

Εργασία 2 – Αξιολόγηση κριτικών ταινιών

Συνοπτική περιγραφή

Σκοπός της εργασίας είναι να αναπτύξετε μια εφαρμογή που συλλέγει δεδομένα από κριτικές ταινιών και (α) λειτουργεί ως βάση δεδομένων κριτικών επιτρέποντας την προσθήκη, αναζήτηση και διαγραφή κριτικών, και (β) χρησιμοποιεί τα δεδομένα για να αξιολογήσει αυτόματα αν μια νέα κριτική είναι θετική, αρνητική ή ουδέτερη.

>> Υπολογισμοί

Το σύστημα αξιολόγησης βασίζεται στον υπολογισμό ενός μέσου όρου βαθμολογίας για κάθε λέξη που εμφανίζεται σε μια κριτική.

Η κλίμακα αξιολόγησης των ταινιών είναι [0, 4] με 0 να αντιστοιχεί σε ιδιαίτερα αρνητική κριτική, 4 σε ιδιαίτερα θετική κι ενδιάμεσες τιμές σε αντίστοιχες διαβαθμίσεις.

- Βαθμολογία λέξης: Για κάθε λέξη, υπολογίζονται και αποθηκεύονται το *συνολικό άθροισμα των βαθμών* των κριτικών στις οποίες εμφανίζεται, και το *πλήθος εμφανίσεων* της λέξης σε όλες τις κριτικές.
 - Παράδειγμα: Αν η λέξη "eric" εμφανίζεται σε δύο κριτικές με βαθμούς 3.5 και 4.0 αντίστοιχα, τότε ο μέσος βαθμός της είναι $(3.5 + 4.0) / 2 = 3.75$ ("θετική" λέξη)
- Αξιολόγηση νέας κριτικής: Ο βαθμός μιας νέας κριτικής υπολογίζεται ως ο *μέσος όρος των βαθμών των λέξεων* που την αποτελούν. Αν μια λέξη της νέας κριτικής δεν υπάρχει στις υπάρχουσες κριτικές, θεωρείται πως έχει βαθμό 2.0.

>> Δεδομένα

- Αποθήκευση κριτικών: Τα δεδομένα των κριτικών (τίτλος, κείμενο, όνομα κριτικού, βαθμός) αποθηκεύονται σε δυναμική δομή δεδομένων.
- Αποθήκευση λέξεων: Οι λέξεις και οι βαθμολογίες τους αποθηκεύονται σε πίνακα κατακερματισμού ώστε να επιτευχθεί γρήγορη πρόσβαση σε αυτές.

Οργάνωση και αρχεία

Η εργασία θα υλοποιηθεί με διαχωρισμό αρμοδιοτήτων στις παρακάτω ενότητες:

Ενότητα	Αρχεία	Περιγραφή
Βιβλιοθήκη	libhw2.h, libhw2.a	Ορισμοί και υλοποίηση έτοιμων συναρτήσεων.
Κριτικές	review.h, review.c	Ορισμοί και υλοποίηση της δομής που αποθηκεύονται οι κριτικές.
Λίστα	list.h, list.c	Ορισμοί και υλοποίηση της βασικής δομής λίστας που χρησιμοποιεί ο πίνακας κατακερματισμού για τα δοχεία.
Πίνακας κατακερματισμού	htable.h, hash	Ορισμοί και υλοποίηση του πίνακα κατακερματισμού.
Εφαρμογή	hw2.h, hw2.c	Το κυρίως πρόγραμμα (main) που συντονίζει τη ροή δεδομένων μεταξύ των παραπάνω.

Σε κάθε περίπτωση, το αρχείο .h πρέπει να περιέχει ορισμούς τύπων (π.χ. struct) και επικεφαλίδες συναρτήσεων και το αντίστοιχο αρχείο .c τις υλοποιήσεις των συναρτήσεων. Πιο συγκεκριμένα:

>> Βιβλιοθήκη

Σας δίνονται μια στατική βιβλιοθήκη libhw2.a και το αρχείο libhw2.h. Μη μεταβάλετε αυτά τα αρχεία.

