Συστήματα Διαχείρισης και Ανάλυσης Δεδομένων

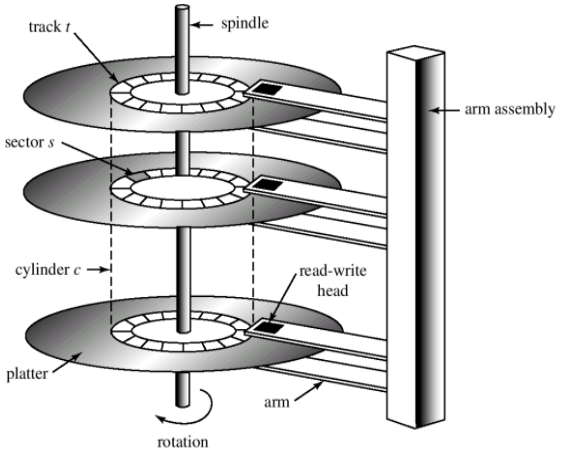
1η Σειρά Ασκήσεων

Παντελίδης Ιπποκράτης – p3210150

Άσκηση 1)

Για τα πρώτα δύο ερωτήματα της άσκησης θα χρησιμοποιήσουμε την εικόνα την δομής του μαγνητικού δίσκου που φαίνεται παρακάτω.

1. Για να βρούμε το μέγεθος του τομέα του δίσκου θα ξεκινήσουμε από το γενικό που είναι ο δίσκος και θα πηγαίνουμε προς το πιο ειδικό που είναι οι τομείς και θα υπολογίζουμε κάθε φορά τα στοιχεία που λείπουν από αυτά που έχουν δοθεί στην εκφώνηση.



Μας δίνεται για τον δίσκο ότι έχει Συνολική Χωρητικότητα = 80GB, 4 πλακέτες διπλής όψης, 500 tracks ανά επιφάνεια και 1024 sectors ανά ίχνος.

Οπότε έχουμε :

i) **bytes/δίσκο = bytes/επιφάνεια * επιφάνειες/πλακέτα * αριθμός πλακετών**

δηλαδή 80GB = bytes/επιφάνεια * 2 * 4 και άρα bytes/επιφάνεια = 10GB

ii) **bytes/επιφάνειας = bytes/ίχνος * ίχνη/επιφάνεια**

δηλαδή 10GB = bytes/ίχνος * 512 και άρα bytes/ίχνος = $10 * 2^{21}$

iii) **bytes/ίχνος = bytes/τομέα * τομείς/ίχνος**

δηλαδή $10 * 2^{21}$ = bytes/τομέα * 1024 και άρα bytes/τομέα = $10 * 2^{11}$

Τέλος μετατρέπουμε τα $10 * 2^{11}$ bytes σε KB οπότε διαιρούμε με 2^{10} και έχουμε τελικά ότι το μέγεθος του τομέα είναι $10 * 2^1$ KB δηλαδή **20KB**.

2. Ξέρουμε ότι η χωρητικότητα του κυλίνδρου είναι το γινόμενο του αριθμού των επιφανειών του δίσκου και του μεγέθους του ενός ίχνους. Στο παράδειγμα έχουμε 8 επιφάνειες και από το προηγούμενο ερώτημα βρήκαμε ότι το κάθε ίχνος έχει μήκος $10 * 2^{21}$ bytes, δηλαδή $20 * 2^{20}$ bytes. Άρα η **χωρητικότητα** του **κυλίνδρου** είναι 8 *

$20 * 2^{20}$ bytes = **$160 * 2^{20}$ bytes**. Επίσης η **σχέση R** έχει **μέγεθος** 8192 * 40KB = 2^{13} * $40 * 2^{10}$ = **$40 * 2^{23}$ bytes**. Για να βρούμε τώρα πόσος κυλίνδρους χωράει η σχέση R αρκεί να διαιρέσουμε την χωρητικότητα της σχέσης με αυτή του κυλίνδρου και έχουμε $40 * 2^{23}$ ~~bytes~~ / $160 * 2^{20}$ ~~bytes~~ = $2^3 / 4 = 2^3 / 2^2$ = **2 κύλινδροι**.

3. Ξέρουμε ότι ο απαιτούμενος χρόνος για την ανάγνωση ολόκληρης της σχέσης R προκύπτει από τον ακόλουθο τύπο :

Συνολικός Χρόνος για το Διάβασμα της Σχέσης =
Μέσο Χρόνος Μετακίνησης Κεφαλής (I) +
Μέση Καθυστέρηση Περιστροφής (II) +
Χρόνος Ανάγνωσης των Μπλοκ της Σχέσης (III)

Θεωρούμε ότι ο χρόνος μετακίνησης της κεφαλής στο επόμενο ίχνος καθώς και άλλες καθυστερήσεις είναι μηδαμινές.

