

1η Άσκηση Σ.Δ.Α.Δ

Τσιτζος Ιωάννης
P3200211

Άσκηση (1)

1. Συνολική Χωρητικότητα Δίσκου = $10 \cdot 10.000 \cdot 1.000 \cdot 512 = 512 \cdot 10^8 = \boxed{51,2 \text{ GB}}$

↑ επιμέρειες
↓ κίτρινες
↑ τογείς/ίχτες
↓ bytes/τογίος

2. Μέση Καθυστερήση Περίστροφης (Average Rotation Latency)

↳ Υπολογίζεται ως ένα μισό Περίστροφης του Δίσκου

Έχουμε :

60 sec	10.000 rpm
R	$\frac{1}{2}$ rpm

$$30 = 10.000 R \Rightarrow R = \frac{30}{10.000} \Rightarrow R = 3 \cdot 10^{-3} \text{ sec} \Rightarrow \boxed{R = 3 \text{ msec}}$$

3. Χρόνος Μεταφοράς ενός Μπλόκ (Block transfer Time)

Έχουμε :

- 10.000 rpm
- 1000 sectors/track
- 8 sectors/block

Για 1 Περίστροφή διαβάζω 1000 sectors

Για να διαβάσω 1 block = 8 διαδοχικοί sectors θέλω :

$$\left(\frac{8}{1000} \right) \cdot \left(\frac{60}{10.000} \right) = 0,000048 \text{ sec} = \boxed{0,048 \text{ msec}}$$

4.) Συνολικός Χρόνος για Διάβασμα του αρχείου =

$$= (\text{Block Transfer Time} + \text{Average Seek Time}) \cdot \# \text{ Blocks} =$$

$$= (0,048 + 8) \cdot 10^6 = 8048 \cdot 10^3 \text{ ms} = \boxed{8048 \text{ sec}}$$

5.) 1 κόμβος αντιστοιχεί σε 1 block

3 επίπεδα

N τυχαίες εγγραφές

- 3ο Επίπεδο $\rightarrow N$ κόμβοι
- 2ο Επίπεδο $\rightarrow \left\lceil \frac{N}{\lceil N+1/2 \rceil} \right\rceil$ κόμβοι
- 1ο Επίπεδο (Ρίζα) $\rightarrow 1$ κόμβος

$$\text{Blocks} = N + \frac{2N}{N+1} + 1$$

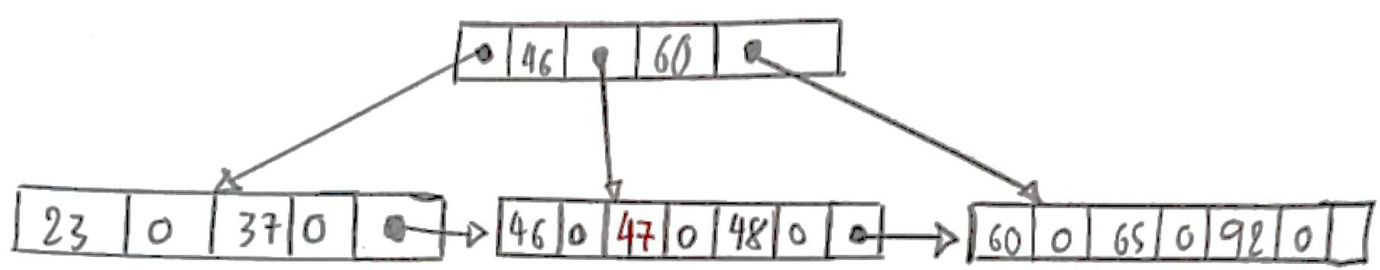
$$\text{Total Transfer Time} = 8,048 \cdot \left(N + \left\lceil \frac{2N}{N+1} \right\rceil + 1 \right) \text{ ms}$$

1. Η εντολή ενημέρωσης UPDATE δεν επηρεάζεται από την ύπαρξη των ευρετηρίων, καθώς δεν αλλάζει το πεδίο empid που είναι το πεδίο του clustered index θείσιν με την αύξηση του age όλων των εγγραφών. Δεν επηρεάζεται η ταξινόμηση τους στον πίνακα. Το άλλο ευρετήριο του age επίσης δε θα επηρεαστεί, αφού οι αλλαγές αφορούν ^{όλα} τα δεδομένα του πίνακα και όχι τη δομή του ευρετηρίου.