Το αρχείο libhw2.h περιλαμβάνει τις επικεφαλίδες των παρακάτω συναρτήσεων (οι οποίες υλοποιούνται στο libhw2.a):

- `print_menu`: Εκτυπώνει ένα μενού επιλογών. Χρησιμοποιήστε τη στη `main`.
- `read_line`: Παίρνει ως παράμετρο ένα δείκτη προς συμβολοσειρά (`char **`) και την πηγή από την οποία πρέπει να διαβάσει (`NULL` για ανάγνωση από τη συμβατική είσοδο ή όνομα αρχείου για ανάγνωση από αρχείο), διαβάζει μία γραμμή κειμένου, αφαιρεί τυχόν λευκούς χαρακτήρες από την αρχή και το τέλος της, δεσμεύει δυναμικά επαρκή μνήμη και αποθηκεύει το κείμενο στη συμβολοσειρά. Επιστρέφει 0 αν η ανάγνωση είναι επιτυχής, ή -1 σε περίπτωση λάθους.

>> Κριτικές

Για κάθε κριτική αποθηκεύονται το *όνομα της ταινίας*, το *όνομα του/της κριτικού*, το *κείμενο* της κριτικής και ο *βαθμός αξιολόγησης* της ταινίας.

Ορίστε ένα `struct` που αναπαριστά μία κριτική. Μπορείτε να ορίσετε επιπλέον `struct` εάν χρειάζεται.

Όλες οι κριτικές πρέπει να αποθηκεύονται σε μια δυναμικά δεσμευμένη δομή. Η μορφή αυτής της δομής είναι δική σας επιλογή (π.χ. λίστα, δυναμικός πίνακας), αλλά σε κάθε περίπτωση πρέπει να είναι δυναμική με μέγεθος ανάλογο του πλήθους δεδομένων που περιέχει.

Πρέπει να παρέχετε συναρτήσεις για τις παρακάτω λειτουργίες πάνω στην δομή:

- (α) Δημιουργία (άδειας) δομής
- (β) Προσθήκη νέας κριτικής.
- (γ) Διαγραφή κριτικής ενός/μιας συγκεκριμένου/ης κριτικού. Η συνάρτηση αφαιρεί κι επιστρέφει την πρώτη κριτική που βρίσκει για τον/την δεδομένο/η κριτικό, ή ένδειξη αποτυχίας αν δεν βρεθεί τέτοια κριτική. Η συνάρτηση θα πρέπει να κληθεί πολλαπλές φορές για να διαγράψει όλες τις κριτικές του ίδιου ατόμου. Επιτρέπεται η χρήση `static` τοπικών μεταβλητών σε αυτή τη συνάρτηση.
- (δ) Εκτύπωση κριτικών με κριτήριο μια συμβολοσειρά. Η συμβολοσειρά μπορεί να είναι ένας τίτλος ταινίας ή ένας αστερίσκος ("*"). Αν είναι τίτλος και υπάρχουν κριτικές γι' αυτή την ταινία, τις εκτυπώνει ως εξής: Για κάθε κριτική εκτυπώνει το βαθμό αξιολόγησής της με δύο δεκαδικά ψηφία, κόμμα, κενό, το όνομα του/της κριτικού, κόμμα, κενό, το κείμενο της κριτικής και τέλος χαρακτήρα '\n'. Αν είναι αστερίσκος, τότε για κάθε αποθηκευμένη κριτική εκτυπώνει '>', κενό, τον τίτλο της ταινίας στην οποία αναφέρεται και στην επόμενη γραμμή εκτυπώνει τα στοιχεία της κριτικής όπως περιγράφεται παραπάνω. Σε κάθε περίπτωση, η σειρά εκτύπωσης κριτικών δεν έχει σημασία.
- (ε) Απελευθέρωση όλης της δυναμικά δεσμευμένης μνήμης για τη δομή κριτικών.