Από την εκφώνηση μας δίνεται ότι ο **(I)** είναι 8ms και άρα **0,008 sec**. Την **(II)** μπορούμε να την υπολογίσουμε εύκολα και χρησιμοποιώντας την ταχύτητα περιστροφής που μας δίνεται στην εκφώνηση βρίσκουμε ότι έχει τιμή (60 sec / 7200 rpm) / 2 = **0,00415 sec**. Για τον υπολογισμό του **(III)** θα υπολογίσουμε αρχικά τον χρόνο ανάγνωσης ενός μόνο ίχνους που έχει τιμή 60 sec / 7200 rpm = 0,0083 sec. Επίσης ο κάθε κύλινδρος έχει 8 ίχνη (4 επιφάνειες * 2 ίχνη) οπότε για τον διάβασμα ενός κυλίνδρου θέλουμε 0,0083 * 8 = 0,0664 sec. Επειδή ακόμα έχουμε 2 κυλίνδρους θέλουμε συνολικά χρόνο για το (III) 0.0664 * 2 = **0,1328 sec**.

Οπότε αθροίζοντας παραπάνω έχουμε 0,008 + 0,00415 + 0,1328 = **0,14495 sec**.

4. Για αυτό το ερώτημα γνωρίζουμε ότι ο απαιτούμενος χρόνος για την ανάγνωση μια τυχαίας εγγραφής είναι :

Συνολικός Χρόνος για το Διάβασμα ενός Μπλοκ =
Μέσο Χρόνος Μετακίνησης Κεφαλής (I) +
Μέση Καθυστέρηση Περιστροφής (II) +
Χρόνος Μεταφοράς του Μπλοκ (III)

Θεωρούμε όπως και στο ερώτημα 3 ότι ο χρόνος μετακίνησης της κεφαλής στο επόμενο ίχνος καθώς και άλλες καθυστερήσεις είναι μηδαμινές.

Επίσης οι χρόνοι **(I)** και **(II)** μένουν ακριβώς οι ίδιοι με το προηγούμενο ερώτημα και είναι αντίστοιχα **0,008 sec** και **0,00415 sec**. Για τον υπολογισμό του **(III)** ξέρουμε από την εκφώνηση ότι ένα ίχνος έχει 1024 τομείς. Μας δίνεται επίσης στην εκφώνηση ότι κάθε εγγραφή της σχέσης R αποθηκεύεται ολόκληρη σε ένα μπλοκ και αφού το μέγεθος κάθε εγγραφής είναι 40KB και το μέγεθος του μπλοκ είναι 40KB. Οπότε προκύ-

πτει ότι 1 μπλοκ ισοδυναμεί με 2 τομείς (αφού ένας τομέας έχει 20KB) και άρα ένα ίχνος έχει 512 μπλοκ. Έχουμε λοιπόν ότι διαβάζουμε ένα τέτοιο ίχνος σε 0.0083 sec οπότε διαβάζουμε 1 μπλοκ σε 0.0083 / 512 = 0.000016 sec. Ο συνολικός χρόνος για την **ανάγνωση** του **ενός μπλοκ** με την χρήση του παραπάνω τύπου είναι 0.008 + 0.00415 + 0.000016 = 0.012166 sec δηλαδή στρογγυλοποιώντας **0.01217 sec**. Επειδή όμως θέλουμε τον χρόνο **ανάγνωσης 100 εγγραφών**, που αποθηκεύονται όμως η κάθε μία σε 1 μπλοκ, έχουμε τελικά 0.012166 * 100 = **1.217 sec**.

Άσκηση 2)

Πριν προχωρήσουμε στην απάντηση των ερωτημάτων θα υπολογίσουμε το **μέγεθος εγγραφής** της **σχέσης** ΦΟΡΟΛΟΓΟΥΜΕΝΟΣ και το **μέγεθος εγγραφής** του **ευρετηρίου**. Για να βρούμε το πρώτο αθροίζουμε το μήκος όλων των πεδίων της σχέσης και έχουμε 9 + 30 + 10 + 9 + 8 + 1 + 40 + 4 + 4 = 115 bytes. Για το δεύτερο θα χρειαστούμε μόνο το πεδίο πάνω στο οποίο θα χτιστεί το ευρετήριο (ΑΦΜ) και το μήκος του δείκτη μπλοκ, που από εκφώνηση έχει μήκος 6 bytes. Οπότε το μέγεθος εγγραφής του ευρετηρίου είναι 9 + 6 = 15 bytes.

1. Εφόσον το κάθε μπλοκ χωράει 1024 bytes και η κάθε εγγραφή της σχέσης έχει μέγεθος 115 bytes, βρίσκουμε ότι σε κάθε μπλοκ χωράνε floor(1024/115) = 8 εγγραφές. Άρα για την αποθήκευση των 60.000 εγγραφών της σχέσης θα χρειαστούμε 60.000 / 8 = 7500 μπλοκ. Επομένως, επειδή το ευρετήριο θα περιέχει μία εγγραφή της μορφής <πρωτεύον κλειδί, δείκτης> για καθένα μπλοκ από αυτά τα 7500 της σχέσης προκύπτει ότι θα έχει και **7500 καταχωρήσεις**.