2. Η εντολή αυτή επηρεάζεται από την ύπαρξη του clustered index στο πεδίο empid, καθώς το ερώτημα χρησιμοποιεί το πεδίο empid για να περιορίσει τις εγγραφές που θα αλλάξουν. Επομένως, η ύπαρξη του ευρετηρίου επιταχύνει την ενημέρωση.

3. Η εντολή αυτή δεν περιλαμβάνει τα πεδία empid και age, καθώς ενημερώνει το πεδίο salary με βάση το departmentid του κάθε υπαλλήλου (πρέπει να είναι 10).
Άρα δεν επηρεάζεται από την ύπαρξη ευρετηρίων.

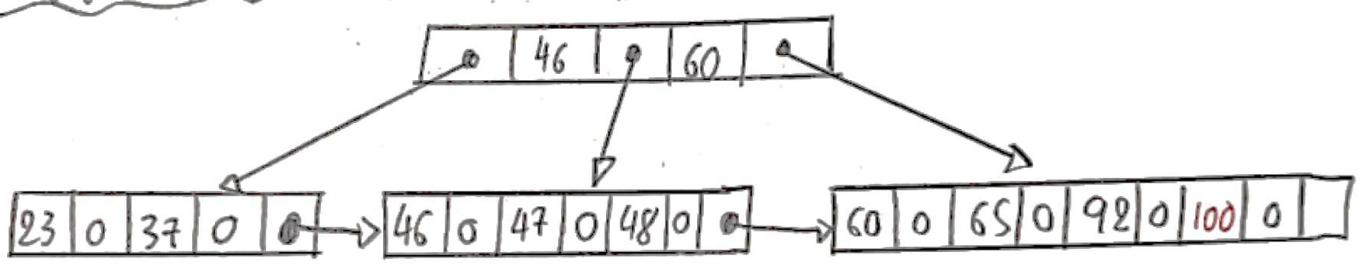
Άσκηση (3)



