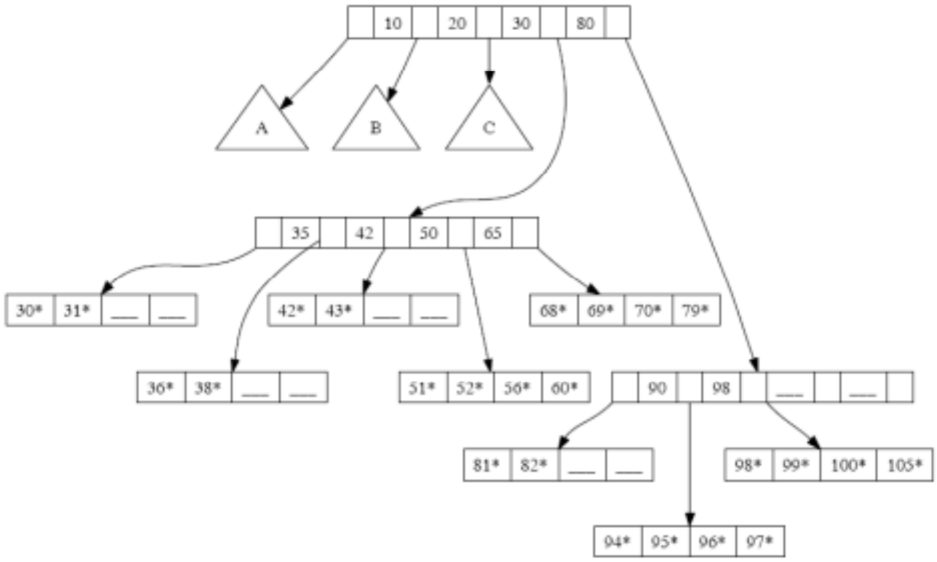
ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΛΥΣΗ ΣΤΗΝ ΔΕΥΤΕΡΗ ΑΣΚΗΣΗ

ΜΑΘΗΜΑ	**ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ**
ΑΚΑΔ. ΕΤΟΣ	2012-13
ΔΙΔΑΣΚΟΝΤΕΣ	Ιωάννης Βασιλείου *Καθηγητής*, Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών Τιμολέων Σελλής *Καθηγητής*, Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

Ερώτημα 1.

Θεωρείστε το B+ δέντρο της Εικόνας 1.
(i) Σε ποιο υπο-δέντρο θα αναζητήσετε την τιμή 9 και σε ποιο την τιμή 25;
(ii) Εισάγετε την τιμή 101 και δώστε το αποτέλεσμα της εισαγωγής.
(iii) Διαγράψτε την τιμή 43 και δώστε το αποτέλεσμα της διαγραφής.
(iv) Εισάγετε την τιμή 80 και δώστε το αποτέλεσμα της εισαγωγής.
(v) Διαγράψτε την τιμή 30 και δώστε το αποτέλεσμα της διαγραφής.
Οι παραπάνω εισαγωγές και διαγραφές να γίνουν όλες στο αρχικό ευρετήριο της Εικόνας 1.

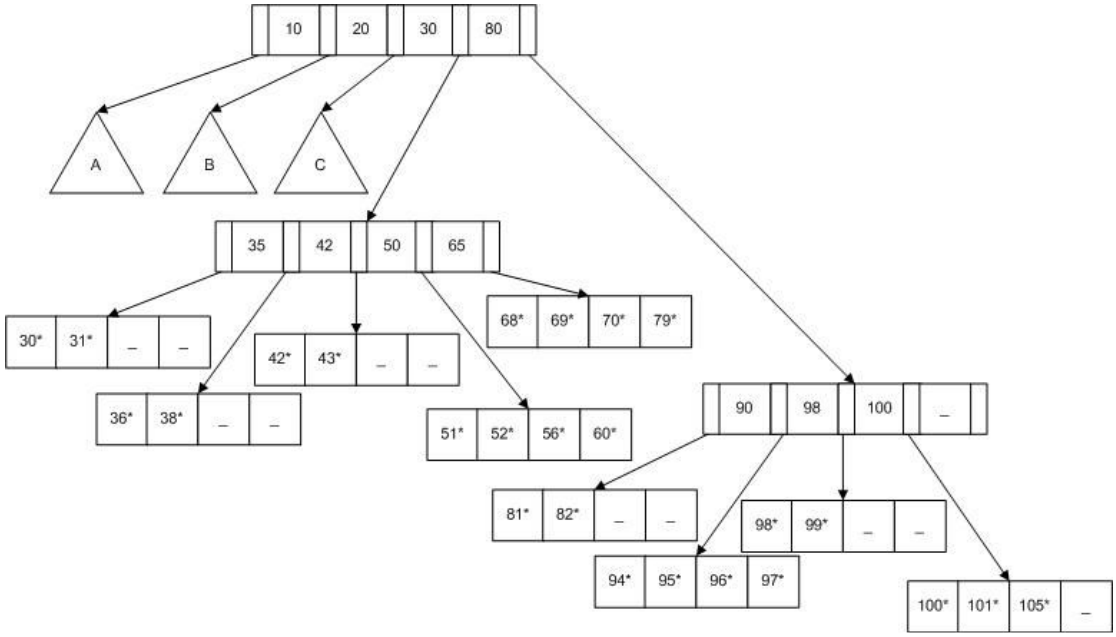


Εικόνα 1. B+ δέντρο για το Ερώτημα 1. Ο μέγιστος αριθμός δεικτών για τους εσωτερικούς κόμβους είναι 5 και για τα φύλλα 4. Τα φύλλα είναι διπλά συνδεδεμένα μεταξύ τους (δε φαίνεται στην εικόνα). Τα A, B, C είναι υπο-δέντρα στα οποία δείχνουν οι αντίστοιχοι δείκτες.

