

Bloom Filter

Αρχεία: bloom_filter.h, bloom_filter.c

Δομές: Bloom_Filter: Είναι μια δομή που περιέχει το έναν πίνακα int και το μέγεθός του.

Συναρτήσεις: Έχουν υλοποιηθεί συναρτήσεις για την αρχικοποίηση ενός bloom filter, την εισαγωγή μιας εγγραφής σ' αυτό και την αναζήτηση.

//=====

Skip List

Αρχεία: skiplist.h, skiplist.c

Δομές: Skip_List: λειτουργεί σαν header και περιέχει τον μέγιστο αριθμό των επιπέδων που θα υπάρχουν στην λίστα (int levels), έναν counter για τον αριθμό των στοιχείων που περιέχονται (int count), και έναν πίνακα δεικτών σε κόμβους της λίστας που λειτουργεί σαν πίνακας από headers στην λίστα κάθε επιπέδου (SLNode** level_array).

SLNode: είναι η δομή κόμβου της λίστας, περιέχει τον αριθμό των επιπέδων στα οποία συμμετέχει (int levels), έναν πίνακα δεικτών στους επόμενους κόμβους του αντίστοιχου (SLNode** level_array) και έναν δείκτη στην αντίστοιχη εγγραφή (Entry* entry).

Συναρτήσεις: Έχουν υλοποιηθεί συναρτήσεις για την αρχικοποίηση των level_array, και των δομών, την εισαγωγή και την αναζήτηση εγγραφών, για την αφαίρεση και την διαγραφή κόμβων και της Skip List, και την εκτύπωση. Τα επίπεδα κάθε κόμβου υπολογίζεται τυχαία κατά την αρχικοποίηση του.

Κατά την αναζήτηση και την εισαγωγή γίνεται διάσχιση της λίστας από το μεγαλύτερο επίπεδο στο μικρότερο μέχρι να βρεθεί η εγγραφή ή, αντίστοιχα, η θέση που θα πρέπει να γίνει η εισαγωγή της. Για την εισαγωγή φτιάχνεται ένας πίνακας update_array που αποθηκεύει τους κόμβους του κάθε επιπέδου που θα πρέπει να ενημερωθούν για να εισαχθεί το στοιχείο στο επίπεδο.

//-----

Δομές Αποθήκευσης Δεδομένων Εγγραφών

Αποθήκευση Εγγραφών

Entry

Αρχεία: entries.h, entries.c

Δομές: Entry: περιέχει δείκτες σε κόμβους πινάκων κατακερματισμού, έναν για κάθε πεδίο της εγγραφής. Για τα πεδία virusName και date υπάρχουν πίνακες από δείκτες σε hash tables έτσι ώστε κάθε εγγραφή να μπορεί να κρατάει δεδομένα για παραπάνω από έναν ιό. Μεταξύ των δύο πινάκων Virus και Date υπάρχει αντιστοιχία στις θέσεις, δηλ η ημερομηνία εμβολιασμού για τον Ιό που βρίσκεται στο Virus[i] είναι στο Date[i]. Σε περίπτωση που κάποιος δεν έχει εμβολιαστεί, το Date[i] είναι NULL. Το virus_count είναι μετρητής των ιών που υπάρχουν στην εγγραφή.

Συναρτήσεις: Έχουν υλοποιηθεί συναρτήσεις για την αρχικοποίηση της δομής, που όταν δημιουργείται όλα τα στοιχεία είναι κενά, και συμπληρώνονται με χρήση ξεχωριστών συναρτήσεων (πχ. Entry_Add_Id, Entry_Add_Virus κλπ). Υπάρχουν συναρτήσεις που επιστρέφουν κάποια από τα πεδία της εγγραφής (πχ. Entry_Get_Country, Entry_Get_Date), και μία που κάνει αναζήτηση στον πίνακα με τους ιούς της εγγραφής και επιστρέφει την θέση του (Entry_Get_Virus_position).