>> Λίστα

Υλοποιήστε μια δομή λίστας (η οποία αργότερα θα χρησιμοποιηθεί για τα δοχεία (buckets) του πίνακα κατακερματισμού). Κάθε κόμβος της λίστας, πέρα από ό,τι δείκτες χρειάζονται για την διασύνδεση με άλλους κόμβους, πρέπει να περιέχει ως δεδομένα μια λέξη (ως δυναμικά δεσμευμένη συμβολοσειρά, με όλα τα γράμματα μικρά), το πλήθος κριτικών στις οποίες εμφανίζεται αυτή η λέξη, και το άθροισμα βαθμών αξιολόγησης αυτών των κριτικών.

Η λίστα πρέπει να είναι διπλά διασυνδεδεμένη, κυκλική με τερματικό κόμβο (sentinel) και να διατηρείται ταξινομημένη ως προς τις λέξεις, σε αύξουσα λεξικογραφική σειρά.

Πρέπει να παρέχονται συναρτήσεις για τις παρακάτω λειτουργίες πάνω σε τέτοιες λίστες:

- (α) Δημιουργία άδειας λίστας
- (β) Απλή αναζήτηση λέξης σε λίστα. Βοηθητική συνάρτηση που επιστρέφει ένδειξη επιτυχίας και δείκτη σε κόμβο. Ο δείκτης λειτουργεί ως σημείο αναφοράς για την ολοκλήρωση άλλων λειτουργιών που απαιτούν βήμα αναζήτησης. Για παράδειγμα, αν η λέξη βρεθεί, τότε μέσω του δείκτη μπορεί να γίνει άμεση ενημέρωση ή διαγραφή της, χωρίς καινούργια αναζήτηση. Αν δε βρεθεί, ο δείκτης μπορεί να υποδεικνύει τη θέση που πρέπει να γίνει εισαγωγή της.
- (γ) Εισαγωγή/Διαγραφή λέξης. Μία από τις παραμέτρους πρέπει να είναι ο βαθμός της λέξης. Αν κατά την εισαγωγή διαπιστωθεί πως η λέξη υπάρχει ήδη, τότε ανανεώνονται το άθροισμα βαθμών της και το πλήθος εμφανίσεων. Αν κατά τη διαγραφή διαπιστωθεί πως το πλήθος εμφανίσεων έγινε μηδέν, τότε διαγράφεται ο κόμβος αυτής της λέξης, διαφορετικά ανανεώνονται το άθροισμα βαθμών της και το πλήθος εμφανίσεων.
- (δ) Προσθήκη/Αφαίρεση κόμβου. Βοηθητικές συναρτήσεις που παίρνουν ως παράμετρο δείκτη προς ένα κόμβο (και ότι άλλο χρειάζεται), κι εκτελούν την αντίστοιχη λειτουργία.
- (ε) Εκτύπωση περιεχομένων. Για κάθε λέξη εκτυπώνονται: κενό, χαρακτήρας '[', κενό, η λέξη, κενό, το πλήθος εμφανίσεων της, κενό, το συνολικό άθροισμα βαθμολογιών της με δύο δεκαδικά ψηφία, κενό και χαρακτήρας ']'. Μετά το τέλος εκτύπωσης των περιεχομένων της λίστας (ανεξαρτήτως αν είναι άδεια ή όχι), εκτυπώνονται δύο χαρακτήρες '\n'.
- (στ) Ανάκτηση βαθμολογίας λέξης. Παίρνει ως παράμετρο μια λέξη κι επιστρέφει το μέσο βαθμό της εφόσον αυτή υπάρχει ή ένδειξη αποτυχίας.
- (ζ) Απελευθέρωση όλης της δυναμικά δεσμευμένης μνήμης για τη λίστα.

>> Πίνακας κατακερματισμού

Οι λέξεις που εμφανίζονται στις κριτικές και οι πληροφορίες αξιολόγησης αυτών πρέπει να αποθηκεύονται σε έναν πίνακα κατακερματισμού ο οποίος πρέπει να υλοποιηθεί ως δυναμικός πίνακας από λίστες (όπως αυτές ορίστηκαν παραπάνω).

