Κεφάλαιο 12

ΔΟΜΕΣ ΜΕ ΔΥΑΔΙΚΗ ΑΝΑΠΑΡΑΣΤΑΣΗ

Κεφάλαιο 12

ΔΟΜΕΣ ΜΕ ΔΥΑΔΙΚΗ ΑΝΑΠΑΡΑΣΤΑΣΗ

12.1 Εισαγωγή

Η εξέλιξη των υπολογιστικών συστημάτων βοήθησε ώστε να δημιουργηθούν διεπιφάνειες χρήστη σε υψηλότερο επίπεδο, δηλαδή με λιγότερη σχέση προς το υλικό της μηχανής. Έτσι η επικοινωνία έγινε φιλικότερη και αυξήθηκε η παραγωγικότητα. Ωστόσο, στο κεφάλαιο αυτό σε μία αντίθετη κατεύθυνση από την εξέλιξη αυτή, θα εξετασθούν οργανώσεις αρχείων που έχουν το δυαδικό ψηφίο (bit) ως δομικό στοιχείο για την αναπαράστασή τους. Το κέρδος από μία τέτοια προσέγγιση είναι διπλό: οικονομία στο χώρο και ταχύτητα στην επεξεργασία. Κατά την υλοποίηση των δομών αυτών ο προγραμματιστής των εφαρμογών μπορεί είτε να χρησιμοποιήσει μία γλώσσα χαμηλού επιπέδου για την αποθήκευση και την επεξεργασία των δεδομένων είτε να χρησιμοποιήσει μία γλώσσα υψηλού επιπέδου και να μετατρέψει τα bits σε bytes. Ας σημειωθεί επίσης ότι πολλοί υπολογιστές έχουν ενσωματωμένες ταχύτατες διαδικασίες αναζήτησης σε δυαδικές συμβολοσειρές (bitstring) μεγέθους byte, λέξης κλπ.

Η οργάνωση και επεξεργασία **βάσεων κειμένων** (textual databases) είναι ένα ευρύ αντικείμενο που εξετάζεται κυρίως από τον κλάδο της Ανάκτησης Πληροφοριών. Βέβαια, στον αναγνώστη είναι ήδη γνωστές δύο κατηγορίες μεθόδων:

- οι μέθοδοι πλήρους σάρωσης κειμένου που εξετάσθηκαν στο βιβλίο των Δομών Δεδομένων (αλγόριθμοι Boyer-Moore, Knuth-Morris-Pratt κλπ.), και

- η μέθοδος των αντεστραμμένων καταλόγων που εξετάσθηκε στο προηγούμενο κεφάλαιο.

Η κατηγορία των **αρχείων υπογραφών** (signature files) αποτελεί μία ακόμη οικογένεια μεθόδων για την οργάνωση και την επεξεργασία κειμένων, με ιδιαίτερο γνώρισμα ότι είναι δομές με δυαδική αναπαράσταση. Η μέθοδος αυτή δεν είναι νέα. Χρησιμοποιήθηκε για πρώτη φορά με σκοπό την ανάκτηση δεδομένων το 1949 από τον Mooers και αναφέρεται ήδη από το 1963 σε διδακτικά βιβλία (δες βιβλίο Bourne).

Στη συνέχεια του κεφαλαίου θα εξετασθούν δενδρικές και τυχαίες δομές αρχείων που χρησιμεύουν για την οργάνωση και επεξεργασία δομημένων δεδομένων (όπως οι κλασικές εγγραφές που θεωρήθηκαν μέχρι τώρα) καθώς και μη δομημένων δεδομένων (όπως πχ. το ελεύθερο κείμενο ή πολυμεσικά δεδομένα, όπως ήχος, εικόνα, κινούμενη εικόνα κλπ). Τα θέματα της υλοποίησης των σύνθετων αυτών δομών δεν θα εξετασθούν σε βάθος, αλλά περισσότερη έμφαση θα δοθεί στην παρουσίαση των εννοιών.

12.2 Εξαγωγή υπογραφών

Πριν εξετασθεί οποιαδήποτε δομή καταλόγου που στηρίζεται σε **υπογραφές** (signatures), πρέπει πρώτα να εξετασθεί η έννοια της υπογραφής των δεδομένων και οι τρόποι **εξαγωγής της υπογραφής** (signature extraction) από τα δεδομένα, είτε δομημένα είτε μη δομημένα.

Αρχικά, λοιπόν, ας θεωρήσουμε την περίπτωση του ελεύθερου (μη δομημένου) κειμένου. Η μέθοδος των υπογραφών μπορεί να χρησιμοποιηθεί σε πληθώρα εφαρμογών οργάνωσης και επεξεργασίας βάσεων κειμένων, όπως για παράδειγμα σε εγκυκλοπαίδειες, νομικά κείμενα, γραφεία πατεντών κλπ. Επίσης, μπορεί να χρησιμοποιηθεί και σε πολυμεσικά συστήματα (multimedia systems) εξάγοντας υπογραφές από τα διάφορα πολυμεσικά χαρακτηριστικά. Εν πάσει περιπτώσει, στο περιβάλλον της οργάνωσης και επεξεργασίας κειμένου η μέθοδος εφαρμόζεται ως εξής.

Το πλήρες κείμενο σαρώνεται ώστε να εξαλειφθούν οι κοινές λέξεις (για παράδειγμα, άρθρα, προθέσεις κλπ.) και να απομείνουν οι λεγόμενες `χρήσιμες΄ λέξεις. Οι χρήσιμες λέξεις μετατρέπονται στη ρίζα τους (δηλαδή ενικός αριθμός, ονομαστική πτώση κλπ). Οι λέξεις που χωρούν σε μία φυσική σελίδα του δίσκου (physical block) θεωρείται ότι αποτελούν μία λογική

ομάδα (logical block). Σε κάθε λέξη μίας λογικής ομάδας εφαρμόζεται κάποια τεχνική βασισμένη στον κατακερματισμό, οπότε η λέξη μετατρέπεται σε μία δυαδική συμβολοσειρά σταθερού μήκους, η οποία περιέχει ένα μέγιστο σταθερό αριθμό άσσων. Η συμβολοσειρά αυτή είναι η υπογραφή της συγκεκριμένης λέξης και αποτελεί μία αφαιρετική `περίληψή΄ του.

Μία συνηθισμένη πρακτική για την απεικόνιση λέξεων σε δυαδικές συμβολοσειρές είναι η εξής. Θεωρούμε τα γράμματα της λέξης κατά κυλιόμενες τριάδες. Δηλαδή, από τη λέξη signature λαμβάνονται οι τριάδες sig, ign, gna, nat, atu, tur και ure. Κατόπιν, σε κάθε τριάδα εφαμόζεται μία συνάρτηση κατακερματισμού. Για παράδειγμα, λαμβάνοντας το άθροισμα των κωδικών ASCII των γραμμάτων κάθε τριάδας, για τις προηγούμενες τριάδες προκύπτει το σύνολο των αριθμών (83+73+71), (73+71+78), (71+78+65), (78+65+84), (65+84+85), (84+85+82) και (85+82+69), δηλαδή: 227, 232, 214, 227, 234, 251 και 236. Παρατηρούμε ότι η πρώτη και η τέταρτη τριάδα (δηλαδή, sig και nat), παρ΄ ότι είναι διαφορετικές φθάνουν στο ίδιο αποτέλεσμα. Ωστόσο, μία τέτοια κατάσταση μπορεί να προκύψει με οποιαδήποτε συνάρτηση κατακερματισμού.

Κατόπιν, οι ακέραιοι της προηγούμενης ομάδας εισάγονται σε μία νέα συνάρτηση κατακερματισμού ώστε να προκύψει η υπογραφή της λέξης. Για παράδειγμα, αν το μήκος της υπογραφής είναι 50 bits, τότε θεωρώντας τη συνάρτηση `mod 50΄ καταλήγουμε στο σύνολο των αριθμών: 27, 32, 14, 27, 34, 1, 36. `Ετσι, από τα 50 bits της υπογραφής, τα οποία είναι αριθμημένα από 0 μέχρι 49 και αρχικοποιημένα με 0, θα μετατρέψουμε σε άσσους το 1ο, το 14ο, 27ο, το 32ο, το 34ο και το 36ο. Επομένως, τελικά η υπογραφή της λέξης signature είναι η συμβολοσειρά

$$01000000000000100000000000010000101010000000000000$$

δηλαδή, ο αριθμός των άσσων είναι έξι (και όχι επτά). Ο αριθμός αυτός ονομάζεται **βάρος** (weight) της υπογραφής. Επίσης, είναι προφανές ότι λόγω της φύσης του κατακερματισμού, είναι δυνατόν δύο διαφορετικές λέξεις να έχουν την ίδια υπογραφή, ενώ επίσης δεν είναι δυνατόν από μία υπογραφή να βγει συμπέρασμα για την αρχική λέξη εφαρμόζοντας μία αντίστροφη διαδικασία.

Στη συνέχεια, οι επιμέρους υπογραφές των λέξεων μίας συγκεκριμένης λογικής ομάδας συνδυάζονται με τη λογική πράξη oring, οπότε δημιουργείται μία μόνο υπογραφή σε επίπεδο λογικής ομάδας κειμένου. Η πράξη αυτή

Αρχείο	001	100	001	010
Εγγραφή	000	101	011	000
Πεδίο	001	000	001	110
Υπογραφή	001	101	011	110

Σχήμα 12.1: Κωδικοποίηση με υπέρθεση.

λέγεται επίσης **κωδικοποίηση με υπέρθεση** (superimposed coding). Ένα παράδειγμα της κωδικοποίησης αυτής παρουσιάζεται στο Σχήμα 12.1.

Για να είναι αποτελεσματική η αναζήτηση σε υπογραφές πρέπει οι υπογραφές να έχουν εξαχθεί κατά το βέλτιστο τρόπο. Έχει αποδειχθεί αναλυτικά ότι μία υπογραφή φέρει τη μέγιστη ποσότητα πληροφορίας αν ο αριθμός των άσσων ισούται με τον αριθμό των μηδενικών. Αυτό επιτυγχάνεται αν ισχύει η σχέση:

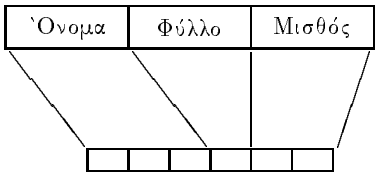
$$F \times \ln 2 \;=\; m \times D$$

όπου F είναι το μήκος της υπογραφής σε bits, m το βάρος σε κάθε υπογραφή λέξης, ενώ D είναι το πλήθος των λέξεων που αποτελούν μιά λογική ομάδα. Επομένως, το βάρος της υπογραφής της κάθε λέξης πρέπει να είναι αρκετά μικρότερο από το βάρος της υπογραφής της λογικής ομάδας. Στην αντίθετη περίπτωση, η υπογραφή που θα προκύψει μετά την υπέρθεση θα είναι γεμάτη από άσσους και δεν θα έχει διακριτική ικανότητα. Στο παράδειγμα του Σχήματος 12.1 οι τιμές των παραμέτρων είναι: μήκος υπογραφής F=12 bits, βάρος υπογραφής λέξεων m=4 και πληθικός αριθμός λογικής ομάδας D=3. Έτσι, μετά την υπέρθεση το βάρος της υπογραφής στο επίπεδο των τριών λέξεων είναι έξι (δηλαδή, δώδεκα δια δύο).

