

ΕΡΓΑΣΙΑ 2

ΕΡΓΑΣΤΗΡΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Ι, ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2023-2024

Προθεσμία: 15/1/2024, 23:59

Γενικές οδηγίες

Διαβάστε ΟΛΗ την εκφώνηση προσεκτικά πριν ξεκινήσετε να γράφετε κώδικα. Σχεδιάστε τις δομές (πίνακες) στο χαρτί για να καταλάβετε τι σχέση μεταξύ τους και πώς χρησιμοποιούνται σε κάθε στάδιο της άσκησης.

Κάθε φορά που ολοκληρώνετε ένα στάδιο, επιβεβαιώστε ότι το πρόγραμμά σας λειτουργεί σωστά, πριν προχωρήσετε παρακάτω.

Αν χρειάζεστε διευκρινίσεις ή έχετε προβλήματα, στείλτε μήνυμα στο χώρο συζητήσεων στο eclass. Προσοχή: δεν επιτρέπεται η δημοσίευση κώδικα στο eclass.

Η εργασία αυτή πρέπει να γίνει ατομικά. Μπορείτε να συζητάτε τις ασκήσεις με συμφοιτητές σας αλλά δεν επιτρέπεται η ανταλλαγή κώδικα με οποιονδήποτε τρόπο.

Η εργασία καλύπτει όλη την ύλη του μαθήματος μέχρι και τη διάλεξη πάνω σε δείκτες/πίνακες ως παραμέτρους. Το πρόγραμμα σας πρέπει να επιδεικνύει κατανόηση όλων των σχετικών εννοιών και θα βαθμολογηθεί πάνω στα παρακάτω (χωρίς ιδιαίτερη σειρά):

- Ορθότητα υπολογισμών
- Αποτελεσματική χρήση κατάλληλων μεταβλητών, τελεστών, σταθερών, επαναλήψεων κτλ.
- Γενική μορφοποίηση προγράμματος (στοίχιση, ονόματα μεταβλητών και σταθερών, κτλ.)
- Σωστή δημιουργία και χρήση συναρτήσεων
- Σωστή χρήση συναρτήσεων από το string.h
- Συμμόρφωση με τις προδιαγραφές της εκφώνησης
- Αποτελεσματικά σχόλια

Μπορείτε να κατασκευάσετε περισσότερες συναρτήσεις από αυτές που ζητούνται, αλλά όχι λιγότερες.

Απαγορεύεται αυστηρά η χρήση goto και η χρήση καθολικών μεταβλητών (δηλαδή μεταβλητών δηλωμένων έξω από τη συνάρτηση main).

Μπορείτε να υποθέσετε ότι η είσοδος από το πληκτρολόγιο θα δίνεται πάντα στη σωστή μορφή. Για παράδειγμα, αν ζητείται αριθμός δε θα δοθεί συμβολοσειρά, ή αν ζητείται ακέραιος δε θα δοθεί αριθμός με υποδιαστολή.

Ξεκινήστε νωρίς! Ο προγραμματισμός είναι πάντα ΠΟΛΥ πιο χρονοβόρος από ότι περιμένετε.

Εισαγωγή

Κάθε φορά που βλέπετε ένα βίντεο σε social media ή κάνετε κάποια αγορά σε online κατάστημα, η εφαρμογή σας συστήνει εναλλακτικά κλιπ ή προϊόντα τα οποία πιστεύει ότι θα σας ενδιαφέρουν.

Ένας απλός τρόπος να υλοποιηθεί ένας μηχανισμός συστηνόμενων προϊόντων είναι ο εξής: αν έχετε αξιολογήσει μια σειρά από αυτά και κάποιος άλλος χρήστης έχει παρόμοιες αξιολογήσεις, τότε είναι πολύ πιθανό να σας αρέσουν κάποια προϊόντα που άρεσαν και στον άλλο χρήστη.

Σε αυτή την άσκηση θα γράψετε ένα πρόγραμμα το οποίο διαβάσει τις αξιολογήσεις ενός μεγάλου αριθμού χρηστών για ένα σύνολο βιβλίων και τις χρησιμοποιεί για να συστήσει σε κάποιον από αυτούς ένα βιβλίο που δεν έχει διαβάσει ακόμη και πιθανώς να του αρέσει.