2. Για να βρούμε τα μπλοκ του πρώτου επιπέδου του ευρετηρίου πρέπει να βρούμε αρχικά πόσες εγγραφές ευρετηρίου χωράνε σε κάθε μπλοκ. Εφόσον το μέγεθος του μπλοκ είναι 1024 bytes και το μέγεθος της εγγραφής του ευρετηρίου είναι 15 bytes βρίσκουμε ότι χωράνε floor(1024 / 15) = 68 εγγραφές ευρετηρίου σε κάθε μπλοκ. Αφού επιπλέον το πρώτο επίπεδο του ευρετηρίου έχει 7500 εγγραφές, από ερώτημα 1, βρίσκουμε τελικά ότι το πρώτο επίπεδο έχει ceil(7500/68) = **111 μπλοκ**.

3. Μέχρι τώρα το ευρετήριο μας έχει ένα επίπεδο, όμως επειδή θέλουμε πολυεπίπεδο μπορούμε με τον ίδιο τρόπο που δημιουργήσαμε το πρώτο επίπεδο να το διασπάσουμε ακόμα παραπάνω. Ξέρουμε από το προηγούμενο ερώτημα ότι σε κάθε μπλοκ χωράνε 68 εγγραφές ευρετηρίου και ότι το **πρώτο επίπεδο** έχει **111 μπλοκ**, επομένως το δεύτερο επίπεδο του ευρετηρίου πρέπει να έχει πάλι εγγραφές της μορφής <πρωτεύον κλειδί, δείκτης> αλλά αυτήν την φορά μία για κάθε μπλοκ από τα 111 του επόμενου επιπέδου. Οπότε το **δεύτερο επίπεδο** θα έχει ceil(111/68) = **2 μπλοκ**. Με την ίδια λογική το **τρίτο επίπεδο** θα έχει ceil(2/68) = **1 μπλοκ** το οποίο χωράει ακρι-

βώς σε μία σελίδα και σύμφωνα με την εκφώνηση είναι το τελευταίο επίπεδο. Άρα το ευρετήριο έχει **3 επίπεδα**.

4. Ο συνολικός αριθμός των μπλοκ του πολυεπίπεδου ευρετηρίου προκύπτει αθροίζοντας τα μπλοκ που απαιτούνται σε κάθε επίπεδο. Επομένως έχουμε

   Συνολικός αριθμός μπλοκ = μπλοκ 1ου + μπλοκ 2ου + μπλοκ 3ου

   Σύμφωνα με ότι έχουμε βρει στο προηγούμενα ερώτημα έχουμε τελικά ότι συνολικός αριθμός είναι 111 + 2 + 1 = **114 μπλοκ**.

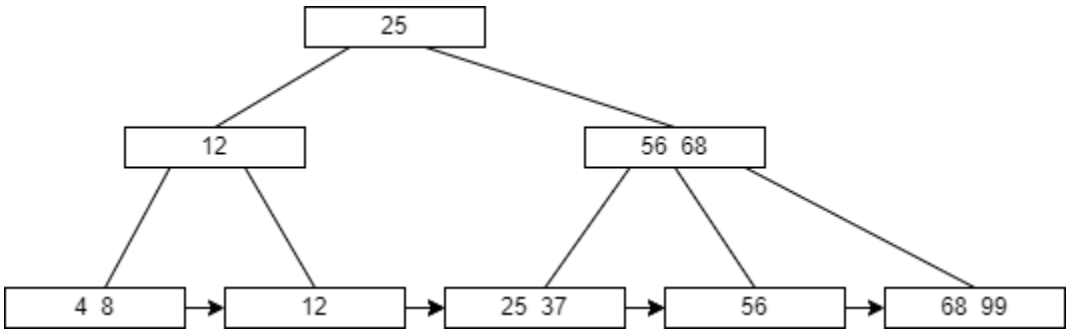
5. Όταν μας δίνεται μία τιμή ΑΦΜ για αναζήτηση ξεκινάμε από το τελευταίο επίπεδο του ευρετηρίου και διαβάζουμε το ένα και μοναδικό μπλοκ του. Βρίσκουμε την εγγραφή που μας εξυπηρετεί και μέσω του δείκτη της πάμε στο επόμενο επίπεδο. Αυτό επαναλαμβάνεται άλλες δύο φορές μέχρι να βρούμε την τιμή που αναζητούμε στο κατάλληλο μπλοκ της σχέσης. Μέχρι τώρα έχουμε προσπελάσει 3 μπλοκ και διαβάζουμε άλλο ένα για την ανάκτηση της εγγραφή από το αρχείο. Οπότε συνολικά θα προσπελάσουμε 3 + 1 = **4 μπλοκ**.

