

Laborator 4

Fisierele trebuie impartite astfel:

- student.h → declararea clasei
 - student.cpp → implementarea clasei
 - main.cpp → testarea functiilor
-

Fisier: Carte.h

```
class Student{
    char *nume;
    char *facultate;
    int an_studiu;
    float medie;
public:
    Student(); //constructor fara parametri
    Student(const char* nume, const char* facultate, int an_studiu, float medie);
    Student(const Student&); //constructor de copiere
    Student& operator=(const Student&); //operatorul de asignare
    ~Student(); //destructorul

    void afisare() const; //afiseaza attributele clasei
    char* getNume() const;
    char* getFacultate() const;
    int getAnStudiu() const;
    float getMedie() const;

    void setNume(const char* nume);
    void setFacultate(const char* facultate);
    void setAnStudiu(int an);
    void setMedie(float medie);
};
```

Cerinte:

5p Implementati clasa Student (constructori, destructor, operator=, getteri, setteri, afisare).

1p Creati un vector alocat dinamic de tip Student.

1p Afisati studentii cu media mai mare decat o valoare citita de la tastatura.

1p Testati constructorul de copiere si operatorul de asignare(operator=).

1p Eliberati memoria alocata dinamic (vectorul).

1p din oficiu

Cum se compileaza si ruleaza

1. Compileaza programul:

- `g++ -g main.cpp student.cpp -o lab4`

2. Ruleaza simplu:

- `./lab4`

3. Ruleaza cu Valgrind:

- `valgrind ./lab4`

Exemple de mesaje:

- “definitely lost” → memorie ne-eliberata
 - “Invalid read/write” → acces invalid dupa eliberare sau in afara vectorului
-