**Nr. 1**

Se considera urmatoarele clase:

***BonDeMasa*** ce contine atributele: **id**(int), **emitent**(sir de caractere de lungime variabila), **valoare**(float).

***Angajat*** ce contine atributele: **id**(int), **nume**(sir de caractere de lungime fixa 50), **nrBonuri**(int), **bonuri** (vector de obiecte de tip BonDeMasa).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

BonDeMasa b1(1,"GustoPass",9.35), b2, b3;

b3 = b1;

Angajat a1(22,"Ionescu"), a2;

cout<<a1;//afisarea tuturor informatiilor despre un angajat

cin>>b2;

cin>>a2;//citirea tuturor informatiilor despre un angajat

Angajat a3(a2), a4;

a3 \*= b1; //atribuie bonul de masa b1 angajatului a3 prin salvarea lui b1 in vectorul bonuri din a3

a4 = a3;

if (a1 < a2) cout<<"Angajatul "<<a1<<" are id-ul mai mic decat "<<a2;

a3.afiseaza(); //afisarea tuturor informatiilor despre un angajat.

**Nr. 2**

Se considera urmatoarele clase:

***Ingredient*** ce contine atributele: **nume**(sir de caractere de lungime variabila), **pret**(float), **cantitate**(int).

***Pizza*** ce contine atributele: **denumire**(sir de caractere de lungime fixa 50), **nrIngrediente**(int), **Ingr**(vector de obiecte de tip Ingredient).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Ingredient i1("Piper",0.5, 10), i2("Nimic", 0.0, 0);

cout<<i1;

cin>>i2;

i1 += 500; // aduna o valoare la cantitatea disponibila

Ingredient i3 = i1, i4, i5(i1);

i4 = i1;

Pizza pizza1("Clasic"), pizza2;

pizza1 = i1 + pizza1 // adaugare ingredient 1 in pizza 1

cin>>pizza2;

cout<<pizza1<<pizza2;

if (pizza1 > pizza2) cout<<"pizza1 are mai multe ingrediente decat pizza2";

else cout<<"pizza2 are mai multe ingrediente decat pizza1";

**Nr. 3**

Se considera urmatoarele clase:

***Data*** ce contine atributele: **zi**(int), **luna**(sir de caractere de lungime fixa 20), **an**(int).

***Proprietar*** ce contine atributele: **nume**(sir de caractere de lungime variabila), **data\_nasterii**(Data).

***Casa*** ce contine atributele: **oras**(sir de caractere de lungime fixa 50), **nrProprietari**(int), **Prop**(vector de obiecte de tip Proprietar).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Data d1(15,”iulie”,1975), d2=d1;

Proprietar p1(“Ionescu”,d1), p2(“Popescu”,2,”ianuarie”,1980);

Casa c1(“Bucuresti”), c2, c3;

c1+=p1; // adauga proprietarul p1 in lista de proprietari din casa c1

c1+=p2; // adauga proprietarul p2 in lista de proprietari din casa c1

cout<<c1;

cin>>c2;

c3 = c2;

cout<<c3;

**Nr. 4**

Se considera urmatoarele clase:

***Data*** ce contine atributele: **zi**(int), **luna**(sir de caractere de lungime fixa 20), **an**(int).

***Actor*** ce contine atributele: **nume**(sir de caractere de lungime variabila), **data\_nasterii**(Data).

***Film*** ce contine atributele: **titlu**(sir de caractere de lungime fixa 50), **nrActori**(int), **Act**(vector de obiecte de tip Actor).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Data d1(15,”iulie”,1975), d2=d1;

Actor a1(“Ionescu”,d1), a2(“Popescu”,2,”ianuarie”,1980);

Film f1(“Matrix”), f2, f3;

f1+=p1; // adauga actorul a1 in lista filmului f1

f1+=p2; // adauga actorul a2 in lista filmului f1

cout<<f1;

cin>>f2;

f3 = f2;

cout<<f3;

**Nr. 5**

Se considera urmatoarele clase:

***Data*** ce contine atributele: **zi**(int), **luna**(sir de caractere de lungime fixa 20), **an**(int)

***Medicament*** ce contine atributele: **denumire**(sir de caractere de lungime variabila), **pret**(float), **fabricat**(Data).