Άσκηση 3)

1. Διαδοχικές Εισαγωγές σε B+ Δέντρο

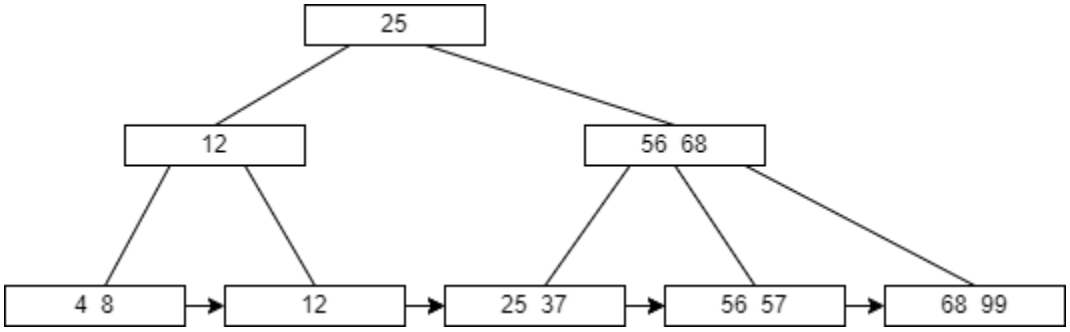
Βήμα 1ο : Εισαγωγή 8

Βρισκόμαστε στην απλή περίπτωση, όπου απλά κάνουμε αναζήτηση της τιμής 8 στο δέντρο και επειδή υπάρχει χώρος στην κατάλληλη θέση (αριστερότερο φύλλο) το τοποθετούμε εκεί. Οπότε έχουμε το ακόλουθο δέντρο :



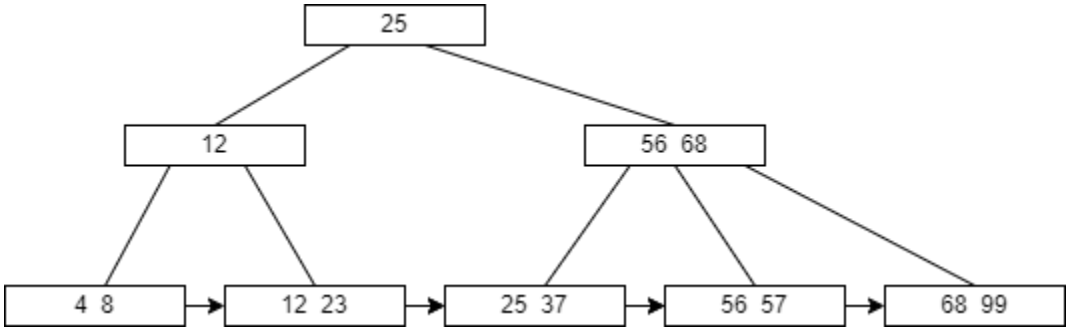
Βήμα 2ο : Εισαγωγή 57

Όμοια με το Βήμα 1, κάνουμε αναζήτηση και βλέπουμε ότι υπάρχει χώρος για ακόμα μία τιμή στο 4ο φύλλο οπότε και το τοποθετούμε εκεί. Το δέντρο φαίνεται παρακάτω :



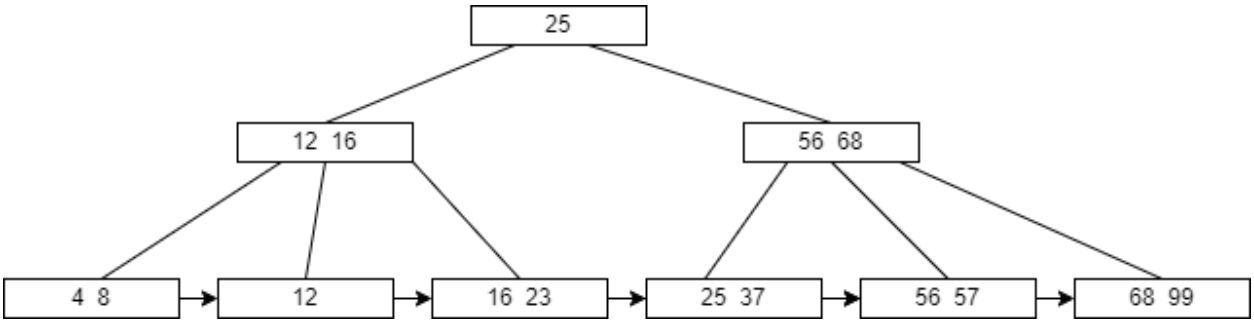
Βήμα 3ο : Εισαγωγή 23

Πάλι βρισκόμαστε στην απλή περίπτωση και καταλήγουμε να προσθέσουμε την τιμή στο 2ο φύλλο. Οπότε προκύπτει το ακόλουθο δέντρο :



