

## Proiect de Programare / C++

**-- Sistem gestiune & interacțiune bibliotecă --**

**Student 1:** Popovici Carlos-Patric

**Student 2:** Plugar Viorel-Nicolae

### I. Enunț

**Studentul 1** se ocupă de gestionarea inventarului unei biblioteci (similar cu gestionarea unui magazin ):

- adăugare, ștergere, modificare cărți (în principal schimbarea numărului de exemplare disponibile).
- vizualizare a tuturor împrumuturilor active (similar cu vizualizarea comenzilor ).

**Studentul 2** se ocupă de interacțiunea cu biblioteca din perspectiva unui membru (similar cu interacțiunea unui client cu un magazin ):

- adăugarea și ștergerea unei cărți dintr-o listă temporară de împrumut (echivalentul unui coș de cumpărături ).
- crearea unui împrumut pe baza listei, finalizând astfel cererea (echivalentul creării unei comenzi ).

### II. Structura datelor folosite de echipă

Pentru a folosi `char[ ]` în loc de `string`, trebuie să definim dimensiuni maxime pentru fiecare câmp text.

C++

```
// Definim constante pentru dimensiunile maxime
const int MAX_ISBN = 20;
const int MAX_TITLU = 100;
const int MAX_AUTOR = 50;
const int MAX_ID_MEMBRU = 30;
const int MAX_CARTI_PER_IMPRUMUT = 10;
```

Vom folosi clasele adaptate, analog cu clasele `Produs`, `Data` și `Comanda`:

- **Carte:**
  - `char isbn[MAX_ISBN];`

- char titlu[MAX\_TITLU];
- char autor[MAX\_AUTOR];
- int exemplare\_disponibile;
- **Data:**
  - int zi, luna, an;
- **Imprumut:**
  - char id\_membru[MAX\_ID\_MEMBRU];
  - Data data\_imprumut;
  - char lista\_isbn\_carti[MAX\_CARTI\_PER\_IMPRUMUT][MAX\_ISBN];
  - int numar\_carti\_imprumutate;

### III. Structura fișierelor

Vom folosi fișiere text pentru a stoca datele, similar cu stoc.txt, comenzi.txt și cos\_cumparaturi.txt:

- **catalog.txt** - fișier în care stocăm detaliile despre cărțile din bibliotecă.

```
<numar total de titluri>
<isbn_carte1> <titlu_carte1> <autor_carte1> <exemplare_disponibile1>
<isbn_carte2> <titlu_carte2> <autor_carte2> <exemplare_disponibile2>
...
```

- **imprumuturi.txt** - fișier în care stocăm detalii despre împrumuturile active.

```
<id_membru1> <data_imprumut1> <lista_isbn_carti_imprumutate1>
<id_membru2> <data_imprumut2> <lista_isbn_carti_imprumutate2>
...
```

- **cerere\_imprumut.txt** - fișier temporar în care Studentul 2 reține cărțile pe care dorește să le împrumute.

```
<isbn_carte1>
<isbn_carte2>
...
```

### IV. Interacționarea cu executabilele

Aplicațiile vor expune funcționalități prin linia de comandă, la fel ca în exemplul original.

**Aplicația 1 (Bibliotecar) va expune următoarele posibilități:**

- `./app_bibliotecar.exe vizualizare_catalog`
- `./app_bibliotecar.exe adaugare_carte <isbn> <titlu> <autor> <exemplare>`
- `./app_bibliotecar.exe stergere_carte <isbn>`
- `./app_bibliotecar.exe modificare_exemplare <isbn> <numar_nou_exemplare>`
- `./app_bibliotecar.exe vizualizare_imprumuturi`

**Aplicația 2 (Membru) va expune următoarele posibilități:**

- `./app_membru.exe vizualizare_cerere`
- `./app_membru.exe adaugare_cerere <isbn>`
- `./app_membru.exe stergere_cerere <isbn>`
- `./app_membru.exe imprumuta <id_membru>`