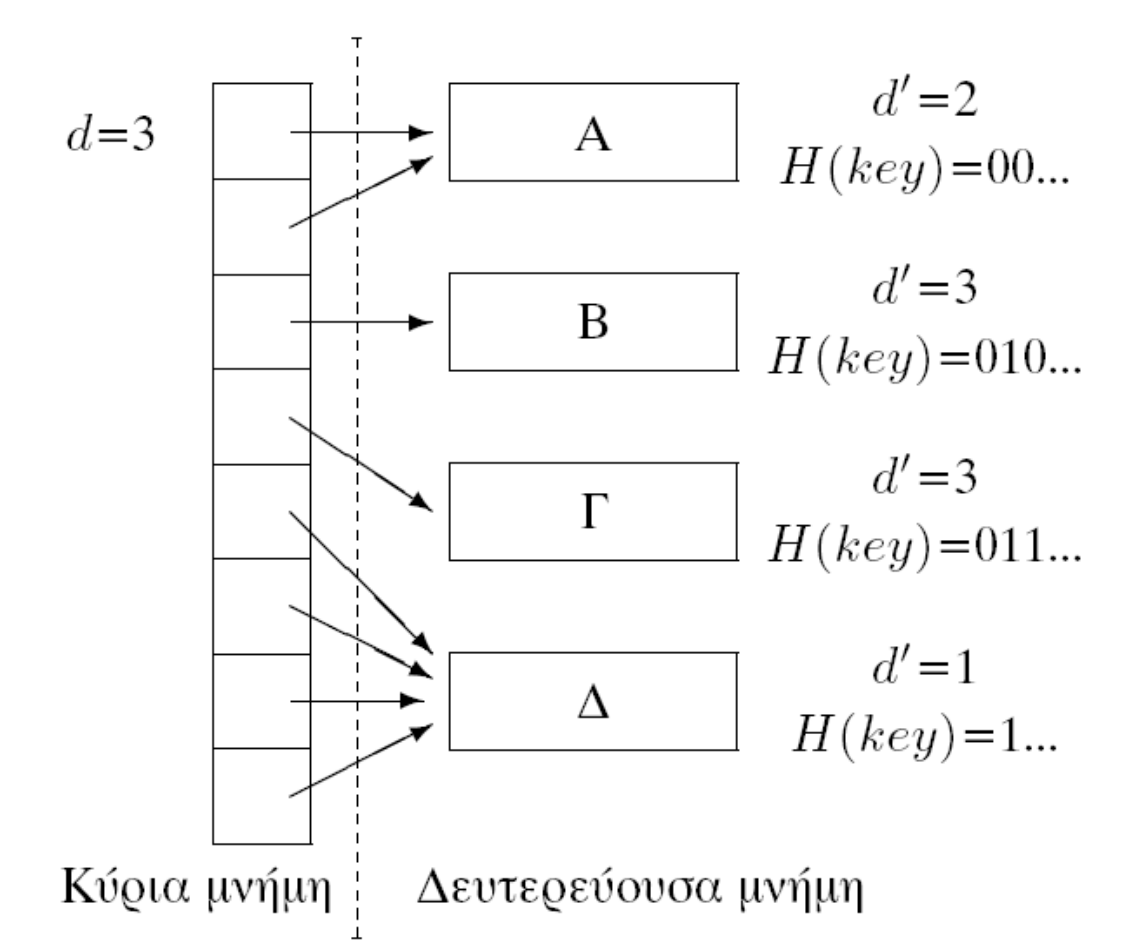
- Τα βήματα που ακολουθούνται στον επεκτατό κατακερματισμό είναι τα εξής:
  1. εφαρμόζεται συνάρτηση κατακερματισμού στο κλειδί (και γίνεται δυαδική αναπαράσταση του αποτελέσματος),
  2. εξάγονται τα πρώτα d bits,
  3. χρησιμοποιείται ο κατάλογος για τον εντοπισμό του κατάλληλου δείκτη,
  4. με βάση τον δείκτη γίνεται προσπέλαση στον κάδο, και
  5. γίνεται αναζήτηση στον κάδο για την εύρεση του κλειδιού.

Επεκτατός κατακερματισμός

- Η περιεκτικότητα ενός κάδου δεν πρέπει να είναι μικρότερη από ένα ποσοστό που ορίζεται από το χρήστη, π.χ 50% και μεγαλύτερη από 100%.
- Η επεξεργασία των εισαγωγών και διαγραφών είναι αρκετά πολύπλοκη διαδικασία.