Πλέον, η υπογραφή αυτή αποτελεί τη μονάδα σύγκρισης κατά τις αναζητήσεις. Για τη διαπίστωση αν μία λέξη ανήκει σε μία λογική ομάδα ακολουθείται η εξής διαδικασία. Εξάγεται η υπογραφή της συγκεκριμένης λέξης και συγκρίνεται με την υπογραφή της λογικής ομάδας, bit προς bit. Αν κάποιος άσσος της υπογραφής της λέξης αντιστοιχεί σε μηδέν της υπογραφής της ομάδας, τότε είναι σαφές ότι η λέξη αυτή δεν συμπεριλαμβάνεται μεταξύ των λέξεων της λογικής ομάδας. Όμως αν όλοι οι άσσοι της υπογραφής της λέξης αντιστοιχούν κάποιους από τους άσσους της υπογραφής της λογικής ομάδας, τότε συμπεραίνεται ότι *πιθανώς* η λέξη αυτή να ανήκει στη λογική ομάδα. Στην περίπτωση αυτή είναι αναγκαίο να εξετασθούν όλες οι λέξεις της λογικής ομάδας, ώστε να υπάρξει θετική ή αρνητική απάντηση με

βεβαιότητα. Αν τελικά η συγκεκριμένη λέξη δεν ανήκει στη λογική ομάδα, τότε λέγεται ότι συνέβη μία **λανθασμένη πτώση** (false drop).

Ωστόσο, η μέθοδος των υπογραφών μπορεί να εφαρμοσθεί και σε δομημένα δεδομένα, δηλαδή σε εγγραφές που διακρίνονται σε διάφορα επιμέρους πεδία. Για παράδειγμα, έστω η εγγραφή ενός υπαλλήλου που περιλαμβάνει τα εξής τρία πεδία: `Ονομα, Φύλλο και Μισθός. Με τη βοήθεια μίας συνάρτησης κατακερματισμού η τιμή κάθε πεδίου μίας εγγραφής μετατρέπεται σε μία επιμέρους υπογραφή δεδομένου σταθερού μήκους. Βέβαια, δεν είναι αναγκαίο οι υπογραφές αυτές να έχουν το ίδιο μήκος, ούτε οι συναρτήσεις κατακερματισμού να είναι ίδιες για όλα τα πεδία. `Ετσι, το πεδίο `Ονομα μπορεί να παρασταθεί με τρία bits, το πεδίο Φύλλο μπορεί να παρασταθεί με ένα bit, ενώ το πεδίο Μισθός με δύο bits, εφαρμόζοντας κάθε φορά μία διαφορετική συνάρτηση. Η συνολική υπογραφή της εγγραφής σχηματίζεται από τις τρεις επιμέρους υπογραφές με **παράθεση** (concatenation), οπότε προκύπτει μία υπογραφή μήκους 6 bits, όπως παρουσιάζεται στο Σχήμα 12.2. Από το σημείο αυτό μπορεί να ακολουθηθεί η διαδικασία που περιγράφηκε προηγουμένως για την περίπτωση του ελεύθερου κειμένου. Δηλαδή, οι υπογραφές των εγγραφών που ανήκουν σε μία σελίδα του δίσκου μπορούν να συνδυασθούν με υπέρθεση, ώστε να εξαχθεί η υπογραφή της σελίδας.



Σχήμα 12.2: Εξαγωγή υπογραφής από εγγραφή.

Ο Roberts (1975) χρησιμοποίησε ένα απλό σειριακό αρχείο για την οργάνωση των δεδομένων ενός τηλεφωνικού καταλόγου. Μία χρήσιμη παρατήρηση του ήταν ότι το αρχείο των υπογραφών θα μπορούσε να αποθηκευθεί κατά `φέτες´ (bit-slices) σε ξεχωριστές δομές, δηλαδή πρώτα όλα τα πρώτα bits όλων των υπογραφών, ύστερα τα δεύτερα bits όλων των υπογραφών κοκ. Το αποτέλεσμα αυτής της μεθόδου ήταν πολύ καλή επίδοση κατά την αναζήτηση με αντιστάθμισμα μέτρια επίδοση κατά την ανανέωση. Στο πρόβλημα της φυσικής αποθήκευσης ενός αρχείου υπογραφών έχουν δοθεί αρκετές λύσεις, που όμως δεν θα εξετασθούν στα πλαίσια του βιβλίου αυτού. `Ομως, αναφέρεται ότι οι υπογραφές των λέξεων είναι αραιές, με την έννοια

ότι περιέχουν πολλά μηδενικά, και μπορεί να εφαρμοσθεί κάποια μέθοδος συμπίεσης από αυτές που θα εξετασθούν σε επόμενο κεφάλαιο. Ο αναγνώστης μπορεί να ανατρέξει στις αναφορές για περισσότερα ενδιαφέροντα στοιχεία (Faloutsos 1985).

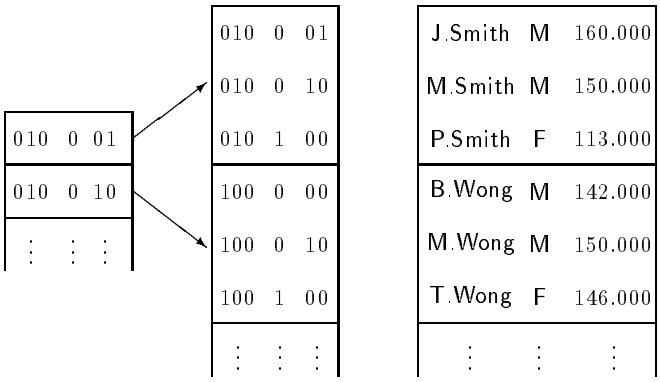
12.3 Δένδρα υπογραφών

Ερωτήσεις μερικής ταύτισης μπορεί να ικανοποιηθούν με αρκετές από τις δομές που εξετάσθηκαν στο προηγούμενο κεφάλαιο. Στη συνέχεια θα θεωρηθεί το παράδειγμα της τελευταίας παραγράφου, δηλαδή του αρχείου με εγγραφές που αποτελούνται από τα τρία γνωστά πεδία, και από τις οποίες με παράθεση και υπέρθεση σχηματίζονται οι αντίστοιχες υπογραφές. Στο αρχείο αυτό θα εξετασθεί η περίπτωση ερωτήσεων μερικής ταύτισης με τη βοήθεια των υπογραφών.

Έστω, λοιπόν, ότι κατ΄ αρχήν οι εγγραφές αποθηκεύονται σε ένα σειριακό αρχείο. Οι υπογραφές τους σχηματίζουν ένα δεύτερο σειριακό αρχείο που είναι μία συμπυκνωμένη έκφραση του κύριου αρχείου και ονομάζεται **αρχείο υπογραφών** (signature file). Επίσης, έστω ότι τίθεται μία απλή ερώτηση ως προς οποιοδήποτε από τα τρία πεδία ή μία ερώτηση μερικής ταύτισης ως προς οποιονδήποτε συνδυασμό των τριών πεδίων λαμβανομένων ανά δύο (ή και μία ερώτηση επακριβούς ταύτισης). Αρχικά οι τιμές των υπ΄ όψη πεδίων που αναζητώνται με την ερώτηση μετατρέπονται στις αντίστοιχες υπογραφές και παρατίθενται για το σχηματισμό της συνολικής υπογραφής της ερώτησης. Αν δεν υπάρχει ενδιαφέρον ως προς κάποιο πεδίο (ή κάποια πεδία) από τα τρία, τότε οι τιμές των bits στις αντίστοιχες θέσεις θεωρούνται αδιάφορες (don't care bits) και συμβολίζονται με ερωτηματικό (?). Έτσι, η αναζήτηση αρχίζει με την προσπέλαση του αρχείου υπογραφών, όπου η προσπέλαση είναι σειριακή. Αν η υπογραφή της αναζητούμενης εγγραφής ταυτισθεί με κάποια από τις υπογραφές του αρχείου, τότε πρέπει να προσπελασθεί το κύριο αρχείο για να διαπιστωθεί αν πραγματικά η αντίστοιχη εγγραφή ικανοποιεί το χρήστη ή αποτελεί μία λανθασμένη πτώση. Η αναζήτηση τελειώνει με την εξάντληση των υπογραφών του αντίστοιχου αρχείου.

Η επίδοση μπορεί να βελτιωθεί αν κατασκευασθεί ένας κατάλογος, όπως συμβαίνει με την κατασκευή ενός καταλόγου στα απλά σειριακά αρχεία. Η δομή αυτή αναλογικά προς τη μέθοδο ISAM ονομάζεται μέθοδος IDAM

(Indexed Descriptor Access Method). Ο κατάλογος στην περίπτωση αυτή σχηματίζεται εξάγοντας από τις υπογραφές ενός σταθερού αριθμού διαδοχικών αριθμών μία νέα υπερ-υπογραφή με τη μέθοδο της κωδικοποίησης με υπέρθεση. Έτσι, από το σύνολο των υπογραφών προκύπτει ένα νέο σύνολο υπερυπογραφών υποπολλαπλάσιου μεγέθους. Στο Σχήμα 12.3 παρουσιάζεται ένα **δένδρο υπογραφών** (signature tree) με δύο επίπεδα, όπου από κάθε τρεις υπογραφές εξάγεται μία υπερ-υπογραφή. Γενικά η διαδικασία αυτή μπορεί να επαναληφθεί για όσα επίπεδα χρειάζεται, ώστε εφαρμόζοντας την κωδικοποίηση με υπέρθεση να προκύψει ένα σύνολο υπερ-...-υπερ-υπογραφών που να χωρά για αποθήκευση στην κύρια μνήμη. Σε πρακτικές υλοποιήσεις το μήκος των υπογραφών μπορεί να είναι από 100 ως 200 bytes, ενώ ο παράγοντας ομαδοποίησης είναι της τάξης του 100. Με άλλα λόγια το μέγεθος του δένδρου των υπογραφών είναι μία επιβάρυνση της τάξης του 10% περίπου σε σχέση με το μεγέθος του κύριου αρχείου.



Σχήμα 12.3: Δένδρο υπογραφών.

Ο τρόπος αναζήτησης στο δένδρο αυτό για ερωτήσεις με βάση δευτερεύον κλειδί αλλά και γενικότερα ερωτήσεις μερικής ταύτισης είναι πλέον προφανής. Ας σημειωθεί επίσης ότι μία ανεπιτυχής αναζήτηση μπορεί να τερματισθεί στα ανώτερα επίπεδα του δένδρου χωρίς να καταστεί αναγκαία η προσπέλαση στο κύριο αρχείο.