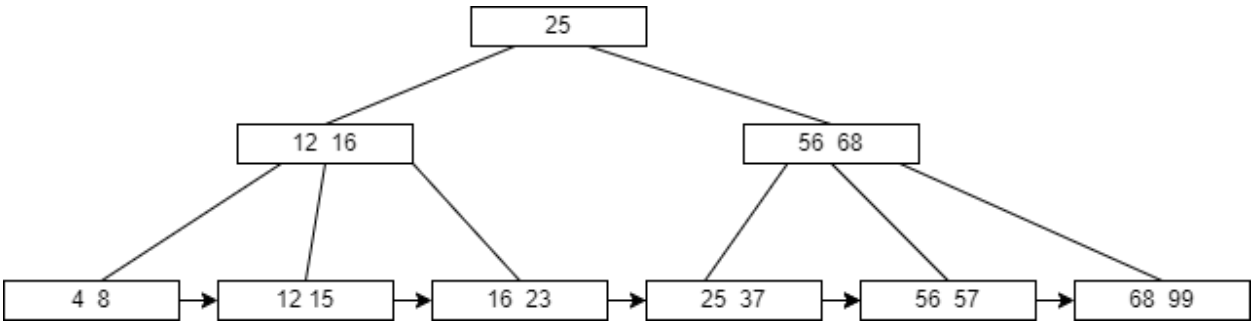
Βήμα 4ο : Εισαγωγή 16

Αρχικά αναζητούμε την τιμή 16 στο δέντρο και βλέπουμε ότι πρέπει να τοποθετηθεί στο 2ο φύλλο. Ωστόσο το δεύτερο φύλλο δεν έχει άλλο χώρο, καθώς κάθε κόμβος χωράει το πολύ δύο κλειδιά, οπότε έχουμε leaf overflow. Σε αυτή την περίπτωση, σπάμε το φύλλο και σύμφωνα με την υπόδειξη της εκφώνησης το ένα κλειδί πάει σαν αριστερό παιδί και τα άλλα δύο πάνε σαν δεξιά. Οπότε τώρα έχουμε το δέντρο :



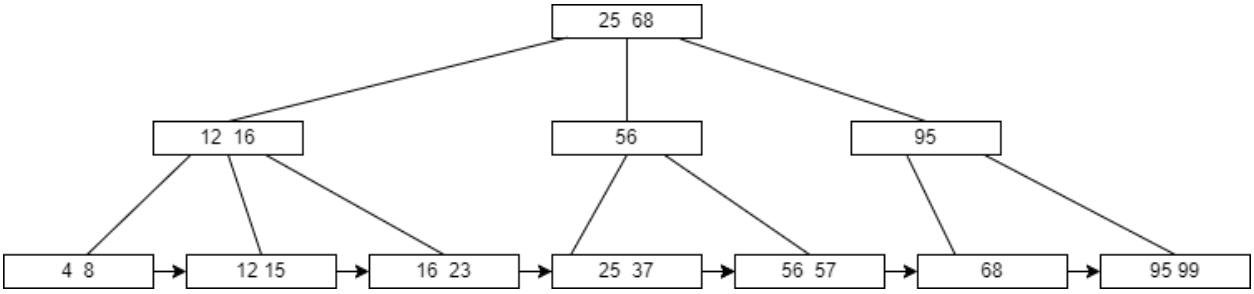
Βήμα 5ο : Εισαγωγή 15

Επανερχόμαστε στην απλή περίπτωση εισαγωγής όπου το κλειδί 15 αναζητείται και εν τέλει τοποθετείται στο 2ο φύλλο διότι υπάρχει χώρος για ακόμα μία τιμή. Προκύπτει :



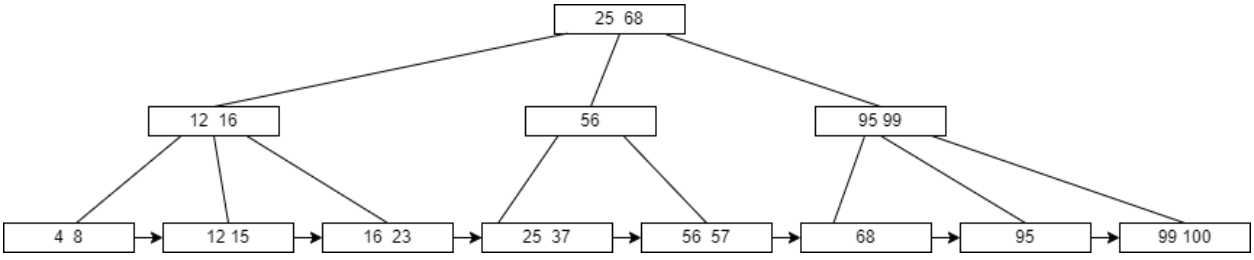
Βήμα 6ο : Εισαγωγή 95

Αναζητούμε την τιμή 95 και βλέπουμε ότι πρέπει να τοποθετηθεί στο δεξιότερο φύλλο, το οποίο δεν έχει χώρο. Οπότε έχουμε leaf overflow, ανεβάζοντας την τιμή 95 πάνω, η οποία με την σειρά της προκαλεί non leaf overflow στο ενδιάμεσο κόμβο. Σε αυτήν την περίπτωση ανεβαίνει η τιμή 68 στην ρίζα και προκύπτει το παρακάτω δέντρο :



