



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΚΑΙ ΑΝΑΛΥΣΗΣ ΔΕΔΟΜΕΝΩΝ

1^η Σειρά Ασκήσεων

Ονοματεπώνυμο:

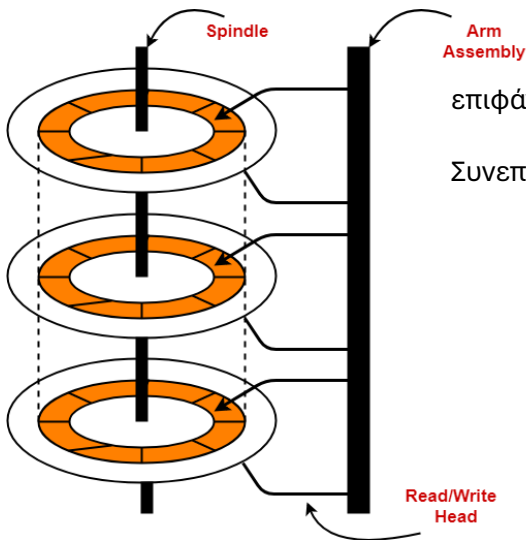
Μαρία Σχοινάκη

Αριθμός Μητρώου:

3210191

Email:

p3210191@aueb.gr

Άσκηση 1

1) Εφόσον ο δίσκος αποτελείται από 4 πλακέτες διπλής όψης, αυτό σημαίνει ότι ο δίσκος συνολικά έχει 8 επιφάνειες (surfaces), αφού κάθε πλακέτα έχει 2 επιφάνειες. (2*4)

Συνεπώς:

$$\begin{aligned} \text{Capacity} &= \text{surfaces} * \frac{\text{tracks}}{\text{surface}} * \frac{\text{sectors}}{\text{track}} * \frac{\text{KB}}{\text{sector}} \\ \Leftrightarrow 80 \text{ GB} &= 8 * 512 * 1024 * \frac{\text{KB}}{\text{sector}} \\ \Leftrightarrow 10 \text{ GB} &= 512 * 1024 * \frac{\text{KB}}{\text{sector}} \\ \Leftrightarrow 10.485.760 \text{ KB} &= 524.288 * \frac{\text{KB}}{\text{sector}} \\ \Leftrightarrow \frac{\text{KB}}{\text{sector}} &= \frac{10.485.760 \text{ KB}}{524.288} \\ \Leftrightarrow \frac{\text{KB}}{\text{sector}} &= 20 \text{ KB} \end{aligned}$$

2) Η σχέση R, καταλαμβάνει 327.680 KB του δίσκου (40KB * 8192). Κάθε κύλινδρος έχει χωρητικότητα:

$$\begin{aligned} \text{Cylinder capacity} &= \text{surfaces} * \text{track capacity} \\ &= \text{surfaces} * \text{sectors per track} * \text{sector capacity} \\ &= 8 * 1024 * 20 \text{ KB} \\ &= 163.840 \text{ KB} \end{aligned}$$

και αφού η σχέση R είναι αποθηκευμένη σε συνεχόμενα μπλοκ του δίσκου,

$$\text{Cylinders for R} = \frac{\text{R to KB}}{\text{cylinder capacity}} = \frac{327.680 \text{ KB}}{163.840 \text{ KB}} = 2$$

3) Έχουμε ότι η ταχύτητα περιστροφής είναι 7200 rpm. Άρα: $\frac{7200}{60} = 120 \text{ rps}$, οπότε

$$\text{Rotation Delay} = \frac{1}{120} = 0.0083 \text{ s και}$$

$$\text{Average Rotation Delay} = \frac{\text{Rotation Delay}}{2} = \frac{0.0083 \text{ s}}{2} = 0.00415 \text{ s} = 4.15 \text{ ms}$$

Επίσης, από προηγούμενο ερώτημα (2), βρήκαμε ότι η σχέση R καταλαμβάνει 2 κυλίνδρους.

Για το διάβασμα ενός ίχνους χρειάζονται $60 / 7200 = 0,0083 \text{ s}$. Για το διάβασμα ενός κυλίνδρου απαιτούνται $0,0083 * 8 = 0,0664 \text{ s}$. Για το διάβασμα των 2 κυλίνδρων απαιτούνται $2 * 0,0664 = 0,1328 = 132,8 \text{ ms}$

$$\begin{aligned} T &= \text{Average Seek Time} + \text{Average Rotation Delay} + \text{Read Blocks Of R Time} \\ &= 8 \text{ ms} + 4.15 \text{ ms} + 132,8 \text{ ms} \\ &= 12,15 \text{ ms} + 132,8 \text{ ms} \\ &= 144,95 \text{ ms} \end{aligned}$$

4) Το ελάχιστο δυνατό μέγεθος ενός block είναι 40 KB, ώστε να μπορεί να χωράει μία ολόκληρη εγγραφή, όπως αναφέρει η εκφώνηση. Επίσης, έχουμε ότι 1 sector είναι 20 KB (1). Οπότε, 1 block αποτελείται από $\frac{40 \text{ KB}}{20 \text{ KB}} = 2 \text{ sectors}$. Σε $\frac{60}{7200} = 0,0083 \text{ s}$, γίνεται μια περιστροφή και η κεφαλή περνάει από 1024 sectors.

