

Πανεπιστήμιο Κρήτης, Τμήμα Επιστήμης Υπολογιστών

HY240: Δομές Δεδομένων

Χειμερινό Εξάμηνο – Ακαδημαϊκό Έτος 2025-26

1η Φάση Πρότζεκτ

Ημερομηνία Παράδοσης: 30 Οκτωβρίου 2025, ώρα 23.59.

Τρόπος Παράδοσης: Μέσα από το eLearn του μαθήματος.

1. Γενική Περιγραφή

Στην εργασία αυτή θα υλοποιήσετε ένα **σύστημα διαχείρισης βιβλιοθήκης**, το οποίο προσομοιώνει τη λειτουργία ενός δικτύου μελών που δανείζονται και αξιολογούν βιβλία.

Σκοπός είναι να αναπτύξετε ένα πρόγραμμα το οποίο επεξεργάζεται **σειρά εντολών** από αρχείο εισόδου και διαχειρίζεται δυναμικά τις πληροφορίες για τα μέλη, τα βιβλία, και τις **κατηγορίες** (είδη) τους.

Κάθε βιβλίο ανήκει σε μία κατηγορία και μπορεί να δανειστεί από πολλά μέλη. Όταν ένα μέλος επιστρέψει ένα βιβλίο, μπορεί να το **βαθμολογήσει**: οι βαθμολογίες αυτές καθορίζουν τη **δημοτικότητα** του βιβλίου μέσα στην κατηγορία του. Το σύστημα οφείλει να υποστηρίζει **αποδοτικές ενημερώσεις** των ταξινομημένων λιστών, ώστε κάθε νέο score να επαναποθετεί το βιβλίο στη σωστή θέση με ελάχιστες αλλαγές δεικτών.

Στο τέλος, για κάθε κατηγορία υπολογίζεται ποια βιβλία θα προβληθούν στις «βιτρίνες» της βιβλιοθήκης. Η επιλογή γίνεται βάσει δημοτικότητας, χρησιμοποιώντας τη μέθοδο του **μέτρου και των μεγαλύτερων υπολοίπων** (όπως στις αναλογικές εκλογές): κάθε κατηγορία παίρνει έναν αριθμό «θέσεων προβολής» ανάλογο της συνολικής της βαθμολογίας.

Η εργασία δίνει έμφαση:

- στη σωστή χρήση **συνδεδεμένων λιστών** (μονοσυνδεδεμένων και διπλά συνδεδεμένων),
- στην **αποδοτική ενημέρωση** κόμβων κατά την προσθήκη νέων δανεισμών ή αξιολογήσεων,
- και στην **ορθή εκτέλεση** αριθμητικών υπολογισμών και κατατάξεων με βάση πραγματικά δεδομένα (βαθμολογίες, πλήθος διαθέσιμων θέσεων, υπόλοιπα κατηγοριών).

Η όλη προσομοίωση υλοποιείται σε γλώσσα **C**, με έμφαση στη διαχείριση δυναμικής μνήμης, τη διασύνδεση δομών, και την αποφυγή ανακατασκευής λιστών.

2. Οντότητες του Συστήματος

Το σύστημα της βιβλιοθήκης οργανώνεται γύρω από **τρεις βασικές οντότητες** — **Books** (Βιβλία), **Members** (Μέλη) και **Genres** (Κατηγορίες) — καθώς και μία **βοηθητική οντότητα Loan** (Δανεισμός) που συνδέει τα μέλη με τα βιβλία.

2.1. Book (Βιβλίο)

Κάθε βιβλίο είναι ένα αντικείμενο που μπορεί να δανειστεί, να επιστραφεί και να βαθμολογηθεί. Ανήκει σε μία μόνο κατηγορία (genre) και συγκεντρώνει στατιστικά σχετικά με τη δημοτικότητά του.

Πεδία:

- `bid` – Μοναδικός κωδικός βιβλίου.
- `gid` – Κωδικός κατηγορίας στην οποία ανήκει.
- `title` – Τίτλος του βιβλίου.
- `sum_scores` – Άθροισμα όλων των έγκυρων βαθμολογιών.
- `n_reviews` – Πλήθος έγκυρων αξιολογήσεων.
- `avg` – Μέση βαθμολογία ($\text{floor}(\text{sum_scores} / \text{n_reviews})$).
- `lost_flag` – 1 αν το βιβλίο έχει δηλωθεί ως χαμένο, αλλιώς 0.
- `next`, `prev` – Δείκτες για διπλά συνδεδεμένη λίστα ταξινομημένη κατά `avg` (φθίνουσα).