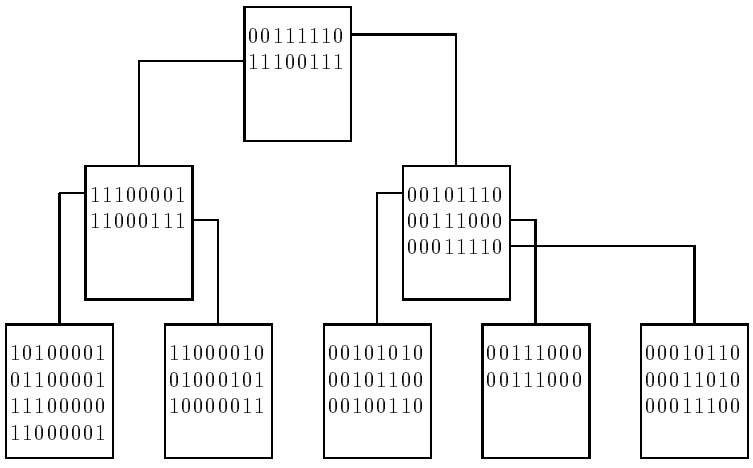
Ωστόσο, η προσέγγιση της δομής IDAM είναι στατική, όπως εξ άλλου στατική είναι και η δομή των αρχείων ISAM. Στη συνέχεια θα εξετασθεί μία ακόμη δενδρική δομή για υπογραφές, η οποία έχει τα χαρακτηριστικά του B-δένδρου, δηλαδή είναι ισοζυγισμένη και διακρίνεται από δυναμικότητα που

εκφράζεται με διασπάσεις κόμβων μετά από εισαγωγή και υπερχείλιση, και επαναεισαγωγές εγγραφών μετά από διαγραφή και υποχείλιση. Η δομή αυτή ονομάζεται S-δένδρο (S-tree) και προτάθηκε από τον Deppisch το 1986.

S-δένδρο τάξης (k,K) είναι το ετερογενές δένδρο με τα ακόλουθα χαρακτηριστικά:

- η ρίζα περιέχει (εκτός αν είναι φύλλο) τουλάχιστο δύο ζεύγη και το μέγιστο K ζεύγη του τύπου (p,s), όπου p είναι ένας δείκτης προς ένα παιδί, s είναι η υπογραφή του συγκεκριμένου κόμβου, ενώ η υπογραφή αυτή παράγεται με υπέρθεση των υπογραφών όλων των παιδιών,
- οι εσωτερικοί κόμβοι (εκτός της ρίζας) περιέχουν το ελάχιστο k ζεύγη και το μέγιστο K ζεύγη του τύπου (p,s), όπου $1 \leq k \leq K/2$,
- ένας εσωτερικός κόμβος με l ζεύγη έχει l παιδιά, και
- τα φύλλα βρίσκονται στο ίδιο επίπεδο και περιέχουν ζεύγη του τύπου (p',s), όπου p' είναι ένας δείκτης προς το αντίστοιχο αντικείμενο στο κυρίως αρχείο και s είναι η υπογραφή του σχετικού αντικειμένου.

Επειδή οι υπογραφές παράγονται με κατακερματισμό, μία υπογραφή μπορεί να είναι αποθηκευμένη περισσότερο από μία φορά μέσα στο S-δένδρο. Επίσης, σημειώνεται ότι δεν υπάχει καμία διάταξη για τα ζεύγη των κόμβων. Ένα παράδειγμα S-δένδρου παρουσιάζεται στο Σχήμα 12.4, όπου οι υπογραφές αποτελούνται από οκτώ bits, ενώ το βάρος είναι τρία bits.



Σχήμα 12.4: Παράδειγμα S-δένδρου (k=2, K=4).

Τα ποσοτικά χαρακτηριστικά του S-δένδρου είναι διαφορετικά από τα αντίστοιχα του B-δένδρου, αλλά εξάγονται με παρόμοιο τρόπο. Είναι προφανές ότι:

$$U_{min} = k/K$$

και όχι 50% όπως στην περίπτωση του B-δένδρου. Βέβαια, αυτό σημαίνει ότι δεν ισχύει ούτε ότι E[U]=69%. Η τιμή τους ύψους, h, περιορίζεται από τη σχέση:

$$h \leq \lceil \log_k n \rceil - 1$$

όπου n είναι ο αριθμός των υπογραφών, ενώ ο ελάχιστος και ο μέγιστος αριθμός κόμβων δίνεται από τις σχέσεις:

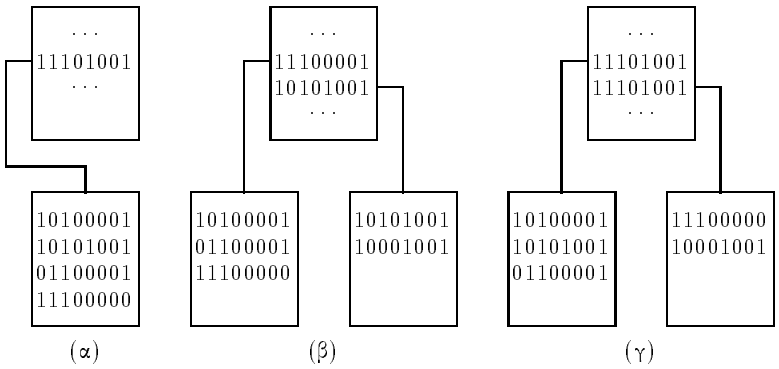
$$nod_{min} = 1 + \sum_{i=0}^{h-2} k^i \qquad\qquad nod_{max} = 1 + 2\frac{k^{h-1} - 1}{k - 1}$$

Η διαδικασία αναζήτησης μίας υπογραφής σε ένα S-δένδρο είναι σχεδόν προφανής για τον αναγνώστη. Έστω ότι η αναζήτηση αφορά στην υπογραφή 00100110. Η διαδικασία αρχίζει από τη ρίζα. Εκ των δύο υπογραφών της ρίζας ταιριάζει η 00111110, γιατί έχει άσσους εκεί όπου έχει άσσους και η αναζητούμενη. Έτσι, η διαδικασία συνεχίζει στο δεξιό κόμβο του δευτέρου επιπέδου, ο οποίος περιέχει τρεις υπογραφές. Εξ αυτών των υπογραφών ταιριάζει η 00101110, για τον ίδιο λόγο όπως προηγουμένως. Με τον τρόπο αυτό η διαδικασία καταλήγει στο αντίστοιχο φύλλο, όπου διαπιστώνεται ότι η αναζητούμενη υπογραφή πράγματι υπάρχει. Η διαδικασία που περιγράφηκε αφορά σε μία ερώτηση επακριβούς ταύτισης και ήταν επιτυχής. Αν η αναζητούμενη υπογραφή ήταν η 00001110, τότε θα επρόκειτο για μία ανεπιτυχή αναζήτηση που θα κατέληγε στο δεξιότερο φύλλο.

Με τη δομή των S-δένδρων μπορούν να απαντηθούν και ερωτήσεις μερικής ταύτισης. Κατά τη διάσχιση του δένδρου σε μία τέτοια περίπτωση, το μονοπάτι από τη ρίζα προς τα φύλλα δεν είναι απαραίτητα μοναδικό, αλλά μπορεί να ακολουθηθούν περισσότερα του ενός μονοπάτια. Για παράδειγμα, έστω ότι αναζητώνται οι υπογραφές ??1???1?. Έτσι, από τη ρίζα θα πρέπει να προσπελασθούν και οι δύο κόμβοι του δευτέρου επιπέδου. Εξετάζοντας το περιεχόμενο των κόμβων αυτών, αρχικά διαπιστώνεται ότι είναι αδύνατο από τον αριστερό κόμβο να βρεθεί κάποια υπογραφή με τις ζητούμενες προδιαγραφές, ενώ από το δεξιό κόμβο υπάρχει μόνο μία υπογραφή με άσσους στην τρίτη, την έκτη και την έβδομη θέση. Επομένως, πρακτικά η διαδικασία συνεχίζεται στο αριστερότερο παιδί του δεξιού κόμβου, όπου

διαπιστώνεται και η τελική απάντηση που αποτελείται από τις υπογραφές 00101010 και 00100110. Γενικώς, όσο λιγότεροι άσσοι προσδιορίζονται σε μία ερώτηση μερικής ταύτισης, τόσο περισσότερα είναι τα μονοπάτια που πρέπει να ακολουθηθούν.

Η διαδικασία της αναζήτησης σε ένα S-δένρο δεν είναι ιδιαίτερα δύσκολη. Περισσότερο δύσκολη είναι η διαδικασία των εισαγωγών, οπότε διασπώνται οι υπερχειλίζοντες κόμβοι. Η δυσκολία έγκειται στο γεγονός ότι αν υπογραφές δεν κατανεμηθούν στους δύο κόμβους με ένα έξυπνο τρόπο, τότε οι υπογραφές των κόμβων αυτών που θα ανέλθουν στο ανώτερο επίπεδο θα έχουν πολλούς άσσους. Έτσι, σε επόμενες αναζητήσεις θα ακολουθούνται πολλά μονοπάτια του δένδρου ακόμη και αν η υπογραφή της ερώτησης δεν περιέχει λίγους άσσους. Στη συνέχεια δίνεται ένα παράδειγμα καλής και κακής διάσπασης. Έστω ότι στον κόμβο του Σχήματος 12.5α, ο οποίος έχει μέγιστη χωρητικότητα τεσσάρων υπογραφών, εισάγεται η υπογραφή 10001001 και πρέπει να γίνει διάσπαση λόγω υπερχείλισης. Στο Σχήμα 12.5β και στο Σχήμα 12.5γ παρουσιάζονται δύο εναλλακτικά σενάρια κατανομής των υπογραφών σε δύο κόμβους. Προφανώς, το σενάριο του Σχήματος 12.5γ δεν πρέπει να προτιμηθεί γιατί οι υπογραφές που ανέρχονται στον πατέρα (α) έχουν βάρος έξι άσσων, και (β) είναι ταυτόσημες. Αντίθετα, οι υπογραφές που ανέρχονται στον πατέρα των δύο κόμβων του Σχήματος 12.5β έχουν βάρος τεσσάρων άσσων.



Σχήμα 12.5: Παράδειγμα διάσπασης σε S-δένδρο.