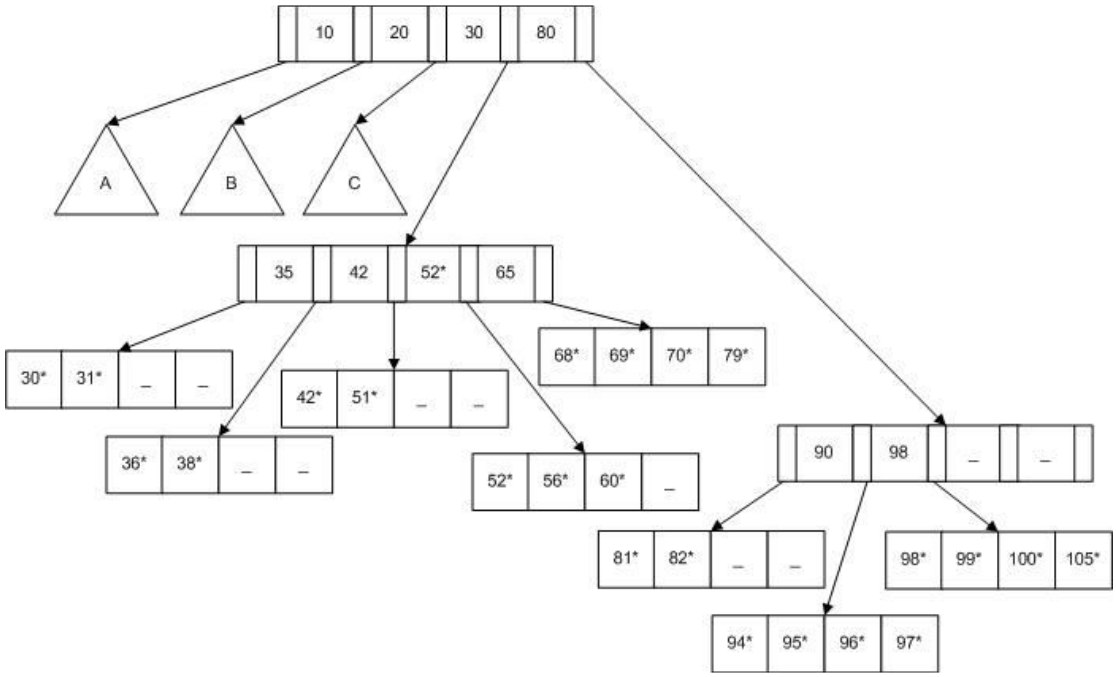
Λύση

(i) Η τιμή 9 θα βρίσκεται στο υπο-δέντρο A, και η τιμή 25 στο υπο-δέντρο C.

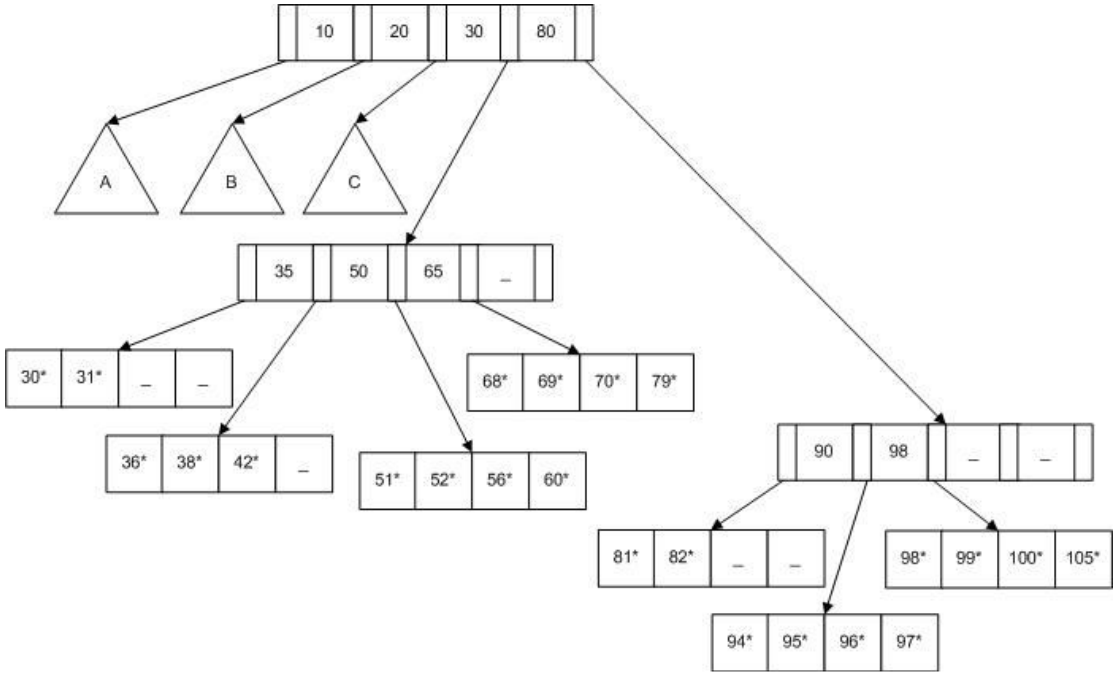
(ii) Εισάγουμε την τιμή 101, οπότε διασπάται το δεξιότερο φύλλο και ενημερώνεται κατάλληλα ο κόμβος-πατέρας.