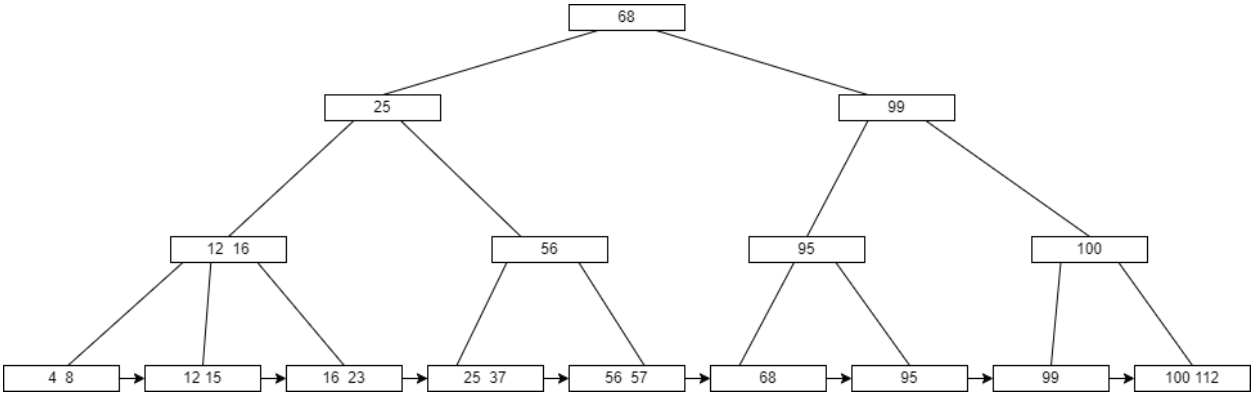
Βήμα 7ο : Εισαγωγή 100

Η τιμή 100 τοποθετείται πάλι στο δεξιότερο φύλλο όπως στο προηγούμενο βήμα αλλά έχουμε ξανά leaf overflow οπότε πάλι το ένα κλειδί (95) πάει αριστερά και τα άλλα δύο (99, 100) μπαίνουν δεξιά. Έχουμε λοιπόν :



Βήμα 8ο : Εισαγωγή 112

Όμοια με προηγουμένως, η τιμή 112 πρέπει να μπει στο δεξιότερο φύλλο και αυτό προκαλεί διαδοχικές διασπάσεις κόμβων μέχρι την ρίζα. Αρχικά, έχουμε leaf overflow, το οποίο προκαλεί στον πατέρα non leaf overflow και αυτό με την σειρά του σπάει την ρίζα. Τελικά μετά από όλες τις εισαγωγές έχουμε το δέντρο :



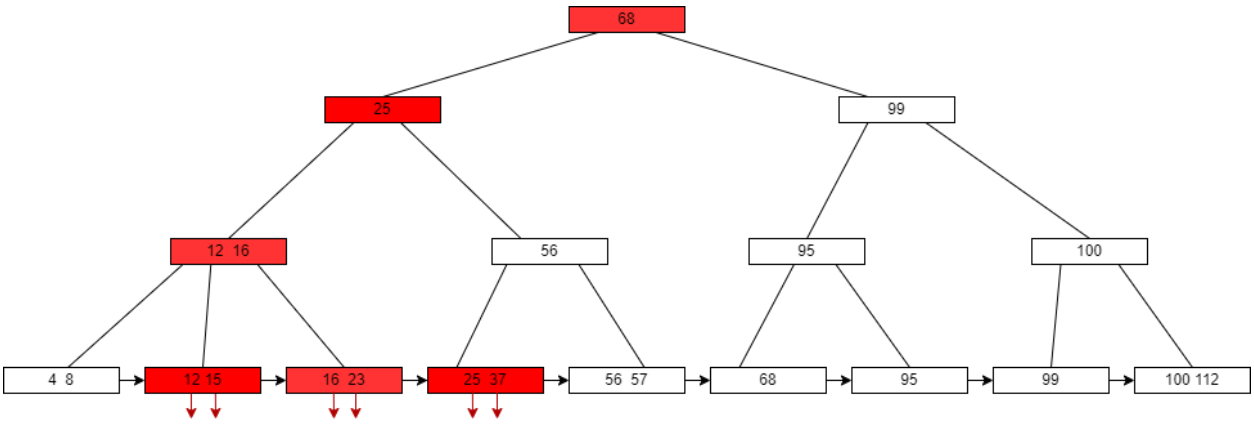
2. Για να βρούμε πόσα μπλοκ πρέπει να προσπελαστούν στον δίσκο για να ανακτηθούν όλες οι εγγραφές με κλειδί A>=12 AND A<=37 θα ακολουθήσουμε το ακόλουθο μονοπάτι :

   [68] → [25] → [12, 16] → [12, 15] → [16, 23] → [25, 37]

   Οι προσπελάσεις μας λοιπόν θα είναι αυτές οι 6 καθώς και οι προσπελάσεις που θα γίνουν στον δίσκο για κάθε τιμή των φύλλων που είναι εντός του διαστήματος [12, 37], οι οποίες είναι και αυτές 6. Άρα συνολικά έχουμε **12 προσπελάσεις**.