Οι αξιολογήσεις εκφράζονται ως πραγματικοί αριθμοί με τιμές -5 (το μισώ), -3 (δε μου άρεσε καθόλου), -1 (δε μου άρεσε πολύ), 0 (δεν το έχω διαβάσει), 1 (μέτριο), 3 (μου άρεσε), 5 (μου άρεσε πάρα πολύ).

Σας δίνονται 2 αρχεία C:

- Το hw2.h περιέχει τους ορισμούς των σταθερών BOOK_INFO_SIZE, NAME_SIZE, MAX_USERS, MAX_BOOKS.
- Στο hw2.c θα γράψετε το πρόγραμμα σας. Περιέχει ήδη την εντολή #include "hw2.h" μέσω της οποίας θα συμπεριληφθούν οι ορισμοί των παραπάνω σταθερών στο hw2.c

Μπορείτε, στα πλαίσια των ελέγχων ορθότητας των προγραμμάτων σας, να αλλάξετε τις τιμές των σταθερών που ορίζονται στο hw2.h. Δεν επιτρέπονται άλλες αλλαγές ή προσθήκη επιπλέον κώδικα σε αυτό το αρχείο. Δεν επιτρέπεται να ορίσετε εκ νέου αυτές τις σταθερές στο hw2.c.

ΜΗΝ προσπαθήσετε να γράψετε όλο το πρόγραμμα σε ένα βήμα γιατί θα κάνετε λάθη και θα σας πάρει πολύ περισσότερο χρόνο. Κάθε φορά που ολοκληρώνετε ένα στάδιο, να επιβεβαιώνετε ότι λειτουργεί σωστά πριν προχωρήσετε στο επόμενο.

Δημιουργία δομών και εισαγωγή δεδομένων

Σε αυτό το στάδιο, το πρόγραμμα διαβάσει από το πληκτρολόγιο και αποθηκεύει σε κατάλληλους πίνακες πληροφορίες για τα βιβλία, τους χρήστες και τις αξιολογήσεις που έδωσαν οι χρήστες στα βιβλία.

Κατασκευάστε τους παρακάτω πίνακες:

- Ένα δισδιάστατο πίνακα «**χρηστών**» από χαρακτήρες, με MAX_USERS γραμμές και NAME_SIZE στήλες, όπου σε κάθε γραμμή r θα αποθηκευτεί το όνομα ενός χρήστη ως συμβολοσειρά.
- Ένα δισδιάστατο πίνακα «**βιβλίων**» από χαρακτήρες, με MAX_BOOKS γραμμές και BOOK_INFO_SIZE στήλες, όπου σε κάθε γραμμή c θα αποθηκευτούν τα στοιχεία ενός βιβλίου ως συμβολοσειρά.
- Ένα δισδιάστατο πίνακα «**αξιολογήσεων**» από πραγματικούς αριθμούς, με MAX_USERS γραμμές και MAX_BOOKS+1 στήλες, όπου σε κάθε κελί [r][c] για c μέχρι και MAX_BOOKS-1 θα αποθηκευτεί η αξιολόγηση που έδωσε ο χρήστης r στο βιβλίο c, ενώ σε κάθε κελί [r][MAX_BOOKS] θα αποθηκευτεί μια διαφορετική πληροφορία που περιγράφεται [παρακάτω](#). Το όνομα του χρήστη r είναι αυτό που έχει αποθηκευτεί στη γραμμή r του πίνακα χρηστών. Το βιβλίο c είναι αυτό που έχει αποθηκευτεί στη γραμμή c του πίνακα βιβλίων.