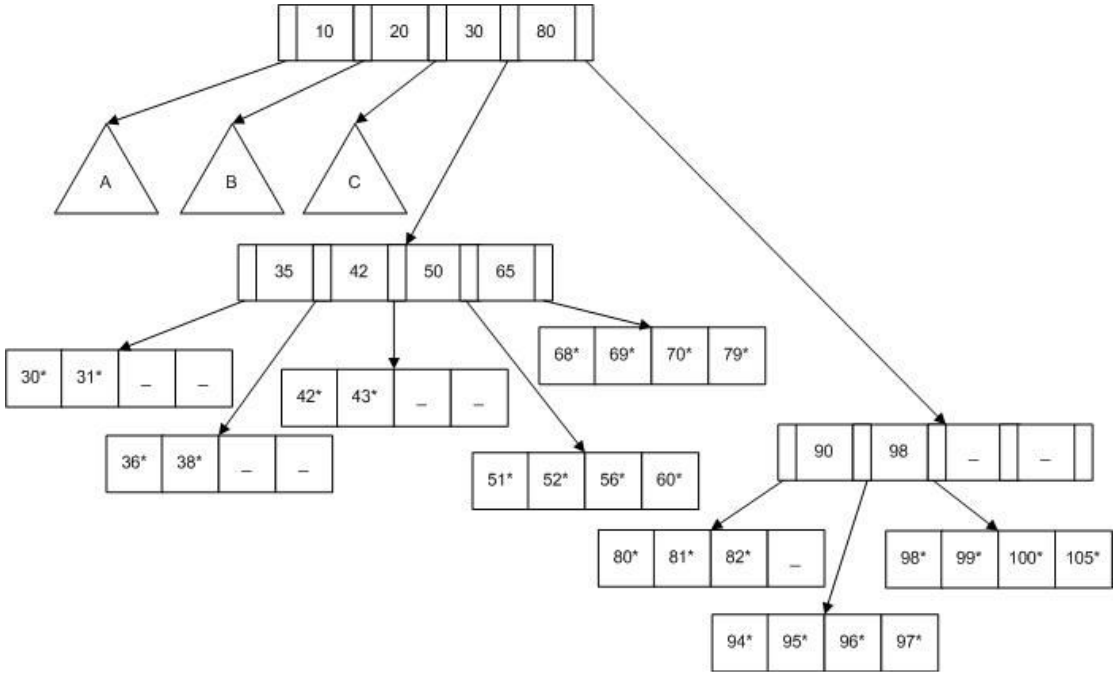
(iii) Διαγράφουμε την τιμή 43, οπότε γίνεται ανακατανομή με το δεξί φύλλο-αδελφό και ενημερώνεται κατάλληλα ο κόμβος-πατέρας.



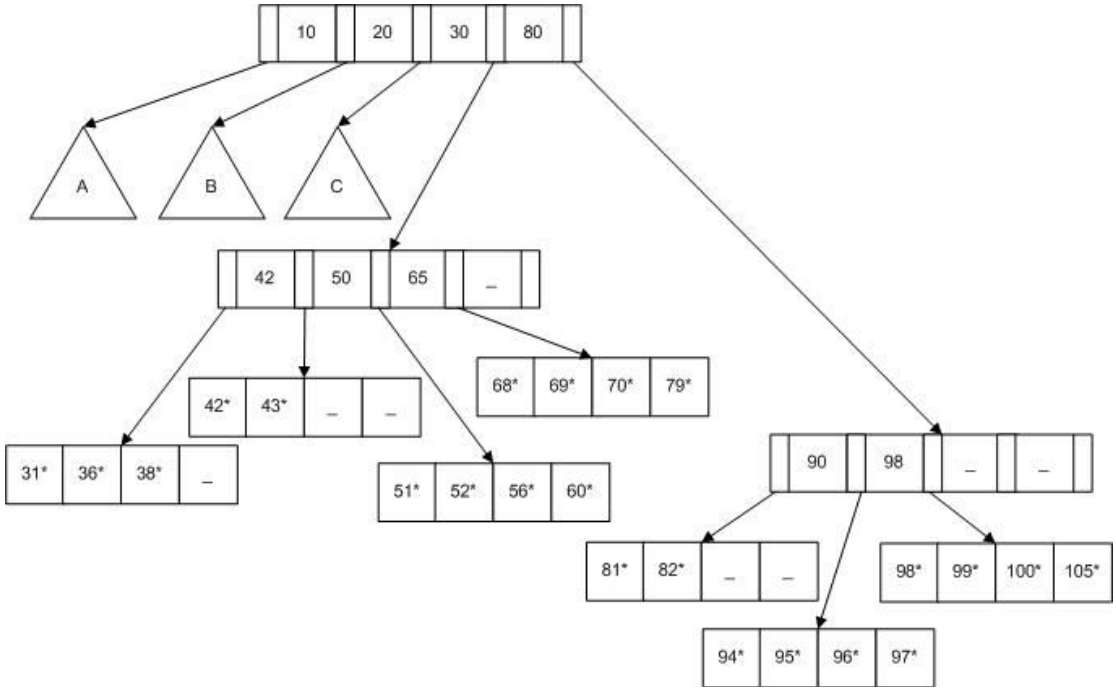
Εναλλακτικά, μπορεί να γίνει συγχώνευση με το αριστερό φύλλο-αδερφό. Οπότε έχουμε:



(iv) Εισάγουμε την τιμή 80 στο κατάλληλο φύλλο. Στο εν λόγω φύλλο υπάρχει χώρος για την εισαγωγή της.



(v) Διαγράφουμε την τιμή 30, οπότε απαιτείται συγχώνευση με το δεξί φύλλο-αδελφό και κατάλληλη ενημέρωση του κόμβου-πατέρα.



Ερώτημα 2.