Επεκτατός κατακερματισμός

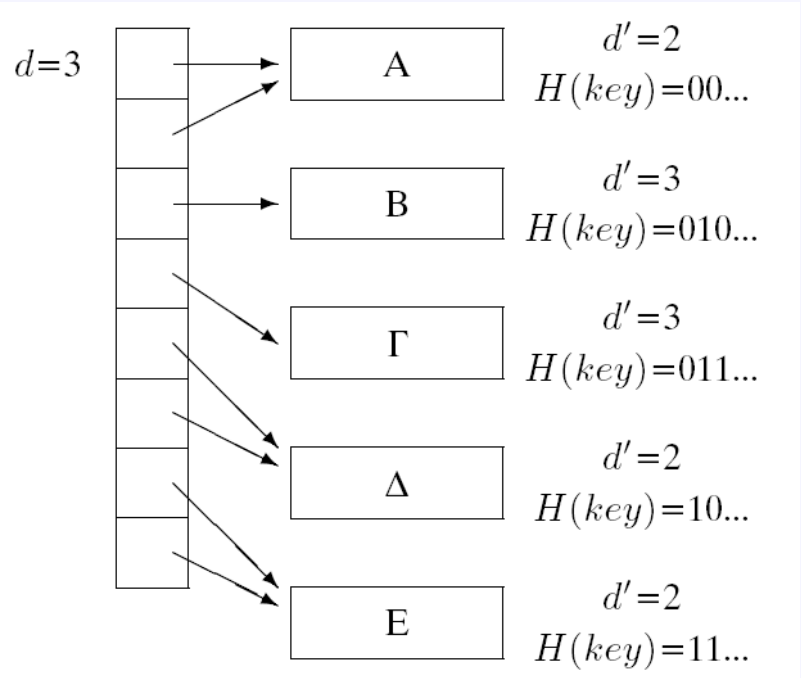
- Έστω ότι στον κάδο Δ πρέπει να εισαχθεί μια εγγραφή.



Επεκτατός κατακερματισμός

Διαδικασία εισαγωγής σε γεμάτο κάδο (>1 δείκτες δείχνουν στο κάδο)

1. Προκαλείται επέκταση του αρχείου
2. Προστίθεται νέος κάδος
3. Οι μισοί δείκτες αλλάζουν και αναφέρονται στο νέο κάδο, ενώ μεταφέρονται και οι αντίστοιχες εγγραφές



