

Laborator 13 – The Final Lab

Tema: Catalog studenti 

Autor: ing. Eduard Donea

Poveste

În cadrul unui laborator, un asistent dorește să gestioneze rapid o listă de studenți, împreună cu o singură notă finală pentru fiecare student. Pentru a păstra informația organizată și ușor de prelucrat, vom construi un mini-sistem de tip **catalog** folosind clase C++.

Scopul laboratorului

- definirea și utilizarea mai multor clase în C++;
 - folosirea compozitiei (o clasă conține obiecte din alta clasă);
 - alocare dinamică pentru un tablou de studenți;
 - afisarea informației în consolă;
 - testarea rapidă în `main.cpp`.
-

Structura proiectului

- **Nota** – clasă simplă care reține o valoare numerică (0–10);
 - **Student** – clasă care reține nume + o notă finală;
 - **Catalog** – clasă care gestionează un tablou dinamic de studenți;
 - **main.cpp** – fișier de testare.
-

Clasa Nota

Clasa **Nota** este o clasa foarte simpla care contine:

- o valoare intre 0 si 10 (int sau float).

Se cere:

- constructor cu parametru (valoare);
- metoda `getValue()`.

Clasa Student

Clasa **Student** va contine:

- un nume (sir de caractere / `std::string`);
- o nota finala (obiect de tip **Nota**).

Se cere:

- constructor cu parametri (nume, nota);
- metoda de afisare simpla (ex: `print()` sau operator `<<`).

Clasa Catalog

Clasa **Catalog** gestioneaza un tablou dinamic de studenti si contine:

- un tablou dinamic de **Student**;
- numarul de studenti.

Se cere:

- constructor cu parametru (**n**) care alocă tabloul;
- metoda `setStudent(i, ...)` pentru completarea unui student;
- metoda `printAll()` care afiseaza tot catalogul;
- metoda `mediaCatalogului()` care calculeaza media notelor.

Important: Clasa **Catalog** trebuie sa aiba **destructor** care elibereaza memoria.

Fișierul main.cpp

Fișierul `main.cpp` este folosit pentru testarea functionalitatii.

Model minim de test:

- creati un obiect `Catalog` cu 3 studenti;
- completati studentii folosind `setStudent`;
- afisati catalogul;
- afisati media catalogului.

Cerințe (10p)

1. (3p) Implementarea claselor `Nota` si `Student`.
2. (3p) Implementarea clasei `Catalog` cu alocare dinamica.
3. (2p) Afisare corecta a studentilor si calculul mediei.
4. (2p) Demonstratie in `main.cpp`.

Compilare și rulare

```
g++ -std=c++17 *.cpp -o lab13
./lab13
```

Bonus GitHub Classroom

Pentru obtinerea punctajului bonus:

1. Acceptati tema de pe GitHub Classroom: <https://classroom.github.com/a/pAubWfK>
2. Implementati rezolvarea completa.
3. Incarcati proiectul in repository-ul generat.

Atenție: Repository-urile sunt verificate automat. Codul copiat intre studenti duce la anularea punctajului bonus.