Οπότε, για 1 block, απαιτούνται $\frac{8,3 \text{ ms}}{1024} * 2 = 0,01621 \text{ ms}$

Time to read one random block = Average Seek Time + Average Rotation Delay + Transfer Time

$$= 8 + 4,15 + 0,01621 \text{ ms}$$

$$= 12,16624 \text{ ms}$$

Άρα ο χρόνος ανάγνωσης 100 block είναι:

Time to read one random block * 100

$$= 12,16624 * 100$$

$$= 1,216624 \text{ s}$$

Άσκηση 2

1) Το μέγεθος μιας εγγραφής στο αρχείο είναι το άθροισμα των μεγεθών των πεδίων.

Άρα μέγεθος εγγραφής = $9 + 30 + 10 + 9 + 8 + 1 + 40 + 4 + 4 = 115 \text{ bytes}$.

Γνωρίζουμε ότι κάθε μπλοκ έχει 1024 bytes, άρα σε ένα μπλοκ χωράνε $\lfloor \frac{1024}{115} \rfloor = 8,9$ άρα 8 εγγραφές.

Το πλήθος των καταχωρήσεων του πρώτου επιπέδου του ευρετηρίου είναι το πλήθος των μπλοκ που χρειάζονται για την αποθήκευση όλων των εγγραφών (60.000) του αρχείου.

Άρα $\frac{60.000}{8} = 7.500$ καταχωρήσεις.

2) Το μέγεθος κάθε καταχώρησης στο αρχείο του ευρετηρίου είναι το άθροισμα του πεδίου κλειδιού διάταξης αρχείου και του μήκους του δείκτη μπλοκ. (R_i)

Άρα είναι ίσο με: $9 + 6 = 15 \text{ bytes}$. (δεδομένα εκφώνησης)

Το πλήθος των μπλοκ του πρώτου επιπέδου του ευρετηρίου είναι $\lceil \frac{r_1}{f_0} \rceil$.

Όπου, r_1 το πλήθος των καταχωρήσεων του πρώτου επιπέδου του ευρετηρίου (ερώτημα 1) και f_0 ο παράγοντας διακλάδωσης, που είναι ίδιος με τον παράγοντα ομαδοποίησης. ($f_0 = bfr_i$)

Έχουμε $bfr_i = \lfloor \frac{B}{R_i} \rfloor$, όπου B = μέγεθος μπλοκ και R_i = μέγεθος καταχώρησης στο αρχείο του ευρετηρίου.

$$bfr_i = \lfloor \frac{B}{R_i} \rfloor = \lfloor \frac{1024}{15} \rfloor = \lfloor 68,2 \rfloor = 68 = f_0 \text{ καταχωρήσεις ανα μπλοκ.}$$

Άρα το πλήθος των μπλοκ του πρώτου επιπέδου του ευρετηρίου είναι $\lceil \frac{r_1}{f_0} \rceil = \lceil \frac{7.500}{68} \rceil = \lceil 110,294 \rceil = 111$.

3) Το πλήθος των επιπέδων του ευρετηρίου είναι τόσο όσο μέχρι όλες οι εγγραφές του ευρετηρίου να χωράνε σε 1 μπλοκ.

Άρα πρέπει να υπολογίζουμε τον αριθμό των μπλοκ κάθε επιπέδου, μέχρι να βρούμε επίπεδο με πλήθος μπλοκ = 1.

- Αριθμός μπλοκ πρώτου επιπέδου $b_1 = \lceil \frac{r_1}{f_0} \rceil = 111$
- Αριθμός μπλοκ δευτέρου επιπέδου $b_2 = \lceil \frac{b_1}{f_0} \rceil = \lceil \frac{111}{68} \rceil = \lceil 1,632 \rceil = 2$
- Αριθμός μπλοκ τρίτου επιπέδου $b_3 = \lceil \frac{b_2}{f_0} \rceil = \lceil \frac{2}{68} \rceil = \lceil 0,0294 \rceil = 1$

Άρα το πλήθος των επιπέδων του ευρετηρίου είναι 3.

Σημείωση: Μπορούσαμε επίσης να εφαρμόσουμε τον τύπο $\lceil \log_{f_0}(r_1) \rceil = \lceil \log_{68}(7500) \rceil = \lceil 2,1146 \rceil = 3$.

4) Ο συνολικός αριθμός μπλοκ που απαιτούνται για το πολυεπίπεδο ευρετήριο είναι το άθροισμα του πλήθους των μπλοκ των επιπέδων.

Άρα είναι από το προηγούμενο ερώτημα: $111 + 2 + 1 = 114$ μπλοκ.

Σημείωση: Στο άθροισμα των μπλοκ για το πολυεπίπεδο ευρετήριο **δεν** αθροίζουμε τα μπλοκ που χρειάζεται το αρχείο για αποθήκευση, καθώς η άσκηση ζητάει μόνο τα επιπλέον μπλοκ που θα γραφτούν λόγω της δημιουργίας του πολυεπίπεδου ευρετηρίου.

5) Προκειμένου να αναζητηθεί και να ανακτηθεί μία εγγραφή από το αρχείο, όταν δίνεται η τιμή του ΑΦΜ στο πολυεπίπεδο ευρετήριο, χρειάζεται να προσπελαστούν 4 μπλοκ, ένα από κάθε επίπεδο του ευρετηρίου και 1 ακόμη μπλοκ από το αρχείο.

