

Tema laborator 2

Mod de lucru:

Veti deschide un proiect nou (eg. DevCpp: New->Project->Console Application; bifati C++ Project; dati un nume proiectului; precizati calea unde vreti sa fie salvat).

In proiect, pe langa fisierul main.cpp, generat automat, veti adauga un fisier header Angahat.h ce contine interfata clasei data mai jos si un fisier sursa Angajat.cpp ce contine implementarea metodelor clasei (in el adaugati #include "Angajat.h").

In main.cpp – veti face toate testele necesare si implementarile aferente cerintelor. Veti include "Angajat.h" ca sa aveti acces la acest tip de date.

Incarcarea temei:

In contul Lab 2, semigrupa voastra, incarcati fisierele Angajat.h, Angajat.cpp si main.cpp.

Cerinta:

O firma doreste sa tina evidenta persoanelor angajate. Se doreste:

1. ca angajatii sa fie stocati intr-un vector (alocat dinamic), sortati dupa nume
2. sa fie afisate toate detaliile despre toti angajatii
3. sa se poata modifica datele unui angajat, folosind ca parametru datele unui alt angajat
4. sa se afle cate femei, respectiv, barbati sunt angajati
5. sa fie afisati toti angajatii cu salariu>3000

Pentru a tine evidenta este nevoie de tipul de date (clasa) Angajat cu attributele de mai jos.

//fisier Angajat.h

```
class Angajat{
```

```
    char*nume;
```

```
    char CNP[14];
```

```
    int salariu;
```

```
public:
```

```
    Angajat(); //constr. fara param; util pt. alocarea spatiului pt. elem. din vectorul cu angajati;
```

```
        // nume=NULL; CNP este de ex: "000..0"; salariu=0
```

```
    Angajat(char*, char[], int); //constr. cu param; pentru a initializa cu valori toate attributele;
```

```
    Angajat(const Angajat&); //constructor de copiere – se apeleaza automat la transmiterea
```

```
        //parametrilor in fctii prin valoare; returnarea obj din fctii prin valoare;
```

```
        // puneti cout<<"const de copiere"; in impl. ca sa vedeti unde e apelat;
```

```

~ Angajat(); //elibereaza spatiu ocupat de obiecte

void afisare() const; //afiseaza toate attributele unui angajat

void modificare(const Angajat&); // face o copie profunda a parametrului si o stocheaza in
                                //objectul cu care se face apelul;

int getSalariu() const;

bool eFemeie() const;

}; //si orice alta metoda considerati ca mai este necesara

```

```

//Angajat.cpp
//aici o sa faceti implementarea metodelor

```

Exemplu de utilizare a obiectelor si metodelor:

```

//main.cpp
Angajat p, p1("aa","11",4000);
Angajat p2(p1);
p2.afisare();
p.modificare(p2);
p2.afisare();

```