Γράψτε μια συνάρτηση αρχικοποίησης που παίρνει ως παράμετρο τον πίνακα βιβλίων και, μέχρι είτε να γεμίσει ο πίνακας είτε να διαβαστεί η συμβολοσειρά "No_more_books", διαβάζει από το πληκτρολόγιο μια σειρά από συμβολοσειρές κάθε μία εκ των οποίων περιέχει τις πληροφορίες ενός βιβλίου, και τις αποθηκεύει στον πίνακα βιβλίων, μία ανά γραμμή. Το "No_more_books" δεν αποθηκεύεται στον πίνακα. Οι πληροφορίες κάθε ενός βιβλίου δίνονται ως μια συμβολοσειρά (χωρίς κενά). Υποθέστε ότι οι πληροφορίες ενός βιβλίου είναι πάντα το πολύ BOOK_INFO_SIZE - 1 χαρακτήρες, και το ότι δε θα δοθεί πληροφορία για το ίδιο βιβλίο περισσότερες από μία φορές. Η συνάρτηση επιστρέφει το πλήθος των βιβλίων που έχουν αποθηκευτεί στον πίνακα.

Γράψτε μια συνάρτηση αναζήτησης που παίρνει ως παραμέτρους τον πίνακα χρηστών, το πλήθος χρηστών που είναι αποθηκευμένοι σε αυτόν και το όνομα ενός χρήστη και αναζητά το όνομα του χρήστη στους ήδη υπάρχοντες στον πίνακα. Αν αυτό βρεθεί, επιστρέφει τη θέση του στον πίνακα, διαφορετικά επιστρέφει -1.

Γράψτε μια συνάρτηση αρχικοποίησης που παίρνει ως παραμέτρους τους πίνακες χρηστών και αξιολογήσεων και το πλήθος βιβλίων που είναι αποθηκευμένα στον πίνακα βιβλίων και, μέχρι είτε να γεμίσει ο πίνακας χρηστών είτε να διαβαστεί η συμβολοσειρά "-" ως όνομα χρήστη, διαβάζει από το πληκτρολόγιο τα ονόματα των χρηστών και τις αξιολογήσεις που έδωσαν και τα αποθηκεύει στους αντίστοιχους πίνακες. Η συμβολοσειρά "-" δεν αποθηκεύεται στον πίνακα χρηστών. Υποθέστε ότι τα ονόματα των χρηστών είναι πάντα το πολύ NAME_SIZE - 1 χαρακτήρες. Η συνάρτηση επιστρέφει το πλήθος των χρηστών που αποθηκεύτηκαν στον πίνακα χρηστών. Η ανάγνωση γίνεται ως εξής:

1. Διαβάζεται το όνομα ενός χρήστη και χρησιμοποιείται η συνάρτηση αναζήτησης χρήστη για να εντοπιστεί αν αυτός βρίσκεται ήδη στον πίνακα χρηστών. Εάν βρίσκεται, εκτυπώνεται το μήνυμα "User exists.\n" και το βήμα 1 επαναλαμβάνεται.
2. Εάν ο χρήστης δε βρίσκεται στον πίνακα χρηστών, τότε αποθηκεύεται σε αυτόν κι ακολούθως διαβάζονται με τη σειρά οι αξιολογήσεις που έδωσε ο χρήστης στα βιβλία. Οι αξιολογήσεις των βιβλίων δίνονται με την ίδια σειρά που τα βιβλία είναι αποθηκευμένα στον πίνακα βιβλίων και αποθηκεύονται με αυτή την σειρά στον πίνακα αξιολογήσεων. Υποθέστε ότι οι τιμές των αξιολογήσεων είναι πάντα έγκυρες.

Γράψτε μια συνάρτηση που παίρνει ως παραμέτρους τους τρεις παραπάνω πίνακες και δείκτες σε δύο ακεραίους. Ακολούθως, εκτυπώνει το μήνυμα "Book info:\n", καλεί τη συνάρτηση αρχικοποίησης για τον πίνακα των βιβλίων, εκτυπώνει το μήνυμα "User info:\n" και καλεί τη συνάρτηση αρχικοποίησης για τους πίνακες χρηστών και αξιολογήσεων. Τέλος, αποθηκεύει το πλήθος των βιβλίων και το πλήθος των χρηστών στις θέσεις μνήμης που δείχνουν οι αντίστοιχες παράμετροι (που είναι δείκτες προς ακεραίους). Η συνάρτηση δεν επιστρέφει κάτι.