Το πρόβλημα, λοιπόν, της εισαγωγής αντιμετωπίζεται ως εξής. Δεδομένης της υπογραφής προς εισαγωγή, γίνεται μία διάσχιση από τη ρίζα προς τα φύλλα έτσι ώστε σε κάθε επίπεδο να επιλέγεται ως επόμενος κόμβος, ο

κόμβος του οποίου η υπογραφή αν υπερτεθεί με την εισαγόμενη υπογραφή να προκύψει η μικρότερη αύξηση βάρους. Προφανώς, η στρατηγική αυτή αποσκοπεί στην ελαχιστοποίηση των διαφορετικών μονοπατιών που θα πρέπει να διασχισθούν σε μελλοντικές αναζητήσεις. Τελικώς, λοιπόν, προσπελάζεται ένα φύλλο. Αν το φύλλο έχει διαθέσιμο χώρο, τότε η υπογραφή εισάγεται στο φύλλο αυτό και ταυτόχρονα ελέγχεται αν πρέπει να ενημερωθεί η υπογραφή του πατέρα, καθώς και οι υπογραφές των κόμβων των ανωτέρων επιπέδων προς τη ρίζα του δένδρου. Αν το φύλλο όπου καταλήγει η εισαγωγή είναι πλήρες, τότε γίνεται διάσπαση σύμφωνα με την εξής ευριστική τεχνική. Κατ΄ αρχήν επιλέγονται δύο υπογραφές s_1 και s_2, που ονομάζονται **σπόροι** (seed) των δύο σελίδων. Ως σπόρος s_1 επιλέγεται η υπογραφή με το μεγαλύτερο βάρος, ενώ ως σπόρος s_2 επιλέγεται η υπογραφή με το μεγαλύτερο αριθμό άσσων στις θέσεις όπου ο σπόρος s_2 έχει μηδέν. Έτσι, δεδομένης της υπογραφής s_1, η υπογραφή s_2 είναι εκείνη που διαφέρει περισσότερο προς την s_1. Ως μέτρο **διαφοράς** (dissimilarity) επιλέγεται η απόσταση Hamming, δηλαδή ο αριθμός των θέσεων όπου η μία υπογραφή έχει άσσο και η άλλη έχει μηδέν. Αν με $\delta(s_1, s_2)$ συμβολίζεται η απόσταση Hamming δύο υπογραφών s_1 και s_2, ενώ με γ συμβολίζεται το βάρος μίας υπογραφής, τότε ισχύει η σχέση:

$$\delta(s_1, s_2) \;=\; \gamma(s_1 \vee s_2) - \gamma(s_1 \wedge s_2)$$

Κατόπιν, οι υπόλοιπες υπογραφές λαμβάνονται μία-μία με τυχαίο τρόπο και συγκρίνονται με τους δύο σπόρους. Έτσι, κάθε υπογραφή καταλήγει στη σελίδα, όπου με το σπόρο της δίνει ελάχιστη αύξηση βάρους σε περίπτωση υπέρθεσης. Όμως υπάρχει περίπτωση, καθώς γίνεται η ανάθεση των υπογραφών στις δύο σελίδες, κάποια σελίδα να γεμίσει από K υπογραφές. Τότε όλες οι υπόλοιπες αναθέτονται στην άλλη σελίδα χωρίς περαιτέρω εξέταση. Βέβαια, στη συνέχεια πρέπει να εξαχθούν οι υπογραφές των δύο φύλλων και να ανέλθουν στο ανώτερο επίπεδο. Αυτό σημαίνει ότι αν ο πατέρας είναι πλήρης, τότε πρέπει να γίνει νέα διάσπαση σε ανώτερο επίπεδο κοκ. μέχρι πιθανώς να διασπασθεί η ρίζα και να αυξηθεί το ύψος του δένδρου.

Σημειώνεται ότι αυτή η μέθοδος διάσπασης σελίδων δεν είναι η βέλτιστη από την άποψη της δημιουργίας σελίδων (α) με περίπου τον ίδιο αριθμό υπογραφών, οπότε θα αναβάλονταν χρονικά η μελλοντική διάσπαση των ιδίων, και (β) με το χωρισμό τους σε δύο υποσύνολα έτσι ώστε οι υπογραφές των δύο υποσυνόλων να έχουν το ελάχιστο βάρος. Το δεύτερο χαρακτηριστικό θα μπορούσε να ικανοποιηθεί με έναν αλγόριθμο εκθετικής πολυπλοκότητας, ενώ ο αλγόριθμος που περιγράφηκε προηγουμένως είναι γραμμικός.

Οι διαγραφές εκτελούνται επίσης με δυναμικό τρόπο. Δηλαδή, σε περίπτωση διαγραφής είναι δυνατόν να απαιτείται ενημέρωση της υπογραφής του ανωτέρου ή των ανωτέρων επιπέδων μέχρι το επίπεδο της ρίζας. Πλέον σύνθετη είναι η περίπτωση όπου ένας κόμβος μείνει με k–1 ζεύγη. Στην περίπτωση αυτή ο κόμβος αποδίδεται στο σύστημα, ενημερώνονται οι υπογραφές του πατέρα (και ίσως αναδρομικά οι κόμβοι στο μονοπάτι προς τη ρίζα), ενώ οι k–1 υπογραφές επανα-εισάγονται ώστε να τοποθετηθούν σε νέα φύλλα.

Τα στατικά (IDAM) και τα δυναμικά (S-trees) δένδρα υπογραφών σε σχέση με τα αντεστραμμένα αρχεία και τις πολλαπλές λίστες έχουν συγκεκριμένα πλεονεκτήματα:

- στο δένδρο υπογραφών το κόστος προσπέλασης παραμένει σχεδόν σταθερό ανεξάρτητα από τον αριθμό των πεδίων που καθορίζονται από την ερώτηση μερικής ταύτισης, σε αντίθεση με τις άλλες μεθόδους, όπου αυξάνοντας τον αριθμό των πεδίων αυξάνει και το κόστος απάντησης,

- η μέθοδος των αντεστραμμένων αρχείων και των πολλαπλών λιστών ή υλοποιείται ή δεν υλοποιείται για κάποιο συγκεκριμένο πεδίο. Στα δένδρα υπογραφών αν κάποιο πεδίο είναι πιο σημαντικό από τα υπόλοιπα και αναζητάται περισσότερο συχνά, τότε το μήκος της υπογραφής για το πεδίο αυτό ρυθμίζεται ώστε να είναι μεγαλύτερο και να υπάρχει μεγαλύτερη αξιοπιστία (δηλαδή, λιγότερες λανθασμένες πτώσεις), και τέλος

- ο επιπρόσθετος χώρος για τα δένδρα υπογραφών ποικίλει από 5% ως 40%, ενώ στις άλλες μεθόδους μπορεί να φθάσει το 100% του όγκου του κύριου αρχείου. Βέβαια, υπάρχει και το κόστος ενημέρωσης, που στα δένδρα υπογραφών είναι μεγαλύτερο απ΄ ότι στις άλλες οργανώσεις αν η ενημερώνονται λίγα πεδία (για παράδειγμα ένα ή δύο πεδία), αλλά είναι μικρότερο στην αντίθετη περίπτωση.

12.4 Ανάκτηση με υπογραφές σελίδων

Τα δένδρα υπογραφών της προηγούμενης μεθόδου λειτουργούν σαν ένας πολυεπίπεδος μηχανισμός φίλτρου, ώστε να μειωθεί ο όγκος των υπογραφών που πρέπει να εξετασθούν. Όμως είναι δυνατό χρησιμοποιώντας ένα διαφορετικό τρόπο απεικόνισης στη δευτερεύουσα μνήμη, οι εγγραφές να μην

αποθηκεύονται τυχαία στις σελίδες του αρχείου, αλλά οι εγγραφές με ίδιες τιμές στα διάφορα χαρακτηριστικά να κατευθύνονται στην ίδια σελίδα. Έτσι, σε μία ερώτηση μερικής ταύτισης οι εγγραφές θα είναι συγκεντρωμένες (clustered) και θα προσπελάζονται μόνο οι σχετικές σελίδες αντί να γίνεται σάρωση ολόκληρου του αρχείου. Η **ανάκτηση μερικής ταύτισης με υπογραφές σελίδων** (partial match retrieval with page signatures), που προτάθηκε από τον Ramamohanarao (1983), εφαρμόζει μία τέτοια τεχνική.

Η μέθοδος βασίζεται σε δύο μηχανισμούς φίλτρων προσπέλασης που βοηθούν, ώστε να γίνει ταχύτερα η αναζήτηση επειδή μειώνεται ο αντίστοιχος χώρος αναζήτησης (φιλοσοφία 'διαίρει και βασίλευε'). Κατ' αρχήν με βάση τις εγγραφές κάθε σελίδας κατασκευάζεται ένα μονοεπίπεδο (σειρακό) αρχείο υπογραφών. Σε δεύτερο στάδιο, από κάθε εγγραφή εξάγεται μία συμβολοσειρά πολύ περιορισμένου μήκους σε σχέση με την προηγούμενη υπογραφή, η οποία δίνει και τη διεύθυνση της σελίδας του αρχείου όπου θα γίνει η αποθήκευση. Η δυαδική αυτή συμβολοσειρά εξάγεται χρησιμοποιώντας νέες συναρτήσεις κατακερματισμού με μικρότερο μήκος συμβολοσειράς για κάθε χαρακτηριστικό. Στη συνέχεια αυτές οι επιμέρους συμβολοσειρές των χαρακτηριστικών παρατίθενται για να σχηματισθεί κατά τα γνωστά μία νέα συμπυκνωμένη υπογραφή της εγγραφής.

Αρχείο υπογραφών	Βασικό αρχείο	Υπογραφή σελίδας	Αριθμός σελίδας
010 0 01	J.Smith M 146.000	000	0
		001	1
010 0 10	M.Smith M 150.000	010	2
010 1 00	P.Smith F 113.000	011	3
100 0 00	B.Wong M 142.000	100	4
100 0 10	M.Wong M 150.000	101	5
		110	6
100 1 00	T.Wong F 146.000	111	7
⋮ ⋮ ⋮	⋮ ⋮ ⋮	⋮	⋮

Σχήμα 12.6: Υπογραφές και διευθύνσεις σελίδων.

Στη συνέχεια θα εξετασθεί και πάλι το προηγούμενο παράδειγμα της εγγραφής που περιλαμβάνει τα πεδία: `Ονομα, Φύλλο και Μισθός. Και πάλι με τη βοήθεια μίας συνάρτησης κατακερματισμού η τιμή κάθε πεδίου μίας εγγραφής μετατρέπεται σε μία δυαδική συμβολοσειρά σταθερού μήκους. `Εστω, λοιπόν, ότι το κάθε πεδίο παρίσταται με ένα bit, συνεπώς η περιγραφή της σελίδας έχει μήκος τρία bits. Οι έξι γνωστές εγγραφές του Σχήματος 12.3 τοποθετούνται σε οκτώ σελίδες με χωρητικότητα μίας εγγραφής όπως παρουσιάζεται στο Σχήμα 12.6.

Ο τρόπος αναζήτησης στη δομή αυτή για ερωτήσεις με βάση δευτερεύον κλειδί, αλλά και ερωτήσεις μερικής ταύτισης είναι παρόμοια με την προηγούμενη μέθοδο. Πρώτα, λοιπόν, εξάγονται οι δύο συμβολοσειρές σύμφωνα με τα δύο σύνολα των συναρτήσεων κατακερματισμού και διαπιστώνεται σε ποιές πιθανές σελίδες του αρχείου μπορεί να είναι αποθηκευμένη η αναζητούμενη εγγραφή. Στη συνέχεια προσπελάζονται οι κατάλληλες σελίδες από το αρχείο υπογραφών και ελέγχονται οι υπογραφές των αποθηκευμένων εγγραφών του κυρίου αρχείου σε σχέση με την υπογραφή της αναζητούμενης εγγραφής. Σε περίπτωση μη ταύτισης, γίνεται αντιληπτό ότι πρόκειται για ανεπιτυχή αναζήτηση και η διαδικασία τερματίζει. Σε περίπτωση ταύτισης, η διαδικασία συνεχίζεται με προσπέλαση στο κύριο αρχείο για τον τελικό έλεγχο. `Εστω, λοιπόν για παράδειγμα, ότι δίνεται μία ερώτηση με βάση το πεδίο Φύλλο που πρέπει να έχει τιμή `Μ΄. `Ετσι, εξάγεται ότι οι διεύθυνσεις των σελίδων είναι ?0?, οπότε τελικά προσπελάζονται οι σελίδες 0, 1, 4 και 5. Είναι προφανές ότι όσα περισσότερα χαρακτηριστικά καθορίζονται σε μία ερώτηση μερικής ταύτισης, τόσο ταχύτερα γίνεται ο εντοπισμός των σχετικών σελίδων. `Εστω, τώρα, ότι τίθεται μία ερώτηση μερικής ταύτισης, όπου πρέπει να ισχύει: `Φύλλο=Γ΄ και `Μισθός>145.000΄. Στην περίπτωση αυτή εξάγεται ότι οι διευθύνσεις των σελίδων είναι ?11, οπότε προσπελάζονται οι σελίδες 3 και 7. Η συνέχεια της επεξεργασίας είναι πλέον ευνόητη.