Χρησιμοποιούμε το ευρετήριο του τρίτου επιπέδου, ώστε να βρούμε το μπλοκ αναφοράς της εγγραφής, το οποίο υπάρχει στο δεύτερο επίπεδο. Αυτή η αναζήτηση απαιτεί την πρόσβαση σε ένα μόνο μπλοκ του τρίτου επιπέδου. (*κορυφαίο*)

Έπειτα, χρησιμοποιούμε το ευρετήριο του δευτέρου επιπέδου, ώστε να βρούμε το μπλοκ αναφοράς της εγγραφής, το οποίο υπάρχει στο πρώτο επίπεδο. Αυτή η αναζήτηση απαιτεί την πρόσβαση σε ένα μόνο μπλοκ του δευτέρου επιπέδου.

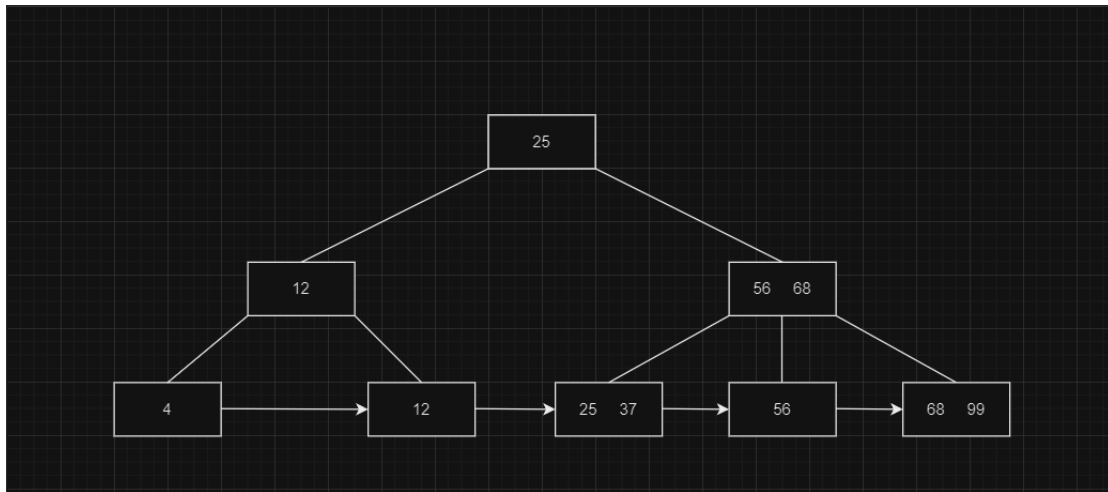
Έπειτα, χρησιμοποιούμε το ευρετήριο του πρώτου επιπέδου, ώστε να βρούμε το μπλοκ της εγγραφής, στο αρχείο. Αυτή η αναζήτηση απαιτεί την πρόσβαση σε ένα μόνο μπλοκ του πρώτου επιπέδου.

Για να ανακτήσουμε την εγγραφή, προσπελάζουμε άλλο ένα μπλοκ του δίσκου, αυτό που περιέχει την εγγραφή, από το αρχείο δεδομένων, στο οποίο οδηγηθήκαμε μέσω της προσπέλασης των μπλοκ των 3 επιπέδων του πολυεπίπεδου ευρετηρίου.

Άρα, συνολικά προσπελούνται 3 μπλοκ επιπέδων και 1 μπλοκ αρχείου, δηλαδή 4 μπλοκ του δίσκου.

