1η Σειρά Ασκήσεων Σχεδιασμός Βάσεων Δεδομένων:

Γεωργίου Αλέξιος-Λάζαρος 3180027

Άσκηση 1:

a)

Χωρητικότητα ίχνους (track):

$$Sector\ Size\ =\ 4096\ bytes, Track\ =\ 256\ sectors$$

$$Track\ Size\ =\ sectors\ *\ Sector\ size\ =\ 256\ *\ 4096\ =\ \mathbf{1,048,576\ B}\ =\ 1\ MB$$

Χωρητικότητα επιφάνειας (surface):

$$Track\ Size\ =\ 1{,}048{,}576\ B, Surface\ =\ 65{,}536\ tracks$$

$$Surface\ Size\ =\ Track\ Size\ *\ tracks\ =\ 1{,}048{,}576\ *\ 65{,}536\ =\ \mathbf{68,719,476,736\ B}\ =\ 64\ GB$$

Χωρητικότητα δίσκου (disk):

$$Disk\ =\ 8\ platters\ =\ 16\ surfaces$$

$$Disk\ size\ =\ Surface\ Size\ *\ surfaces\ =\ 16\ *\ 68{,}719{,}476{,}736\ =\ \mathbf{1,099,511,627,776\ B}\ =\ 1\ TB$$

b)

Οι κύλινδροι είναι ίσοι με τον αριθμό ιχνών σε μια επιφάνεια, οπότε έχουμε **65,536** κυλίνδρους στο δίσκο.

c)

Ταχύτητα περιστροφής 7200 rpm, άρα κάθε 60 sec έχουμε 7200 rotations.

Η **μέγιστη** καθυστέρηση περιστροφής είναι ο χρόνος για μια ολόκληρη περιστροφή = 60 / 7200 =

0.00833.. sec = **8.33 ms**

Η **μέση** καθυστέρηση περιστροφής είναι ο χρόνος για μισή περιστροφή άρα ο μισός της μέγιστης, **0.00416 sec** = **4.17 ms**

d)

$$Track\ size\ =\ 1\ MB, Track\ time\ =\ 0.00833..sec$$

$$Transfer\ rate\ =\frac{Track\ size}{Track\ time}=\ 120\ MBps$$

Άσκηση 2:

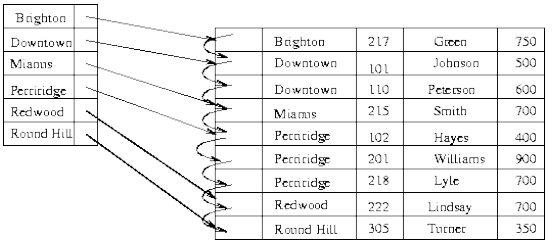
Έχουμε 128 B για κάθε εγγραφή άρα σε μια σελίδα χωράνε 1024/128 = 8 εγγραφές.

Επομένως, για την αποθήκευση της σχέσης R θέλουμε $\left\lceil \frac{N}{8} \right\rceil$ σελίδες (μπλοκ).

Στα ευρετήρια, χρειάζομαι 16B για κάθε εγγραφή τους (10 bytes για το #a και άλλα 6 για τον pointer).

Χωράνε 1024/16 = 64 εγγραφές ανά σελίδα.

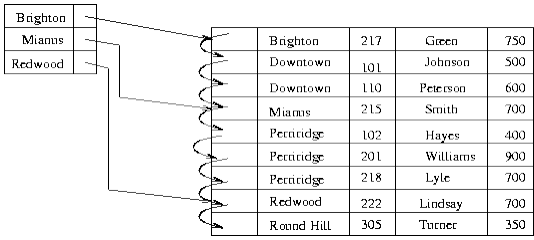
a) **Dense** Index



Στο πυκνό ευρετήριο πρέπει να φτιάξω εγγραφή για κάθε εγγραφή της σχέσης. Άρα για την αποθήκευση του ευρετηρίου θέλουμε $\left\lceil \frac{N}{64} \right\rceil$ σελίδες (μπλοκ).

Σύνολο $D(N) = \left\lceil \frac{N}{8} \right\rceil + \left\lceil \frac{N}{64} \right\rceil \cong 0.14N \; \forall \; \mathrm{N} \in [0, 2^{48}]$ σελίδες.

b) **Sparse** Index



Στο αραιό ευρετήριο μπορώ να φτιάξω όσες εγγραφές θέλω, κατά σύμβαση μια για κάθε σελίδα της σχέσης είναι καλά. Στη σχέση έχουμε $\left\lceil \frac{N}{8} \right\rceil$ σελίδες.