Η μέθοδος αυτή παρουσιάζει σημαντικά πλεονεκτήματα σε σχέση με τη μέθοδο του δένδρου υπογραφών:

- η δομή με τις υπογραφές των σελίδων δεν είναι απαραίτητο να βρίσκεται μαζί με το κυρίως αρχείο, οπότε αν κατά την επεξεργασία είναι αποθηκευμένο στην κύρια μνήμη, τότε δεν υπάρχει επιβάρυνση για είσοδο/έξοδο των δεδομένων,

- όταν μία εγγραφή εισάγεται, διαγράφεται ή ανανεώνεται αλλάζει μόνο η υπογραφή της σελίδας, χωρίς να προκαλούνται αλυσιδωτές αλλαγές στις υπογραφές των ανώτερων επιπέδων του δένδρου,

- αν και αρχικά σχεδιάσθηκε για στατικά δεδομένα και για ερωτήσεις ανάκτησης με δευτερεύον κλειδί, η ιδέα της υλοποίησης ενός αρχείου περιγραφών μπορεί να συνδυασθεί με τα δυναμικά τυχαία αρχεία, οπότε μπορούν να εξυπηρετηθούν και ερωτήσεις με βάση το πρωτεύον κλειδί.

12.5 Κατακερματισμός με υπογραφές

Από τον Larson (1984) προτάθηκε ένα στατικό αρχείο **κατακερματισμού με υπογραφές** (signature hashing), που επιτυγχάνει την επιτυχή και ανεπιτυχή αναζήτηση με μία και μόνο μία προσπέλαση στο δίσκο. Είναι αντιληπτό ότι η μέθοδος αυτή είναι πολύ ενδιαφέρουσα γιατί καμία δομή απ' όσες εξετάσθηκαν μέχρι τώρα δεν μπορεί να εγγυηθεί την επίδοση αυτή. Για παράδειγμα, πολλές από τις μεθόδους των δυναμικών τυχαίων αρχείων διακρίνονται για αυτήν την επίδοση με την προϋπόθεση ότι ο κατάλογος είναι μικρός και χωρά στην κύρια μνήμη. Βέβαια, στην προκειμένη περίπτωση τίποτε δεν είναι δωρεάν. Τα μειονεκτήματα της μεθόδους είναι:

- η πολυπλοκότητα του λογισμικού της,
- η εισαγωγή μπορεί να κοστίσει αρκετά περισσότερο από μία προσπέλαση στο δίσκο, και
- δεσμεύεται χώρος στην κύρια μνήμη για την αποθήκευση του αρχείου των υπογραφών που έχει το ίδιο μήκος (δηλαδή, αριθμός διευθύνσεων) με το κύριο αρχείο.

Για την κατανόηση της μεθόδου θα εξετασθεί αμέσως ένα παράδειγμα που διασαφηνίζει τις διαδικασίες εισαγωγής και αναζήτησης. Για την ευκολία του παραδείγματος υποτίθεται ότι το κύριο αρχείο αποτελείται από 11 σελίδες με χωρητικότητα μία μόνο εγγραφή (b=11). Το ίδιο μήκος έχει και το αρχείο των υπογραφών που ονομάζονται **διαχωριστές** (separators). Υποτίθεται επίσης ότι το μέγεθος των υπογραφών είναι τέσσερα bits, ενώ αρχικά σε κάθε θέση του αρχείου υπογραφών αποθηκεύεται η τιμή 1111. Κατά την εισαγωγή χρησιμοποιείται ως συνάρτηση κατακερματισμού η μέθοδος της διαίρεσης:

$$f(key) \;=\; key \bmod b$$

ενώ σε περίπτωση σύγκρουσης ως δεύτερη συνάρτηση χρησιμοποιείται η σχέση:

$$i(key) \ = \ \lfloor \frac{key}{b} \rfloor \ \mathsf{mod}\ b$$

ώστε να προκύψει η απόσταση της νέας θέσης του αρχείου που θα πρέπει να εξετασθεί.

Έστω ότι στο αρχείο πρόκειται να εισαχθούν εγγραφές με κλειδιά 52, 19, 71, 56, 68, 5 και 12. Σύμφωνα με τη μέθοδο απαιτείται για κάθε κλειδί να υπολογισθεί μία αντίστοιχη ακολουθία υπογραφών. Μάλιστα κάθε υπογραφή της ακολουθίας αντιστοιχεί σε κάθε διαδοχική εξέταση που μπορεί να γίνει για ένα κλειδί εξαιτίας των συγκρούσεων. Δηλαδή, η μέθοδος αυτή μοιάζει με τη μέθοδο της ανοικτής διεύθυνσης γιατί δεν χρησιμοποιεί δείκτες ή ψευδοδείκτες. Οι υπογραφές συνήθως εξάγονται από το κλειδί με τη βοήθεια μίας ψευδοτυχαίας γεννήτριας αριθμών που είναι μοναδική για κάθε κλειδί. Ωστόσο, για την ευκολία του παραδείγματος στη συνέχεια θα χρησιμοποιηθεί και πάλι μία συνάρτηση κατακερματισμού, όπως για παράδειγμα η σχέση:

$$s_1(key) \ = \ key \ \mathsf{mod}\ 15$$

όπου το 15 χρησιμοποιείται ως διαιρέτης γιατί το μήκος των υπογραφών είναι τέσσερα bits, ενώ γενικά για κάθε επόμενη προσπάθεια εξέτασης θα χρησιμοποιηθεί η σχέση:

$$s_i(key) \ = \ ((s_{i-1}(key) + 1) \times key) \ \mathsf{mod}\ 15$$

Έτσι, στον Πίνακα 12.1 φαίνεται η ακολουθία των υπογραφών των προη-

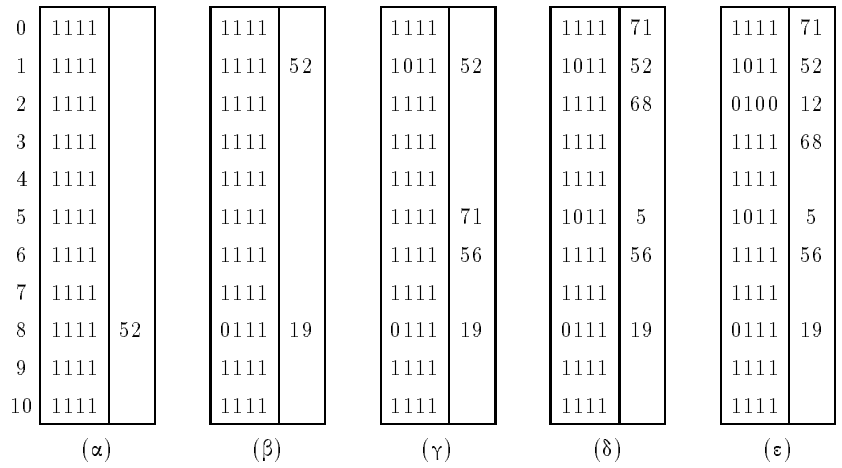
Κλειδί	Εξέταση			
	1	2	3	4
52	7 (0111)	11 (1001)	9 (1001)	10 (1010)
19	4 (0100)	5 (0101)	9 (1001)	10 (1010)
71	11 (1011)	12 (1100)	8 (1000)	9 (1001)
56	11 (1011)	12 (1100)	8 (1000)	9 (1001)
68	8 (1000)	12 (1100)	14 (1110)	0 (0000)
5	10 (1010)	10 (1010)	10 (1010)	10 (1010)
12	12 (1100)	6 (0110)	9 (1001)	0 (0000)

Πίνακας 12.1: Ακολουθία υπογραφών κλειδιών.

γούμενων κλειδιών. Οι υπογραφές αυτές ονομάζονται **υπογραφές θέσης-κλειδιού** (position-key signatures). Είναι φανερό ότι αν προκύψει μηδέν, τότε στη συνέχεια προκύπτει και πάλι η ίδια ακολουθία τιμών. Ο λόγος που χρησιμοποιείται μία ολόκληρη ακολουθία τιμών υπογραφών είναι ο διαφορισμός δύο συνωνύμων σε μία επόμενη εξέταση.

Όταν πρόκειται να εισαχθεί μία εγγραφή, τότε κατ΄ αρχήν εξάγεται η υπογραφή της και με τη βοήθεια της συνάρτησης κατακερματισμού εντοπίζεται η κατάλληλη θέση του αρχείου υπογραφών. Αν η υπογραφή της εγγραφής είναι μεγαλύτερη ή ίση προς την αντίστοιχη υπογραφή του αρχείου, τότε με τη βοήθεια της δεύτερης συνάρτησης κατακερματισμού εντοπίζεται η νέα θέση του αρχείου. Έτσι, τη φορά αυτή η δεύτερη υπογραφή του κλειδιού από τη γνωστή ακολουθία υπογραφών συγκρίνεται με την αντίστοιχη υπογραφή της νέας θέσης του αρχείου. Η διαδικασία αυτή συνεχίζεται μέχρις ότου η υπογραφή του κλειδιού είναι μικρότερη από την αντίστοιχη του αρχείου, οπότε προσπελάζεται το κύριο αρχείο. Αν η θέση του αρχείου αυτού είναι κενή, τότε η εγγραφή αποθηκεύεται και η διαδικασία τελειώνει. Αν κάποια άλλη εγγραφή είναι αποθηκευμένη στη θέση αυτή, τότε τελικά τη θέση καταλαμβάνει εκείνη η εγγραφή που έχει τη μικρότερη υπογραφή, ενώ η αντίστοιχη θέση του αρχείου υπογραφών ενημερώνεται με την τιμή της μεγαλύτερης από τις δύο υπογραφές των υπ΄ όψη εγγραφών. Έτσι, η διαδικασία συνεχίζεται για την εισαγωγή της άλλης εγγραφής σε μία νέα θέση.

