

Structuri de date (Seria CB)

Tema 3 - Bibliotecă online

Responsabili temă	Roxana Știucă, Florin Iancu, Bianca Mihaela Cauc
Data publicării	10.05.2021
Termen predate	27.05.2021 (23:59) Se acceptă teme trimise cu penalizare de 10 puncte/zi (din maxim 150 de puncte) până la data de 30.05.2021 (ora 23:59)
Versiune document	1.4
Versiune checker	1.4

1. Introducere

După ce a urmărit toate serialele de pe Netflix, Andrei și-a găsit o nouă ocupație: cititul. Prietenii săi i-au dat foarte multe recomandări de cărți, iar el nu le poate reține pe toate. Așa că vom crea un program în care Andrei își poate nota cărțile pe care vrea să le citească.

Fiecare carte va avea: un **titlu** unic, un **autor**, un **rating** (număr întreg, oferit de prietenul lui Andrei odată cu recomandarea) și **numărul de pagini**.

Andrei trebuie să poată **să introducă** cărți noi și **să caute** după cărți deja introduse într-un mod eficient. De asemenea, programul trebuie să aibă o opțiune de **auto-complete** pentru când Andrei nu își amintește în totalitate titlul sau autorul unei cărți.

2. Implementare

Pentru gestionarea cărților, vom folosi următoarele structuri de date:

1. **T1**: un arbore de prefixe (trie), în care cheia este titlul cărții, iar informația reținută la finalul căii este un pointer către o structură cu informațiile cărții;
2. **T2**: un alt arbore de prefixe (trie), în care cheia este un autor, iar informația reținută este un alt arbore de prefixe (trie), în care sunt reținute cărțile scrise de acel autor, reținute asemenea structurii 1.

2.1. Arbori de prefixe (Trie)

Este un arbore multi-căi (fiecare nod poate avea oricât descendenți).

Este un tip de arbore de căutare cu următoarele proprietăți:

- Fiecare nod poate avea între 0 și $|\Sigma|$ descendenți ($|\Sigma|$ reprezintă dimensiunea alfabetului).
- Spre deosebire de un arbore binar de căutare, unde fiecărui nod i se asociază o cheie, la arborele de prefixe cheile se asociază căii până la nod.

Exemplu:

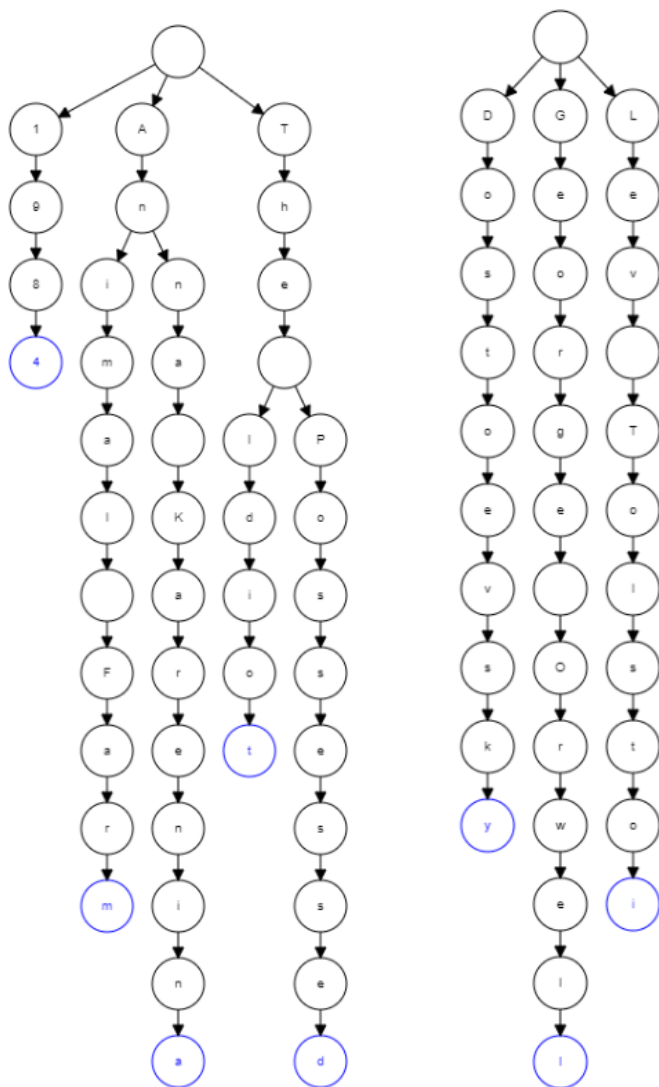
Andrei a primit următoarele recomandări de cărți:

- "1998", George Orwell;
- "Anna Karenina", Lev Tolstoi;
- "The Idiot", Dostoevsky;
- "Animal Farm", George Orwell;
- "The Possessed", Dostoevsky.