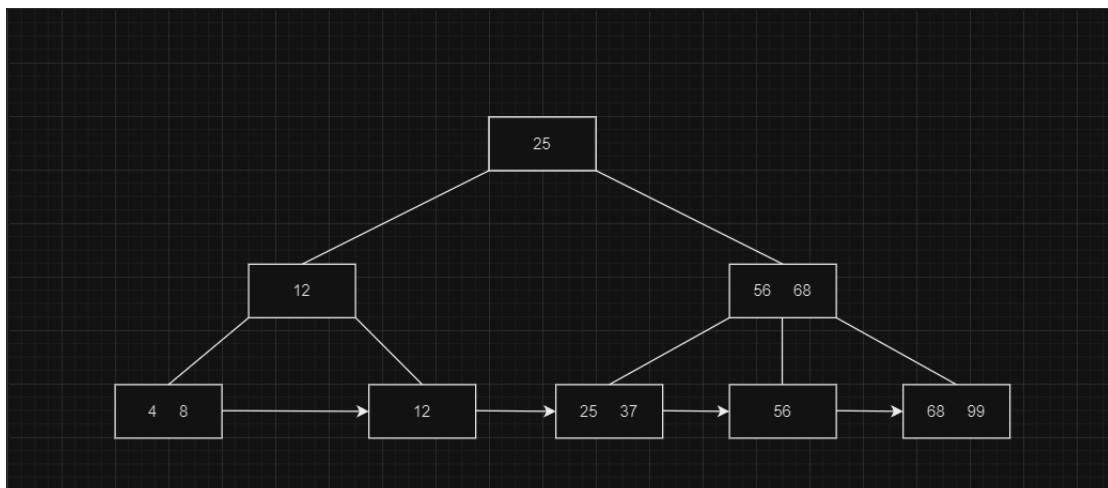
Άσκηση 3

Αρχικό δέντρο



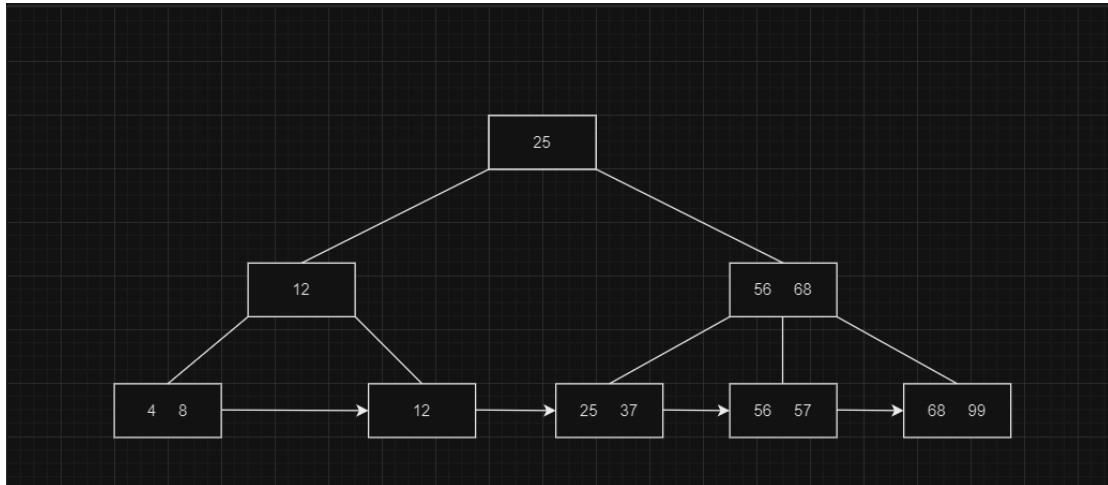
1) Εισαγωγή 8

- Αναζήτηση τιμής 8 που θέλουμε να εισάγουμε.
- Εισάγουμε την τιμή στο φύλλο που καταλήξαμε, αφού βρισκόμαστε στην απλή περίπτωση όπου η τιμή χωράει στο φύλλο.



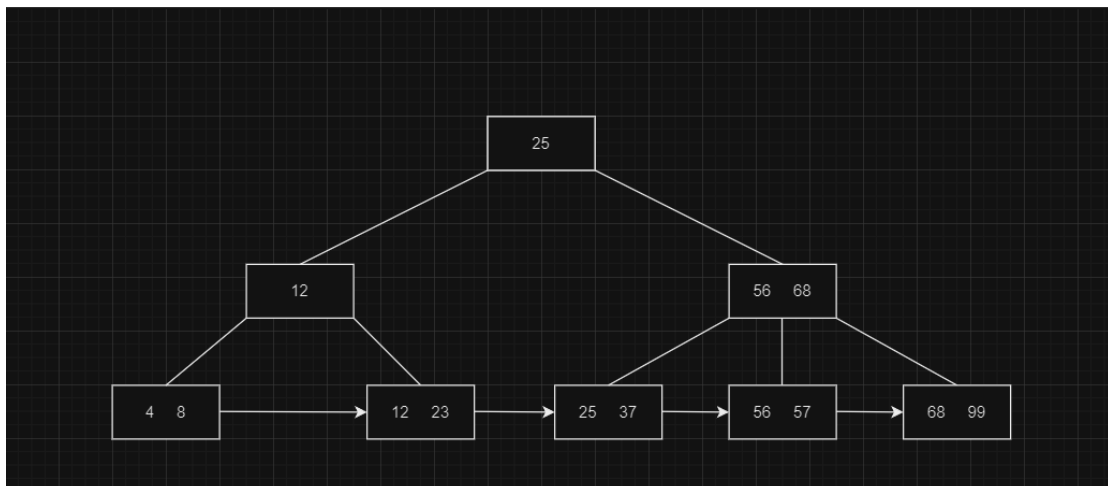
Εισαγωγή 57

- Αναζήτηση τιμής 57 που θέλουμε να εισάγουμε.
- Εισάγουμε την τιμή στο φύλλο που καταλήξαμε, αφού βρισκόμαστε στην απλή περίπτωση όπου η τιμή χωράει στο φύλλο.



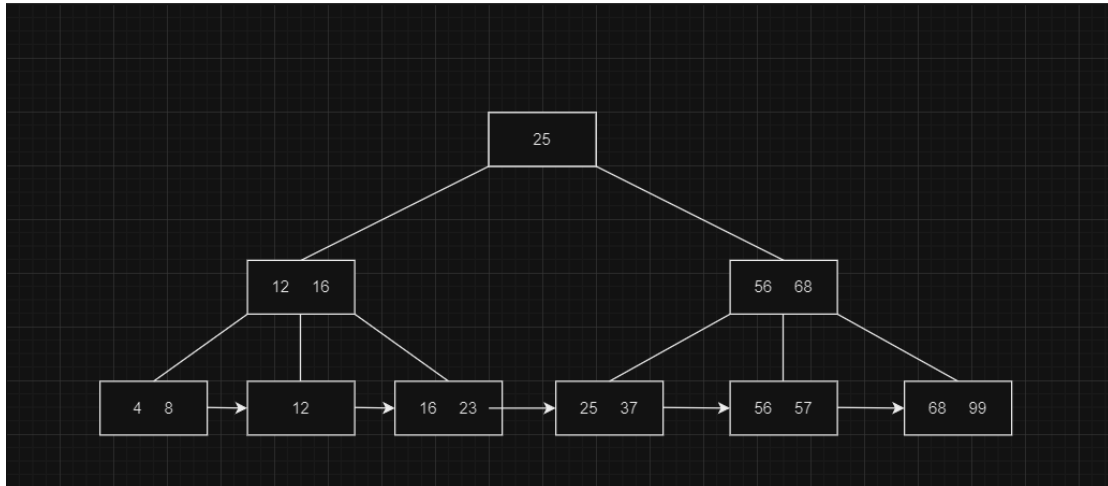
Εισαγωγή 23

- Αναζήτηση τιμής 23 που θέλουμε να εισάγουμε.
- Εισάγουμε την τιμή στο φύλλο που καταλήξαμε, αφού βρισκόμαστε στην απλή περίπτωση όπου η τιμή χωράει στο φύλλο.