Θεωρείστε στατικό πίνακα κατακερματισμού (οργάνωση αρχείου) όπου κάθε κάδος χωρά μέχρι 2 εγγραφές. Η συνάρτηση κατακερματισμού είναι $h(k) = kmod3$. Ως συνήθως ένας κάδος έχει την χωρητικότητα ενός μπλοκ στον δίσκο.
(i) Δείξτε τα περιεχόμενα του πίνακα κατακερματισμού σε κάθε βήμα κατά την εισαγωγή των ακόλουθων κλειδιών: 27, 5, 18, 30, 10, 32, 38. Για τη διαχείριση υπερχειλίσεων χρησιμοποιείστε αλυσιδωτή σύνδεση (chaining).
(ii) Πόσο κοστίζει (σε I/O) η προσπέλαση της εγγραφής με κλειδί 30 και πόσο η προσπέλαση της εγγραφής με κλειδί 32;
(iii) Ποιος ο μέσος αριθμός I/O μιας αποτυχημένης αναζήτησης;

Λύση

(i) Με βάση τη συνάρτηση κατακερματισμού ισχύει:

k	h(k)
27	0
5	2
18	0
30	0
10	1
32	2
38	2

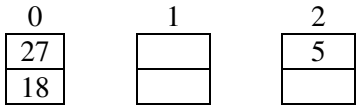
Εισαγωγή 27:

0	1	2
27		

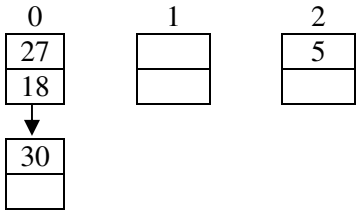
Εισαγωγή 5:

0	1	2
27		5

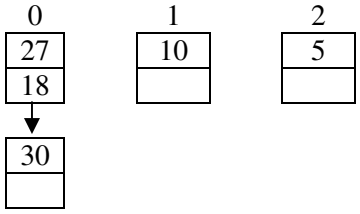
Εισαγωγή 18:



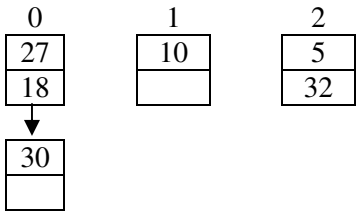
Εισαγωγή 30:



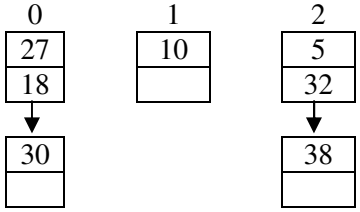
Εισαγωγή 10:



Εισαγωγή 32:



Εισαγωγή 38:



(ii)Η εγγραφή με τιμή κλειδιού 30 βρίσκεται στην περιοχή υπερχείλισης του bucket 0 του πίνακα κατακερματισμού. Συνεπώς απαιτούνται 2 προσβάσεις στο δίσκο για την προσπέλασή της, άρα 2 I/O.

Η εγγραφή με τιμή κλειδιού 32 βρίσκεται στο bucket 2 του πίνακα κατακερματισμού (όχι όμως σε περιοχή υπερχείλισης). Συνεπώς απαιτείται 1 πρόσβαση στο δίσκο για την προσπέλασή της, άρα 1 I/O.

(iii) Ο πίνακας κατακερματισμού έχει τρία bucket (ως κύρια στοιχεία), εκ των οποίων τα δύο έχουν ξεχωριστές περιοχές υπερχείλισης. Επομένως για κάθε ένα από τα δύο bucket που έχουν και περιοχή υπερχείλισης απαιτούνται δύο προσβάσεις στην περίπτωση αποτυχημένης αναζήτησης. Τέλος έχουμε το μεσαίο bucket (χωρίς περιοχή υπερχείλισης), το οποίο απαιτεί μονάχα μία πρόσβαση.

Συνολικά, για όλα τα bucket του πίνακα απαιτούνται 5 προσβάσεις. Υπολογίζοντας τον μέσο όρο έχουμε:

$$Μέσος\ αριθμός\ προσβάσεων\ σε\ αποτυχημένη\ αναζήτηση = 5/3 = 1.67$$

Ερώτημα 3.

Θεωρείστε μια σχέση R που έχει 500.000 εγγραφές (πλειάδες) και είναι αποθηκευμένη σε ένα αρχείο σωρού. Κατασκευάζουμε ένα ευρετήριο για ένα γνώρισμα (πεδίο) A της σχέσης R, που δεν είναι κλειδί για τη σχέση. Έστω ότι υπάρχουν 1.000 διαφορετικές τιμές του A και οι εγγραφές είναι ομοιόμορφα κατανεμημένες ως προς αυτές. Το μέγεθος του πεδίου A είναι 8 bytes και των δεικτών 16 bytes (όλων των ειδών δεικτών). Θεωρείστε μέγεθος block 2048 bytes.

(α) Έστω ότι το ευρετήριο είναι ένα δευτερεύον ευρετήριο. Θεωρείστε ότι χρησιμοποιείται ένα επιπλέον επίπεδο έμμεσων δεικτών, εφόσον το κλειδί αναζήτησης (A) δεν είναι υποψήφιο κλειδί (βλέπε σελ. 483-485 από το βιβλίο των Silberschatz, Korth, Sudarshan).
   (i) Δώστε το συνολικό μέγεθος του αρχείου ευρετηρίου (συμπεριλαμβανομένου και του επιπλέον επιπέδου) σε αριθμό blocks.
   (ii) Υπολογίστε το κόστος σε αριθμό I/O (δηλαδή, σε αριθμό προσπελάσεων σελίδων) της αναζήτησης A = α, όπου α μια τιμή από το πεδίο ορισμού του A.
   (iii) Έστω ότι εισάγουμε μια νέα πλειάδα στη σχέση R. Εξηγείστε πως αλλάζει το ευρετήριο, όταν η εγγραφή έχει μια τιμή στο A η οποία (1) δεν εμφανίζεται ήδη σε άλλη πλειάδα και (2) εμφανίζεται ήδη σε άλλη πλειάδα. Δώστε μια εκτίμηση του κόστους (πολυπλοκότητας) αυτών των αλλαγών (σε αριθμό I/O) για κάθε μία από τις δύο περιπτώσεις.