   Note: Σταματάμε σε αυτό το σημείο διότι βρήκαμε το 37 που είναι η μεγαλύτερη τιμή που δεχόμαστε και ξέρουμε ότι δεν υπάρχει άλλη μιας και η εκφώνηση λέει ότι το A γνώρισμα είναι πρωτεύον κλειδί.

   Παρακάτω βλέπουμε και σχηματικά αυτές τις **προσπελάσεις** :



Άσκηση 4)

Βήμα 1ο : Εισαγωγή 1000

1000	

0 **1** **utilization=1/6, m=1, i=1**

Βήμα 2ο : Εισαγωγή 0000

0000	
1000	

0 **1** **utilization=2/6, m=1, i=1**

Βήμα 3ο : Εισαγωγή 1101

0000	
1000	1101

0 **1** **utilization=3/6, m=1, i=1**

Βήμα 4ο : Εισαγωγή 0010

0010	
0000	
1000	1101

0 **1** **utilization=4/6, m=1, i=1**

Βήμα 5ο : Εισαγωγή 0010

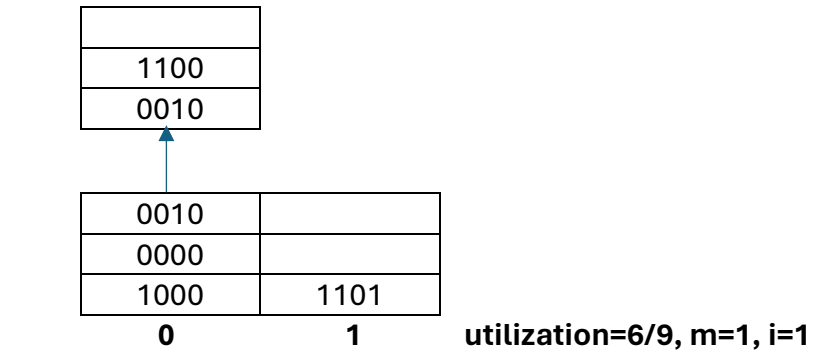
0010

0010	
0000	
1000	1101

0 **1** **utilization=5/9, m=1, i=1**

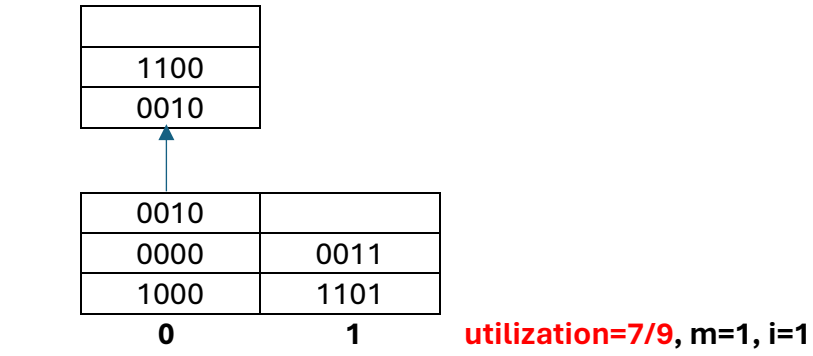
Για τον λόγο ότι ο κάδος ‘0’ γέμισε και το utilization είναι κάτω από το 70% στο Βήμα 4 βάζουμε μία σελίδα υπερχείλισης και αυτή με 3 εγγραφές.

Βήμα 6ο : Εισαγωγή 1100



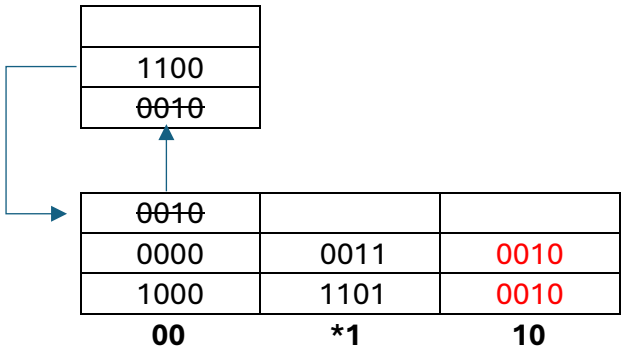
utilization=6/9, m=1, i=1

Βήμα 7ο : Εισαγωγή 0011



utilization=7/9, m=1, i=1

Επειδή με την εισαγωγή του 0011 το utilization γίνεται 7/9 = 77,7% > 70% αυξάνουμε το m, που παίρνει την αμέσως επόμενη δυαδική τιμή, η οποία είναι η 10, οπότε πρέπει να αυξήσουμε και την τιμή του του i κατά 1. Σε αυτό το σημείο μπορούμε να αναπροσαρμόσουμε τους κάδους και παρατηρούμε παρακάτω ότι δεν χρειάζεται πλέον η σελίδα υπερχείλισης.



Οπότε οι ανανεωμένοι κάδοι φαίνονται παρακάτω :

1100		
0000	0011	0010
1000	1101	0010
00	***1**	**10**

utilization=7/9, m=10, i=2

Παρατηρούμε ότι το utilization παραμένει πάνω από 70% οπότε πρέπει να αυξήσουμε το m ακόμα μία φορά. Θα γίνει 11 και το i θα παραμείνει το ίδιο.