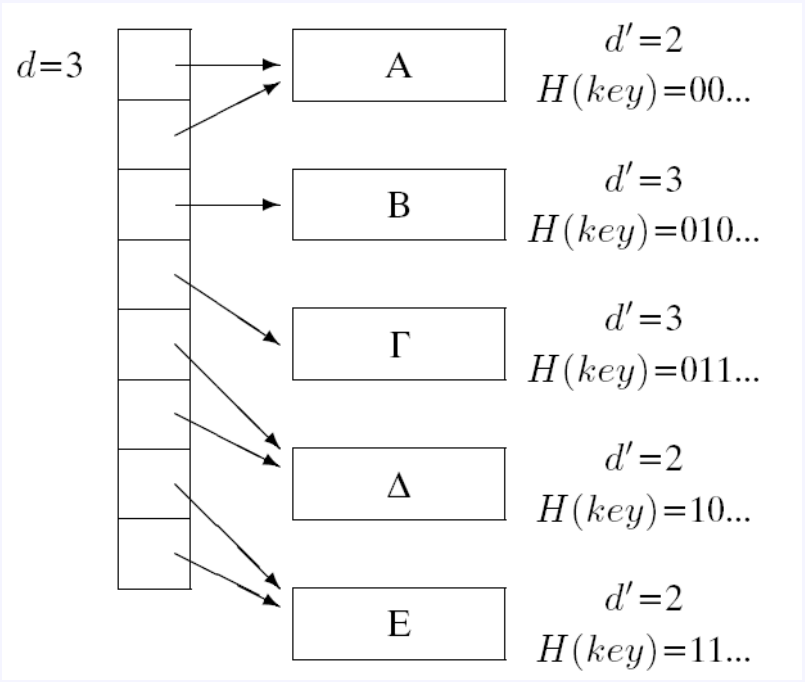
Επεκτατός κατακερματισμός

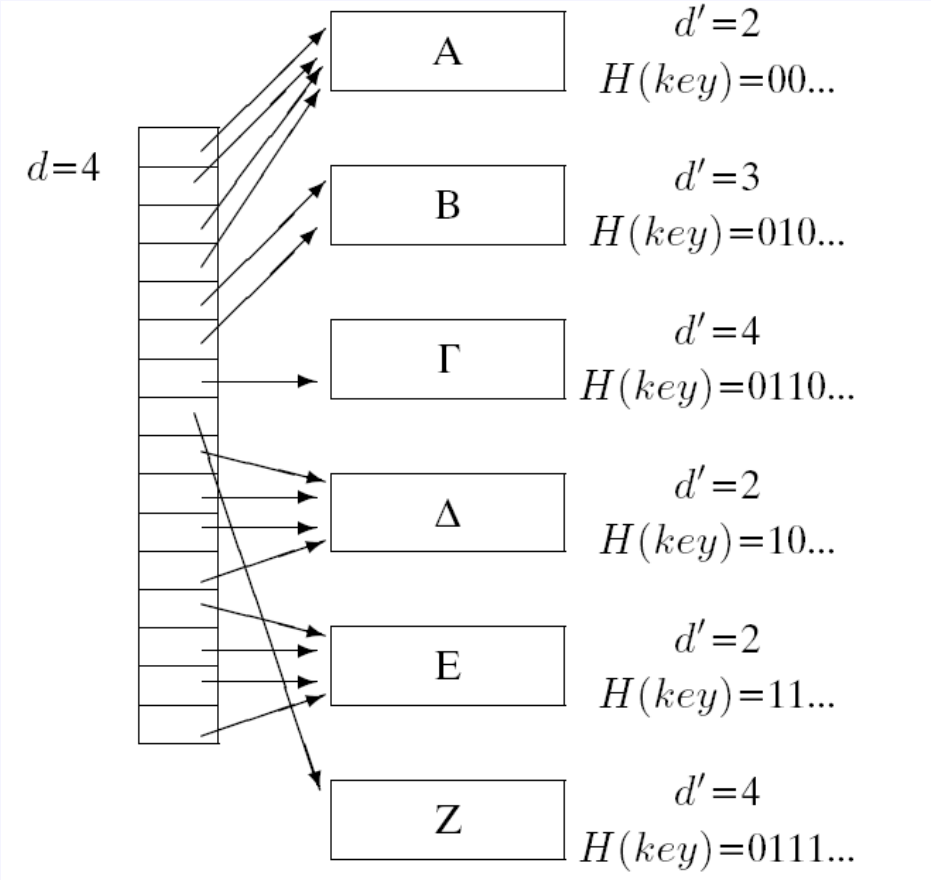
Διαδικασία εισαγωγής σε γεμάτο κάδο (ένας δείκτης δείχνει στο κάδο)

1. Όταν ο κάδος που πρόκειται να διασπασθεί αναφέρεται με ένα δείκτη (**d=d'**), τότε γίνεται **επέκταση καταλόγου**.
2. Ο κατάλογος (2^d) διπλασιάζεται όπου **d=d+1**.
3. Κάθε δείκτης καταλαμβάνει τώρα δύο θέσεις του νέου καταλόγου.
4. Κάθε κάδος έχει δύο ή περισσότερους δείκτες, οπότε μπορεί να ακολουθήσει διάσπαση.

Επεκτατός κατακερματισμός

- Έστω ότι στον κάδο Γ πρέπει να εισαχθεί μια εγγραφή.