Structurile sunt populate ca mai jos. Fiecare nod *albastru* din T2 are o legătură către un alt arbore de prefixe, asemănător arborelui T1, dar conținând doar cărțile scrise de autorul respectiv.

Structura T1:

Structura T2:



3. Cerință

Va trebui să gestionați cărțile adăugate de Andrei pe baza unui set de comenzi citite dintr-un fișier.

3.1. Adăugare recomandare carte

Sintaxă: `add_book <titlu>:<autor>:<rating>:<numar_pagini>`

Mod de funcționare: cartea este adăugată atât în T1, cât și în arborele de căți corespunzător autorului său din T2. Dacă pentru autor nu există încă o cheie în T2, atunci va fi adăugat.

Dacă în T1 și T2 deja există cartea, atunci structurile rămân nemodificate.

Exemplu: `add_book Wuthering Heights:Emily Bronte:9:391 /* se parcurge arborele T1 pe calea dată de titlul "Wuthering Heights". Dacă nodurile din această cale nu existau deja, ele sunt create. În ultimul nod, se păstrează informația legată de carte; similar, se parcurge arborele T2 pe calea dată de numele autorului "Emily Bronte", se crează arborele corespondent cărților sale și apoi în acela se adaugă pe calea "Wuthering Heights" informația legată de carte. */`

3.2. Căutare carte

Sintaxă: `search_book <titlu>`

Mod de funcționare: cartea este căutată în T1. Dacă nu există, va fi întors un mesaj de forma:

Cartea <titlu> nu exista in recomandarile tale.

Altfel, dacă este găsită cartea, se va afișa un mesaj de forma:

Informatii recomandare: <titlu>, <autor>, <rating>, <numar_pagini>

Exemplu: `search_book Wuthering Heights /* se caută în T1 calea "Wuthering Heights" */`

Cartea Wuthering Heights nu exista in recomandarile tale. /* în caz că nu se găsește în T1 */

Informatii recomandare: Wuthering Heights, Emily Bronte, 9, 391 /* în caz că a fost găsită */

3.3. Auto-complete pentru căutare carte

Sintaxă: `search_book <prefix_titlu>~`

Mod de funcționare: se caută în T1 cărți al căror titlu începe cu <prefix_titlu>. Se afișează pe câte un rând primele maxim 3 titluri de cărți care încep cu <prefix_titlu> (primele din punct de vedere lexicografic).

<titlu_cartel1>

<titlu_carte2>

<titlu_carte3>

Dacă nu există nicio carte care începe cu prefix_titlu, se afișează mesajul:

Nicio carte gasita.

Exemplu: `search_book Wuthering~ /* caută în T1 calea "Wuthering" și afișează primele maxim 3 chei până la un nod ce conține o carte */`

Wuthering Heights

3.4. Afișare cărți autor

Sintaxă: `list_author <autor>`

Mod de funcționare: se va căuta în T2 după numele autorului. După vor fi afișate toate cărțile din arborele de căutare corespondent lui, în ordine lexicografică:

<titlu_cartel1>

<titlu_carte2>

<titlu_carte3>

...

Dacă autorul nu există în T2, atunci se va afișa un mesaj de forma:

Autorul <autor> nu face parte din recomandările tale.

Exemplu: `list_author Emily Bronte /* caută în T2 după cheia "Emily Bronte" și parcurge arborele corespondent cărților sale, afișând toate titlurile */`
Wuthering Heights

3.5. Auto-complete pentru afișare cărți autor

Sintaxă: `list_author <prefix_autor>~`

Mod de funcționare: se caută în T2 autori al căror nume începe cu <prefix_autor>. Se afișează pe câte un rând primii maxim 3 autori care încep cu <prefix_autor> (primii din punct de vedere lexicografic).

<nume_autor1>

<nume_autor2>

<nume_autor3>

Dacă nu există niciun autor care începe cu <prefix_autor>, se va afișa mesajul:
Niciun autor gasit.