Το αρχικό μέγεθος του πίνακα κατακερματισμού είναι αποθηκευμένο στη συμβολική σταθερά INIT_HSIZE η οποία μπορεί να οριστεί κατά τη διάρκεια της μεταγλώττισης με χρήση της επιλογής -D. Εάν δεν έχει οριστεί κατά τη μεταγλώττιση, τότε η τιμή της πρέπει να ορίζεται από τον κώδικα σας σε 1. Χρησιμοποιήστε `#ifndef/#endif` για τον έλεγχο.

Ο μέγιστος κι ελάχιστος επιτρεπτός βαθμός πληρότητας (load factor) του πίνακα κατακερματισμού είναι αποθηκευμένοι στις συμβολικές σταθερές HIGH_LOAD και LOW_LOAD αντίστοιχα. Οι τιμές τους μπορούν να οριστούν κατά τη διάρκεια της μεταγλώττισης με χρήση της επιλογής -D. Εάν δεν έχουν οριστεί κατά τη μεταγλώττιση, τότε πρέπει να ορίζεται από τον κώδικα σας σε 4.0 και 1.0 αντίστοιχα.

Η πολιτική ανακατακερματισμού (re-hashing) σε συνδυασμό με την δυναμική αυξομείωση του πίνακα βασίζεται στο βαθμό πληρότητας (load factor) του πίνακα. Συγκεκριμένα: Αν μετά την εισαγωγή νέας εγγραφής ο βαθμός πληρότητας είναι μεγαλύτερος ή ίσος της τιμής HIGH_LOAD τότε ο πίνακας διπλασιάζεται. Αν μετά τη διαγραφή μιας εγγραφής ο βαθμός πληρότητας είναι μικρότερος ή ίσος της τιμής LOW_LOAD, τότε ο πίνακας υποδιπλασιάζεται, μόνο αν αυτό δεν επιφέρει μείωση κάτω από το αρχικό μέγεθος INIT_HSIZE. Σε περίπτωση που αποτυγχάνει η δέσμευση μνήμης κατά το διπλασιασμό του πίνακα κατακερματισμού, τότε το πρόγραμμα συνεχίζει κανονικά χωρίς να γίνει ανακατακερματισμός, με τον πίνακα να παραμένει στο μέγεθος που είχε πριν την εισαγωγή της νέας εγγραφής.

ΠΡΟΣΟΧΗ: Για τη διαχείριση των λιστών πρέπει να καλούνται αποκλειστικά οι αντίστοιχες συναρτήσεις που έχουν οριστεί στο list.h. Ο κώδικας που υλοποιεί τον πίνακα κατακερματισμού δεν πρέπει να έχει άμεση πρόσβαση στους δείκτες next/prev της λίστας ούτε να πρέπει να γνωρίζει την υλοποίηση αυτών των συναρτήσεων.

Πρέπει να παρέχονται συναρτήσεις για τις παρακάτω λειτουργίες:

- (α) Δημιουργία άδειου πίνακα κατακερματισμού
- (β) Προσθήκη/Διαγραφή λέξης
- (γ) Ανακατακερματισμός
- (δ) Αναζήτηση λέξης στον πίνακα.
- (ε) Εκτύπωση περιεχομένων πίνακα. Αρχικά εκτυπώνονται δύο χαρακτήρες '\n'. Στην επόμενη γραμμή εκτυπώνονται με κενό ανάμεσα τους το μέγεθος του πίνακα κατακερματισμού, ο συνολικός αριθμός των εγγραφών, το load factor (πλήθος εγγραφών/μέγεθος πίνακα) με δύο δεκαδικά ψηφία, και το μέγεθος του μεγαλύτερου δοχείου. Στην συνέχεια, ξεκινώντας από καινούργια γραμμή, για κάθε θέση του πίνακα, εκτυπώνονται σε μια γραμμή ο αριθμός της θέσης, χαρακτήρας ': ', κενό, το πλήθος λέξεων του δοχείου και στη συνέχεια τα περιεχόμενα της λίστας.
- (στ) Απελευθέρωση όλης της δυναμικά δεσμευμένης μνήμης για τον πίνακα κατακερματισμού.