Προσθέστε κώδικα στη main που κάνει τα εξής:

1. Καλεί την παραπάνω συνάρτηση για να διαβαστούν κι αποθηκευτούν τα δεδομένα του προγράμματος στους κατάλληλους πίνακες.
2. Εκτυπώνει το μήνυμα "\nBooks:\n" και τα περιεχόμενα του πίνακα βιβλίων με την σειρά που είναι αποθηκευμένα στον πίνακα, με "\n" μετά από την πληροφορία κάθε βιβλίου.
3. Εκτυπώνει το μήνυμα "\nReviews:\n" και σε κάθε γραμμή που ακολουθεί το όνομα του χρήστη με πλάτος 8 χαρακτήρων, κενό και τις αξιολογήσεις του με πλάτος 3 χαρακτήρων χωρίς δεκαδικά ψηφία, με κενά ανάμεσα τους και "\n" μετά την τελευταία αξιολόγηση για κάθε χρήστη. Και πάλι οι πληροφορίες πρέπει να εκτυπώνονται με την σειρά που είναι αποθηκευμένες στους αντίστοιχους πίνακες.
4. Εκτυπώνει το μήνυμα "\n##\n". Αυτό είναι σημαντικό για τη βαθμολόγηση της άσκησης.

Εντοπισμός πελάτη και υπολογισμός συσχέτισης αξιολογήσεων

Διαβάστε προσεκτικά την περιγραφή όλου του σταδίου πριν ξεκινήσετε.

Στην συνέχεια, προσθέστε κώδικα στη `main` που κάνει τα εξής:

1. Εκτυπώνει το μήνυμα: `"Enter your username:\n"`
2. Διαβάζει από το πληκτρολόγιο το όνομα ενός χρήστη που ενδιαφέρεται να μάθει ποιο βιβλίο του συστήνει το πρόγραμμα. Ας πούμε αυτόν τον χρήστη «πελάτη».
3. Ελέγχει, χρησιμοποιώντας την αντίστοιχη συνάρτηση, αν ο πελάτης υπάρχει στον πίνακα χρηστών.
4. Αν ο πελάτης δεν υπάρχει στον πίνακα χρηστών, εκτυπώνει το μήνυμα `"No user \"X\"\n"`, όπου `X` το όνομα του πελάτη κι επαναλαμβάνει την διαδικασία από το βήμα 1.
5. Αν βρεθεί χρήστης με αυτό το όνομα, εκτυπώνει το μήνυμα `"Hello X!\n"` όπου `X` το όνομα του πελάτη.
6. Υπολογίζει την συσχέτιση αξιολογήσεων ανάμεσα σε κάθε χρήστη `r` και τον πελάτη, και την αποθηκεύει στο κελί `[r][MAX_BOOKS]` του πίνακα αξιολογήσεων. Βλέπε αναλυτική περιγραφή παρακάτω.
7. Εκτυπώνει το μήνυμα `"\nCorrelations:\n"` και για κάθε χρήστη, το όνομα του, κενό, την συσχέτιση αξιολογήσεων που υπολογίστηκε για τον πελάτη με ένα δεκαδικό ψηφίο, και `"\n"`. Η πληροφορία πρέπει να εκτυπώνεται με την σειρά που είναι αποθηκευμένοι οι χρήστες στον πίνακα χρηστών.
8. Εκτυπώνει το μήνυμα `"\n##\n"`. Αυτό είναι σημαντικό για τη βαθμολόγηση της άσκησης.