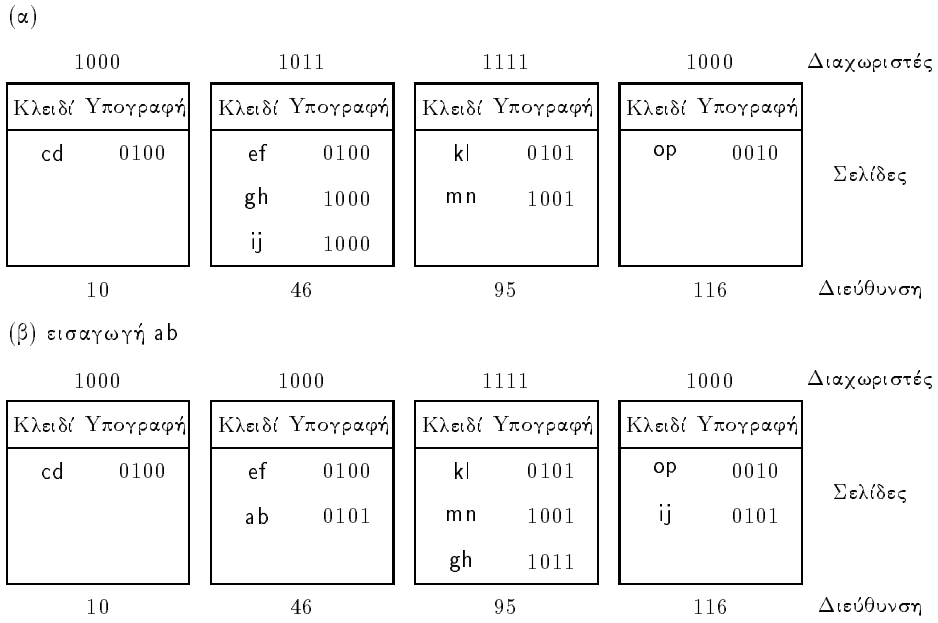
Η διαδικασία που αναπτύχθηκε θεωρητικά φαίνεται καλύτερα στο Σχήμα 12.7 που δείχνει την εξέλιξη της δομής με τις διαδοχικές εισαγωγές των εγγραφών με τα προηγούμενα κλειδιά. Η εγγραφή 52 κατευθύνεται στη θέση 8. Η υπογραφή της 0111 είναι μικρότερη από την τιμή 1111 που είναι αποθηκευμένη στην αντίστοιχη θέση του αρχείο υπογραφών. Άρα γίνεται προσπέλαση στη θέση 8 του αρχείου και η εγγραφή αποθηκεύεται γιατί η θέση είναι κενή. Το αποτέλεσμα παρουσιάζεται στο Σχήμα 12.7α. Στη συνέχεια εισάγεται η εγγραφή 19 που επίσης κατευθύνεται στη θέση 8. Η υπογραφή της θέσης είναι 1111, άρα προσπελάζεται η θέση 8 και διαπιστώνεται ότι είναι κατειλημμένη. Μία από τις δύο εγγραφές πρέπει να μείνει στη θέση και η άλλη να φύγει. Μένει η εγγραφή 19 γιατί έχει μικρότερη υπογραφή (δηλαδή 0100 σε σχέση με το 0111), ενώ η εγγραφή 52 απομακρύνεται κατά $\lfloor \frac{52}{11} \rfloor \bmod 11 = 4$ θέσεις, και αποθηκεύεται στη θέση $(8+4) \bmod 11 = 1$. Ταυτόχρονα ενημερώνεται η υπογραφή της θέσης 8 από 1111 σε 0111, όπως φαίνεται στο Σχήμα 12.7β. Η εγγραφή 71 εισάγεται εύκολα στη θέση 5. Κατόπιν η εγγραφή 56 κατευθύνεται στη θέση 1, όπου η υπογραφή



Σχήμα 12.7: Εισαγωγές σε αρχείο κατακερματισμού με υπογραφές.

είναι 1111, δηλαδή μεγαλύτερη από την υπογραφή 1011 της συγκεκριμένης εγγραφής. Άρα προσπελάζεται η θέση 1, που είναι κατειλημμένη από την εγγραφή 52. Η εγγραφή 52 παραμένει στη θέση αυτή γιατί έχει μικρότερη υπογραφή (δηλαδή 0111 έναντι 1011). Η εγγραφή 56 αφήνει την υπογραφή της στη θέση 1 και αποθηκεύεται μετά από $\lfloor \frac{56}{11} \rfloor \bmod 11 = 5$ θέσεις, δηλαδή στη θέση 6 όπως φαίνεται στο Σχήμα 12.7γ. Η εγγραφή 68 αποθηκεύεται χωρίς πρόβλημα στη θέση 2. Όμως πρόβλημα παρουσιάζεται στην εισαγωγή της εγγραφής 5, που κατευθύνεται στη θέση 5 που είναι κατειλημμένη από την εγγραφή 71. Εκ των δύο αυτών εγγραφών, η εγγραφή 5 αποθηκεύεται στη θέση 5 γιατί έχει μικρότερη υπογραφή από την εγγραφή 71 (1010 έναντι 1011), και η εγγραφή 71 αποθηκεύεται στη θέση 0 εφαρμόζοντας τη δεύτερη συνάρτηση κατακερματισμού, ενώ ταυτόχρονα ενημερώνεται και η υπογραφή της θέσης 5. Στο Σχήμα 12.7δ φαίνεται το αποτέλεσμα μετά την εισαγωγή της εγγραφής 5. Τέλος η εγγραφή 12 κατευθύνεται στη θέση 1 που είναι κατειλημμένη από την εγγραφή 52. Η εγγραφή αυτή παραμένει στη θέση της γιατί έχει μικρότερη υπογραφή, ενώ η εγγραφή 12 κατευθύνεται στην επόμενη θέση που είναι κατειλημμένη από την εγγραφή 68. Στο σημείο αυτό συγκρίνεται η πρώτη υπογραφή της εγγραφής 68 (δηλαδή 1000) με τη δεύτερη υπογραφή της εγγραφής 12 (δηλαδή 0110). Έτσι, η εγγραφή 12 καταλαμβάνει τη θέση 2, ενώ η εγγραφή 68 αφήνει την υπογραφή της στη θέση 2 και κατευθύνεται στη θέση 8. Η υπογραφή της θέσης αυτής είναι 0111, που είναι μικρότερη από την τιμή της δεύτερης υπογραφής (δηλαδή

1100) της εγγραφής 68. Έτσι, τελικά η εγγραφή 68 αποθηκεύεται στη θέση 3 μετά από δύο προσπάθειες. Ο αναγνώστης μπορεί πλέον να συμπεράνει πως προκύπτει η τελική μορφή της δομής στο Σχήμα 12.7ε. Η αναζήτηση ακολουθεί μία ανάλογη λογική και γι΄ αυτό το λόγο δεν δίνεται περισσότερη έμφαση στη διαδικασία αυτή.



Σχήμα 12.8: Εισαγωγές σε αρχείο με μεγάλη χωρητικότητα.

Αν η χωρητικότητα των σελίδων είναι μεγαλύτερη της μίας εγγραφής, τότε η διαδικασία δεν διαφέρει σημαντικά. Έστω ένα αρχείο με σελίδες χωρητικότητας 3 εγγραφών. Στο Σχήμα 12.8α παρουσιάζονται μερικές σελίδες του αρχείου, όπου πρέπει να εισαχθεί η εγγραφή ab. Δίνεται ότι η ακολουθία των θέσεων όπου το κλειδί πρέπει να εισαχθεί και οι αντίστοιχες υπογραφές είναι:

$$f(ab) = (10, 46, ...) \qquad \text{και} \qquad s(ab) = (1011, 0101, ...)$$

Έτσι, λοιπόν η εγγραφή ab με υπογραφή 1011 δεν μπορεί να αποθηκευθεί στη σελίδα 10 που χαρακτηρίζεται από μικρότερο διαχωριστή (1000). Επομένως η διαδικασία συνεχίζεται στη σελίδα 46, που έχει τον ίδιο διαχωριστή, αλλά είναι πλήρης. Οι εγγραφές gh και ij εξάγονται από τη σελίδα αυτή,

ώστε να εισαχθεί το κλειδί ab. Ταυτόχρονα ενημερώνεται ο διαχωριστής της σελίδας σε 1000, όπως φαίνεται στο Σχήμα 12.8β. Τώρα οι εγγραφές gh και ij πρέπει να επανα-εισαχθούν. Δίνεται ότι

$$f(gh) = (46, 95, ...) \qquad \text{και} \qquad s(gh) = (1000, 1011, ...)$$

$$f(ij) = (..., 46, 116, ...) \qquad \text{και} \qquad s(gh) = (..., 1000, 0101, ...)$$

Είναι ευνόητο πλέον πως προκύπτει το Σχήμα 12.8β. Έτσι, για μία εισαγωγή έπρεπε να προσπελασθούν τρεις σελίδες και να αλλάξει ένας διαχωριστής.

Ανακεφαλαιώνοντας μερικά πρακτικά συμπεράσματα από υλοποιήσεις της μεθόδου αυτής παρατηρείται ότι:

- η επιτυχής και η ανεπιτυχής αναζήτηση εκτελούνται εγγυημένα με μία προσπέλαση στο δίσκο,
- η διαγραφή γίνεται πολύ εύκολα με τη μέθοδο της σημαίας, που δηλώνει τη λογική διαγραφή της αντίστοιχης εγγραφής,
- η φυσική διαγραφή είναι χρονοβόρα διαδικασία, γιατί ίσως θα έπρεπε να ενημερωθεί ο πίνακας υπογραφών και να μετακινηθούν μερικές εγγραφές του κύριου αρχείου,
- η εισαγωγή είναι ακριβή αν ο παράγοντας φόρτισης πλησιάζει τη μονάδα, γιατί μπορεί να προκαλέσει διαδοχικές επαναεισαγωγές ήδη αποθηκευμένων εγγραφών,
- ο παράγοντας φόρτισης ωστόσο μπορεί να είναι υψηλός (για παράδειγμα, να ισούται με 80%), το μέγεθος της σελίδας είναι 10 εγγραφές και το μήκος του διαχωριστή είναι 8 bits, τότε μία εισαγωγή απαιτεί προσπέλαση 1,5 σελίδων,
- υπάρχει απειροστή πιθανότητα κατά την εισαγωγή η διαδικασία των διαδοχικών δοκιμών να βρεθεί εκτός ελέγχου ή και να είναι εντελώς αδύνατη, οπότε δημιουργείται υπερχείλιση με ταυτόχρονη αύξηση του κόστους προσπέλασης,
- ο απαιτούμενος χώρος στην κύρια μνήμη είναι ιδιαίτερα μικρός, όσο μεγαλύτερη είναι η χωρητικότητα των σελίδων, για παράδειγμα όπως προκύπτει και αναλυτικά (Gonnet 1988) λιγότερο από ένα byte ανά εγγραφή του κύριου αρχείου, και τέλος

- υπάρχει το ανάλογα τίμημα της πολυπλοκότητας του λογισμικού.

Η μέθοδος που παρουσιάσθηκε είναι στατική, δηλαδή το μέγεθος του αρχείου δεν μεταβάλλεται ανάλογα με το πλήθος των εισαγωγών και των διαγραφών. Όπως είναι γνωστό η μέθοδος του γραμμικού κατακερματισμού είναι δυναμική, διακρίνεται όμως από υπερχείλιση και δεν εγγυάται ότι η επιτυχής ή η ανεπιτυχής αναζήτηση μπορεί να εκτελεσθεί με μία προσπέλαση. Από τον Larson (1988) έχει προταθεί μία δυναμική εκδοχή της μεθόδου χρήσης υπογραφών (διαχωριστών), η οποία βασίζεται στο γραμμικό κατακερματισμό. Η μελέτη από τη βιβλιογραφία της σύνθετης αυτής δομής αφήνεται στον αναγνώστη.

