

Laborator 4

Fisierele trebuie impartite astfel:

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

Fisier: Carte.h

```
class Carte{
    char *titlu;
    char *autor;
    int an_publicare;
    int numar_pagini;
public:
    Carte(); //constructor fara parametri
    Carte(const char* titlu, const char* autor, int an_publicare, int numar_pagini);
    Carte(const Carte&); //constructor de copiere
    Carte& operator=(const Carte&); //operatorul de asignare
    ~Carte(); //destructorul

    void afisare() const; //afiseaza attributele clasei
    char* getTitlu() const;
    char* getAutor() const;
    int getAnPublicare() const;
    int getNumarPagini() const;

    void setTitlu(const char* titlu);
    void setAutor(const char* autor);
    void setAnPublicare(int an);
    void setNumarPagini(int numar);
};
```

Cerinte:

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

1p Creati un vector alocat dinamic de tip Carte.

1p Afisati cartile publicate dupa un anumit an citit 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 carte.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
-