**ΣΤΕΛΙΟΣ ΚΟΚΚΟΚΥΡΗΣ - 3160063**

# ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ
# ΔΕΥΤΕΡΗ ΣΕΙΡΑ ΑΣΚΗΣΕΩΝ

## Άσκηση 1

Γνωρίζουμε ότι T(R1) = 20000 εγγραφές, T(R2) = 50000 εγγραφές, B(R1) = 20000 / 100 = 200 σελίδες, B(R2) = 50000 / 100 = 500 σελίδες και M = 51 σελίδες.

1) α) Ως εξωτερική σχέση επιλέγουμε την μικρότερη εκ των R1,R2, δηλαδή την R1.Το κόστος (σε I/O) της σύζευξης R1 $\bowtie$ R2 είναι :

Cost = B(R1) + ceil[ B(R1) / (M - 1) ] * B(R2) = 200 + (200 / 50) * 500 = 200 + 4 * 500 = 200 + 2000 = 2200 I/O's

β) Αν (B(R1) + B(R2)) <= $M^2$ τότε θα χρησιμοποιήσουμε τον efficient SMJ.Έχουμε, (200 + 500) <= $51^2$ δηλαδή 700 <= 2601 ισχύει.
Άρα, Cost = 3 * (B(R1) + B(R2)) = 3 * (200 + 500) = 3 * 700 = 2100 I/O's

γ) Πρέπει min(B(R1),B(R2)) <= $M^2$ δηλαδή 200 <= 2601 ισχύει.
Άρα, Cost = 3 * (B(R1) + B(R2)) = 3 * (200 + 500) = 3 * 700 = 2100.

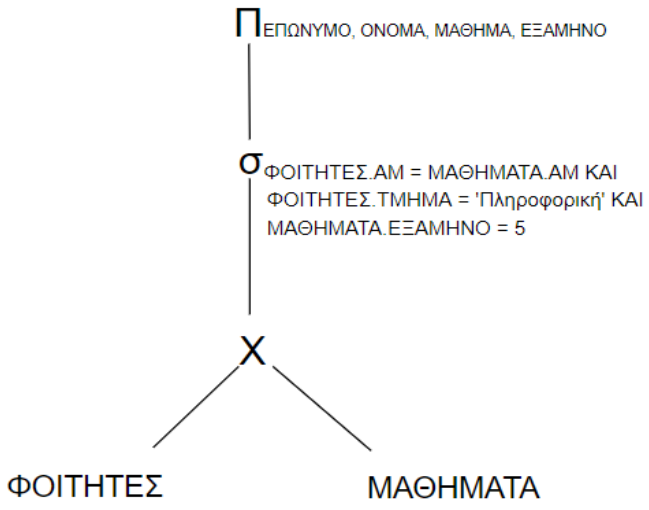
2) Για την παραπάνω σύζευξη το ελάχιστο κόστος που μπορούμε να έχουμε ισούται με 700 I/O's με χρήση του αλγορίθμου NLJ.Για την επίτευξη του ελάχιστου δυνατού κόστους το μέγεθος του ενταμιευτή θα πρέπει να είναι >= 200.Σε αυτή την περίπτωση το ceil[ B(R1) / (M - 1) ] θα είναι πάντα ίσο με 1 (δηλαδή το μικρότερο δυνατό αριθμό που θα μπορούσαμε να έχουμε δεδομένου του ceil).

Άρα, minCost = B(R1) + ceil[ B(R1) / (M - 1) ] * B(R2) = 200 + (200 / 200) * 500 = 200 + 1 * 500 = 200 + 500 = 700 I/O's.