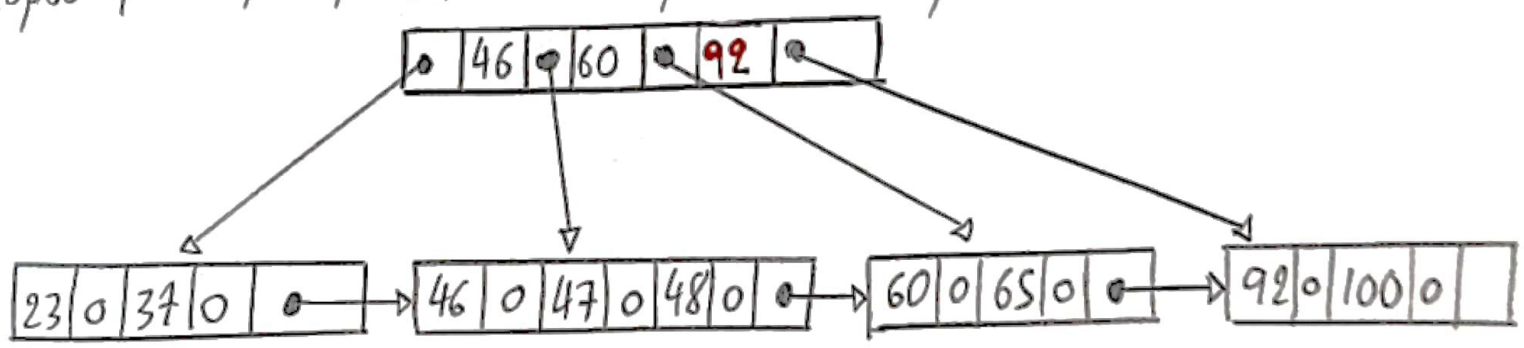
Εισαγωγή τιμής 47

Η τιμή 47 τοποθετείται στον 2ο κόμβο-φύλλο μετά το 46.
 Εφόσον $n=3$ και ο κόμβος-φύλλο περιέχει 2 τιμές χωράει και η τιμή 47
 ανάμεσα στις 46 και 48.

Εισαγωγή τιμής 100

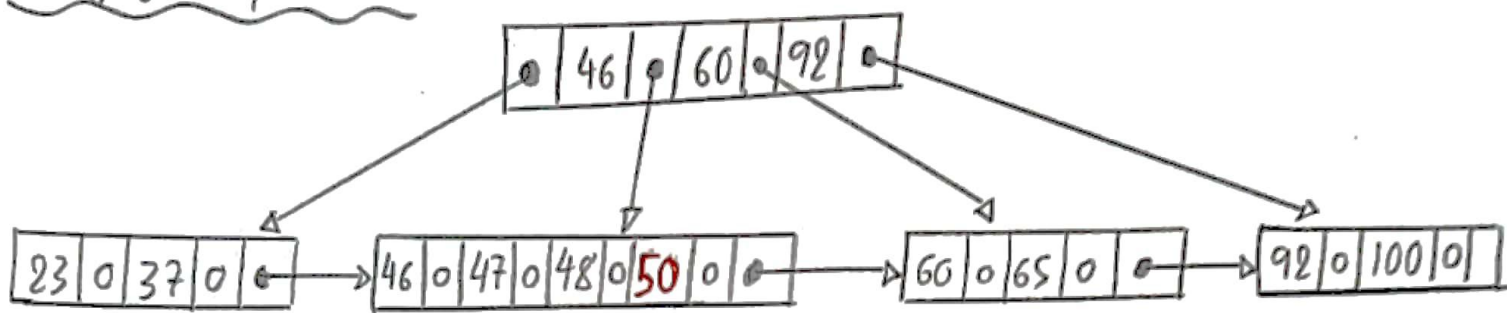


Η τιμή 100 τοποθετείται στον 3ο κόμβο φύλλο μετά το 92.
 Όμως, η χωρητικότητα του κόμβου είναι $n=3$, άρα η εισαγωγή του 100
 προκαλεί υπερχείλιση. Θα δημιουργηθεί νέος κόμβος, όπου θα καταχωρηθούν οι
 τιμές 92 και 100, ενώ ταυτόχρονα η τιμή 92 θα ανησυχράει στον
 κόμβο ρίζα μαζί με έναν δείκτη προς το νέο κόμβο



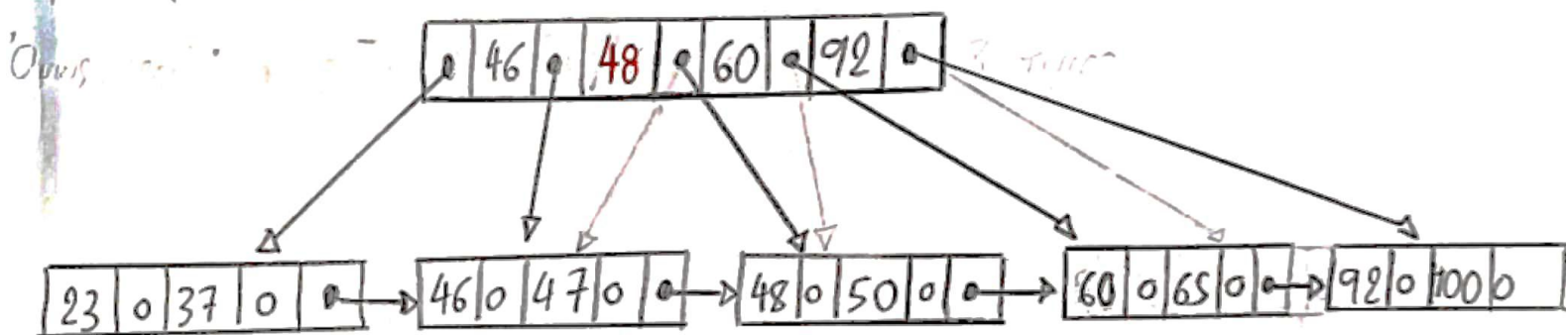
Επίσημη (3) (συνέχεια)