Άρα για την αποθήκευση του ευρετηρίου θέλουμε $\left\lceil \frac{\left\lceil \frac{N}{8} \right\rceil}{64} \right\rceil$ σελίδες (μπλοκ).

Σύνολο $S(N) = \left\lceil \frac{N}{8} \right\rceil + \left\lceil \frac{\left\lceil \frac{N}{8} \right\rceil}{64} \right\rceil \cong 0.1269N \; \forall \; \mathrm{N} \in [0, 2^{58}]$ σελίδες.

Το αραιό χρειάζεται λιγότερο χώρο και είναι ευκολότερο το update του, αλλά θέλει περισσότερη ώρα για την εύρεση των εγγραφών της σχέσης.

Ο περιορισμός του N είναι οι συνδυασμοί των εγγραφών που μπορεί να περιγράψει ο pointer.

Άσκηση 3:

1) Αναζήτηση της εγγραφής με κλειδί 41.

- Ξεκινάμε από την ρίζα του δέντρου, βλέπουμε ότι 41 > 13.
- Κατευθυνόμαστε στον δεξιά κόμβο (πάντα μέσω του pointer της ρίζας/μαύρη κουκίδα).
- Βρίσκουμε σε ποιο διάστημα είναι (ελέγχουμε αν 41<23, δεν είναι, μετά 41 < 31, μετά 41<43 ισχύει).
- Ακολουθούμε τον pointer προς τον επόμενο κόμβο (31,37,41).
- Εξετάζουμε τα στοιχεία του κόμβου με τη σειρά και βρίσκουμε το 41 στο τέλος.
- Ακολουθούμε τον pointer προς την εγγραφή του 41 (άσπρη κουκίδα).

2) Αναζήτηση της εγγραφής με κλειδί 40.

- Ξεκινάμε από την ρίζα του δέντρου, βλέπουμε ότι 40 > 13.
- Κατευθυνόμαστε στον δεξιά κόμβο.
- Βρίσκουμε σε ποιο διάστημα είναι (ελέγχουμε αν 40 < 23, δεν είναι, μετά 40 < 31, μετά 40 < 43 ισχύει).
- Ακολουθούμε τον pointer προς τον επόμενο κόμβο (31,37,41).
- Εξετάζουμε τα στοιχεία του κόμβου με τη σειρά και δεν το βρίσκουμε.
- Δεν υπάρχει η εγγραφή.

3) Αναζήτηση των εγγραφών με κλειδιά μικρότερα του 30.

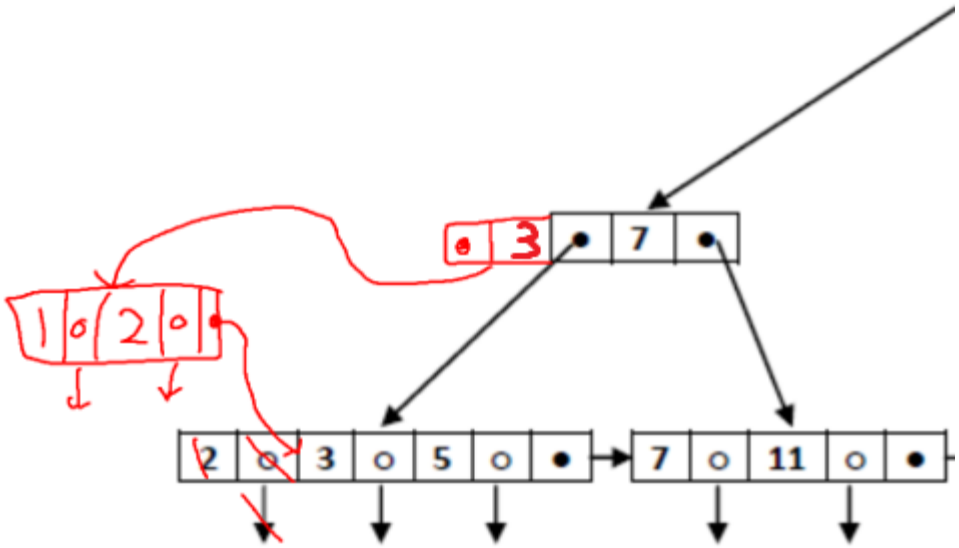
- Αρχίζουμε από την ρίζα και ακολουθούμε τον πρώτο pointer κάθε κόμβου μέχρι να φτάσουμε στο φύλλο και στην συνέχεια επιστρέφουμε τους δείκτες των εγγραφών κάθε αριθμού εάν αυτός είναι μικρότερος του 30, σταματάμε την πρώτη φορά που η σύγκριση δεν θα ισχύει. Χρησιμοποιούμε τους pointer κάθε κόμβου φύλλου στον επόμενο κόμβο φύλλου.
- Η διαδρομή που θα γίνει είναι (13),(7),(2,3,5)*, (7,11)*,(13,17,19)*,(23,29)* (οι κόμβοι με αστερίσκο, σημαίνει ότι επιστρέφουμε τους δείκτες εγγραφών τους).