***Farmacie*** ce contine atributele: **denumire**(sir de caractere de lungime fixa 50), **nrMedicamente**(int), **Med**(vector de obiecte de tip Medicament).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Data d(1,”ianuarie”,2018);

Medicament m1("Parasinus",9.5, 10), m2("Aspirina", d),m3;

cin>>m3;

cout<<m1;

if (m1<m2) cout<<”m1 este fabricat inaintea m2”;

else cout<<”m2 este fabricat inaintea m1”;

Farmacie f1(“Farmac”);

f1 += m1; //adaugare medicament m1 in lista de medicamente a farmaciei

f1 = m2 + f1; //adaugare medicament m2 in lista de medicamente a farmaciei

Farmacie f2 = f1;

cout<<f2; //afisarea tuturor medicamentelor

**Nr. 6**

Se considera urmatoarele clase:

***Materie*** ce contine atributele: **denumire**(sir de caractere de lungime fixa 50), **optional**(true=1/false=0)

***Student*** ce contine atributele: **nrMatricol**(int), **nume**(sir de caractere de lungime variabila), **grupa**(int).

***Curs*** ce contine atributele: **mat**(Materie), **nrStudenti**(int), **St**(vector de obiecte de tip Student).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Materie m1(“POO”,0), m2(“Pedagogie”,1),m3;

if (m1==m2) cout<<”Cursuri identice”;

else cout<<”Denumiri diferite”;

Student s1(12,“Ionescu”,133), s2;

cin>>s2; cout<<s1<<s2;

Curs c1(m1);

c1+=s1;//adauga student s1 in lista studentilor cursului c1

c1 = c1 + s2; //adauga student s2 in lista studentilor cursului c1

Curs c2 = c1;

cout<<c2;

m3 = c2.materie(); cout<<m3;

**Nr. 7**

Se considera urmatoarele clase:

***Caracteristica*** - atributele: **domeniu**(sir de caractere de lungime fixa 20), **taxa**(0/1 – 0 valoare default);

***Student -*** (**nr**(int), **nume**(sir de caractere de lungime variabila), **car**(Caracteristica), **grupa**(int));

***Facultate*** - (**denumire**(sir de caractere de lungime fixa 50), **nrStudenti**(int), **St**(vector de obiecte Student)).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Caracteristica c1(“Informatica”,1), c2(“Matematica”);

Student s1(1,”Ionescu”,c1,132), s2(2,”Popescu”,”Matematica”,1,102), s3(s1), s4,s5[3];

s4 = s2; cout<<s3<<s4;

for (int i=0;i<3;i++) cin>>s5[i];

Facultate f1(“Automatica”), f2(“FMI”,3,s5), f3;

f3 = f1;

f3 += s1; // se adauga s1 listei de studenti ai facultatii f3

f2 += s2; // se adauga s2 listei de studenti ai facultatii f2

f2.taxa(); // afiseaza toti studentii la taxa ai facultatii f2

if (f2 != f3) cout<<”Nu exista studenti inscrisi la ambele facultati”;

else cout<<”Exista studenti la ambele facultati”;

cout<<f3.

**Nr. 8**

Se considera urmatoarele clase:

***Caracteristica*** - atributele: **functie**(sir de caractere de lungime fixa 20), **vechime**(int);

***Angajat*** ce contine atributele: **id**(int), **nume**(sir de caractere de lungime fixa 50), **car**(Caracteristica);

***Banca*** - (**denumire**(sir de caractere de lungime fixa 50), **nrAngajati**(int), **Ang**(vector de obiecte Angajat)).

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Caracteristica c1(“Director”,10), c2(“Casier”), c3 = c1;

c3 = 12 + c3; // actualizeaza vechimea adaugand 12 ani

Angajat a1(1,”Ionescu”,c1), a2(2,”Popescu”,”Functionar”,25), a3(a1), a4,a5[3];

for (int i=0;i<3;i++) cin>>a5[i];

cout<<a5[1];

Banca b1(“Bank”,50), b3, b4, b5(“Bank2”,3,a5);

b4 = b3 = b1;

for (int i=0;i<3;i++) b3 += a5[i]; // se adauga angajatii din vectorul a5 listei de angajati ai bancii b3

Banca b2(b3);

b2.modificare(1,-5); // scade vechimea celui de-al doilea angajat cu 5 ani

cout<<b3.