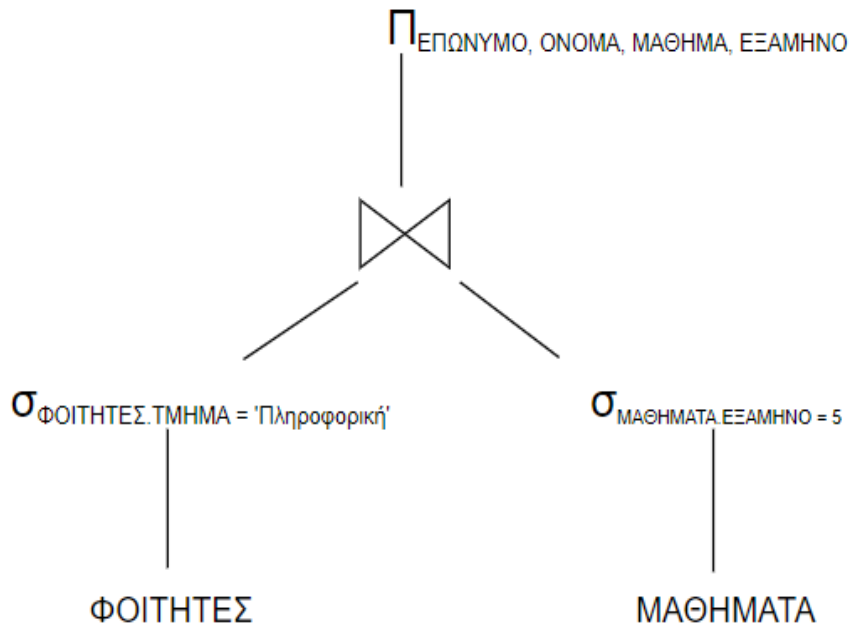
3) Δεδομένου ότι το γνώρισμα c είναι πρωτεύον κλειδί στη σχέση R2 και ξένο κλειδί στη σχέση R1, τότε το πλήθος των εγγραφών του αποτελέσματος της σύζευξης R1 $\bowtie$ R2, έστω T(W), όπου W η σχέση που προκύπτει από την σύζευξη των R1,R2, θα ισούται με T(R1) αφού στη χειρότερη περίπτωση κάθε εγγραφή της σχέσης R1 κάνει join με μία ακριβώς εγγραφή της σχέσης R2 με βάση την τιμή του κλειδιού c.Άρα T(W) = T(R1).Το πλήθος των σελίδων που απαιτούνται για την αποθήκευση των εγγραφών ισούται με B(W) = T(W) / T(R1) = 20000 / 100 = 200 σελίδες.

## Άσκηση 2

1)



2)




3) Γνωρίζουμε ότι Τ(ΦΟΙΤΗΤΕΣ) = 20000, Β(ΦΟΙΤΗΤΕΣ) = 1000, Τ(ΜΑΘΗΜΑΤΑ) = 100000, Β(ΜΑΘΗΜΑΤΑ) = 2500, V(ΦΟΙΤΗΤΕΣ, ΤΜΗΜΑ) = 20, V(ΜΑΘΗΜΑΤΑ, ΕΞΑΜΗΝΟ) = 10 και Μ = 16.

Έστω $X = \sigma_{\text{ΦΟΙΤΗΤΕΣ.ΤΜΗΜΑ = 'Πληροφορική'}}$
Τ(X) = Τ(ΦΟΙΤΗΤΕΣ) / V(ΦΟΙΤΗΤΕΣ.ΤΜΗΜΑ) = 20000 / 20 = 1000
Τ(ΦΟΙΤΗΤΕΣ) / Β(ΦΟΙΤΗΤΕΣ) = 20000 / 1000 = 20 records/page
Β(X) = 1000 / 20 = 50
Λόγω Clustering index στο X, Cost(X) = 50 = Β(X)

Έστω $Y = \sigma_{\text{ΜΑΘΗΜΑΤΑ.ΕΞΑΜΗΝΟ = 5}}$
Τ(Y) = Τ(ΜΑΘΗΜΑΤΑ) / V(ΜΑΘΗΜΑΤΑ, ΕΞΑΜΗΝΟ) = 100000 / 10 = 10000
Τ(ΜΑΘΗΜΑΤΑ) / Β(ΜΑΘΗΜΑΤΑ) = 100000 / 2500 = 40 records/page
Β(Y) = 10000 / 40 = 250
Λόγω non-Clustering index στο Y, Cost(Y) = 10000 = Τ(Y)

α) Αν (Β(X) + Β(Y)) <= $M^2$ τότε μπορούμε να χρησιμοποιήσουμε τον efficient SMJ.Έχουμε, (50 + 250) <= $16^2$ δηλαδή 300 <= 256 δεν ισχύει.Οπότε θα χρησιμοποιήσουμε τον απλό SMJ αφού max{Β(X),Β(Y)} <= $M^2$ είναι 250 <= 256 ισχύει.Άρα Cost = 5 * (Β(X) + Β(Y)) + Cost(X) + Cost(Y) = 5 * (50 + 250) + 50 + 10000 = 5 * 300 + 10050 = 1500 + 10050 = 11550 I/O's