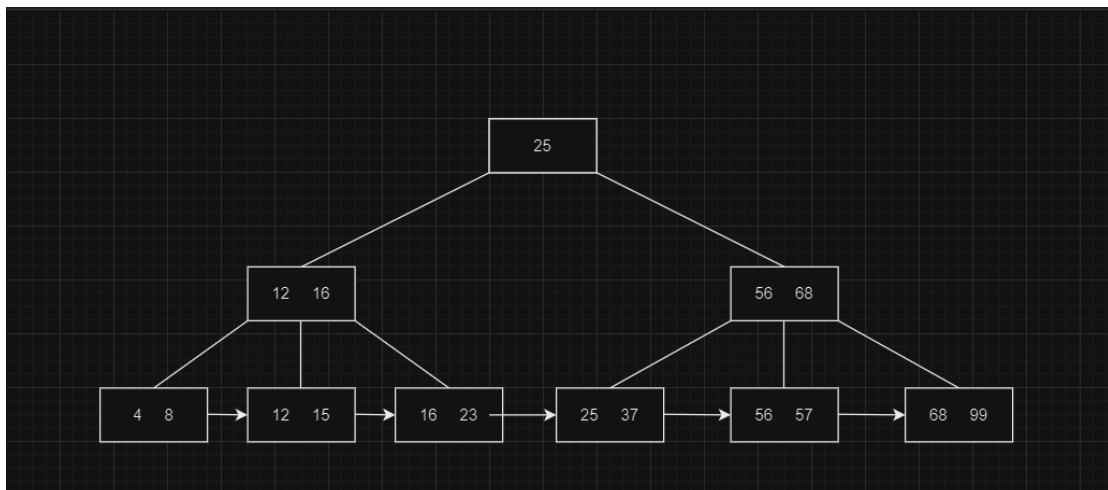
Εισαγωγή 16

- Αναζήτηση τιμής 16 που θέλουμε να εισάγουμε.
- Βρισκόμαστε στην περίπτωση όπου η τιμή δεν χωράει στο φύλλο, άρα έχουμε υπερχείλιση. Δημιουργούμε νέο κόμβο-φύλλο και τοποθετούμε στον γονέα του φύλλου τον μεσσαίο αριθμό του υπερχειλισμένου κόμβου, δηλαδή την τιμή 16. Μοιράζουμε τις τιμές του υπερχειλισμένου κόμβου, ώστε στον αριστερά κόμβο να βρίσκεται η μικρότερη τιμή και στον δεξιό κόμβο οι 2 υπόλοιπες. (εκφώνηση)



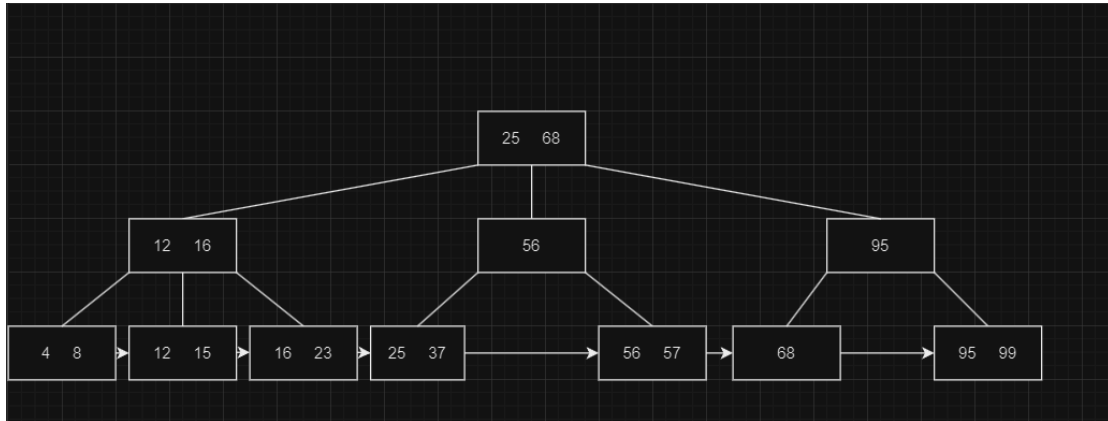
Εισαγωγή 15

- Αναζήτηση τιμής 15 που θέλουμε να εισάγουμε.
- Εισάγουμε την τιμή στο φύλλο που καταλήξαμε, αφού βρισκόμαστε στην απλή περίπτωση όπου η τιμή χωράει στο φύλλο.



