



ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

5^ο εξ., υποχρ. κατευθ. ΠΣΥ / ΤΛΕΣ

Διδάσκων:
καθ. Γιάννης Θεοδωρίδης

Εργαστηριακοί βοηθοί:
Γιώργος Θεοδωρόπουλος, Γιάννης Κοντούλης, Ανδρέας Τριτσαρώλης (ΚΕΚΤ εργ. 205)

ΕΡΓΑΣΙΑ ΜΑΘΗΜΑΤΟΣ

Εισαγωγή

Στο πλαίσιο της εργασίας του μαθήματος, θα χρησιμοποιήσετε την πλατφόρμα διαχείρισης σχεσιακών βάσεων δεδομένων miniDB. Στην τρέχουσα έκδοση, η miniDB υποστηρίζει ένα σχετικά περιορισμένο ρεπερτόριο λειτουργιών-εντολών, αναφορικά με τη φυσική οργάνωση της ΒΔ (αρχεία και ευρετήρια), την επεξεργασία και βελτιστοποίηση επερωτήσεων, τη διαχείριση δοσοληψιών (ζητήματα συνδρομικότητας και ανάνηψης) αλλά και διάφορων άλλων DBMS λειτουργιών. Σε αυτή την κατεύθυνση, καλείστε να εμπλουτίσετε τον κώδικα της miniDB με τα παρακάτω tasks.

Σημειώσεις:

- Τα tasks ανήκουν σε διαφορετικά επίπεδα δυσκολίας (το επίπεδο δηλώνεται από τον αριθμό των “*” δίπλα στον τίτλο του task (1 έως 3). Tasks με δυσκολία 1* απευθύνονται κυρίως σε φοιτητές παλαιότερων ετών και φοιτητές που δεν έχουν ομάδα (ατομικά παραδοτέα). Κάθε τέτοιο task μπορεί να δηλωθεί από μέχρι 3 άτομα/ομάδες και θα **τηρηθεί σειρά προτεραιότητας**. Tasks με δυσκολία 2* και 3* απευθύνονται σε ομάδες μέχρι 3 ατόμων (βέβαια, αν κάποιος/α το επιθυμεί, μπορεί να εργαστεί και ατομικά). Όσα από αυτά έχουν δυσκολία 3* θα βαθμολογηθούν με άριστα το 7 (+2 μονάδες bonus δηλαδή στον τελικό βαθμό).
- Μια ομάδα μπορεί να επιχειρήσει να υλοποιήσει περισσότερα από 1 tasks, αν θέλει να βελτιώσει περαιτέρω τη συνολική της βαθμολογία (εξαιρούνται tasks με δυσκολία 1*).

Εκδήλωση ενδιαφέροντος:

Κάθε ομάδα καλείται να στείλει μήνυμα στις ακόλουθες διευθύνσεις ηλεκτρονικού ταχυδρομείου: gstheo@unipi.gr, ikontoulis@unipi.gr το οποίο θα έχει θέμα “Εκδήλωση ενδιαφέροντος ΣΔΒΔ miniDB 2020-21” (προτείνεται να συμπεριλαμβάνονται στο μήνυμα (.cc) όλα τα μέλη της ομάδας, ώστε να διευκολυνθεί η όποια επικοινωνία) και στο οποίο θα αναφέρονται:

1. Τα ονοματεπώνυμα των μελών της ομάδας (1-3 άτομα)
2. Τα μητρώα των μελών της ομάδας
3. Τα github profiles των μελών της ομάδας
4. Τα τρία (3) tasks που θα ενδιέφεραν περισσότερο την ομάδα (σε φθίνουσα σειρά ενδιαφέροντος). Θα γίνει προσπάθεια κάθε ομάδα να αναλάβει ένα τουλάχιστον task που είναι όσο το δυνατόν πιο “ψηλά” στη λίστα της

Αν μια ομάδα δεν επιθυμεί να χρησιμοποιήσει το github workflow, μπορεί να μην αποστείλει τα σχετικά github links. Ωστόσο, αυτό δεν συστήνεται και, κατα κανόνα, δεν θα γίνει δεκτό από φοιτητές με AM>=Π17XXX