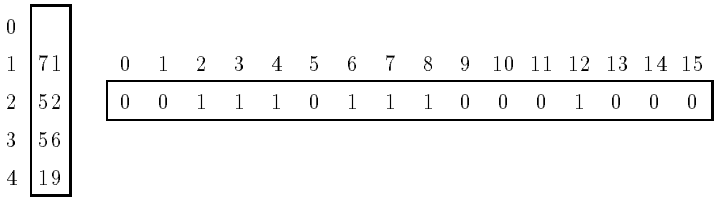
Στο σημείο αυτό αναφέρεται επιγραμματικά ότι από τον Larson (1985) προτάθηκε μία άλλη ενδιαφέρουσα δομή που ονομάζεται **εξωτερικός τέλειος κατακερματισμός** (external perfect hashing), γιατί δεν διακρίνεται από συγκρούσεις και υπερχειλίσεις. Η δομή αυτή είναι εξαιρετικά σύνθετη, αφού συνδυάζει τον επανακατακερματισμό, τον τέλειο κατακερματισμό και τα B-δένδρα. Στον αναγνώστη επαφίεται η μελέτη από τη βιβλιογραφία των αλγορίθμων διαχείρισης και αυτής της σύνθετης αυτής δομής. Ωστόσο, φαίνεται ότι παρά το γεγονός ότι τα τελευταία χρόνια έχουν προταθεί τόσο πανίσχυρες και αποτελεσματικές δομές, εντούτοις στα εμπορικά συστήματα διαχείρισης βάσεων δεδομένων το `πανταχού παρόν' B-δένδρο (και η παραλλαγή του B^+-δένδρου) είναι η πιο δημοφιλής και αναντικατάστατη δομή.

12.6 Φίλτρο Bloome

Σε πολλές εφαρμογές είναι δυνατόν η μεγάλη πλειοψηφία των ερωτήσεων να αφορά σε εγγραφές που είναι ανύπαρκτες. Όπως είναι γνωστό, συνήθως η ανεπιτυχής αναζήτηση κοστίζει ακριβότερα από την επιτυχή αναζήτηση επειδή η διαδικασία διαπίστωσης μη ύπαρξης ενός κλειδιού είναι σχετικά πιο χρονοβόρα. Το φίλτρο Bloome (1970) είναι μία μέθοδος που χρησιμεύει στην ταχύτερη διαπίστωση μίας ανεπιτυχούς αναζήτησης. Σύμφωνα με τη μέθοδο αυτή για κάθε κλειδί εγγραφής που είναι αποθηκευμένη στη δευτερεύουσα μνήμη προκύπτει μία συμπληρωματική δυαδική δομή εφαρμόζοντας μερικές συναρτήσεις κατακερματισμού. Κάθε συνάρτηση κατακερματισμού εξυπηρετεί στο να επιλέξει τυχαία κάποιο bit του φίλτρου και να το μετατρέψει από μηδέν σε άσσο. Η δομή αυτή είναι αρκετά μικρή ώστε να χωρά στην κύρια μνήμη. Επομένως η επεξεργασία της είναι πολύ γρήγορη και

θεωρείται ότι δεν συνεισφέρει στο συνολικό κόστος. Οι πιθανές αποχρίσεις σε μία ερώτηση είναι δύο: `το κλειδί δεν υπάρχει΄ ή `το κλειδί ίσως υπάρχει΄. Στην πρώτη περίπτωση δεν απαιτείται προσπέλαση στο κύριο αρχείο, ενώ στη δεύτερη η προσπέλαση εκτελείται αν και μπορεί να είναι περιττή (περίπτωση λανθασμένης πτώσης). `Ετσι, η δομή αυτή μπορεί να θεωρηθεί ως μία υπογραφή όλου του κύριου αρχείου.

Ακολουθεί ένα μικρό παράδειγμα για την καλύτερη κατανόηση της μεθόδου. Δίνεται στατικό αρχείο κατακερματισμού 5 θέσεων με χωρητικότητα μίας εγγραφής, όπου τα δεδομένα εισάγονται με εφαρμογή της συνάρτησης: $f(key) = key \bmod 5$, ενώ σε περίπτωση σύγχρουσης εφαρμόζεται η τεχνικής της γραμμικής εξέτασης. Συγχρόνως κατασκευάζεται ένα φίλτρο Bloome μεγέθους 16 ψηφίων. Για κάθε εισαγόμενη εγγραφή εφαρμόζονται άλλες δύο συναρτήσεις κατακερματισμού: η συνάρτηση `$key \bmod 16$΄ και η συνάρτηση `$key \bmod 14 + 2$΄. Στο Σχήμα 12.9 παρουσιάζεται το περιεχόμενο του κύριου αρχείου και του αντίστοιχου φίλτρου μετά την εισαγωγή των εγγραφών 52, 19, 71 και 56. `Εστω τώρα, ότι αγνοείται ο μηχανισμός του φίλτρου και ότι αναζητούνται δύο ανύπαρκτες εγγραφές 12 και 68. Η ανεπιτυχής αναζήτησης θα τερματίσει μετά από τέσσερις και τρεις προσπελάσεις στο κύριο αρχείο, αντίστοιχα. Ο αναγνώστης μπορεί να διαπιστώσει ότι με τη χρήση του φίλτρου οι προσπελάσεις αυτές θα αποφεύγονταν. Σε περίπτωση αναζήτησης της εγγραφής 44, εύκολα προκύπτει ότι το φίλτρο θα οδηγούσε σε λανθασμένη πτώση.



Σχήμα 12.9: Αρχείο και αντίστοιχο φίλτρο Bloome.

Μία τέτοια μέθοδος μπορεί να φανεί πολύ χρήσιμη σε πληθώρα εφαρμογών, όπου είναι πολύ πιθανό ότι η ερώτηση θα είναι ανεπιτυχής, όπως για παράδειγμα,

- κατά την αναζήτηση από έναν καταστηματάρχη σε μία λίστα κλεμμένων ή χαμένων πιστωτικών καρτών,

- κατά την αναζήτηση από κάποιον τελωνειακό υπάλληλο σε μία λίστα κλεμμένων διαβατηρίων,
- κατά την αναζήτηση από ένα πρόγραμμα επεξεργασίας κειμένου για τις λανθασμένες λέξεις (spelling check), ή
- κατά τη χρήση της επιλογής DISTINCT για την απάλειψη των διπλοεγγραφών, που μπορεί να προκύψουν από μία ερώτηση με SELECT της γλώσσας ερωτήσεων SQL.

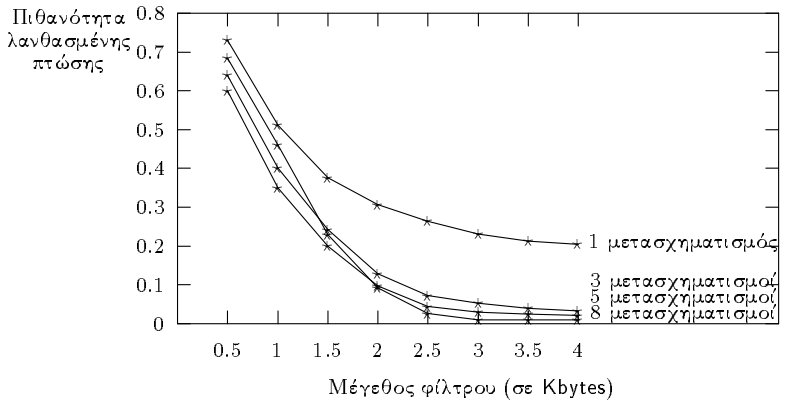
Ωστόσο, ιδιαίτερα αναφέρεται ότι ένα φίλτρο Bloome μπορεί κάλλιστα να εφαρμοσθεί και στη μέθοδο του κατακερματισμού με υπογραφές. Όπως φάνηκε με τη μέθοδο αυτή είναι δυνατό μία εισαγωγή να είναι αρκετά χρονοβόρα αν προκαλέσει τη διαδοχική διαγραφή και επανα-εισαγωγή άλλων εγγραφών. Στη χειρότερη περίπτωση μπορεί θεωρητικά η εκτέλεση μίας εισαγωγής να μην τερματισθεί ποτέ. Με σκοπό την πρόληψη του εκφυλισμού της επίδοσης κατά την εισαγωγή μπορεί να δημιουργηθεί ένα αρχείο υπερχείλισης, όπου να αποθηκεύονται οι εγγραφές που δεν έγινε κατορθωτό να αποθηκευθούν στο κύριο αρχείο μετά από ένα συγκεχριμένο μικρό αριθμό προσπελάσεων. Είναι προφανές ότι το αρχείο αυτό με τις εγγραφές υπερχείλισης θα είναι σχετικά μικρό, οπότε και το αντίστοιχο φίλτρο θα είναι επίσης σχετικά μικρό και επομένως αποτελεσματικό. Η επεξεργασία του φίλτρου αυτού στην κύρια μνήμη θα είναι λιγότερο χρονοβόρα από την επεξεργασία του αρχείου των υπογραφών και τις διαδοχικές συγκρίσεις με τους διάφορους διαχωριστές, που εκτελούνται επίσης στην κύρια μνήμη. Στη συνέχεια εξετάζεται αναλυτικότερα μία σπουδαία πρακτική εφαρμογή του φίλτρου Bloome.

Αν τα δεδομένα ενός αρχείου χρησιμοποιούνται κυρίως για αναγνώσεις, ενώ οι εισαγωγές και διαγραφές εγγραφών είναι λιγότερο συχνές, τότε μπορεί να χρησιμοποιηθεί μία μέθοδος που προτάθηκε από τον Severance (1976). Σύμφωνα με τη μέθοδο αυτή το κύριο αρχείο παραμένει στατικό, ενώ σε ένα δευτερεύον αρχείο, που ονομάζεται **αρχείο διαφορών** (differential file), αποθηκεύονται όλες οι ενημερώσεις των εγγραφών. Το αρχείο διαφορών μπορεί να είναι ένα σειριακό αρχείο με την προϋπόθεση ότι δεν θα μεγαλώσει υπέρμετρα. Έτσι, τα δεδομένα είναι αποθηκευμένα σε δύο αρχεία, που μπορεί να συγχωνευθούν σε περιόδους χαμηλού φόρτου επεξεργασίας. Ανάλογη διαδικασία συμβαίνει στο κύριο αρχείο αρχείο και στο αρχείο συναλλαγών που εξετάσθηκαν στο πρώτο κεφάλαιο, ωστόσο στην προκειμένη περίπτωση τα δεδομένα θα πρέπει να είναι διαθέσιμα κάθε στιγμή στους χρήστες.

Μία απλοϊκή μέθοδος αναζήτησης προτείνει την εξέταση κατ΄ αρχήν του αρχείου διαφορών. Αν η εγγραφή δεν υπάρχει στο αρχείο αυτό, τότε στη συνέχεια εξετάζεται το κύριο στατικό αρχείο. Μία περισσότερο εξελιγμένη μέθοδος αναζήτησης ανήκει στον Gremillion (1982), που συνιστά τη δημιουργία ενός φίλτρου Bloome για την ταχύτερη επεξεργασία του αρχείου διαφορών με σκοπό την πιθανή πρόβλεψη ανεπιτυχούς αναζήτησης. Έτσι, αρχικά η εξέταση του φίλτρου μπορεί να δώσει δύο πιθανές απαντήσεις:

- η αναζητούμενη εγγραφή δεν είναι αποθηκευμένη στο αρχείο διαφορών, οπότε αμέσως εξετάζεται το κύριο αρχείο, ή
- η αναζητούμενη εγγραφή μπορεί να είναι αποθηκευμένη στο αρχείο διαφορών, οπότε η εξέταση αρχίζει από αυτό. Αν η εγγραφή δεν βρεθεί στο αρχείο αυτό, τότε και πάλι η διαδικασία συνεχίζεται στο κύριο αρχείο.