(β) Υποθέστε τώρα ότι κατασκευάζουμε ένα B+-δέντρο. Θεωρείστε ότι κάθε τιμή κλειδιού αναζήτησης αποθηκεύεται μόνο μια φορά σε κάποιο από τα φύλλα του δέντρου και για κάθε τιμή διατηρείται ένας "κάδος" από δείκτες εγγραφών με αυτήν την τιμή κλειδιού αναζήτησης. Αυτοί οι κάδοι αποθηκεύονται σε ξεχωριστά block, δημιουργώντας ένα επιπλέον επίπεδο. Οι δείκτες των φύλλων δείχνουν στο επιπλέον επίπεδο (βλέπε σελ. 497-499 από το βιβλίο των Silberschatz, Korth, Sudarshan).
   Επαναλάβατε τα υπο-ερωτήματα (i)-(iii) του ερωτήματος (α) θεωρώντας ότι το δέντρο είναι όσο το δυνατόν πιο γεμάτο.

(γ) Υποθέστε τώρα ότι κατασκευάζουμε ένα δευτερεύον ευρετήριο όπου δεν υπάρχει το επιπλέον επίπεδο. Επαναλάβατε τα υπο-ερωτήματα (i) και (ii) του ερωτήματος (α). (βλέπε σελ. 483-485 από το βιβλίο των Silberschatz, Korth, Sudarshan και διαφάνειες διαλέξεων).

(δ) Υποθέστε μια αναζήτηση A > α, όπου α μια τιμή από το πεδίο ορισμού του A, η οποία ικανοποιείται από το 10% των εγγραφών της σχέσης. Εξηγείστε πως μπορείτε να χρησιμοποιήσετε τα ευρετήρια των ερωτημάτων (α)-(γ) για αυτήν την αναζήτηση και δώστε μια εκτίμηση του κόστους της σε κάθε περίπτωση.

Λύση

(α) Το αρχείο του ευρετηρίου περιλαμβάνει εγγραφές της μορφής <key, pointer>, μια για κάθε διαφορετική τιμή του κλειδιού αναζήτησης (A). Κάθε δείκτης (pointer) μιας καταχώρησης ευρετηρίου δείχνει σε ένα block από δείκτες εγγραφών στο επιπλέον επίπεδο. Κάθε δείκτης του block αυτού δείχνει προς μια εγγραφή του αρχείου δεδομένων με τιμή στο πεδίο ευρετηριοποίησης ίση με το αντίστοιχο key. Οι δείκτες προς εγγραφές με ίδια τιμή στο κλειδί αναζήτησης (key) βρίσκονται στο ίδιο block, και αν είναι τόσο πολλές οι εγγραφές ώστε οι δείκτες να μην χωρούν σε ένα block, χρησιμοποιείται μια συνδεδεμένη λίστα από block. Οι δείκτες προς εγγραφές με διαφορετική τιμή στο κλειδί αναζήτησης είναι σε διαφορετικές λίστες από block.
(i) Το μέγεθος κάθε εγγραφής του ευρετηρίου είναι $8 + 16 = 24$ bytes. Δεδομένου ότι κάθε block έχει μέγεθος 2048 bytes, ένα block ευρετηρίου χωρά $\lfloor 2048\ bytes/24\ bytes \rfloor = 85$ εγγραφές ευρετηρίου. Υπάρχουν 1000 διαφορετικές τιμές του A, οπότε το ευρετήριο θα έχει 1000 εγγραφές, μια για κάθε διαφορετική τιμή. Εφόσον 85 εγγραφές ευρετηρίου χωρούν σε κάθε block δίσκου, για το ευρετήριο χρειάζονται $\lceil 1000/85 \rceil = 12$ block.

Όσον αφορά το ενδιάμεσο επίπεδο, ισχύουν τα εξής. Εφόσον οι εγγραφές της σχέσης R είναι ομοιόμορφα κατανεμημένες ως προς τις τιμές του Α, σε κάθε διαφορετική τιμή του κλειδιού αναζήτησης αντιστοιχούν $500000/1000 = 500$ εγγραφές, οπότε θα πρέπει να κρατούνται στο ενδιάμεσο επίπεδο 500 δείκτες προς τις αντίστοιχες εγγραφές. Κάθε block χωρά $2048\,bytes/16\,bytes = 128$ δείκτες. Άρα, για κάθε διαφορετική τιμή του Α απαιτούνται $\lceil 500/128 \rceil = 4$ block για την αποθήκευση δεικτών προς εγγραφές που έχουν τη συγκεκριμένη τιμή στο Α. Συνεπώς, συνολικά απαιτούνται $4 \times 1000 = 4000$ block για το ενδιάμεσο επίπεδο.

Το συνολικό μέγεθος του ευρετηρίου είναι 4012 block.

(ii) Αρχικά θα αναζητηθεί η εγγραφή με τιμή κλειδιού αναζήτησης Α = α στο ευρετήριο. Με δυαδική αναζήτηση (εφόσον το αρχείο είναι ταξινομημένο) το κόστος είναι $\lceil \log_2 12 \rceil = 4$ block. Ακολουθώντας τον δείκτη αυτής της εγγραφής προς το επιπλέον επίπεδο, θα προσπελάσουμε τα 4 block που περιέχουν τους δείκτες προς τις εγγραφές με Α = α. Το κόστος αναζήτησης του ευρετηρίου είναι $4 + 4 = 8$ I/O.

Υπάρχουν 500 εγγραφές της R με Α=α. Αν υποθέσουμε ότι βρίσκονται σε διαφορετικά block, θα πρέπει να διαβαστούν 500 block. Οπότε το συνολικό κόστος αναζήτησης είναι $8 + 500 = 508$ I/O. Αν υποθέσουμε ότι υπάρχουν δύο ή περισσότερες εγγραφές στο ίδιο block, τότε γενικά υπάρχουν ν block με τις ζητούμενες εγγραφές και θεωρώντας επαρκούς μεγέθους buffer στη μνήμη κάθε block μπορεί να διαβάζεται μια φορά από το δίσκο (και την επόμενη φορά που θα ζητηθεί θα βρίσκεται ήδη στον buffer). Οπότε, το συνολικό κόστος αναζήτησης είναι $8 + \nu$ I/O.