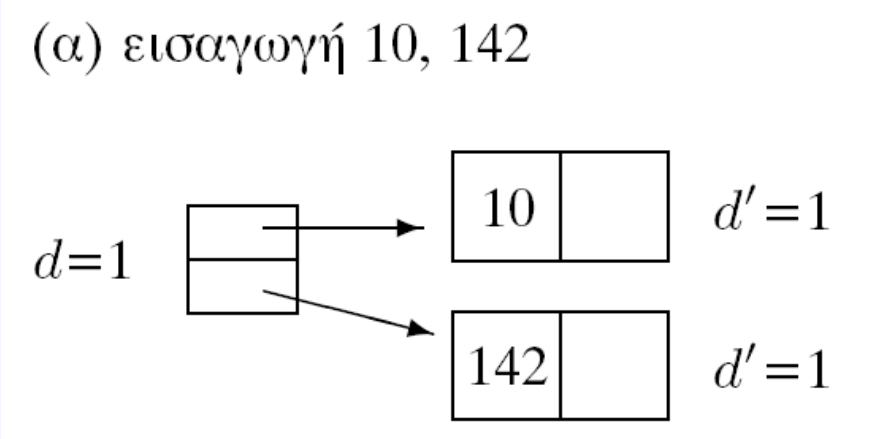


Επεκτατός κατακερματισμός

- Ο κατάλογος διπλασιάζεται κάθε φορά που διασπάται ένας κάδος, για τον οποίο ισχύει η σχέση d=d', ανεξάρτητα από την κατάσταση των άλλων κάδων.
- Αυτή η τεχνική μπορεί να οδηγήσει σε παθολογικές καταστάσεις.

Παράδειγμα εισαγωγής

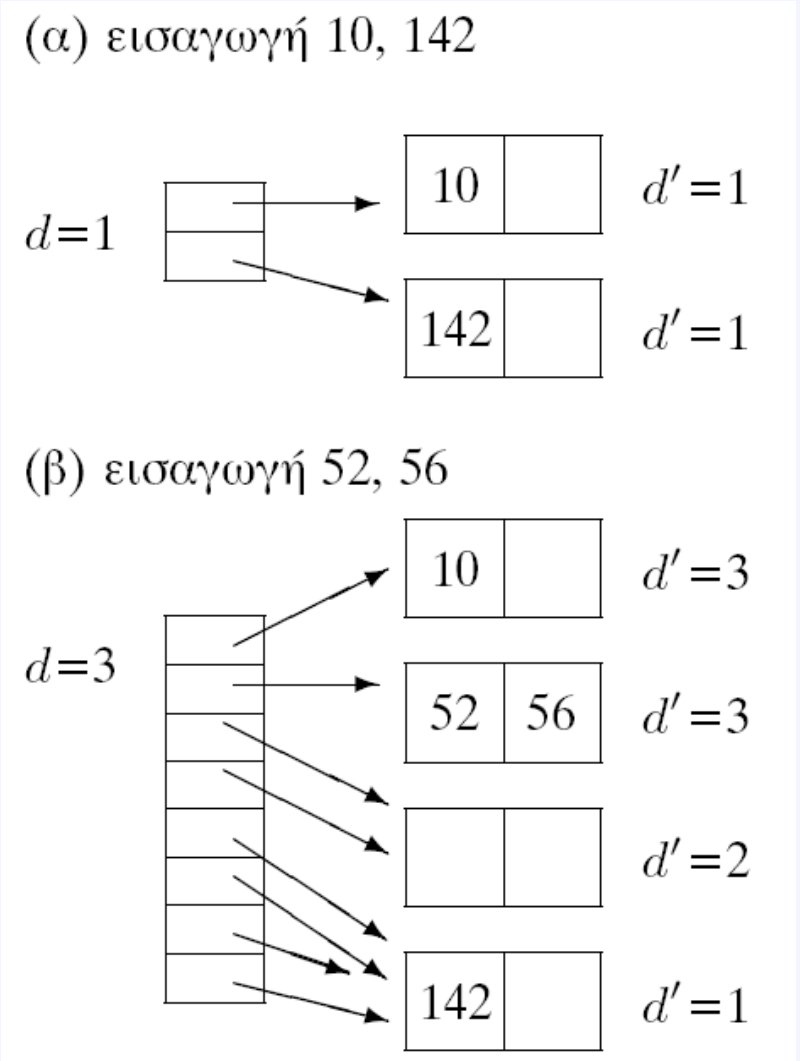
$10 = (00001010)_2$
$142 = (10001110)_2$