Εισαγωγή τιμής 50

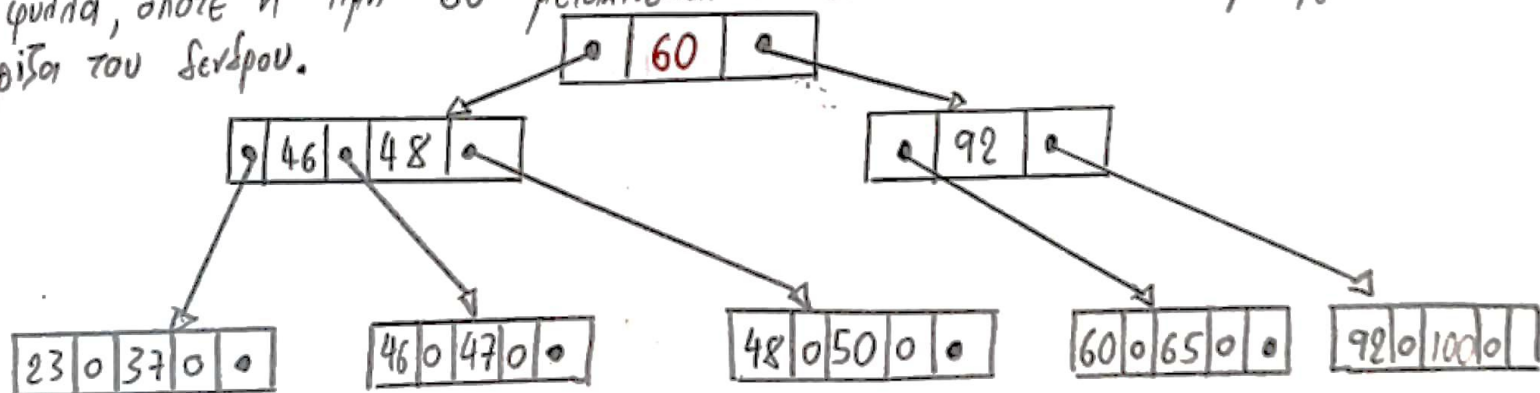


Η τιμή 50 τοποθετείται στον 2ο κόμβο-φύλλο μετά το 48.

Όμως, η χωρητικότητα του κόμβου είναι $n=3$, άρα η εισαγωγή του 50 προκαλεί υπερχείλιση. Θα δημιουργηθεί νέος κόμβος, όπου θα καταχωρηθούν οι τιμές 48 και 50. Ταυτόχρονα, η τιμή 48 θα αντιγραφεί στον κόμβο ρίζα μαζί με ένα δείκτη προς το νέο κόμβο.



Επειδή η ρίζα του δέντρου περιέχει ήδη 3 τιμές θα υπερχείλισει. Άρα θα δημιουργήσουμε ένα νέο κόμβο, ο οποίος θα πάρει τιμές 60 και 92. Στο νέο κόμβο χρειαζόμαστε μια τιμή και 2 δείκτες που θα δείχνουν στα φύλλα, οπότε η τιμή 60 μετακινείται ένα επίπεδο πάνω και δημιουργεί τη ρίζα του δέντρου.