3.6. Căutare în cărțile unui autor

Sintaxă: `search_by_author <autor>:<titlu>`

Mod de funcționare: se caută în T2 după numele autorului, iar apoi în arborele de căutare corespondent lui, după titlul cărții. Dacă autorul nu există, se afișează mesajul:
Autorul <autor> nu face parte din recomandările tale.

Altfel, mesajele sunt aceleași ca pentru comanda `search_book <titlu>`.

Exemplu: `search_by_author Emily Bronte:Wuthering Heights /* se parcurge T2 pe calea "Emily Bronte", iar în arborele corespondent autoarei, se caută calea "Wuthering Heights" */`

3.7. Auto-complete pentru căutare în cărțile unui autor

Sintaxă: `search_by_author <prefix_autor>~`

Mod de funcționare: idem comanda `list_author <prefix_autor>~`.

Sintaxă: `search_by_author <autor>:<prefix_titlu>~`

Mod de funcționare: idem comanda `search_book <prefix_titlu>~`, doar că se va căuta mai întâi în T2 după numele autorului, apoi în arborele de căutare corespondent lui, după <prefix_titlu>.

Dacă autorul nu există în T2, atunci se va afișa un mesaj de forma:

Autorul <autor> nu face parte din recomandările tale.

3.8. Ștergere recomandare

Sintaxă: delete_book <titlu>

Mod de funcționare: comanda șterge atât informațiile despre carte, cât și cheile din T1 și din arborele aferent autorului din T2. Dacă nu mai există alte cărți pentru acel autor, atunci trebuie să fie ștearsă și cheia autorului din T2, precum și eliberată memoria ocupată de arborele de căutare corespondent cărților sale.

Dacă titlul nu există, atunci se va afișa un mesaj de forma:
Cartea <titlu> nu exista în recomandările tale.

4. Restricții și precizări

- Titlul unei cărți este un șir de maxim 50 de caractere.
- Nu există două cărți cu același nume.
- Ordinea lexicografică este dată de următorul alfabet (fără ghilimele):
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789.-'?! "
- Numele unui autor este un șir de maxim 40 de caractere. Pentru simplitate, vom considera că toate caracterele din numele autorului aparțin alfabetului de mai sus.
- Programul va fi rulat astfel: ./tema3 input_file output_file.
- T1 și T2 respectă structura descrisă. Datele nu vor fi păstrate în niciun alt mod.

5. Exemple

Intrare	Ieșire
add_book Bosnia - A Short History:Noel Malcolm:4:384 add_book Wild Horses:Dick Francis:4:352 add_book Inferno:Dante Alighieri:4:396 add_book Ports of Call:Jack Vance:4:300 add_book Going Native:Stephen Wright:4:320 add_book Seventeen and In-Between:Barthe DeClements:4:176 add_book Personal Finance for Dummies:Eric Tyson:4:454 search_book Seventeen and In-Between search_book Young Warriors - Stories of Strength search_book Ports of Call list_author Jack Vance list_author Arthur C. Clarke search_by_author Jack Vance:Ports of Call search_by_author Anita Shreve:Eden Close list_author D~ list_author Step~ search_book Ab~	Informatii recomandare: Seventeen and In-Between, Barthe DeClements, 4, 176 Cartea "Young Warriors - Stories of Strength" nu exista in recomandările tale. Informatii recomandare: Ports of Call, Jack Vance, 4, 300 Ports of Call Autorul Arthur C. Clarke nu face parte din recomandările tale. Informatii recomandare: Ports of Call, Jack Vance, 4, 300 Autorul Anita Shreve nu face parte din recomandările tale. Dante Alighieri Dick Francis Stephen Wright Nicio carte gasita. Personal Finance for Dummies Jack Vance Seventeen and In-Between

<pre> search_book Per~ search_by_author Jack~ search_by_author Barthe DeClements:Seven~ delete_book Ports of Call delete_book The Anderson Tapes - Deadly Sins no 1 delete_book Wild Horses search_book Shakespeare's Romeo and Juliet search_book Ports of Call search_book Open Secrets search_book Personal Finance for Dummies </pre>	<p>Cartea "The Anderson Tapes - Deadly Sins no 1" nu exista in recomandarile tale.</p> <p>Cartea "Shakespeare's Romeo and Juliet" nu exista in recomandarile tale.</p> <p>Cartea "Ports of Call" nu exista in recomandarile tale.</p> <p>Cartea "Open Secrets" nu exista in recomandarile tale.</p> <p>Informatii recomandare: Personal Finance for Dummies, Eric Tyson, 4, 454</p>
---	---