Το αρχείο htable.h περιλαμβάνει ήδη την επικεφαλίδα της συνάρτησης κατακερματισμού hash η υλοποίηση της οποίας βρίσκεται στο htable.c. Σημειώστε πως η τιμή που επιστρέφει η συνάρτηση κατακερματισμού πρέπει να προσαρμόζεται κατάλληλα με βάση το τρέχον μέγεθος του πίνακα κατακερματισμού (modulo το μέγεθος του πίνακα) έτσι ώστε να προκύπτουν θέσεις δοχείων μέσα στα επιτρεπτά όρια του πίνακα.

>> Εφαρμογή

Η συνάρτηση main πρέπει να βρίσκεται στο αρχείο hw2.c. Το πρόγραμμα αρχικοποιεί κατάλληλα τις δομές δεδομένων και κατόπιν σε επανάληψη, καλεί τη συνάρτηση print_menu, διαβάζει την επιλογή του χρήστη (χωρίς να κάνει διάκριση κεφαλαίων/μικρών), και πραγματοποιεί τις απαιτούμενες ενέργειες όπως περιγράφεται παρακάτω για κάθε μια από τις επιλογές. Σε όλες τις περιπτώσεις ανάγνωσης κειμένου (π.χ. όνομα αρχείου, κριτική, κτλ.) χρησιμοποιήστε τη συνάρτηση read_line.

-
- 'A' Εκτυπώνει το μήνυμα `"Enter filename: "` και διαβάζει το όνομα ενός αρχείου. Ακολούθως, διαβάζει από αυτό τα δεδομένα των κριτικών. Για κάθε κριτική το αρχείο περιέχει σε ξεχωριστή γραμμή (α) τον τίτλο ταινίας, (β) το όνομα του/της κριτικού, (γ) το κείμενο της κριτικής και (δ) το βαθμό αξιολόγησης της ταινίας. Κάθε κριτική αποθηκεύεται στη δομή κριτικών ενώ οι επιμέρους λέξεις της κριτικής αποθηκεύονται στον πίνακα κατακερματισμού ανανεώνοντας κατάλληλα τους βαθμούς τους. Δε χρειάζεται να κάνετε έλεγχο για τυχόν διπλότυπες κριτικές. Στο τέλος εκτυπώνει το μήνυμα `"Added N reviews.\n"` όπου N το πλήθος κριτικών που προστέθηκαν στη δομή κριτικών.
-
- 'S' Εκτυπώνει το μήνυμα `"Enter word: "`, διαβάζει μια λέξη από το πληκτρολόγιο, την εντοπίζει στον πίνακα κατακερματισμού κι εκτυπώνει το μήνυμα `"W: Z\n"` όπου W η λέξη που δόθηκε με όλα τα γράμματα μικρά και Z ο μέσος βαθμός αξιολόγησης αυτής με δύο δεκαδικά ψηφία. Αν η λέξη δεν υπάρχει στον πίνακα κατακερματισμού, εκτυπώνεται το μήνυμα `"W not found.\n"`.
-
- 'I' Εκτυπώνει το μήνυμα `"Enter review: "` και διαβάζει μια νέα κριτική. Η κριτική δεν συνοδεύεται από τίτλο ταινίας / όνομα κριτικού ούτε αποθηκεύεται στη δομή αποθήκευσης κριτικών. Το πρόγραμμα υπολογίζει ένα βαθμό αξιολόγησης για την κριτική όπως περιγράφεται παραπάνω κι εκτυπώνει το μήνυμα `"S review (X).\n"` όπου X είναι ο βαθμός που υπολογίστηκε με 2 δεκαδικά ψηφία και S είναι η λέξη `Positive` αν ο βαθμός είναι στο εύρος [3,4], `Neutral` αν ο βαθμός είναι στο εύρος [2, 3) ή `Negative` αν ο βαθμός είναι στο εύρος [0, 2).
-
- 'P' Εκτυπώνει το μήνυμα `"Enter movie title: "`, διαβάζει μια γραμμή κειμένου, εκτυπώνει το μήνυμα `"\nResults for T:\n"` όπου T η είσοδος, και καλεί τη συνάρτηση εκτύπωσης κριτικών. Αν η συνάρτηση δεν εκτυπώσει κάτι, το πρόγραμμα εκτυπώνει το μήνυμα `"No reviews"`.
-

`found.\n"`. Σημείωση: Στο autolab θα ελεγχθούν μόνο περιπτώσεις που εκτυπώνεται μία ή καμία κριτική.