Github Issues:

Για όλα τα παρακάτω tasks θα δημιουργηθούν και αντίστοιχα issues στο [github repository της miniDB](#). Τυχόν γενικότερες απορίες μπορούν να συζητηθούν και στα comments του εκάστοτε issue, ώστε τυχόν επεξηγήσεις να είναι publically διαθέσιμες. Ομοίως, αν υπάρξει απορία συγκεκριμένα για κάποιο task, μπορείτε να ανατρέξετε και στο σχετικό github issue ώστε να δείτε αν αυτή έχει ήδη συζητηθεί/λυθεί.

Ομάδα εργασιών 1 (File organization, Indexing, Query processing and optimization)

Στην τρέχουσα έκδοση, η miniDB υποστηρίζει μόνο αρχεία σωρού (όσον αφορά στις εναλλακτικές οργανώσεις αρχείων) και ευρετήρια τύπου B+tree (όχι στην πλήρη λειτουργικότητα). Όσον αφορά στις μεθόδους αναζήτησης, υποστηρίζει μόνο γραμμική αναζήτηση και σύνδεση Block-Nested-Loop-Join. Επίσης, δεν διαθέτει καθόλου μηχανισμούς βελτιστοποίησης. Συνεπώς, ο κώδικας της miniDB μπορεί να εμπλουτιστεί με τα παρακάτω:

Task 1.1 - Ordered files ()**

Υλοποίηση ταξινομημένων αρχείων και των παρακάτω σχετικών λειτουργιών:

1. Υλοποίηση Insert Stack στο οποίο αποθηκεύονται νέες εγγραφές. Συγκεκριμένα, τα select στον πρωτότυπο πίνακα πρέπει να λειτουργούν κανονικά, δηλαδή οι εγγραφές του insert stack πρέπει να συμπεριλαμβάνονται σε αυτόν (όταν η συνθήκη το απαιτεί)
2. Υλοποίηση συνάρτησης που ενσωματώνει το insert stack στον πίνακα (π.χ. όταν το μέγεθος του insert stack είναι αρκετά μεγάλο)

3. Υλοποίηση δυαδικής αναζήτησης

Task 1.2 - Btree refactoring ()**

Επανασχεδίαση (refactoring) του υπάρχοντος B+tree καθώς και:

1. Υλοποίηση της εντολής delete
2. Υποστήριξη για multicolumn B+tree indexes

Task 1.3 - Hash Indexes ()**

Υλοποίηση ευρετηρίου hash και σχετικών μεθόδων:

1. Hash indexes με διαφορετικά hashing functions (keys) δική σας επιλογής
2. Hash search
3. Hash join
4. Hashing visualization tool

Task 1.4 - Joins ()**

Υλοποίηση των παρακάτω αλγορίθμων σύνδεσης (joins):

1. Outer Joins
2. Index-Nested-Loop-Join
3. Sort-Merge-Join

Μπορείτε να χρησιμοποιήσετε και εξωτερικές βιβλιοθήκες που αυτοματοποιούν μέρος των joins. Ωστόσο, θα εκτιμηθεί αν οι παραπάνω υλοποιήσεις δεν χρησιμοποιούν εξωτερικές βιβλιοθήκες και “έτοιμο” κώδικα.

Task 1.5 - Cost-based optimization (*)**

Υλοποίηση ενός λειτουργικού τρόπου μέτρησης της απόδοσης των διάφορων I/O functions. Αυτός μπορεί να περιέχει απλή μέτρηση χρόνου (να παρέχεται εξωτερικά και όχι αλλάζοντας κάθε υπάρχουσα μέθοδο) αλλά θα πρέπει επίσης να αναλύει και τον εκάστοτε χρόνο.