Στο Σχήμα 12.10 παρουσιάζονται μερικά πειραματικά αποτελέσματα του Gremillion που δείχνουν μεγάλη αποτελεσματικότητα της μεθόδου, αν το μέγεθος του φίλτρου είναι της τάξης 3 ως 4 Kb και γίνεται χρήση περίπου 5 συναρτήσεων κατακερματισμού. Ενδιαφέρουσα είναι η παρατήρηση ότι για μικρά μεγέθη του φίλτρου η πιθανότητα λανθασμένης πτώσης αυξάνει καθώς αυξάνεται ο αριθμός των συναρτήσεων. Το φαινόμενο αυτό εξηγείται από το γεγονός ότι στις περιπτώσεις αυτές είναι πολλοί οι σχετικοί άσσοι.



Σχήμα 12.10: Πιθανότητα λανθασμένης πτώσης σε αρχείο διαφορών.

Στη συνέχεια ακολουθεί μία απλή αναλυτική εξέταση της μεθόδου που αναφέρεται από τον Mullin (1983). Έστω ότι το φίλτρο Bloome έχει μέγεθος m δυαδικά ψηφία, όπου το m είναι της τάξης των μερικών δεκάδων χιλιάδων. Τα ψηφία αυτά αρχικά ισούνται όλα με μηδέν. Σε κάθε πρωτεύον κλειδί εισαγόμενης εγγραφής στο κύριο αρχείο εφαρμόζονται k συναρτήσεις κατακερματισμού (όπου το k μικρός ακέραιος), οπότε προκύπτουν οι θέσεις των ψηφίων του φίλτρου που θα πάρουν την τιμή του άσσου. Δηλαδή, ο μέγιστος αριθμός των ψηφίων του φίλτρου που θα γίνουν άσσοι για κάθε κλειδί θα είναι k. Επομένως κατά την αναζήτηση του κλειδιού μίας εγγραφής πρώτα εφαρμόζονται οι συναρτήσεις κατακερματισμού και ελέγχονται τα αντίστοιχα ψηφία με τη σειρά. Αν από κάποια συνάρτηση προκύψει η θέση ενός ψηφίου που είναι μηδέν στο φίλτρο, τότε η αναζήτηση τερματίζεται ως ανεπιτυχής. Στην αντίθετη περίπτωση μπορεί να συμβεί μία λανθασμένη πτώση με μία συγκεκριμένη πιθανότητα. Ποιά είναι όμως αυτή η πιθανότητα;

Έστω ότι στο κύριο αρχείο έχουν εισαχθεί n εγγραφές. Η πιθανότητα κάποιο ψηφίο να γίνει άσσος από μία συνάρτηση κατακερματισμού ισούται με $1/m$, ενώ η πιθανότητα να μη γίνει άσσος είναι $1–1/m$. Η πιθανότητα αυτό το ψηφίο να μη γίνει άσσος μετά από την εφαρμογή nk συναρτήσεων μετασχηματισμού είναι $(1-1/m)^{nk}$. Άρα η πιθανότητα το ψηφίο αυτό να γίνει άσσος μετά την εφαρμογή των συναρτήσεων αυτών είναι $1-(1-1/m)^{nk}$. Έστω τώρα, ότι αναζητάται μία εγγραφή. Από την εγγραφή αυτή με τους κατάλληλους μετασχηματισμούς προκύπτουν k άσσοι. Η πιθανότητα οι k αυτοί άσσοι να υπάρχουν στο φίλτρο είναι $\left(1-(1-1/m)^{nk}\right)^k$. Αν a είναι η πιθανότητα η αναζήτηση να είναι ανεπιτυχής, τότε η πιθανότητα να συμβεί μία λανθασμένη πτώση είναι $a \times \left(1-(1-1/m)^{nk}\right)^k$. Η ανάλυση αυτή οδηγεί σε τιμές που βρίσκονται κοντά στα αποτελέσματα του Gremillion και μπορούν να χρησιμοποιηθούν κατά το σχεδιασμό του αρχείου διαφορών και του αντίστοιχου φίλτρου.

Τα πλεονεκτήματα μίας τέτοιας οργάνωσης των δεδομένων με ένα αρχείο διαφορών είναι πολλαπλά, όπως για παράδειγμα:

- καλή επίδοση κατά την αναζήτηση, γιατί το κύριο αρχείο μπορεί να αποθηκευθεί σε ειδική ταχύτερη δευτερεύουσα συσκευή,
- απλοποίηση του λογισμικού επεξεργασίας του κύριου αρχείου,
- δυνατότητα ταυτόχρονης χρήσης του κύριου αρχείου από πολλούς χρήστες για λόγους που θα εξηγηθούν σε επόμενο κεφάλαιο, και τέλος

- μεγάλη αξιοπιστία των δεδομένων του κύριου αρχείου, αφού μάλιστα ο κάθε προγραμματιστής εφραμογών μπορεί να έχει το δικό του αρχείο διαφορών.

Η ίδια φιλοσοφία διατρέχει και τη δομή του **B-δένδρου με περιοχή αλλαγών** (change area B-tree) που προτάθηκε από τον Mullin (1981). Σύμφωνα με αυτή τη δομή και το κύριο αρχείο αλλά και το αρχείο διαφορών είναι δύο απλά B-δένδρα.

12.7 Ασκήσεις

<1> Να κατασκευασθεί ένα δένδρο υπογραφών με εγγραφές που να αποτελούνται από τα πεδία Ειδικότητα, Διεύθυνση, Διευθυντής, και Μισθός. Το συνολικό μήκος της υπογραφής είναι 16 bits και επιμερίζεται σε 3, 4, 4 και 5 bits αντίστοιχα. Οι επόμενες συναρτήσεις κατακερματισμού 'σηκώνουν' ένα bit σε άσσο ως εξής:

$$H_1 = \begin{cases} \text{το 1ο bit αν η ονομασία της ειδικότητας αρχίζει από Α} - \Theta \\ \quad 2ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Ι} - \Pi \\ \quad 3ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Ρ} - \Omega \end{cases}$$

$$H_2 = \begin{cases} \text{το 1ο bit αν η ονομασία της διεύθυνσης αρχίζει από Α} - \text{Ζ} \\ \quad 2ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Η} - \text{Μ} \\ \quad 3ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Ν} - \Sigma \\ \quad 4ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Τ} - \Omega \end{cases}$$

$$H_3 = \begin{cases} \text{το 1ο bit αν η ονομασία του διευθυντή αρχίζει από Α} - \text{Ζ} \\ \quad 2ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Η} - \text{Μ} \\ \quad 3ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Ν} - \Sigma \\ \quad 4ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από Τ} - \Omega \end{cases}$$

$$H_4 = \begin{cases} \text{το 1ο bit αν ο μισθός είναι μέχρι 150.000 δραχμές} \\ \quad 2ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από 150.001 μέχρι 250.000 δραχμές} \\ \quad 3ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από 250.001 μέχρι 350.000 δραχμές} \\ \quad 4ο \qquad\qquad\qquad\qquad\qquad\qquad \text{από 350.001 μέχρι 450.000 δραχμές} \\ \quad 5ο \qquad\qquad\qquad\qquad\qquad\qquad \text{περισσότερο από 450.000 δραχμές} \end{cases}$$

Κατά το σχηματισμό των ανωτέρων επιπέδων του δένδρου να θεωρηθεί ότι ο παράγοντας ομαδοποίησης ισούται με τρία. Να γίνουν παραδείγματα αναζήτησης.

<**2**> Να κατασκευασθεί παράδειγμα μίας δομής με υπογραφές σελίδων με βάση τα δεδομένα της προηγούμενης άσκησης. Για τις υπογραφές των εγγραφών να χρησιμοποιηθούν οι ίδιες συναρτήσεις κατακερματισμού για όλα τα πεδία. Η περιγραφή της σελίδας έχει μήκος οκτώ bits, δύο bits ανά χαρακτηριστικό, και χρησιμοποιούνται οι επόμενες συναρτήσεις που `σηκώνουν' τα αντίστοιχα bits:

$$T_1 = \begin{cases} \text{το 1ο bit αν η ονομασία της ειδικότητας αρχίζει από A} - \text{M} \\ \text{2ο} \qquad\qquad\qquad\qquad\qquad\qquad \text{από N} - \Omega \end{cases}$$

$$T_2 = \begin{cases} \text{το 1ο bit αν η ονομασία της διεύθυνσης αρχίζει από A} - \text{M} \\ \text{2ο} \qquad\qquad\qquad\qquad\qquad\qquad \text{από N} - \Omega \end{cases}$$

$$T_3 = \begin{cases} \text{το 1ο bit αν η ονομασία του διευθυντή αρχίζει από A} - \text{M} \\ \text{2ο} \qquad\qquad\qquad\qquad\qquad\qquad \text{από N} - \Omega \end{cases}$$

$$T_4 = \begin{cases} \text{το 1ο bit αν ο μισθός είναι μέχρι και 350.000 δραχμές} \\ \text{2ο} \qquad\qquad\qquad\qquad\qquad\qquad \text{περισσότερο από 150.001 δραχμές} \end{cases}$$

Να γίνουν παραδείγματα αναζήτησης. Να γίνει ποιοτική σύγκριση με την επίδοση της αναζήτησης της δομής της προηγούμενης άσκησης.

<**3**> Στο παράδειγμα του Σχήματος 12.7 να εισαχθεί η εγγραφή 38 ή η εγγραφή 49. Τι αλλαγές προκαλούνται στο κύριο αρχείο και στο αρχείο των υπογραφών αντίστοιχα;

<**4**> Τα κλειδιά του παραδείγματος του Σχήματος 12.7 να εισαχθούν σε αντίστροφη σειρά. Να σχεδιασθεί διαδοχικά η διαδικασία.

<**5**> Δίνεται αρχείο κατακερματισμού 5 θέσεων με χωρητικότητα μίας εγγραφής, όπου εφαρμόζεται η συνάρτηση key mod 5 και η μέθοδος της γραμμικής εξέτασης για την επίλυση των συγκρούσεων. Φίλτρο Bloome των 16 bits κατασκευάζεται με τη βοήθεια των συναρτήσεων `$key$ mod 16' και `key mod $14+2$'. Να εισαχθούν οι εγγραφές 22, 51, 26 και 85 στο αρχείο και να ενημερωθεί το φίλτρο. Σε ποιές περιπτώσεις κατά τις αναζητήσεις των εγγραφών 81, 27, 46, 23, 51 και 39 συμβαίνει λανθασμένη πτώση; Πόσες προσπελάσεις στο αρχείο εξοικονομούνται λόγω της ύπαρξης του φίλτρου κατά τις αναζητήσεις των κλειδιών αυτών;