Ρόλος:

Το βιβλίο αποτελεί τον βασικό κόμβο πληροφορίας. Κατά την επιστροφή:

- ενημερώνεται το άθροισμα και ο μέσος όρος των βαθμολογιών,
- επανατοποθετείται αποδοτικά στη σωστή θέση στη λίστα ταξινομημένων βιβλίων της κατηγορίας του (με τοπικές αλλαγές δεικτών),
- αν δηλωθεί ως χαμένο, εξαιρείται από τις μελλοντικές προβολές.

2.2. Member (Μέλος)

Αντιπροσωπεύει έναν αναγνώστη της βιβλιοθήκης, ο οποίος μπορεί να δανείζεται, να επιστρέψει και να αξιολογεί βιβλία.

Πεδία:

- `sid` – Μοναδικός κωδικός μέλους.
- `name` – Ονοματεπώνυμο μέλους.
- `loans` – Μονοσυνδεδεμένη λίστα ενεργών δανεισμών με sentinel node.
- `next` – Δείκτης στον επόμενο μέλος.

Ρόλος:

Κάθε μέλος μπορεί να έχει ταυτόχρονα πολλούς ενεργούς δανεισμούς. Κατά την επιστροφή βιβλίου, το μέλος παρέχει βαθμολογία που ενημερώνει το αντίστοιχο book. Η λίστα `loans` είναι **unsorted** και επιτρέπει εισαγωγή/διαγραφή σε **O(1)**. Υλοποιείται με χρήση sentinel node (κόμβος φρουρός).

2.3. Loan (Δανεισμός)

Αντιπροσωπεύει τη σχέση μεταξύ ενός μέλους και ενός βιβλίου που έχει δανειστεί. Είναι προσωρινή οντότητα που διαγράφεται μετά την επιστροφή.

Πεδία:

- sid – Κωδικός μέλους που δανείζεται.
- bid – Κωδικός βιβλίου που δανείζεται.
- next – Δείκτης στον επόμενο ενεργό δανεισμό του ίδιου μέλους.

Ρόλος:

Η ύπαρξη της οντότητας αυτής επιτρέπει τη συσχέτιση member → book και τη διαχείριση των ενεργών δανεισμών χωρίς αναζήτηση σε όλες τις λίστες βιβλίων.

2.4. Genre (Κατηγορία)

Αντιπροσωπεύει μια θεματική ενότητα βιβλίων (π.χ. Πληροφορική, Ιστορία, Φυσική).

Πεδία:

- gid – Κωδικός κατηγορίας.
- name – Όνομα κατηγορίας.
- books – Διπλά συνδεδεμένη λίστα βιβλίων ταξινομημένων κατά φθίνουσα δημοτικότητα (μέσος όρος avg).
- slots – Πλήθος θέσεων προβολής για τη συγκεκριμένη κατηγορία.
- lost_count – Πλήθος χαμένων βιβλίων της κατηγορίας.
- invalid_count – Πλήθος άκυρων αξιολογήσεων.
- next – Δείκτης στην επόμενη κατηγορία.

Ρόλος:

Η κατηγορία συγκεντρώνει βιβλία και στατιστικά δημοτικότητας.

2.5. Library (Σύστημα Βιβλιοθήκης)

Κεντρική οντότητα που συγκεντρώνει όλες τις κατηγορίες και τα μέλη της βιβλιοθήκης.

Πεδία:

- genres – Λίστα όλων των κατηγοριών (sorted κατά gid).
- members – Λίστα όλων των μελών (sorted κατά sid).

Ρόλος:

Αποτελεί το σημείο εκκίνησης για όλες τις εντολές εισόδου (G, BK, M, L, R, D, κ.λπ.).

Μέσω αυτής εντοπίζονται και ενημερώνονται οι κατάλληλες λίστες κατά την εκτέλεση γεγονότων.

3. Τρόπος λειτουργίας προγράμματος

Το εκτελέσιμο διαβάζει **σειριακά** ένα αρχείο εντολών και ενημερώνει τις δυναμικές δομές της βιβλιοθήκης χωρίς επανεκκίνηση ή ανακατασκευή λιστών.

`./library <input-file>`

- Κάθε γραμμή στο <input-file> είναι **μία εντολή**.
- Οι εντολές εκτελούνται **αυστηρά με τη σειρά** που δίνονται.
- Για κάθε **επιτυχή** ενέργεια εκτυπώνεται DONE. Για άκυρα ids/συνθήκες εκτυπώνεται IGNORED και δεν αλλάζει η κατάσταση.
- Όπου ζητείται εκτύπωση κατάστασης, η μορφή εξόδου είναι **ακριβώς** αυτή που ορίζεται παρακάτω.

3.1 Ρύθμιση παραμέτρων

- **S <slots>**

Ορίζει το συνολικό πλήθος θέσεων προβολής (SLOTS) που θα μοιραστούν στις κατηγορίες κατά την D.