'R' Εκτυπώνει το μήνυμα `"Enter name: "`, διαβάζει από το πληκτρολόγιο το όνομα ενός κριτικού και αφαιρεί από τη δομή κριτικών όλες τις κριτικές που γράφτηκαν από αυτόν, ανανεώνοντας κατάλληλα τον πίνακα κατακερματισμού. Στο τέλος, εκτυπώνει το μήνυμα `"\nRemoved X reviews by N.\n"`, όπου X το πλήθος κριτικών που αφαιρέθηκαν και N το όνομα του κριτικού.

'D' Εκτυπώνει τα περιεχόμενα του πίνακα κατακερματισμού.

'Q' Αποδεσμεύει όλη τη δυναμικά δεσμευμένη μνήμη του προγράμματος και τερματίζει.

Σε οποιαδήποτε άλλη επιλογή, το πρόγραμμα εκτυπώνει το μήνυμα `"Invalid selection.\n"`

Για κάθε μία από τις παραπάνω επιλογές, ορίστε μια βοηθητική συνάρτηση (με την επικεφαλίδα στο `hw2.h` και την υλοποίηση στο `hw2.c`). Η διαχείριση των επιλογών πρέπει να γίνει με κατάλληλη χρήση `switch` που στα `cases` περιέχει τις κλήσεις των αντίστοιχων συναρτήσεων και `break`, `continue` ή `return`.

Απαιτήσεις υλοποίησης

- Κατασκευάστε κατάλληλο Makefile. Κάθε αρχείο `.c` πρέπει να μεταγλωττίζεται ξεχωριστά και στο τέλος να συνδέονται όλα τα επιμέρους object files για την παραγωγή εκτελέσιμου με όνομα `hw2`.
- Απαγορεύεται η χρήση global μεταβλητών καθώς και η χρήση `goto`, `gets` ή `fflush`. Η χρήση static μεταβλητών επιτρέπεται μόνο στα πλαίσια της υλοποίησης της διαγραφής κριτικών.
- Απαγορεύεται η χρήση variable length arrays.
- Μπορείτε να ορίσετε επιπλέον συναρτήσεις από αυτές που αναφέρονται στην εκφώνηση.
- Δε γίνεται διάκριση κεφαλαίων/μικρών κατά την εισαγωγή δεδομένων ή κατά τη σύγκριση συμβολοσειρών. Ίσως σας φανεί χρήσιμη η συνάρτηση `strcasecmp`.
- Τα δεδομένα που εισάγονται στη δομή αποθήκευσης κριτικών δε χρειάζονται επεξεργασία πριν την εισαγωγή τους όσον αφορά κεφαλαία/μικρά/κενά. Τα εισάγετε (και αργότερα τα εκτυπώνετε) στη μορφή που δίνονται αρχικά. Η μόνη μετατροπή που πρέπει να κάνετε είναι στις λέξεις που εισάγονται στον πίνακα κατακερματισμού, οι οποίες πρέπει να είναι με όλα τα γράμματα μικρά.
- Κατά την είσοδο δεδομένων από το χρήστη τα οποία δυνητικά αποτελούνται από πολλές λέξεις (π.χ. μία κριτική, ένας τίτλος ταινίας), όλες οι λέξεις δίνονται σε μία γραμμή.
- Για την απόσπαση των επιμέρους λέξεων μιας κριτικής θα σας φανεί χρήσιμη η συνάρτηση `strtok`. Μπορείτε να υποθέσετε ότι οι λέξεις μια κριτικής διαχωρίζονται με κενά.
- Σε περίπτωση αποτυχίας δέσμευσης μνήμης (εκτός από την περίπτωση διπλασιασμού του πίνακα κατακερματισμού) εκτυπώνετε το μήνυμα `"Memory error.\n"` και το πρόγραμμα τερματίζει άμεσα με χρήση εντολής `exit`.
- Αποδεσμεύστε σωστά τη δυναμικά δεσμευμένη μνήμη σε όλες τις περιπτώσεις, εκτός αν το πρόγραμμα πρόκειται να τερματίσει με `exit`.
- Το πρόγραμμα πρέπει να ακολουθεί τους κανόνες μορφοποίησης, σχολιασμού, κτλ. που έχουν αναρτηθεί στο eclass, καθ' όλη τη διάρκεια της ανάπτυξης του.