Επεκτατός κατακερματισμός

52 =$(00110100)_2$
56 =$(00111000)_2$

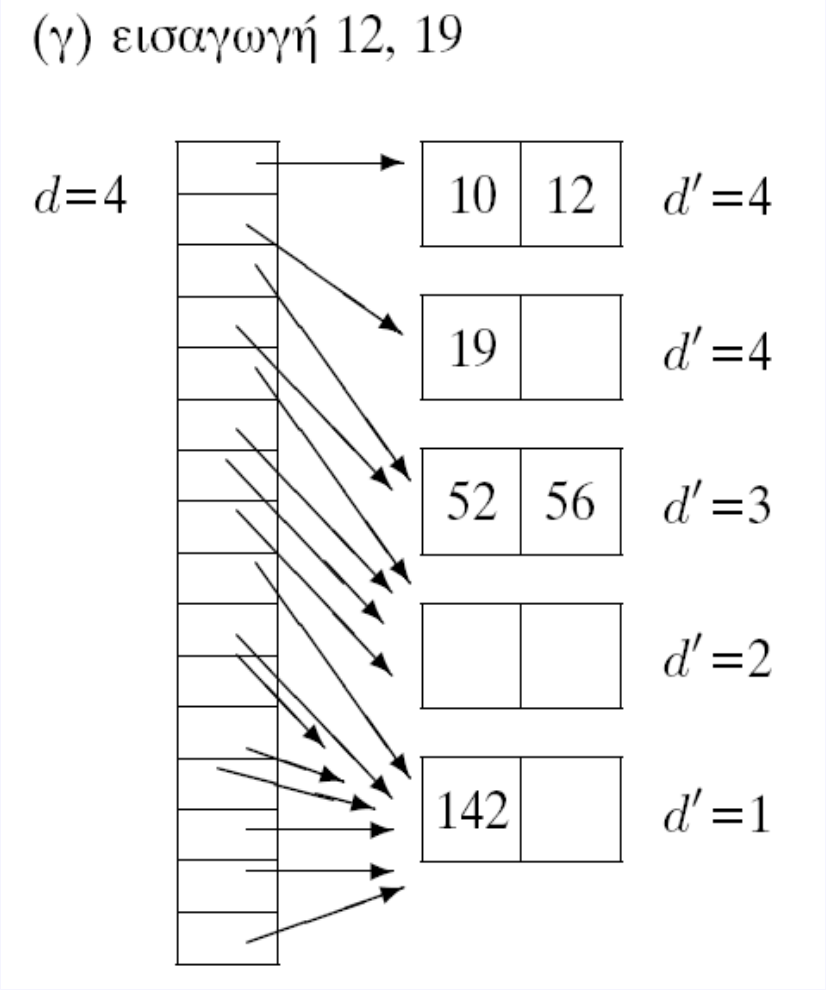
- **Εξαιρετική Περίπτωση:** Επειδή έχουν **ίδια τα πρώτα 2 ψηφία με το 10** γίνεται **διπλασιασμός** του καταλόγου **δύο φορές**.



Επεκτατός κατακερματισμός

$12=(00001100)_2$
$19=(00010011)_2$

- Διπλασιασμός του κατάλόγου γίνεται και με την εισαγωγή των εγγραφών με κλειδιά 12 και 19.
- Ο κατάλογος έχει $2^4=16$ εισόδους.
- Αρκετοί διακριτοί δείκτες δείχνουν είτε σε κενό είτε σε κοινό κάδο.