Περιγραφή βήματος 6:

Γράψτε μια συνάρτηση υπολογισμού συσχέτισης η οποία παίρνει κατάλληλες παραμέτρους (πρέπει να σκεφτείτε ποιες) ώστε να μπορεί για δύο χρήστες να υπολογίσει και να επιστρέψει ένα πραγματικό αριθμό που εκφράζει το βαθμό συσχέτισης ανάμεσα στις αξιολογήσεις τους. Ο αριθμός υπολογίζεται ως εξής:

Αν θεωρήσουμε ότι οι αξιολογήσεις που έδωσαν δύο χρήστες i, j είναι δύο διανύσματα $(i_0, i_1, \dots, i_{k-1})$ και $(j_0, j_1, \dots, j_{k-1})$ όπου k το πλήθος των βιβλίων, τότε ο βαθμός συσχέτισης δίνεται από την Ευκλείδεια απόσταση ανάμεσά τους, δηλαδή την τετραγωνική ρίζα του αθροίσματος των διαφορών των αντίστοιχων βαθμών:

$$\sqrt{((i_0 - j_0)^2 + (i_1 - j_1)^2 + \dots + (i_{k-1} - j_{k-1})^2)}$$

χωρίς να λαμβάνονται υπόψη οι όροι που περιλαμβάνουν «μηδενικές» αξιολογήσεις (βιβλία που δεν έχουν διαβαστεί από έναν ή και τους δύο χρήστες). Όσο πιο μικρή είναι αυτή η ποσότητα τόσο πιο κοντινή είναι η συσχέτιση αξιολογήσεων των δύο χρηστών.

Γράψτε μια συνάρτηση η οποία παίρνει ως παραμέτρους τον πίνακα αξιολογήσεων και ότι άλλο χρειάζεται (πρέπει να σκεφτείτε ποιες άλλες παράμετροι είναι απαραίτητες) ώστε να υπολογίσει για κάθε ένα χρήστη `r` τη συσχέτιση αξιολογήσεων με τον πελάτη και να την αποθηκεύσει στο κελί `[r][MAX_BOOKS]` του πίνακα αξιολογήσεων. Η συνάρτηση αυτή πρέπει να χρησιμοποιεί την προηγούμενη συνάρτηση. Δεν επιστρέφει κάτι. Αυτή η συνάρτηση πρέπει να καλείται μέσα από την `main` στο βήμα 6.

Εύρεση συστηνόμενου βιβλίου

Τώρα που έχουν υπολογιστεί οι βαθμοί συσχέτισης, μπορούν να προταθούν βιβλία στον πελάτη.

Γράψτε μια συνάρτηση που παίρνει ως παραμέτρους τον πίνακα αξιολογήσεων, την θέση του πελάτη στον πίνακα χρηστών, το πλήθος χρηστών και το πλήθος βιβλίων. Η συνάρτηση εντοπίζει τον χρήστη που έχει την κοντινότερη συσχέτιση αξιολογήσεων με τον πελάτη και έχει αξιολογήσει με βαθμό ≥ 3 τουλάχιστον ένα βιβλίο που δεν έχει ήδη διαβάσει ο πελάτης. Εάν υπάρχει ισοψηφία, επιλέγεται όποιος χρήστης εμφανίζεται πρώτος στον πίνακα χρηστών. Η συνάρτηση επιστρέφει την θέση του επιλεγμένου χρήστη στον πίνακα χρηστών ή -1 αν δεν βρεθεί κανένας χρήστης που να πληροί τις προϋποθέσεις.

Προσθέστε κώδικα στη main που κάνει τα εξής:

1. Χρησιμοποιεί την παραπάνω συνάρτηση για να εντοπίσει τον πιο κατάλληλο χρήστη. Αν η συνάρτηση επιστρέφει -1, εκτυπώνει το μήνυμα "No recommendations.\n" και το πρόγραμμα τερματίζει.
2. Διαφορετικά, εκτυπώνει το μήνυμα "X recommends:\n" όπου X το όνομα του χρήστη που βρέθηκε στο βήμα 1.
3. Εντοπίζει όλα τα βιβλία που άρεσαν (βαθμός ≥ 3) σε αυτόν τον χρήστη και που δεν έχει διαβάσει ήδη ο πελάτης. Για κάθε ένα τέτοιο βιβλίο, εκτυπώνει τις πληροφορίες του, κενό, τον βαθμό αξιολόγησης του χρήστη (χωρίς δεκαδικά ψηφία) και "\n".

Υποβολή

Υποβάλετε το αρχείο hw2.c ως έχει (όχι zip!).