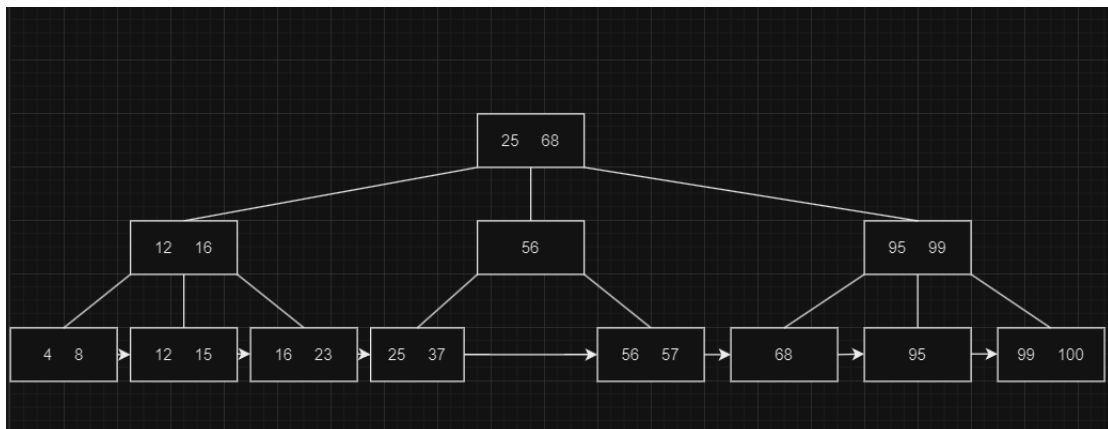
Εισαγωγή 95

- Αναζήτηση τιμής 95 που θέλουμε να εισάγουμε.
- Βρισκόμαστε στην περίπτωση όπου η τιμή δεν χωράει στο φύλλο, άρα έχουμε υπερχειλίση. Δημιουργούμε νέο κόμβο-φύλλο και τοποθετούμε στον γονέα του φύλλου τον μεσσαίο αριθμό του υπερχειλισμένου κόμβου, δηλαδή την τιμή 95. Μοιράζουμε τις τιμές του υπερχειλισμένου κόμβου, ώστε στον αριστερά κόμβο να βρίσκεται η μικρότερη τιμή και στον δεξιό κόμβο οι 2 υπόλοιπες. (εκφώνηση)
- Παρατηρούμε ωστόσο και υπερχειλίση του γονέα, οπότε ακολουθούμε την ίδια διαδικασία διάσπασης κόμβου αναδρομικά.



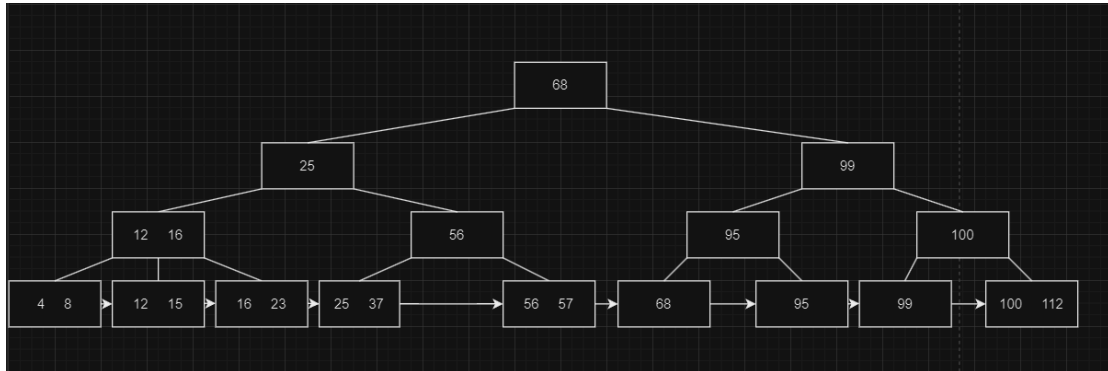
Εισαγωγή 100

- Αναζήτηση τιμής 100 που θέλουμε να εισάγουμε.
- Βρισκόμαστε στην περίπτωση όπου η τιμή δεν χωράει στο φύλλο, άρα έχουμε υπερχειλίση. Δημιουργούμε νέο κόμβο-φύλλο και τοποθετούμε στον γονέα του φύλλου τον μεσσαίο αριθμό του υπερχειλισμένου κόμβου, δηλαδή την τιμή 99. Μοιράζουμε τις τιμές του υπερχειλισμένου κόμβου, ώστε στον αριστερά κόμβο να βρίσκεται η μικρότερη τιμή και στον δεξιό κόμβο οι 2 υπόλοιπες. (εκφώνηση)



Εισαγωγή 112

- Αναζήτηση τιμής 112 που θέλουμε να εισάγουμε.
- Βρισκόμαστε στην περίπτωση όπου η τιμή δεν χωράει στο φύλλο, άρα έχουμε υπερχειλίση. Δημιουργούμε νέο κόμβο-φύλλο και τοποθετούμε στον γονέα του φύλλου τον μεσαίο αριθμό του υπερχειλισμένου κόμβου, δηλαδή την τιμή 100. Μοιράζουμε τις τιμές του υπερχειλισμένου κόμβου, ώστε στον αριστερά κόμβο να βρίσκεται η μικρότερη τιμή και στον δεξιό κόμβο οι 2 υπόλοιπες. (εκφώνηση)
- Παρατηρούμε ωστόσο και υπερχειλίση του γονέα, οπότε ακολουθούμε την ίδια διαδικασία διάσπασης κόμβου αναδρομικά. Επίσης, δημιουργείται υπερχειλίση ρίζας του δέντρου οπότε το δέντρο αυξάνεται κατά ένα επίπεδο.