β)Επιλέγουμε ως εξωτερική σχέση την μικρότερη εκ των Β(X),Β(Y).Άρα, Cost = Cost(X) + ceil[ Cost(X) / (M - 1) ] * Cost(Y) = 50 + ceil[50 / 15] * 10000 = = 50 + ceil[ 3.3 ] * 10000 = 50 + 4 * 10000 = 50 + 40000 = 40050 I/O's

**Άσκηση 3**

α) Η επιλογή $\sigma_{\neg(\text{Πόλη} < \text{"Ιωάννινα"})}(R)$ μετατρέπεται ως εξής :
$\sigma_{(\neg \text{Πόλη} < \text{"Ιωάννινα"})}(R)$ δηλαδή $\sigma_{\text{Πόλη} >= \text{"Ιωάννινα"}}(R)$

Στην αρχική της μορφή η επιλογή του ερωτήματος (α) ήταν αρκετά χρονοβόρα διότι η βάση θα επιχειρούσε να βρει αρχικά όσες πλειάδες ικανοποιούνται από την συνθήκη Πόλη < "Ιωάννινα" και ύστερα λόγω της άρνησης θα επέλεγε εκείνες που δεν συμπεριλαμβάνονται στα αποτελέσματα της συνθήκης αυτής.Ενώ στην τελική της μορφή, όπως προέκυψε παραπάνω, η επιλογή επιχειρεί να βρει απευθείας τις πόλεις εκείνες που ικανοποιούν την συνθήκη Πόλη >= "Ιωάννινα".

β) Ομοίως με το ερώτημα (α) και σε αυτό το ερώτημα η άρνηση καθιστά την επιλογή $\sigma_{\neg (\text{Πόλη} = \text{"Ιωάννινα"})}(R)$ αργή δεδομένου και του ευρετηρίου B+ δέντρου στο γνώρισμα Πόλη.Η αρχική επιλογή μετατρέπεται ως εξής :

$\sigma_{(\neg \text{Πόλη} = \text{"Ιωάννινα"})}(R)$ δηλαδή $\sigma_{\text{Πόλη} <> \text{"Ιωάννινα"}}(R)$

Αρκεί, λοιπόν, να επιλέξει όλες τις πόλεις που είναι διάφορες των Ιωαννίνων.Σε αντίθεση με την αρχική επιλογή, η οποία θα έβρισκε τις πόλεις εκείνες που είναι ίσες με τα Ιωάννινα και ύστερα θα επέλεγε όσες πόλεις δεν περιέχονται σε αυτές που βρήκε.

γ)Έχουμε : $\sigma_{\neg(\text{Πόλη} < \text{"Ιωάννινα"} \vee \text{Αποθεματικό} < 50000)}(R)$ δηλαδή $\sigma_{((\neg \text{Πόλη} < \text{"Ιωάννινα"}) \wedge (\neg \text{Αποθεματικό} < 50000))}(R)$ =>
=> $\sigma_{(\text{Πόλη} >= \text{"Ιωάννινα"} \wedge \text{Αποθεματικό} >= 50000)}(R)$ => $\sigma_{\text{Αποθεματικό} >= 50000}(\sigma_{\text{Πόλη} >= \text{"Ιωάννινα"}})(R)$

Η βέλτιστη επιλογή για το ερώτημα (γ) φαίνεται παραπάνω όπως προέκυψε από την αρχική μέσω μετασχηματισμών.Παρατηρούμε ότι στην τελική επιλογή εφαρμόζεται πρώτα η συνθήκη Πόλη >= "Ιωάννινα" διότι με αυτόν τον τρόπο επιτυγχάνουμε μέγιστη απόδοση αφού στο γνώρισμα Πόλη υπάρχει ευρετήριο B+ δέντρου.Όπως και στα πρώτα δύο ερωτήματα έτσι και σε αυτό στόχος ήταν η απαλοιφή της άρνησης και η εκμετάλλευση του ευρετηρίου μεγιστοποιώντας την απόδοση της επιλογής.