**Nr. 9**

Se considera urmatoarele clase:

***Diagnostic –*** atributele: **diag**(sir de caractere de lungime variabila), **zi**(int), **luna**(int), **an**(int);

***Carnet\_de\_Sanatate*** - atributele: **nr**(int), **nrDiagnostice** (int), **dia**(vector de Diagnostice);

***Pacient*** ce contine atributele: **id**(int), **nume**(sir de caractere de lungime fixa 50), **carnet**(Carnet\_de\_Sanatate);

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Diagnostic d1(“Clinic sanatos”,1,1,2017), d2(“Gripa”,12,3,2017), d3[2];

d3[0] = d1; d3[1] = d2;

Carnet\_de\_Sanatate c1(12345), c2(45,2,d3), c3;

cin>>c3;

c2 \*= “Pneumonie”;//Adauga in vectorul de diagnostice si creste numarul de elemente

Pacient p1(1,”Ionescu”,c1), p2(p1),p3,p4;

cin>>p3;

if (p1==p3); // Afiseaza mesaj daca cei 2 pacienti au acelasi nume sau nu

p4 += c2 // Ataseaza carnetul c2 pacientului p4.

cout<<p4;

**Nr. 10**

Se considera urmatoarele clase:

***Autor –*** atributele: **nume**(sir de caractere de lungime fixa 50), **prenume**(sir de caractere de lungime variabila);

***Carte*** - atributele: **aparitie**(int), **titlu** (sir de caractere de lungime fixa 100)**, nrAutori** (int), **Aut**(vector de Autori);

***Detalii -*** atributele: **isbn**(int), **car** (Carte);

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Autor a1(“Slavici”,”Ioan”), a2(“Seneca”), a3;

cin>>a3;

Carte c1(1881,”Moara cu noroc”, 1, a1), c2(2017,”Dictionar”), c3(c1), c4, c5;

cin>>c4; c5 = c4;

c2 += a1;//Adauga pe Ioan Slavici in vectorul de autori si creste numarul de elemente

c2.modif(1, “Popescu”); //modifica numele primului autor in “Popescu”

if (c3 < c5) // Afiseaza mesaj “cartea” c3 “a aparut inaintea cartii” c5

cout <<c1<<c2<<c4;

Detalii d1(1234), d2(2587,c1),d3;

cin>>d3;

if (d1==d3) // Afiseaza mesaj “cele doua carti sunt identice”

cout<<d2;

**Nr. 11**

Se considera urmatoarele clase:

***Punct2D –*** atributele: **x**(int), **y**(int);

***Figura*** - atributele: **denumire** (sir de caractere de lungime variabila)**, nrPuncte** (int), **P**(vector de Punct2D);

***Geometrie -*** atributele: **nrFiguri**(int), **fig** (vector de obiecte de tip Figura);

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Punct2D p1(1,-1), p2(2), p3[4],p4[2];

for( int x=0;x<4;x++) cin>>p3[x];

p4[0] = p1; p4[1] = p2;

Figura f1(“triunghi dreptunghic”,3), f2(“Segment”,2,p4), f3[2];

cin>>f3[0];

f3[1] = f3[0];

f3[1] += p1;//Adauga un punct unei figuri si schimba denumirea acesteia

if (f1 != f3[1]) // Afiseaza mesaj “figura” f1 “diferita de figura ” f3[1].

Geometrie g1, g2(2,f3), g3(1,”patrat”,4,0,0,0,1,1,1,1,0), g4 = g3;

cin>>g1;

g3 -= 1; //sterge primul punct, decrementeaza nr de puncte si schimba denumirea in “triunghi”

cout<<g3<<g4;

**Nr. 12**

Se considera urmatoarele clase:

***Sofer –*** atributele: **nume**(sir de caractere de lungime fixa 50), **prenume**(sir de caractere de lungime fixa 50);

***Masina*** - atributele: **nr** (sir de caractere de lungime variabila)**, nrSoferi** (int), **So**(vector de Sofer);

***Fisa -*** atributele: **id**(int), **ma** (Masina);

Definiti clasele si completati-le cu metode si supraincarcari de operatori pentru a permite in functia main() urmatoarele operatii:

Sofer s1(“Ionescu”,”Ion”), s2[5];

cin>>s2[0]; s2[1] = s2[0];

for( int x=0;x<2;x++) cout<<s2[x];

Masina m1(“B 23 ABC”),m2(“B 10 DEF”,2,s2),m3(m1);

m3 += s1; //adauga soferul 1 masinii m3

int poz; cin>>poz;

m3 = poz + m3;//adauga un sofer masinii m3 pe pozitia poz (sau la final daca poz > nr soferi)

if (m2 == m3)// afiseaza mesaj “masinile” m1 “si” m2 sunt conduse de aceiasi soferi.

cout<<m1<<m3;

Fisa f1(1,m1), f(2,”B 59 XYZ”,1,”Popescu”,”George”);

f1+=”Georgescu” //adauga soferul Georgescu masinei din fisa f1

if (f1 > f2) //afiseaza mesaj “masina de pe fisa 1 are mai multi soferi asignati”.