Ασκηση (3)

2.) ($key \geq 46$ AND $key \leq 65$)

a) κλειδί Αναζήτησης Μοναδικό

$[60] \rightarrow [46, 48] \rightarrow [46, 47] \rightarrow [48, 50] \rightarrow [60, 65]$ 5 κόμβοι

b) κλειδί Αναζήτησης ΟΧΙ Μοναδικό (Υπάρχουν Διηλοζυμια)

$[60] \rightarrow [46, 48] \rightarrow [46, 47] \rightarrow [48, 50] \rightarrow [60, 65] \rightarrow [92, 100]$ 6 κόμβοι

Άσκηση (4)

Για την αποθήκευση κάθε εγγραφής της σχέσης R χρειαζόμαστε 5 blocks.

Αρα, για αρχή, έχω $\frac{1000}{5} = 200$ blocks

Το ύψος του δέντρου είναι $h_{tree} = \frac{\log_{10}(1000)}{\log_{10}(10)} = 3$

Για την εύρεση του πλήθους κόμβων στο επίπεδο $n=2$ έχω:

$$\frac{1000}{10^2} = 10$$

Αρα για το 3ο επίπεδο έχουμε $10 \cdot 10$ blocks = 100

Για το 1ο επίπεδο (ρίζα) έχουμε 1

Αρα θέλουμε $10 + 100 + 1 = 111$ για ελάχιστη αποθήκευση στο B+ δέντρο

Συνολικά ελάχιστος αριθμός μπλοκ = $200 + 111 = \boxed{311 \text{ blocks}}$

Ασκηση (5)

1.)

$i=1$

0
1

1
0101
0111
0110

1
1111

Στο σημείο αυτό θέλουμε να εισάγουμε την τιμή 0000 όμως ο κόβος υπερχειλίζει

Αρα συνεχίζουμε ως εξής:

$i=2$

00
01
10
11

2
0000

2
0101
0111
0110

1
1111

Στο σημείο αυτό θέλουμε να εισάγουμε την τιμή 0100 όμως ο κόβος υπερχειλίζει

$i=3$

000
001
010
011
100
101
110
111

3
0000
0001

3
0011
0010

3
0111
0110

3
0101
0100

1
1111
1000
1010

- 9.) Για να γίνει συνδιασμός του ευρετηρίου πρέπει να γίνει υπερχειρίση κάρου με τονικό βάθος 3.

Εφόσον επιτρέπονται τα συνότυνα, αν στον 1ο κάρδο προσθέσουμε τις τιμές 0000, 0001 θα υπερχειρίσει και το ευρετήριο πλέον θα πρέπει να έχει $i=4$. Ο κάρδος θα δηλώνει δε 2

κάρδους : Ο ένας θα έχει τις 2 0000 τιμές (τονικό βάθος 4)
Ο άλλος θα έχει τις 2 0001 τιμές (τονικό βάθος 4,

Ο ελάχιστος αριθμός τιμών λοιπόν είναι 2

Άσκηση 6

Η πιθανότητα υπερχείλισης ενός κώδου εξαρτάται από τον αριθμό κλειδών n , το μέγεθος του ερευτηρίου και τη συνάρτηση κατακερματισμού. Ας υποθέσουμε ότι κάθε κλειδί έχει ίση πιθανότητα να καταλήξει σε οποιονδήποτε κώδο.

Ο αριθμός των κλειδών είναι n και ο αριθμός των κώδων είναι m .

$$\text{Πιθανότητα υπερχείλισης κώδου} = \left(\frac{1}{m}\right)^{n-1}$$

Η πιθανότητα μειώνεται όσο αυξάνεται ο αριθμός των κώδων. Όταν ο αριθμός των κλειδών είναι πολύ μεγάλος σε σχέση με τον αριθμό των κώδων, η πιθανότητα αυξάνεται.