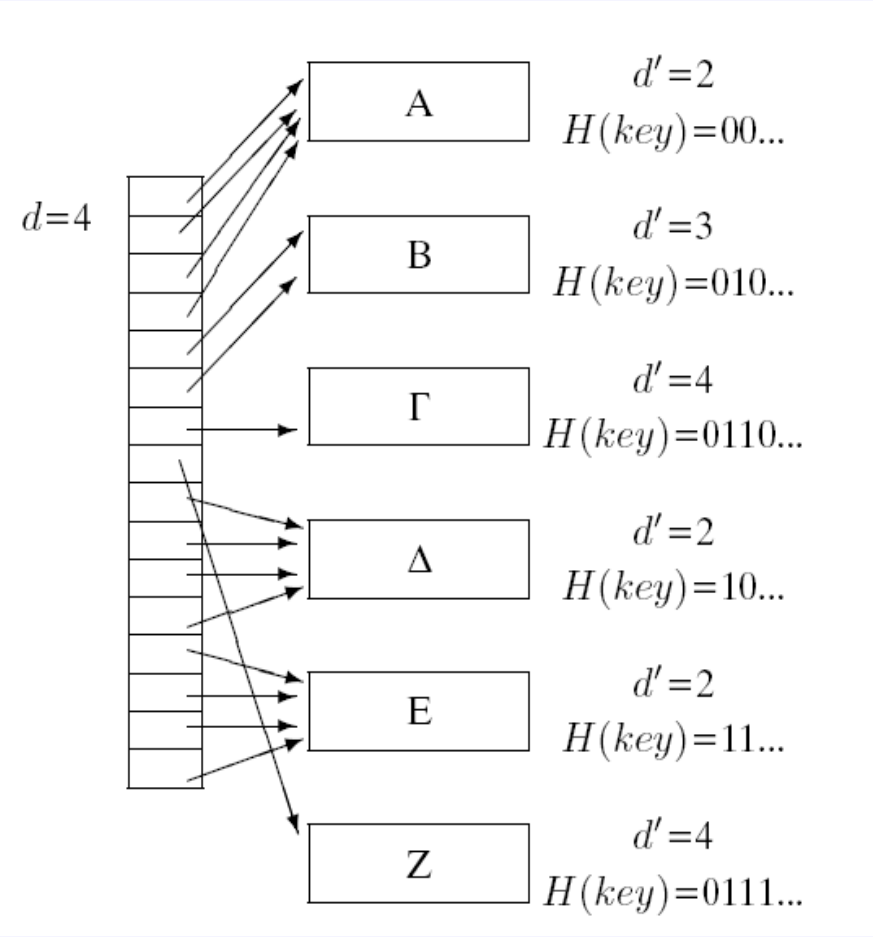
Επεκτατός κατακερματισμός

Διαδικασία διαγραφής - Συσσωμάτωση δύο κάδων

- Μία διαγραφή μπορεί να προκαλέσει συρρίκνωση του αρχείου.
- Η συσσωμάτωση δύο κάδων σε έναν γίνεται υπό τρεις προϋποθέσεις:
  1. **η μέση περιεκτικότητα** των δύο κάδων **δεν ξεπερνά το 50%,**
  2. **οι κάδοι** που πρόκειται να συνδυασθούν χαρακτηρίζονται από την **ίδια τιμή** της παραμέτρου **d'**,
  3. **τα κλειδιά των εγγραφών των δύο κάδων έχουν κοινά τα πρώτα (d'-1) ψηφία** του αποτελέσματος του μετασχηματισμού κατακερματισμού.

Επεκτατός κατακερματισμός

- Η πρώτη προϋπόθεση ισχύει για τους κάδους Α, Β.
- Οι κάδοι όμως δεν είναι συγχωνεύσιμοι γιατί τα κλειδιά των εγγραφών δεν έχουν ίδιο αριθμό κοινών ψηφίων (d').
- Αντίθετα, μπορούν να συγχωνευτούν οι κάδοι Δ, Ε.



Επεκτατός κατακερματισμός

- Όταν όλοι οι δείκτες αποτελούν ζεύγη, τότε ο κατάλογος μπορεί να υποδιπλασιαστεί.

Παράδειγμα: Α,Α,Α,Α,Β,Β,Γ,Γ,Δ,Δ,Δ,Δ,Ε,Ε,Ζ,Ζ

Μπορεί να γίνει υποδιπλασιασμός.

Παράδειγμα: Α,Α,Α,Β,Β,Β,Γ,Γ,Δ,Δ,Δ,Δ,Ε,Ε,Ζ,Ζ

Δεν μπορεί να γίνει υποδιπλασιασμός.

Επεκτατός κατακερματισμός

Παραλλαγή

- Σε μία απλή υλοποίηση του καταλόγου **μπορεί να μην υπάρχουν περιττοί δείκτες**.
- Αν για δεδομένο επίπεδο d του καταλόγου και για κάποιο συνδυασμό d ψηφίων **δεν υπάρχουν αντίστοιχες τιμές κλειδιών, τότε ο αντίστοιχος δείκτης έχει τιμή NULL**.

➕ Έχει ταχύτερη ανεπιτυχή αναζήτηση και απλούστερη διαδικασία διάσπασης κάδων.

➖ Έχει μεγαλύτερες απαιτήσεις χώρου.

Επεκτατός κατακερματισμός

Μειονεκτήματα

- Σε εξαιρετικές περιπτώσεις ο κατάλογος μπορεί να γίνει τόσο μεγάλος, ώστε να μην χωρά στην κύρια μνήμη (ανεξέλεγκτος διπλασιασμός του καταλόγου):
  - οπότε αποθηκεύεται στη δευτερεύουσα μνήμη, και
  - απαιτούνται επιπλέον προσπελάσεις στο δίσκο.
- Δεν προσφέρεται για ερωτήσεις διαστήματος.
- Το κόστος εισαγωγών έχει εξάρσεις που οφείλονται στις πολλές και ταυτόχρονες διασπάσεις των κάδων του αρχείου και του καταλόγου.