(iii) (1) Εφόσον η τιμή του Α δεν εμφανίζεται ήδη σε άλλη πλειάδα, δεν υπάρχει αντίστοιχη εγγραφή στο ευρετήριο για τη συγκεκριμένη τιμή κλειδιού αναζήτησης. Θα πρέπει να καταχωρηθεί μια εγγραφή με τη συγκεκριμένη τιμή κλειδιού αναζήτησης στο κατάλληλο block, αφού το ευρετήριο είναι ταξινομημένο. Αρχικά θα γίνει αναζήτηση στο ευρετήριο για να βρεθεί το block στο οποίο πρέπει να γίνει η εισαγωγή. Γι' αυτό θα διαβαστούν $\lceil \log_2 12 \rceil = 4$ block. Αφού οι εγγραφές είναι τοποθετημένες διαδοχικά και για να διατηρηθεί η ταξινόμηση, θα πρέπει να "ολισθήσουν" οι εγγραφές που βρίσκονται αμέσως μετά από αυτή που θέλουμε να εισάγουμε μια θέση "πιο κάτω". Άρα θα γίνει 1 write του block όπου καταχωρείται η νέα εγγραφή (έχει ήδη διαβαστεί από τη δυαδική αναζήτηση) και θα διαβαστούν και θα εγγραφούν τα επόμενα block. Κατά μέσο όρο, αυτό θα γίνει για τα μισά block, δηλαδή 1 read και 1 write για 6 block ευρετηρίου. Τέλος, πρέπει να καταχωρηθεί κατάλληλος δείκτης στο ενδιάμεσο επίπεδο. Γι' αυτό, θα δεσμευτεί ένα νέο block δίσκου (εφόσον η τιμή κλειδιού δεν εμφανίζεται ήδη σε άλλη πλειάδα), όπου θα εγγραφεί η κατάλληλη καταχώρηση δείκτη (1 write).

Το συνολικό κόστος ενημέρωσης του ευρετηρίου είναι $4 + 1 + 6 \times 2 + 1 = 18$ I/O.

Σημειώνεται ότι στην πράξη για να γίνει η εισαγωγή πιο αποδοτική και να αποφευχθεί η ολίσθηση των εγγραφών ευρετηρίου που απαιτείται για να διατηρηθεί το ευρετήριο ταξινομημένο, συχνά δημιουργείται ένα προσωρινό μη διατεταγμένο αρχείο που λέγεται αρχείο υπερχείλισης. Οι νέες εγγραφές ευρετηρίου δεν εισάγονται στη σωστή τους θέση αλλά στο αρχείο υπερχείλισης. Το αρχείο του ευρετηρίου αναδιοργανώνεται περιοδικά, οπότε το αρχείο υπερχείλισης συγχωνεύεται με αυτό.

(2) Αρχικά, θα γίνει αναζήτηση στο ευρετήριο για να βρεθεί το block το οποίο περιέχει την εγγραφή με τη συγκεκριμένη τιμή του Α. Γι' αυτό θα διαβαστούν $\lceil \log_2 12 \rceil = 4$ block. Στη συνέχεια θα ακολουθηθεί ο δείκτης της συγκεκριμένης εγγραφής, και θα διαβαστούν τα 4 block του ενδιάμεσου επιπέδου, ώστε στο τέταρτο (που έχει κενό) να γίνει η ενημέρωση για τον νέο δείκτη. Γι' αυτό θα διαβαστούν 4 block και θα εγγραφεί 1 block.

Το συνολικό κόστος ενημέρωσης του ευρετηρίου είναι $4 + 4 + 1 = 9$ I/O.

(β) Κάθε διαφορετική τιμή του κλειδιού αναζήτησης (Α) υπάρχει μια φορά σε κάποιο από τα φύλλα του B+-δέντρου και ο αντίστοιχος δείκτης δείχνει στο block του επιπλέον επιπέδου που περιέχει δείκτες προς εγγραφές με αντίστοιχη τιμή στο κλειδί αναζήτησης. Οι δείκτες προς εγγραφές με ίδια τιμή στο κλειδί αναζήτησης βρίσκονται στο ίδιο block, και αν είναι τόσο πολλές οι εγγραφές ώστε οι δείκτες να μην χωρούν σε ένα block, χρησιμοποιείται μια

συνδεδεμένη λίστα από block. Οι δείκτες προς εγγραφές με διαφορετική τιμή στο κλειδί αναζήτησης είναι σε διαφορετικές λίστες από block.

(i) Έστω ότι σε κάθε κόμβο του B+-δέντρου υπάρχουν n τιμές κλειδιού και n+1 δείκτες. Κάθε κόμβος αντιστοιχεί σε ένα block, οπότε θα πρέπει να ισχύει $n \times 8 + (n + 1) \times 16 \leq 2048$, άρα $n = 84$. Άρα κάθε κόμβος του δέντρου χωρά 84 τιμές κλειδιού και 85 δείκτες. Εφόσον υπάρχουν 1000 διαφορετικές τιμές του Α, το δέντρο είναι όσο το δυνατόν πιο γεμάτο και στα φύλλα υπάρχει μια φορά κάθε μια διαφορετική τιμή του Α απαιτούνται $\lceil 1000/84 \rceil = 12$ κόμβοι φύλλα. Ένας κόμβος-ρίζα είναι αρκετός για να δείχνει στα 12 φύλλα. Συνεπώς το δέντρο έχει 2 επίπεδα. Μέχρι στιγμής το ευρετήριο χρειάζεται 13 block.

