



Tehnici avansate de programare POO C++

Conf. univ. ing. dr. Lefkovits Szidónia
E-mail: szidonia.lefkovits@umfst.ro
<http://sites.umfst.ro/lefkovits-szidonia>
utilizator: TAP
parola: CursTAP2020



Laborator 5

Moștenirea claselor



Probleme propuse/ Teme

Problema 1

- 1. Să se creeze clasa Angajat cu:
- proprietatea tarifOrar de tip double și cu valoare inițială 5.5
- proprietate nrOre de tip int
- metoda getSalar care întoarce tarifOrar * nrOre
- metoda getTarifOrar care întoarce valoarea proprietății tarifOrar
- Constructor, copy-constructor, destructor, operator=
- Să se suprascrie operatorii >>, <<



Probleme propuse/ Teme

Problema 1

- Să se creeze main care să conțină metoda main în care să se instanțieze un obiect din clasa Angajat și să afișeze tariful orar și salariul angajatului.
- Să se creeze o clasă Manager care să moștenească Angajat și să aibă o metodă getSalar care să adauge un bonus de 50% față de salariul unui Angajat.
- Clasa Angajat să aibă o proprietate în plus numărul de subordonați.
- În metodele constructor, destructor, copy-constructor și operator=, operator>>, operator<< se apelează metodele corespunzătoare din clasa părinte.
- Tot în clasa Main să se instanțieze un obiect din clasa Manager și să se afișeze tariful orar și salariul managerului.



Probleme propuse/ Teme

Problema 2

2. Fie clasa Punct2D și clasa Punct3D care moștenește Punct2D.

- Fie clasa **Punct2D** care va avea 2 proprietăți x și y abscisa și ordonata.
- Și în acest caz se vor defini 2 constructori și 1 destructor, operator=
- Să se suprascrie operatorii <, <=, >, >=, ==, !=, +, -, *, /, +=, -=, *=, /=, ++, --, <<, >>
- În main să se creeze n instanțe ale clasei Punct2D, într-un vector de Puncte2D. n se citește de la tastatură.
- În clasa Punct2D se va defini o funcție numită distanță, care va calcula distanța între două puncte date în 2D.



Probleme propuse/ Teme

Problema 2

- Fie clasa **Punct3D** care moștenește clasa **Punct2D** și va avea o proprietate z în plus.
- Și în acest caz se vor defini 2 constructori și 1 destructor, operatorul=
- Să se suprascrie operatorii $<, <=, >, >=, ==, !=, +, -, *, /, +=, -=, *=, /=, ++, --, <<, >>$
- În main să se creeze m instanțe ale clasei **Punct3D**, într-un vector de **Puncte3D**. m se citește de la tastatură.
- În clasa **Punct3D** se va suprascrie funcție numită **distanță**, care va calcula distanța între două puncte date în 3D.



Probleme propuse/ Teme

Problema 3

- 3. Să se implementeze Stiva statică și Stiva dinamică folosind moștenirea între clase.
- Să se implementeze operatorii $=$, $<<$ și $>>$
- Clasa Nod să fie o clasă imbricată a clasei ListStack.



Diagrama de clasă