Explicații:

1. Se adaugă cheia "Bosnia - A Short History" în T1 și cheia "Noel Malcolm" în T2. Se rețin informațiile despre carte.
Se procedează la fel și pentru restul comenzilor "add_book".
2. Pentru fiecare comandă search_book, este parcurs arborele T1 de la rădăcină, pe calea indicată de titlul căutat. Dacă nu există calea indicată sau la finalul său nu sunt marcate informații despre o carte, înseamnă că acea carte nu se află în recomandări.
3. Pentru fiecare comandă list_author, este parcurs arborele T2 de la rădăcină, pe calea indicată de numele autorului. Dacă nu există calea indicată sau finalul căii nu se află un arbore aferent autorului, înseamnă că autorul nu se află în recomandări. Altfel, este parcurs întreg arborele aferent autorului și sunt afișate toate cheile (titlurile cărților sale).
4. Pentru fiecare comandă terminată în '~' (auto-complete), se parcurge arborele T1 sau T2 (în funcție de comandă) pe calea indicată de prefix. Apoi, din nodul curent, se încearcă găsirea a maxim 3 chei din subarbore.
5. Pentru fiecare comandă search_by_author, se parcurge întâi arborele T2, apoi arborele trie aferent aceluia autor.
6. Comenzile delete_book șterg informațiile despre carte, cheia dată de titlul cărții din T1 și din arborele aferent autorului din T2. De asemenea, dacă autorul nu mai are nicio altă carte în recomandări, este distrus arborele de căutare aferent cărților sale și ștearsă cheia din T2.

6. Notare

- **135 de puncte** obținute pe testele de pe vmchecker. **Observatie:** Se pot obține punctaje parțiale fara a implementa toate comenzile (fiecare test valoreaza 5 puncte).
 - Testele 1-7 folosesc doar comenzile
 - `add_book <titlu>:<autor>:<rating>:<numar_pagini>`
 - `search_book <titlu>`
 - Testele 8-10 folosesc in plus comanda:
 - `list_author <autor>`
 - Testele 11-13 folosesc in plus comanda:
 - `search_by_author <autor>:<titlu>`

- Testele 14-22 folosesc in plus comenzile:
 - `list_author <prefix_autor>~`
 - `search_book <prefix_titlu>~`
 - `search_by_author <prefix_autor>~`
 - `search_by_author <autor>:<prefix_titlu>~`
- Testele 23-27 folosesc in plus comanda:
 - `delete_book <titlu>`
- **10 puncte:** coding style;
- **5 puncte:** README - va conține detaliile de implementare a temei, precum și **punctajul obținut la teste** (la rularea pe calculatorul propriu)
- **bonus: 20 de puncte** pentru soluțiile care nu au memory leak-uri (bonusul se acordă doar dacă testul a trecut cu succes);
- temele care nu compilează, nu rulează sau obțin punctaj 0 pe vmchecker, vor primi punctaj 0;
- se depun punctează pentru:
 - warninguri la compilare (trebuie compilat cu -Wall);
 - linii mai lungi de 80 de caractere;
 - folosirea incorectă de pointeri;
 - lipsa verificărilor codurilor de eroare întoarse de funcții;
 - alte situații nespecificate aici, dar considerate inadecvate.

7. Reguli de trimitere a temelor

- temele vor trebui încărcate atât pe vmchecker (în secțiunea **Structuri de Date Seria CB**) cât și pe *curs.upb.ro*, în secțiunea aferentă temei 3.
- arhiva cu rezolvarea temei trebuie să fie .zip și să conțină:
 - fișiere sursă (fiecare fișier sursă va trebui să înceapă cu un comentariu de forma: `/* NUME Prenume - grupa */`)
 - fișier **README**, denumit obligatoriu astfel, care să conțină detalii despre implementarea temei;
 - fișier **Makefile**, denumit obligatoriu astfel, cu două reguli:
 - **build**, care va compila sursele și va obține executabilul cu numele **tema3**
 - **clean**, care va șterge executabilele și alte fișiere obiect generate
- arhiva trebuie să conțină doar fișierele sursă (inclusiv Makefile și README); nu se acceptă fișiere executabile sau obiect.
- dacă arhiva nu respectă specificațiile de mai sus nu va fi acceptată la upload și tema nu va fi luată în considerare.