- Έξοδος: DONE

3.2 Δημιουργία βασικών οντοτήτων

- **G <gid> <name>**

Καταχώρηση **Κατηγορίας (Genre)** με κωδικό gid και όνομα name.

- Αν υπάρχει ήδη, αγνοείται.
- Έξοδος: DONE ή IGNORED

- **BK <bid> <gid> <title>**

Καταχώρηση **Βιβλίου** με κωδικό bid, υπαγόμενο στην κατηγορία gid, και τίτλο title.

Αρχικά: sum_scores=0, n_reviews=0, avg=0, lost_flag=0.

- Αν δεν υπάρχει το gid ⇒ IGNORED.
- Αν το bid υπάρχει ήδη ⇒ IGNORED.
- Έξοδος: DONE ή IGNORED

- **M <sid> <name>**

Καταχώρηση **Μέλους** με κωδικό sid και όνομα name.

- Αν υπάρχει ήδη, IGNORED.
- Έξοδος: DONE ή IGNORED

3.3 Δανεισμοί & Επιστροφές

- **L <sid> <bid>**

Το μέλος sid δανείζεται το βιβλίο bid. Δημιουργείται εγγραφή στη **λίστα ενεργών δανεισμών** του μέλους (unsorted, O(1)).

- Προϋποθέτει ύπαρξη sid και bid.
- Αν υπάρχει ήδη ενεργός δανεισμός του ίδιου βιβλίου από το ίδιο μέλος, IGNORED.

- Αν δεν υπάρχει το βιβλίο, IGNORED
 - Έξοδος: DONE ή IGNORED
- **R <sid> <bid> <score|NA> <status>**
 Επιστροφή βιβλίου bid από το μέλος sid. Αφαιρείται ο αντίστοιχος ενεργός δανεισμός.
 Κανόνες:
 - Av status=lost:
 - Θέσει lost_flag(bid)=1 (παραμένει ιστορικά χαμένο),
 - **δεν** ενημερώνονται sum_scores/n_reviews,
 - Έξοδος: DONE.
 - Av status=ok και score=NA:
 - Κλείνει ο δανεισμός χωρίς αξιολόγηση,
 - Έξοδος: DONE.
 - Av status=ok και score \in [0..10]:
 - Πρόσθεση review (append O(1)),
 - Ενημέρωση sum_scores, n_reviews,
avg=floor(sum_scores/n_reviews),
 - **Επαναδιάταξε** το βιβλίο στη ταξινομημένη λίστα της κατηγορίας του (διπλά συνδ., desc κατά avg) με τοπικές αλλαγές δεικτών (**O(hops)**),
 - Έξοδος: DONE.
 - Av status=ok και score εκτός [0..10]:
 - Η αξιολόγηση αγνοείται (δεν αλλάζουν τα stats),
 - Έξοδος: IGNORED.
 - Αν δεν υπάρχει ενεργός δανεισμός sid–bid:
 - Έξοδος: IGNORED.

3.4 Κατανομή θέσεων προβολής & επιλογή τίτλων

- **D**
 Κατανέμει τις SLOTS θέσεις προβολής **στις κατηγορίες με μέτρο & μεγαλύτερα υπόλοιπα** και επιλέγει τους τίτλους που θα προβληθούν ανά κατηγορία.

Βήματα υπολογισμού:

1. Για κάθε κατηγορία g, υπολόγισε
 $points(g) = \sum(sum_scores(b))$ για όλα τα βιβλία b του g με $n_reviews > 0$ και $lost_flag = 0$.
 Ορίστε VALID = points(g). Av VALID=0 \Rightarrow καμία προβολή.

2. $\text{quota} = \text{floor}(\text{VALID} / \text{SLOTS})$ (αν $\text{SLOTS}=0$ ή $\text{VALID}=0$, καμία προβολή).
3. Αρχικές θέσεις: $\text{seats}(g)=\text{floor}(\text{points}(g)/\text{quota})$ (αν $\text{quota}=0$, όλες οι θέσεις παραμένουν 0).
4. Υπόλοιπο: $\text{rem}(g)=\text{points}(g) - (\text{seats}(g) * \text{quota})$. Μοίρασε τις εναπομείνασες θέσεις στις κατηγορίες με **μεγαλύτερο υπόλοιπο**.
Ισοπαλία σπάει με **μικρότερο gid**.
5. Για κάθε κατηγορία g , επίλεξε τα κορυφαία $\text{seats}(g)$ βιβλία (με $\text{lost_flag}=0$) από την ταξινομημένη λίστα της κατηγορίας κατά avg (tie-break μικρότερο bid).
Αν δεν υπάρχουν αρκετά επιλέξιμα βιβλία, οι θέσεις μένουν κενές.
6. Έξοδος: DONE

3.5 Εκτυπώσεις / Αναφορές

- **PG <gid>** — *Print Genre*

Εκτυπώνει την τρέχουσα ταξινομημένη λίστα βιβλίων της κατηγορίας gid (φθίνουσα avg):

```
<bid>, <avg>
<bid>, <avg>
...
```

- Αν το gid δεν υπάρχει \Rightarrow IGNORED.