//=====

Αποθήκευση Δεδομένων των Πεδίων των Εγγραφών

Hash_Data

Αρχεία: dataprocessing.h, dataprocessing.c

Δομές: Hash_Data: περιέχει δείκτες σε πίνακες κατακερματισμού, έναν για κάθε πεδίο της εγγραφής, και αποθηκεύει το μέγεθος του bloom filter που έχει δοθεί.

Συναρτήσεις: Έχουν υλοποιηθεί συναρτήσεις για την αρχικοποίηση της δομής, μία συνάρτηση για την ανάγνωση του αρχείου και την δημιουργία εγγραφών (DataProcessing), συναρτήσεις για την εισαγωγή πεδίων εγγραφών στους hash της HashData, και συναρτήσεις εκτύπωσης και διαγραφής.

DataProcessing: διαβάζει γραμμή-γραμμή το αρχείο εισόδου, δημιουργεί μια εγγραφή, και κάθε λέξη-πεδίο την εισάγει στο αντίστοιχο hash. Εδώ καλούνται οι συναρτήσεις Entry_Add_... που συνδέουν τα πεδία κάθε εγγραφής με στοιχεία που είναι αποθηκευμένα στους hash της Hash_Data. Ακόμα, δημιουργεί και αρχικοποιεί τα αντιστοιχα skiplists και bloomfilters αν χρειάζεται.

//=====

Hash

Αρχεία: hash.h, hahs.c

Δομές: Hash, Bucket...

Node: περιέχει δείκτες για το collision list του bucket, δείκτη στο bloom filter της εγγραφής, δείκτες στις αντίστοιχες skip lists (Skip_List* vaccinated_persons, Skip_List* not_vaccinated_persons), έναν counter για τον αριθμό των συνολικών εγγραφών που υπάρχουν και στις δύο skip lists και ένα δείκτη σε void που αξιοποιείται με διαφορετικούς τρόπους για την απάντηση ερωτημάτων (data4questions).

Τα τελευταία στοιχεία, που δεν έχουν σχέση με την κλασσική υλοποίηση ενός hash, έχουν νόημα και χρησιμοποιούνται μόνο στα hash για τους ιούς (γτ μόνο εκεί υπάρχουν skiplists και bloomfilters), και το data4questions χρησιμοποιείται και στον hash του id για το insertCitizenRecord.

//=====

Ερωτήματα

Αρχεία: questions.h, questions.c

Έχουν υλοποιηθεί συναρτήσεις για την απάντηση των ερωτημάτων.

VaccineStatus : Αν δοθεί ονομασία ιού, ψάχνει τον ιό στον hash, ψάχνει για το id στην skiplist με τα εμβολιασμένα άτομα του ιού, και αν δεν τον βρεί τυπώνει μηνμ ότι δεν είναι εμβολιασμένος, αλλιώς ότι είναι.

Αν δεν δοθεί κάνει την ίδια διαδικασία για κάθε ιό στον hash.

PopulationStatus: Αν δεν δοθεί ονομασία χώρας, ψάχνει για το id στην skiplist με τα εμβολιασμένα άτομα , συγκρίνει ημερομηνίες εμβολιασμού και τις εκτυπώνει.

PopStatusByAge: λειτουργεί με όμοιο τρόπο με την προηγούμενη, απλά χωρίζει τις εγγραφές στις ζητούμενες κατηγορίες και αποθηκεύει το πλήθος σε μια δομή countByAge απο όπου εκτυπώνονται μετά.

InsertCitizenRecord: Γίνεται εισαγωγή νέου id στο αντίστοιχο σκιπ λιστ και στο μπλουμ φιλτερ.

Όταν γίνεται εισαγωγή εμβολιασμένου, γίνεται μετά αναζήτηση στις λίστες των μη εμβολιασμένων και αν βρεθεί εκεί διαγράφεται.

Σε περίπτωση id που δεν υπάρχει απο πριν στα αρχεία, δημιουργείται νέα εγγραφή και εισαγεται κανονικά στις σωστές δομές.