Όσον αφορά το ενδιάμεσο επίπεδο, ο υπολογισμός είναι ίδιος με το ερώτημα (α), οπότε πάλι απαιτούνται 4000 block για αυτό.

Το συνολικό μέγεθος του ευρετηρίου είναι 4013 block.

(ii) Αρχικά θα διαβαστούν 2 block, η ρίζα και το κατάλληλο φύλλο. Στη συνέχεια, θα διαβαστούν τα 4 block του ενδιάμεσου επιπέδου που περιέχουν τους δείκτες προς τις εγγραφές με A = α. Το κόστος αναζήτησης του ευρετηρίου είναι $2 + 4 = 6$ I/O.

Θεωρώντας ίδιες περιπτώσεις με το υποερώτημα (α)(ii), το συνολικό κόστος αναζήτησης είναι $6 + 500 = 506$ I/O ή $6 + v$ I/O.

(iii) (1) Εφόσον η τιμή του Α δεν εμφανίζεται ήδη σε άλλη πλειάδα, θα πρέπει να καταχωρηθεί η συγκεκριμένη τιμή κλειδιού αναζήτησης στο κατάλληλο φύλλο. Στα φύλλα του ευρετηρίου χωρούν συνολικά $84 \times 12 = 1008$ κλειδιά. Είναι ήδη καταχωρημένα 1000 κλειδιά, οπότε υπάρχουν μερικά κενά.

Έστω ότι η νέα τιμή πέφτει σε γεμάτο φύλλο. Θα γίνει αναζήτηση στο ευρετήριο για να βρεθεί το φύλλο-block στο οποίο πρέπει να γίνει η εισαγωγή. Απαιτείται να διαβαστεί η ρίζα και το κατάλληλο φύλλο, άρα 2 I/O. Θα γίνει διάσπαση του φύλλου, οπότε απαιτείται 1 write για την ενημέρωση του φύλλου, 1 write για την εγγραφή του νέου φύλλου και 1 write για την ενημέρωση της ρίζας, δηλαδή 3 I/O. Θα πρέπει να ενημερωθεί και το ενδιάμεσο επίπεδο δεικτών, όπου θα δεσμευτεί ένα νέο block δίσκου, όπου θα εγγραφεί η κατάλληλη καταχώρηση δείκτη (1 write).

Το συνολικό κόστος ενημέρωσης του ευρετηρίου είναι $2 + 3 + 1 = 6$ I/O.

Έστω ότι η νέα τιμή πέφτει σε φύλλο με κενό. Θα γίνει αναζήτηση για να βρεθεί το κατάλληλο φύλλο-block, άρα 2 I/O. Θα γίνει 1 write για την ενημέρωση του εν λόγω φύλλου. Στο ενδιάμεσο επίπεδο δεικτών, θα δεσμευτεί ένα νέο block δίσκου, όπου θα εγγραφεί η κατάλληλη καταχώρηση δείκτη (1 write).

Το συνολικό κόστος ενημέρωσης του ευρετηρίου είναι $2 + 1 + 1 = 4$ I/O.

(2) Θα γίνει αναζήτηση του φύλλου που περιέχει την καταχώρηση με τη συγκεκριμένη τιμή του Α. Γι' αυτό θα διαβαστούν 2 block (η ρίζα και το κατάλληλο φύλλο). Στη συνέχεια θα ακολουθηθεί ο δείκτης της συγκεκριμένης καταχώρησης, και θα διαβαστούν τα 4 block του ενδιάμεσου επιπέδου, ώστε στο τέταρτο (που έχει κενό) να γίνει εγγραφή του νέου δείκτη.

Το συνολικό κόστος ενημέρωσης του ευρετηρίου σε I/O είναι $2 + 4 + 1 = 7$ I/O.

(γ) Το αρχείο του ευρετηρίου θα περιλαμβάνει εγγραφές της μορφής <key, pointer>, μια για κάθε εγγραφή της σχέσης R. Εφόσον το γνώρισμα Α δεν είναι κλειδί, θα υπάρχουν τόσες εγγραφές ευρετηρίου με την ίδια τιμή κλειδιού αναζήτησης (key), όσες οι εγγραφές της σχέσης R με τη συγκεκριμένη τιμή στο γνώρισμα Α.

(i) Το μέγεθος κάθε εγγραφής ευρετηρίου είναι $8 + 16 = 24$ bytes. Δεδομένου ότι κάθε block έχει μέγεθος 2048 bytes, ένα block χωρά $\lfloor 2048\ bytes/24\ bytes \rfloor = 85$ εγγραφές ευρετηρίου. Το ευρετήριο θα έχει 500000 εγγραφές, οπότε χρειάζονται $\lceil 500000/85 \rceil = 5883$ block.

Συνεπώς, το μέγεθος του ευρετηρίου είναι 5883 block.

(ii) Υπάρχουν 500 εγγραφές της σχέσης R με A=α, άρα 500 αντίστοιχες εγγραφές ευρετηρίου. Οι εγγραφές ευρετηρίου αντιστοιχούν σε $\lceil 500/85 \rceil = 6$ block το λιγότερο και το πολύ σε 7 block (ανάλογα το που βρίσκεται η πρώτη εγγραφή με τιμή α μέσα στο block). Η δυαδική αναζήτηση επιστρέφει το πρώτο block που περιέχει εγγραφή με τιμή κλειδιού αναζήτησης α. Το κόστος της δυαδικής αναζήτησης είναι $\lceil \log_2 5883 \rceil = 13$ block. Στη συνέχεια, θα πρέπει

να ανακληθούν τα υπόλοιπα 5 block (αν είναι 6 block συνολικά) ή 6 block (αν είναι 7 block συνολικά) που περιέχουν τις υπόλοιπες εγγραφές οι οποίες είναι σειριακά τοποθετημένες μετά την πρώτη. Το κόστος αναζήτησης του ευρετηρίου είναι $13 + 5 = 18$ I/O ή $13 + 6 = 19$ I/O.