Ομάδα εργασιών 2 (Διαχείριση Δοσοληψιών, Αρχιτεκτονικές ΣΔΒΔ)

Στην τρέχουσα έκδοση, η miniDB υποστηρίζει μια απλοϊκή προσέγγιση του ελέγχου συνδρομικότητας που επιτυγχάνεται με κλείδωμα/ξεκλείδωμα σε επίπεδο πινάκων. Επίσης, είναι μια κλασική κεντριοποιημένη υλοποίηση. Συνεπώς, ο κώδικας της miniDB μπορεί να εμπλουτιστεί με τα παρακάτω:

Task 2.1 - Locks (*)

Υλοποίηση διαφορετικών ειδών κλειδωμάτων (locks) όπως:

1. Exclusive/Share table wide
2. Exclusive/Share row wide

Συγκεκριμένα, η miniDB στην παρούσα φάση υποστηρίζει έναν τύπο κλειδώματος, τον exclusive table wide. Καλείστε να υλοποιήσετε τους υπόλοιπους τύπους, καθώς και να μεριμνήσετε για τις απαραίτητες αλλαγές στα σχετικά meta tables.

Task 2.2 - Journals ()**

Υλοποίηση ενός persistent log για κάθε database ή table σε μορφή WAL, που θα υποστηρίζει εντολές τύπου rollback, καθώς και αναλυτικό logging. Συγκεκριμένα, το log αυτό θα είναι σε θέση αφενός να συγκρατεί το ιστορικό εντολών μιας βάσης, αφετέρου, σε περίπτωση που ο χρήστης το θελήσει, να κάνει rollback N εντολές πριν. Το default max N το αποφασίζετε εσείς λαμβάνοντας υπόψη τον όγκο των δεδομένων, δίνοντας παράλληλα τη δυνατότητα στον χρήστη να μπορεί να το αλλάζει, αν κρίνεται απαραίτητο.

Task 2.3 - Table partitioning and Inheritance + Distributing (*)**

Μέρος (α) υλοποίηση των παρακάτω:

1. Table Inheritance
2. Table Partitioning

Μέρος (β) υλοποίηση των παρακάτω:

1. Distribute a DB
2. Distribute a partitioned table

Task 2.4 - Server - Client (*)**

Υλοποίηση αρχιτεκτονικής που χρησιμοποιεί θύρες (ports) για την επικοινωνία ενός πελάτη (client) με τη ΒΔ. Αυτό προϋποθέτει τη δημιουργία ενός πελάτη (client) και ενός εξυπηρετητή (server) που θα επικοινωνούν μέσω μιας θύρας. Ο client θα “ζητά” μια επερώτηση (query) και ο server θα του επιστρέφει τα αποτελέσματα. Ενδεικτικά, μπορείτε να υλοποιήσετε και έναν πολύ μικρό SQL compiler (π.χ. μόνο μια select), ώστε να δείξετε ότι λειτουργεί η υλοποίησή σας. Για παράδειγμα, ο client θα στέλνει ένα ‘select * from table’ και ο χρήστης θα λαμβάνει ως απάντηση ένα αποτέλεσμα. Το συγκεκριμένο ερώτημα μπορεί να υλοποιηθεί και σε γλώσσα Go, σε περίπτωση που το επιθυμείτε.

Ομάδα εργασιών 3 (Γενικά - Διάφορα)

Προφανώς, η miniDB μπορεί να εμπλουτιστεί με πολλές άλλες λειτουργίες, όπως οι παρακάτω:

Task 3.1 - Extra I/O and Primary Key rethink (*)

Υλοποίηση των παρακάτω λειτουργιών (functionalities):

1. Group by
2. Select Distinct

Επίσης, καλείστε να επανασχεδιάσετε τον τρόπο που αποθηκεύεται το πρωτεύον κλειδί (primary key), καθώς και να εντάξετε τη δυνατότητα ορισμού multicolumn primary keys.

