

Προγραμματισμός Συστήματος, Εαρινό Εξάμηνο 2021-2022 - 1^η Εργασία

Παράδοση (στο eclass): Τρίτη 5/4/2022, 23:59

Η πρώτη εργασία του μαθήματος έχει σκοπό να σας εξασκήσει στην γλώσσα προγραμματισμού C, που είναι και η βασική γλώσσα προγραμματισμού του μαθήματος.

Πιο συγκεκριμένα, στα πλαίσια αυτής της εργασίας θα υλοποιήσετε ένα σύνολο συναρτήσεων που διαχειρίζονται αρχεία εγγραφών, με σκοπό να αναθέσετε αριθμητικά αναγνωριστικά (ids) σε πεδία εγγραφών, καθώς και να τα διαμορφώσετε, για να βολεύει η επεξεργασία τους.

Παράδειγμα αρχείου εισόδου (εγγραφές της μορφής { user item rating timestamp }):

```
Costas tigani 4 1234567
Nikos sompa 3 1123456
Nikos kourtines 1 1122334
Vasilis tigani 5 1111111
Costas kourtines 2 2233445
Costas friteza 1 2233445
```

Τα ids που θα αναθέσει (σειριακά) θα είναι

Users:

Costas -> 0

Nikos -> 1

Vasilis -> 2

Items:

tigani -> 0

sompa -> 1

kourtines -> 2

friteza -> 3

Παράδειγμα αρχείου εξόδου (ομαδοποιημένο ανά χρήστη, όπου πριν τις εγγραφές κάθε χρήστη θα προηγείται μια γραμμή με το id του και το πλήθος των εγγραφών του που ακολουθούν – επιπλέον, η 1^η γραμμή του αρχείου θα αναγράφει πόσους χρήστες έχει συνολικά το αρχείο, καθώς και πόσα αντικείμενα):

```
3 4
0 3
0 0 4 1234567
0 2 2 2233445
0 3 1 2233445
1 2
1 1 3 1123456
1 2 1 1122334
2 0
2 1 5 1111111
```

Προσοχή: Στο αρχείο εξόδου θα βρίσκονται μόνο χρήστες που έχουν τουλάχιστον N αξιολογήσεις (παράμετρος εισόδου) και η χρονική διαφορά της 1^{ης} με την τελευταία αξιολόγηση του κάθε χρήστη, θα απέχουν τουλάχιστον S δευτερόλεπτα (παράμετρος εισόδου). Αν το N=1, τότε το S δε θα λαμβάνεται υπόψη (προφανώς).

Ζητούμενες Συναρτήσεις

int initialize (int B);

Η συνάρτηση αυτή καλείται για να δημιουργήσει και να αρχικοποιήσει τους 2 κατακερματισμένους πίνακες συνδεδεμένων λιστών (ένας για users και ένας για items), όπου καθένας θα αποτελείται από B κάδους. Επιστρέφει 1 αν ολοκληρωθεί ομαλώς, αλλιώς -1 (σε αυτήν την περίπτωση, εκτυπώνει και τι λάθος συνέβη).

int create_num_id (char* input_file, char* output_file, int N, int S, int* plithos_users, int* plithos_items);

Η συνάρτηση αυτή ανοίγει το αρχείο με όνομα input_file, όπου περιέχει γραμμές (εγγραφές) της μορφής {user, item, rating, timestamp} και δημιουργεί το αρχείο output_file, όπως στο παραπάνω παράδειγμα. Επιστρέφει 1 αν επιτύχει, αλλιώς -1 (μαζί με εκτύπωση λάθους). Επίσης, στην 5^η και 6^η παράμετρο επιστρέφει το πλήθος των users και των items, αντίστοιχα, που έχει το αρχείο εξόδου.

int return_id (char* str, char attribute);

Η συνάρτηση αυτή δέχεται ως είσοδο μία συμβολοσειρά (αρχικό id), καθώς και σε ποιο χαρακτηριστικό αντιστοιχεί ('u' για user και 'i' για item) και επιστρέφει το αριθμητικό id που αντιστοιχεί σε αυτήν, αλλιώς επιστρέφει -1 (σε αυτήν την περίπτωση, εκτυπώνει και τι πήγε στραβά).

int hashfunction(char* str);

Η συνάρτηση αυτή δέχεται ως είσοδο μία συμβολοσειρά (αρχικό id) και επιστρέφει τη θέση που της αντιστοιχεί στον πίνακα κατακερματισμού (0 έως B-1).

void destroy();

Η συνάρτηση αυτή καλείται για να αποδεσμεύσει τις δομές (δυναμική μνήμη) που δεσμεύτηκαν για την εργασία.

Λεπτομέρειες εργασίας: Η εργασία αυτή μπορεί να υλοποιηθεί από 1 έως 2 άτομα (στην περίπτωση που η εργασία υλοποιηθεί από 2 άτομα, θα πρέπει και οι 2 να γνωρίζουν το σύνολο της εργασίας).

Προγραμματιστικός Έλεγχος: Θα ακολουθήσει προγραμματιστικός έλεγχος στα εργαστήρια του τμήματος, από τον διδάσκοντα του μαθήματος, σε ημερομηνία και ώρα που θα σας γνωστοποιηθεί μέσω ανακοίνωσης στο eclass.

Περιβάλλον υλοποίησης: Linux (gcc).

Παραδοτέο: Ένα zip που θα περιέχει το .h και το .c με τις συναρτήσεις που υλοποιήσατε, καθώς και αρχείο κειμένου (doc ή αντίστοιχο) **readme**, με σχόλια (δε θα παραδώσετε main, θα σας δοθεί στο eclass), δηλαδή **σύνολο 3 αρχεία**.

Κανόνες εργασίας:

- Για να βαθμολογηθεί η εργασία, ο κώδικας θα πρέπει τουλάχιστον να μεταγλωττίζεται (compile).
- Απαγορεύεται η αποθήκευση στη μνήμη όλου του αρχείου εισόδου (αν αυτό δεν είναι αναγκαίο).
- Απαγορεύεται η σάρωση/διάβασμα του αρχείου εισόδου παραπάνω από 2 φορές.
- Αν κάποια κομμάτια του κώδικά σας προέλθουν από δημόσια πηγή (βιβλία, διαφάνειες, ιστότοπο), θα πρέπει να δώσετε ρητή και λεπτομερή αναφορά στην πηγή αυτή, καθώς και να εξηγήσετε πως τα χρησιμοποιήσατε.
- Η γλώσσα υλοποίησης της εργασίας θα είναι η C (όχι η Java, ούτε η Python).
- Εργασίες δε θα γίνονται δεκτές μέσω email (η κατάθεση θα γίνεται στο eclass).
- Η παρουσία σας στην εξέταση είναι υποχρεωτική. Μη παρουσία στην προγραμματισμένη εξέταση της εργασίας, συνεπάγεται μηδενισμό της εργασίας.
- Αν διαπιστωθεί, κατά την εξέταση, ότι φοιτητής δεν γνωρίζει να εξηγήσει την εργασία του (όλη ή μέρος αυτής), μηδενίζεται στο μάθημα.
- Αν διαπιστωθεί αντιγραφή μεταξύ εργασιών, μηδενίζονται στο μάθημα όλες οι εμπλεκόμενες ομάδες.