4) Αναζήτηση των εγγραφών με κλειδιά στο διάστημα [20,35].

- Αναζητούμε το 20, ακολουθείτε η διαδρομή (13), (23,31,43), (13,17,19)
- Ελέγχουμε κάθε στοιχείο του κόμβου (13,17,19) αν ανήκει στο διάστημα (δεν επιστρέφεται τίποτα).
- Προχωράμε στον επόμενο κόμβο (23,29)* και επιστρέφουμε κάθε αριθμό ελέγχοντας μόνο το αν είναι μικρότερος του 35. Ακολουθούμε τους επόμενους κόμβους μέχρι να συναντήσουμε το πρώτο αριθμό που δεν ισχύει. Εδώ θα επιστρέψουμε και το 31, θα δούμε ότι το επόμενο είναι το 37 και θα τερματίσουμε.

5) Εισαγωγή του κλειδιού με τιμή 1.

- Κάνουμε αναζήτηση για τη τιμή 1.
- Το 1 θα έμπαινε στον (2,3,5) κόμβο αλλά δεν έχει χώρο.
- Φτιάχνουμε έναν καινούργιο κόμβο αριστερά και βάζουμε το 1 και μεταφέρουμε το 2 εκεί (σαν να σπάμε τον κόμβο με τα 4 στοιχεία σε 2 κόμβους με 2).
- Προσθέτουμε στον κόμβο 7 και το 3 και γίνεται (3,7).



6) Διαδοχική εισαγωγή τριών κλειδιών με τιμές 14,15 και 16 αντίστοιχα.

Υποθέτω ότι το δέντρο είναι πάλι το αρχικό (χωρίς την εισαγωγή του 1). Σε κάθε σπάσιμο κόμβου οι pointers των εγγραφών εννοείται ότι μεταφέρονται. Τα χρώματα συμβολίζουν τις αλλαγές.

Insert 14:

- Κάνουμε αναζήτηση για το 14.
- Ο κόμβος είναι γεμάτος, σπάμε τον (13,17,19) σε (13,14) και (17,19).
- Το 17 πρέπει να μπει στον (23,31,43) ο οποίος είναι γεμάτος οπότε, τον σπάμε σε (17,23) και (43).
- Προσθέτουμε το 31 στον (13) και τώρα η ρίζα γίνεται (13,31).

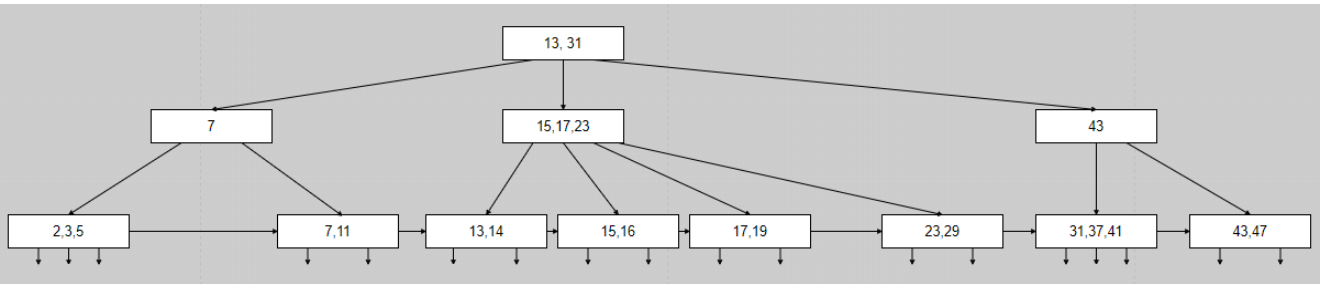
Insert 15:

- Κάνουμε αναζήτηση για το 15.
- Μεταφερόμαστε στον κόμβο (13,14).
- Προσθέτουμε στον κόμβο την εγγραφή με αριθμό 15 (χωράει).

Insert 16:

- Κάνουμε αναζήτηση για το 16.
- Ο κόμβος (13,14,15) είναι γεμάτος, τον σπάμε σε (13,14) και (15,16).
- Προσθέτουμε το 15 στον κόμβο (17,23), χωράει.