Σημείωση: Σε κάποιους κόμβους έγινε λογική αφαίρεση κλειδιών, όπως αναφέρει η θεωρία.

2) Για να μπορέσουμε να φτάσουμε στο αρχείο ώστε να διαβάσουμε τις εγγραφές, αρχικά πρέπει να προσπελάσουμε το ευρετήριο. Συγκεκριμένα, για $A \geq 12$ AND $A \leq 37$, θα πρέπει αρχικά να προσπελάσουμε τους αντίστοιχους κόμβους. Ξεκινάμε την αναζήτηση, με το κάτω όριο του εύρους, δηλαδή την τιμή 12. Προσπελάνουμε την ρίζα, προκειμένου να αναζητήσουμε την τιμή 12. Η ρίζα, θα μας παραπέμψει στο αριστερό υποδέντρο, καθώς η περίπτωση είναι < 68 . Έπειτα στο 1^ο επίπεδο του δέντρου, προσπελάνουμε τον κόμβο με τιμή 25, ο οποίος μας οδηγεί στο αριστερό του υποδέντρο, αφού $12 < 25$. Στο 2^ο επίπεδο του δέντρου, προσπελάνουμε τον κόμβο με τιμές 12|16, ο οποίος μας οδηγεί στο μεσαίο υποδέντρο του, αφού $12 \leq 12 < 16$ και προσπελάνουμε τον κόμβο με κλειδί 12. Άρα η αναζήτηση της τιμής 12, απαιτεί προσπέλαση 4 κόμβων. Στη συνέχεια, μέσω των δεικτών μετακινούμε δεξιά για όσο μας ενημερώνει το εύρος, δηλαδή για όσο δεν συναντάμε τιμή ≥ 37 . Άρα, προσπελάνουμε 2 ακόμα κόμβους, τον κόμβο με τιμές 16|23 και τον κόμβο με τιμές 25|37. Μόλις εντοπίσουμε την τιμή 37, σταματάμε την διάσχιση του δέντρου, καθώς δεν θα υπάρχει αλλού κόμβος με κλειδί του εύρους επειδή το κλειδί αναζήτησης είναι μοναδικό. (Το γνώρισμα A είναι πρωτεύον κλειδί, άρα είναι μοναδικό)

Άρα προσπελάνουμε συνολικά $4 + 2 = 6$ κόμβους, οι οποίοι χωράνε ο καθένας σε 1 μπλοκ. Άρα προσπελάνουμε 6 μπλοκ για την αναζήτηση των κλειδιών του εύρους και επιπλέον προσπελάνουμε 6 μπλοκ στο αρχείο, που αντιστοιχούν στις 6 εγγραφές που δείχνουν οι κόμβοι-φύλλα του εύρους αναζήτησης που προσπελάσαμε. Συγκεκριμένα, κάθε στοιχείο του κόμβου-φύλλου, περιέχει έναν δείκτη δεδομένων, ο οποίος δείχνει στην εγγραφή της οποίας η τιμή στο πεδίο αναζήτησης αναγράφεται ως κλειδί του κόμβου-φύλλου. Αφού είναι 6 τα στοιχεία προσπέλασης κόμβων-φύλλων του εύρους αναζήτησης, άρα έχουμε 6 εγγραφές. Σύμφωνα με την εκφώνηση κάθε εγγραφή αποθηκεύεται σε 1 μπλοκ, άρα για τις εγγραφές προσπελάνουμε 6 μπλοκ. Άρα συνολικά για να ανακτήσουμε όλες τις εγγραφές της R με κλειδί αναζήτησης $A \geq 12$ AND $A \leq 37$, χρειάζεται να προσπελάσουμε $6 + 6 = 12$ μπλοκ δίσκου.

Άσκηση 4

1. Εισαγωγή 1000

1000	

0 1 utilization=1/6 = 16,6% , m=1, i=1

2. Εισαγωγή 0000

0000	
1000	

0 1 utilization=2/6 = 33,33% , m=1, i=1

3. Εισαγωγή 1101

0000	
1000	1101

0 1 utilization=3/6 = 50% , m=1, i=1

4. Εισαγωγή 0010

0010	
0000	
1000	1101

0 1 utilization=4/6 = 66,66% , m=1, i=1

5. Εισαγωγή 0010

0010	
0010	
0000	
1000	1101

0 1 utilization=5/9 = 55,55% , m=1, i=1

6. Εισαγωγή 1100

1100	
0010	
0010	
0000	
1000	1101

0 1 utilization=6/9 = 66.66% , m=1, i=1

7. Εισαγωγή 0011

1100	
0010	
0010	
0000	0011
1000	1101

0 1 utilization=7/9 = 77.77% , m=1, i=1

Αφού το utilization είναι >70%, τότε πρέπει να αυξήσουμε τον αριθμό των κώδων από 2 κώδους($m=1$), σε 3 κώδους($m=2$). Άρα αυξάνουμε και το πλήθος των $i=1$, λιγότερο σημαντικών bits, σε $i=2$.

1100		
0000	0011	0010
1000	1101	0010

00 *1 10 utilization=7/9 = 77.77% , m=2, i=2

Αφού το utilization είναι μετά τις αλλαγές >70%, τότε πρέπει να αυξήσουμε τον αριθμό των κώδων από 3 κώδους($m=2$), σε 4 κώδους($m=3$).