Στάδια υλοποίησης και υποβολή

>> Στάδιο 1:

- Κατασκευάστε ένα Makefile που μεταγλωττίζει μόνο τα αρχεία hw2.c, review.c και τα διασυνδέει με τη βιβλιοθήκη, κατασκευάζοντας εκτελέσιμο με όνομα hw2.
- Αποφασίστε τι είδους δομή θα χρησιμοποιήσετε για την αποθήκευση των κριτικών. Ολοκληρώστε την υλοποίηση στα review.h, review.c και υλοποιήστε εν μέρει τη λειτουργία της επιλογής 'A' (προσθήκη μόνο στη δομή κριτικών).
- Υλοποιήστε την επιλογή 'P' κι επιβεβαιώστε ότι είναι σωστή.
- Υλοποιήστε εν μέρει τη λειτουργία της επιλογής 'R' (αφαίρεση μόνο από τη δομή κριτικών.)
- Υλοποιήστε εν μέρει τη λειτουργία της επιλογής 'Q' (αποδέσμευση δυναμικής μνήμης του προγράμματος έως τώρα κι έξοδος)
- Ολοκληρώστε την υλοποίηση της λίστας. Γράψτε μια προσωρινή, ξεχωριστή συνάρτηση main στην οποία ελέγχετε αποκλειστικά τις λειτουργίες της λίστας. Αυτό το κομμάτι του 1ου σταδίου δε θα ελεγχθεί στο autolab.

Υποβάλετε ένα αρχείο zip που περιέχει τα αρχεία Makefile, hw2.c, hw2.h, review.c, review.c.

Προθεσμία: Παρασκευή 20/3/2026, 21:00

>> Τελικό στάδιο:

- Ολοκληρώστε το Makefile. Το τελικό εκτελέσιμο πρέπει να λέγεται hw2.
- Υλοποιήστε τον πίνακα κατακερματισμού, αρχικά χωρίς ανακατακερματισμό (rehashing).
- Ολοκληρώστε τις υλοποιήσεις των επιλογών 'A', 'R', 'Q' και υλοποιήστε τις λειτουργίες που σχετίζονται με τις υπόλοιπες επιλογές.
- Υλοποιήστε το rehashing στον πίνακα κατακερματισμού.

Κατασκευάστε ένα αρχείο zip με όνομα hw2.zip που περιέχει το Makefile και όλα τα αρχεία .c και .h εκτός του libhw2.h. Μπορείτε να κατασκευάσετε το zip ως εξής:

> (μέσω τερματικού) Μεταβείτε στον κατάλογο που βρίσκονται τα αρχεία και γράψτε την εντολή

```
zip hw2.zip *.c *.h Makefile -x libhw2.h
```

> (ή μέσω GUI) Μεταβείτε στον κατάλογο που βρίσκονται τα αρχεία, επιλέξτε με το ποντίκι (κρατώντας πατημένο το control) τα αρχεία που θέλετε να συμπεριλάβετε, μετά δεξί click και επιλέξτε από το μενού Compress here as ZIP.

Υποβάλετε το hw2.zip στο autolab του μαθήματος. Ο ελάχιστος βαθμός που πρέπει να πετύχετε στο τελικό στάδιο είναι 70.

Προθεσμία: Κυριακή 29/3/2026, 21:00

Το μέγιστο πλήθος υποβολών χωρίς βαθμολογική ποινή είναι 15. Κάθε υποβολή πλέον των 15 έχει ποινή ίση με 5% του βαθμού.