Το τελικό διάγραμμα:



Άσκηση 4:

Σε οποιοδήποτε κόμβο σε ένα B+ δέντρο έχω n κλειδιά θα έχω n+1 pointers.

Ξέρω ότι η σελίδα είναι 2048 B, μέγεθος pointer 12 B, μέγεθος κλειδιού αναζήτησης 8 B.

$$8n + 12(n + 1) = 2048 \leftrightarrow 20n = 2036 \leftrightarrow [n] = 101$$

$$Άρα\ έχουμε\ 101\ κλειδία\ και\ 102\ pointers\ σε\ κάθε\ κόμβο.$$

Επειδή έχουμε 3 επίπεδα, οι pointers στο τελευταίο επίπεδο (στα φύλλα του δέντρου) θα είναι το πολύ 102 * 102 * 101 = **1,050,804 pointers/ αριθμός εγγραφών της σχέσης R.**

Έχω 102 pointers στη ρίζα, 102 pointers σε κάθε κόμβο στο ενδιάμεσο επίπεδο, άρα 102 * 102 κόμβους στα φύλλα που ο καθένας έχει 101 κλειδιά/pointers.

Άσκηση 5:

1. Εισαγωγή 0000

0000	
0	**1**

U = 1/6

2. Εισαγωγή 0001

0000	0001
0	**1**

U = 1/3

3. Εισαγωγή 0001

	0001
0000	0001
0	**1**

U = ½

4. Εισαγωγή 0101

	0101
	0001
0000	0001
0	**1**

U = 2/3

5. Εισαγωγή 0111

Overflow
0111

	0101
	0001
0000	0001
0	**1**

U = 5/9

6. Εισαγωγή 0010

Overflow
0111

	0101
0010	0001
0000	0001
0	**1**

U = 6/9

7. Εισαγωγή 0111

Overflow
0111
0111

	0101
0010	0001
0000	0001
0	**1**

U = 7/9 > 0.7

Αυξάνουμε το i:

Overflow
0111
0111

	0101	
	0001	
0000	0001	0010
00	***1**	**10**

U = 7/12

8. Εισαγωγή 0010

Overflow
0111
0111

	0101	
	0001	0010
0000	0001	0010
00	***1**	**10**

U = 8/12

9. Εισαγωγή 0011

Overflow
0011
0111
0111

	0101	
	0001	0010
0000	0001	0010
00	***1**	**10**

U = 9/12 = ¾ > 0.7

Αυξάνουμε το i:

Overflow
~~0011~~
~~0111~~
~~0111~~

	0101		0011
	0001	0010	0111
0000	0001	0010	0111
00	**01**	**10**	**11**

U = 9/12

Αυξάνουμε το i:

	0101		0011	
	0001	0010	0111	
0000	0001	0010	0111	
000	***01**	***10**	***11**	**100**

U = 9/15

10. Εισαγωγή 0100

	0101		0011	
	0001	0010	0111	
0000	0001	0010	0111	0100
000	***01**	***10**	***11**	**100**

U = 10/15 = 2/3

Άσκηση 6:

a) 2^{10} $buckets$ $(για\ κάθε\ νέο\ βάθος\ διπλάσια\ buckets)$

Έχουμε 2^{10} $εγγραφές\ στο\ ευρετήριο$ $(μια\ για\ κάθε\ bucket)$.

b) $4 * 2^{10}$ $B, 4B$ $ανά\ bucket$

c) Στο μέγιστο χωράνε, 2400 bytes ανά bucket / 400 bytes ανά εγγραφή = **6 εγγραφές ανά bucket**
, 2^{10} $buckets$

$$6 * \ 2^{10}\ εγγραφές\ δεδομένων.$$

Άσκηση 7:

Το σκεπτικό του σχεδιαστή είναι λάθος. Ο μισθός των εργαζομένων δεν είναι ομοιόμορφα κατανεμημένος. Υπάρχουν πολλά άτομα στην κλίμακα των [1000,2000) και πολύ λίγα άτομα στην κλίμακα των [10000,11000).

Γενικά ο μισθός είναι περίπου κανονικά κατανεμημένος γύρω από κάποια μέση τιμή, δεν είναι ομοιόμορφα κατανεμημένος.

Με αυτήν την συνάρτηση κατακερματισμού ο αριθμός των εγγραφών σε κάθε κάδο δεν θα είναι περίπου ίδιος αλλά θα έχουμε τεράστιες διαφορές.