1100			
0000		0010	
1000	1101	0010	0011

00 01 10 11 utilization=7/12 = 58,33% , m=3, i=2

8. Εισαγωγή 1111

1100			
0000		0010	1111
1000	1101	0010	0011

00 01 10 11 utilization=8/12 = 66,66% , m=3, i=2

9. Εισαγωγή 0110

1100		0110	
0000		0010	1111
1000	1101	0010	0011

00 01 10 11 utilization=9/12 = 75% , m=3, i=2

Αφού το utilization είναι >70%, τότε πρέπει να αυξήσουμε τον αριθμό των κάδων από 4 κάδους($m=3$), σε 5 κάδους($m=4$). Άρα αυξάνουμε και το πλήθος των $i=2$, λιγότερο σημαντικών bits, σε $i=3$.

		0110		
0000		0010	1111	
1000	1101	0010	0011	1100

000 *01 *10 *11 100 utilization=9/15 = 60% , m=4, i=3

10. Εισαγωγή 1110

		1110		
		0110		
0000		0010	1111	
1000	1101	0010	0011	1100

000 *01 *10 *11 100 utilization=10/18 = 55,55% , m=4, i=3

Άσκηση 5

1)

α) Για να απαντηθεί μια επερώτηση τύπου Q1, χρειάζεται να προσπελαστούν όλα τα μπλοκ που αντιστοιχούν στον κατακερματισμό του πεδίου a , με βάση την συνάρτηση κατακερματισμού $h1()$. Όπως αναφέρεται η συνάρτηση κατακερματισμού $h1()$ στο πεδίο a , επιστρέφει N bits, για την αποθήκευση της εγγραφής. Από αυτά τα N bits, υπολογίζουμε ότι υπάρχουν 2^N , διαφορετικά αποτελέσματα κατακερματισμού. Επίσης, κάθε εγγραφή, αντιστοιχεί σε μοναδικό αποτέλεσμα κατακερματισμού, καθώς όπως ξέρουμε η συνάρτηση κατακερματισμού είναι ένα προς ένα. Άρα, κάθε επερώτηση για να απαντηθεί, χρειάζεται να προσπελάσει όλα εκείνα τα μπλοκ που αντιστοιχούν στον κατακερματισμό βάση της συνάρτησης κατακερματισμού $h1()$. Άρα από τα 1024 μπλοκ που καταλαμβάνει το αρχείο κατακερματισμού, θα προσπελαστούν εκείνα τα μπλοκ που αντιστοιχούν στον τρέχον κατακερματισμό, ανάλογα με την επερώτηση. Οπότε, ο αριθμός των μπλοκ που χρειάζεται να προσπελαστούν για τις απαντήσεις των επερωτήσεων Q1, είναι $\frac{1024}{2^N}$.

β) Για να απαντηθεί μια επερώτηση τύπου Q2, χρειάζεται να προσπελαστούν όλα τα μπλοκ που αντιστοιχούν στον κατακερματισμό του πεδίου b, με βάση την συνάρτηση κατακερματισμού $h_2()$. Όπως αναφέρεται η συνάρτηση κατακερματισμού $h_2()$ στο πεδίο b, επιστρέφει $10-N$ bits, για την αποθήκευση της εγγραφής. Από αυτά τα $10-N$ bits, υπολογίζουμε ότι υπάρχουν $2^{(10-N)}$, διαφορετικά αποτελέσματα κατακερματισμού. Επίσης, κάθε εγγραφή, αντιστοιχεί σε μοναδικό αποτέλεσμα κατακερματισμού, καθώς όπως ξέρουμε η συνάρτηση κατακερματισμού είναι ένα προς ένα. Άρα, κάθε επερώτηση για να απαντηθεί, χρειάζεται να προσπελάσει όλα εκείνα τα μπλοκ που αντιστοιχούν στον κατακερματισμό βάση της συνάρτησης κατακερματισμού $h_2()$. Άρα από τα 1024 μπλοκ που καταλαμβάνει το αρχείο κατακερματισμού, θα προσπελαστούν εκείνα τα μπλοκ που αντιστοιχούν στον τρέχον κατακερματισμό, ανάλογα με την επερώτηση. Οπότε, ο αριθμός των μπλοκ που χρειάζεται να προσπελαστούν για τις απαντήσεις των επερωτήσεων Q2, είναι $\frac{1024}{2^{(10-N)}}$.

2) Αφού το 30% των επερωτήσεων που αφορούν την σχέση R, είναι της μορφής Q1, ενώ το υπόλοιπο 70% είναι της μορφής Q2, τότε ο μέσος όρος πρέπει να σταθμιστεί και να δώσει την ανάλογη βαρύτητα στις επερωτήσεις. Από το προηγούμενο ερώτημα, γνωρίζουμε πόσα είναι τα μπλοκ που χρειάζεται να προσπελαστούν για την απάντηση των επερωτήσεων Q1 και Q2 αντίστοιχα. Ο μέσος αριθμός μπλοκ που πρέπει να προσπελαστούν για την απάντηση των επερωτήσεων Q1 και Q2, στην σχέση R, είναι:

$$\frac{30\% * 1024}{2^N} + \frac{70\% * 1024}{2^{(10-N)}} = \frac{0,3 * 1024}{2^N} + \frac{0,7 * 1024}{2^{(10-N)}}.$$