1100			
0000	~~0011~~	0010	
1000	1101	0010	0011
00	**01**	**10**	**11**

Οπότε οι ανανεωμένοι κάδοι φαίνονται παρακάτω :

1100				
0000		0010		
1000	1101	0010	0011	
00	**01**	**10**	**11**	**utilization=7/12, m=11, i=2**

Βήμα 8ο : Εισαγωγή 1111

1100				
0000		0010	1111	
1000	1101	0010	0011	
00	**01**	**10**	**11**	**utilization=8/12, m=11, i=2**

Βήμα 9ο : Εισαγωγή 0110

1100		0110		
0000		0010	1111	
1000	1101	0010	0011	
00	**01**	**10**	**11**	**utilization=9/12, m=11, i=2**

Βλέπουμε ξανά ότι το utilization γίνεται 9/12 = 75% > 70% οπότε πρέπει πάλι να αυξήσουμε την τιμή του m στο επόμενο δυαδικό αριθμό. Άρα η τιμή του m γίνεται 100 και σε αυτό το σημείο πρέπει να αυξηθεί και η τιμή του i, η οποία γίνεται 3.

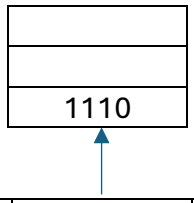
~~1100~~		0110		
0000		0010	1111	
1000	1101	0010	0011	1100
000	***01**	***10**	***11**	**100**

Οπότε οι ανανεωμένοι κάδοι φαίνονται παρακάτω :

		0110		
0000		0010	1111	
1000	1101	0010	0011	1100
000	***01**	***10**	***11**	**100**

utilization=9/15, m=100,i=3

Βήμα 10ο : Εισαγωγή 1110



		0110		
0000		0010	1111	
1000	1101	0010	0011	1100
000	***01**	***10**	***11**	**100**

utilization=10/18, m=100,i=3

Επειδή ο κάδος *10 γέμισε χρησιμοποιούμε μία σελίδα υπερχείλισης με 3 εγγραφές.

Άσκηση 5)

Μας δίνεται ότι η συνάρτηση κατακερματισμού h1 στο πεδίο 'a' χρησιμοποιεί N bits και η h2 στο 'b' χρησιμοποιεί 10-N bits.

1. **a)** Για τις ερωτήσεις τύπου Q1, δίνεται η τιμή του πεδίου 'a' και χρειάζονται να βρεθούν όλες οι εγγραφές στη σχέση R που έχουν ίσο πεδίο 'a'. Γνωρίζουμε την τιμή του 'a' και θα προσπελάσουμε το μπλοκ του, όμως δεν γνωρίζουμε την τιμή του 'b' οπότε θα πρέπει να υπολογίσουμε όλες τις πιθανές τιμές των εναπομεινάντων 10-N bits, οι οποίες είναι $2^{(10-N)}$. Οπότε θα πρέπει να προσπελάσουμε **$2^{(10-N)}$ μπλοκ** για να απαντήσουμε ερωτήσεις τύπου Q1.

   b) Για τις ερωτήσεις τύπου Q1, δίνεται η τιμή του πεδίου 'b' και χρειάζονται να βρεθούν όλες οι εγγραφές στη σχέση R που έχουν ίσο πεδίο 'b'. Γνωρίζουμε την τιμή του 'b' και θα προσπελάσουμε το μπλοκ του, όμως δεν γνωρίζουμε την τιμή του 'a' οπότε θα πρέπει να υπολογίσουμε όλες τις πιθανές τιμές των εναπομεινάντων N bits, οι οποίες είναι 2^N. Οπότε θα πρέπει να προσπελάσουμε **2^N μπλοκ** για να απαντήσουμε ερωτήσεις τύπου Q2.

2. Για να υπολογίσουμε τον μέσο όρο των μπλοκ που πρέπει να προσπελαστούν για να απαντηθούν οι ερωτήσεις (Q1 και Q2) στη σχέση R, χρειάζεται να λάβουμε τον ζυγισμένο μέσο όρο του αριθμού των μπλοκ που ελέγχονται για κάθε τύπο ερώτησης.

Το 30% των ερωτήσεων είναι τύπου Q1, και για αυτές τις ερωτήσεις, χρειάζεται να προσπελαστούν $\mathbf{2^{(10-N)}}$ **μπλοκ** κατά μέσο όρο. Το 70% των ερωτήσεων είναι τύπου Q2, και για αυτές τις ερωτήσεις, χρειάζεται να προσπελαστούν $\mathbf{2^{N}}$ **μπλοκ** κατά μέσο όρο. Ο μέσος όρος των μπλοκ λοιπόν που χρειάζεται να προσπελαστούν για να απαντηθούν οι ερωτήσεις (Q1 και Q2) στη σχέση R είναι: $\boxed{\mathbf{(0.3 * 2^{(10-N)}) + (0.7 * 2^{N})}}$**.**