Task 3.2 - Privileges (*)

Δυνατότητα υποστήριξης διαφορετικών χρηστών (users) στη miniDB με διαφορετικά δικαιώματα πρόσβασης (privileges). Συγκεκριμένα:

1. Διαφορετικοί χρήστες και ομάδες χρηστών
2. Δικαιώματα χρηστών ανά βάση
3. Δικαιώματα χρηστών ανά πίνακα (table)

Task 3.3 - SQL compiler ()**

Υλοποίηση ενός SQL compiler για την υποστήριξη των I/O εντολών που υπάρχουν στην τρέχουσα έκδοση της miniDB. Ενδεικτικά, οι εντολές που μπορούν να υποστηριχθούν είναι οι εξής:

1. Select From Where
2. Update Where
3. Insert Into
4. Delete From Where
5. Create/Drop Table
6. Create Database
7. Select From Join
8. Create Index On

Task 3.4 - Reimplementing performance critical functions (*)**

Εκ νέου υλοποίηση των μεθόδων που επηρεάζουν σε μεγάλο βαθμό την απόδοση της miniDB, χρησιμοποιώντας μια γρηγορότερη (κατα προτίμηση compiled) γλώσσα της επιλογής σας. Ενδεικτικά, μπορείτε να υλοποιήσετε την αποθήκευση και τη φόρτωση αρχείων βάσεων σε ένα format εκτός του pkl. Συνίστανται γλώσσες όπως C/C++/Rust.

=====

Παραδοτέο εργασίας:

Η εργασία θα παραδοθεί αποκλειστικά μέσω email στα: gstheo@unipi.gr, ikontoulis@unipi.gr, ytheod@unipi.gr. Το email θα πρέπει να έχει θέμα “Παραδοτέο ΣΔΒΔ miniDB 2020-21” και στο περιεχόμενό του θα πρέπει να αναγράφονται:

1. Τα ονοματεπώνυμα των μελών της ομάδας (1-3 άτομα)
2. Τα μητρώα των μελών της ομάδας
3. Το/τα task(s) που υλοποίησε η ομάδα
4. Το github link του miniDB fork στο οποίο έχουν υλοποιηθεί τα ζητούμενα του task

Αν μια ομάδα δεν επιθυμεί να χρησιμοποιήσει το github workflow, μπορεί να αποστείλει ένα zip με το ίδιο περιεχόμενο. Ωστόσο αυτό δεν συστήνεται και κατα κανόνα δεν θα γίνει δεκτό από φοιτητές με AM>=Π17XXX

Στα πλαίσια της Continuous Integration φιλοσοφίας του open source, όσοι φοιτητές επιθυμούν μπορούν να επικοινωνούν με τους διδάσκοντες είτε μέσω email, είτε μέσω του Discord Server (<https://discord.gg/UZQXxKCz3w>), ώστε να λαμβάνουν ανατροφοδότηση για τη μέχρι τώρα υλοποίησή τους.

Χρόνος παράδοσης:

Προθεσμία παράδοσης: η τελευταία ημέρα της εξεταστικής περιόδου Ιανουαρίου 2021 ή Σεπτεμβρίου 2021 (δηλαδή, γίνονται δεκτές και εργασίες στην επαναληπτική εξεταστική του Σεπτεμβρίου).

Απορίες σχετικά με την άσκηση

Για οποιαδήποτε απορία αφορά στην άσκηση μπορείτε να απευθυνθείτε στους εργαστηριακούς βοηθούς.

Ζητήματα δεοντολογίας

Είναι προφανές ότι η βαθμολογία πρέπει να αντικατοπτρίζει το επίπεδο της γνώσης που αποκόμισε ο φοιτητής μέσα από το μάθημα και κατάφερε να μεταφέρει αυτή τη γνώση στην εργασία. Για να εξασφαλιστεί όσο είναι δυνατό η παραπάνω αρχή, (α) σε περίπτωση αντιγραφής οι εμπλεκόμενες εργασίες μηδενίζονται, (β) σε περίπτωση αμφιβολίας για το κατά πόσο η ομάδα που αναγράφεται ήταν εκείνη που ανέπτυξε την εργασία, ενδέχεται να της ζητηθεί να την παρουσιάσει για τυχόν διευκρινίσεις.

Καλή Επιτυχία!