Θεωρώντας ίδιες περιπτώσεις με το υποερώτημα (α)(ii), το συνολικό κόστος αναζήτησης είναι $18 + 500 = 518$ I/O ή $19 + 500 = 519$ I/O ή $18 + v$ I/O ή $19 + v$ I/O.

(δ) Η συνθήκη A>α ικανοποιείται από το 10% των εγγραφών της σχέσης R, δηλαδή $500000 \times 10\% = 50000$ εγγραφές. Δεδομένου ότι οι εγγραφές είναι ομοιόμορφα κατανεμημένες ως προς τις τιμές του A, θα ισχύει ότι $1000 \times 10\% = 100$ διαφορετικές τιμές του A αντιστοιχούν στο διάστημα A>α.

Χρησιμοποιώντας το ευρετήριο του ερωτήματος (α):

Με δυαδική αναζήτηση θα βρεθεί το block του ευρετηρίου στο οποίο βρίσκεται η εγγραφή με τιμή κλειδιού αναζήτησης α. Για τη δυαδική αναζήτηση θα προσπελαστούν $\lceil \log_2 12 \rceil = 4$ block. Όλες οι επόμενες εγγραφές στο ευρετήριο έχουν κλειδιά αναζήτησης μεγαλύτερα του α και ξέρουμε ότι θα είναι 100. Αυτές αντιστοιχούν στα δύο τελευταία block του ευρετηρίου. Το 1 block επιστρέφεται από τη δυαδική αναζήτηση και για το επόμενο απαιτείται 1 I/O επιπλέον. Ακολουθώντας τους δείκτες αυτών των εγγραφών ευρετηρίου θα διαβαστούν τα 4 block του ενδιάμεσου επιπέδου που αντιστοιχούν σε κάθε κλειδί αναζήτησης μεγαλύτερο του α. Συνολικά θα διαβαστούν $4 \times 100 = 400$ block του επιπλέον επιπέδου. Το κόστος αναζήτησης του ευρετηρίου είναι $4 + 1 + 400 = 405$ I/O.

Αφού υπάρχουν 50000 εγγραφές της R με A>α, θεωρώντας ίδιες περιπτώσεις με το υποερώτημα (α)(ii), το συνολικό κόστος αναζήτησης είναι $405 + 50000 = 50405$ I/O ή $405 + v$ I/O.

Χρησιμοποιώντας το ευρετήριο του ερωτήματος (β):

Θα διαβαστεί η ρίζα και το κατάλληλο φύλλο του B+-δέντρου, στο οποίο βρίσκεται η καταχώρηση με τιμή κλειδιού αναζήτησης α. Οπότε, οι υπόλοιπες καταχωρήσεις στο εν λόγω φύλλο και σε όλα τα δεξιά φύλλα-siblings έχουν κλειδιά αναζήτησης μεγαλύτερα του α. Οι καταχωρήσεις αυτές είναι 100 και αντιστοιχούν στα δύο δεξιότερα φύλλα του ευρετηρίου. Το πρώτο από τα δύο φύλλα έχει ήδη διαβαστεί (το έχουμε συνυπολογίσει κατεβαίνοντας στο δέντρο), οπότε απαιτείται να διαβαστεί και το δεξί γειτονικό του φύλλο (1 επιπλέον I/O). Οι δείκτες αυτών των φύλλων θα μας οδηγήσουν στα σωστά block του επιπλέον επιπέδου, όπου θα διαβαστούν $4 \times 100 = 400$ block. Το κόστος αναζήτησης του ευρετηρίου είναι $2 + 1 + 400 = 403$ I/O.

Θεωρώντας ίδιες περιπτώσεις με το υποερώτημα (α)(ii), το συνολικό κόστος αναζήτησης είναι $403 + 50000 = 50403$ I/O ή $403 + v$ I/O.

Χρησιμοποιώντας το ευρετήριο του ερωτήματος (γ):

Οι πρώτες 450000 εγγραφές ευρετηρίου έχουν τιμή κλειδιού αναζήτησης μικρότερη ή ίση του α. Αντιστοιχούν στα πρώτα 5295 block: τα πρώτα 5294 block είναι γεμάτα από εγγραφές ευρετηρίου με τιμή κλειδιού αναζήτησης μικρότερη ή ίση του α, και στο 5295 block θα υπάρχουν οι τελευταίες 10. Αυτές οι 10 εγγραφές θα έχουν προφανώς κλειδί ίσο με α. Οι υπόλοιπες $500 - 10 = 490$ εγγραφές με κλειδί α θα βρίσκονται στα αμέσως προηγούμενα 6 block. Δηλαδή, υπάρχουν 7 block ευρετηρίου με εγγραφές με κλειδί ίσο με α. Επίσης, το 5295 block θα περιέχει τις $85 - 10 = 75$ πρώτες εγγραφές με κλειδί μεγαλύτερο του α και τα τελευταία $5883 - 5295 = 588$ block ευρετηρίου θα περιέχουν όλες τις υπόλοιπες.

Η δυαδική αναζήτηση επιστρέφει το πρώτο block ευρετηρίου που περιέχει εγγραφή με τιμή κλειδιού αναζήτησης α. Για τη δυαδική αναζήτηση θα προσπελαστούν $\lceil \log_2 5883 \rceil = 13$ block. Στη συνέχεια θα διαβαστούν όλα τα επόμενα block του ευρετηρίου, δηλαδή $6 + 588$ block. Οπότε, το κόστος αναζήτησης του ευρετηρίου είναι $13 + 6 + 588 = 607$ I/O.

Θεωρώντας ίδιες περιπτώσεις με το υποερώτημα (α)(ii), το συνολικό κόστος αναζήτησης είναι $607 + 50000 = 50607$ I/O ή $607 + v$ I/O.