- **PM <sid>** — *Print Member loans*

Εκτυπώνει όλα τα **ενεργά** δανεισμένα βιβλία του μέλους sid (σειρά εισαγωγής):

```
Loans:
<bid1>
<bid2>
...
```

- Αν δεν υπάρχει μέλος ή δεν έχει δανεισμούς, τυπώνει κενή ενότητα Loans::.

- **PD** — *Print Display (αποτέλεσμα D)*

Εκτυπώνει τις επιλεγμένες προβολές, ομαδοποιημένες κατά κατηγορία (σειρά: αύξουσα gid):

```
Display:
<gid>:
<bid>, <avg>
<bid>, <avg>
...
<gid>:
...
```

Περιλαμβάνει μόνο ό,τι παρήγαγε η τελευταία D. Αν δεν έχει τρέξει D ή δεν υπάρχουν θέσεις/έγκυρα points , τυπώνεται:

```
Display:
(empty)
```

- **PS** — *Print Stats* (προαιρετική αλλά συνιστώμενη για έλεγχο)
Εκτυπώνει για κάθε κατηγορία gid τα $points(g)$ που θα χρησιμοποιούσε η D αν έτρεχε τώρα, καθώς και τα **SLOTS**:

```

SLOTS=<S>
<gid>: points=<p>
<gid>: points=<p>
...

```

Παρατηρήσεις

- Τα ονόματα/τίτλοι δίνονται σε εισαγωγικά όταν περιέχουν κενά.
- $score$ είναι ακέραιος στο $[0..10]$.
- $status \in \{ok, lost\}$.
- Στο σύστημα υπάρχει ένας συνολικός αριθμός “θέσεων προβολής” (**SLOTS**) — δηλαδή, ο αριθμός των βιβλίων που μπορεί να προβληθεί ταυτόχρονα στη “βιτρίνα” της βιβλιοθήκης. Ο αριθμός αυτός δεν είναι ανά κατηγορία, αλλά παγκόσμιος: ισχύει για ολόκληρη τη βιβλιοθήκη.
- Τα avg είναι **ακέραιοι**: $\text{floor}(\text{sum_scores} / n_reviews)$.
- Βιβλίο με $lost_flag=1$ **δεν** συμμετέχει σε $points(g)$ ούτε επιλέγεται σε D .
- Κατά την R , η επανατοποθέτηση γίνεται **τοπικά** (μοναδικές ανταλλαγές/ενημερώσεις δεικτών) στη διπλά συνδεδεμένη λίστα της κατηγορίας.
- Η D **δεν** αλλάζει τα avg μόνο «διαβάζει» τις ήδη ταξινομημένες λίστες και επιλέγει κορυφαία στοιχεία ανά κατηγορία βάσει των υπολογισμών ($quotas$ + υπόλοιπα).

Αποδοτικότητα (ελέγχεται)

- Εισαγωγές σε **μη ταξινομημένες μονοσυνδεδεμένες** λίστες με sentinel : $O(1)$.
- Επανατοποθετήσεις βιβλίων σε **διπλά συνδεδεμένες, ταξινομημένες (φθίνουσα avg)** λίστες: $O(hops)$ με **τοπικές αλλαγές δεικτών** (όχι ανακατασκευή).
- Κατανομή προβολών (εντολή D): μία σάρωση για συγκέντρωση πόντων ανά κατηγορία (**$O(B)$**) + ταξινόμηση υπολοίπων (**$O(G \log G)$**).

4. Απαιτούμενες Δομές Δεδομένων (C)

Βρίσκονται στο αρχείο **Library.h** που συνοδεύει την άσκηση

Επιτρέπεται να επεκτείνετε πεδία αν χρειαστεί, αλλά κρατήστε τη **λογική λιστών** και τη **διάκριση ταξινομημένων/μη.**

5. Σετ Δοκιμών

Συνοδευτικά δίνονται **2 πλήρη σετ δοκιμών** που καλύπτουν βασική ροή, άκρα/σφάλματα, ισοπαλίες/κατανομή και επανατοποθετήσεις.

6. Παράδοση

Ο κώδικα σας, μαζί με ένα τεστ αρχείο που δείχνει ότι δουλεύουν σωστά όλες οι εντολές, τα αποτελέσματα εκτέλεσης των παραπάνω 2 σετ δοκιμών.

Ο κώδικας πρέπει να μεταγλωτίζεται και να λειπουργεί ορθά στα μηχανήματα της σχολής.

Library

