1.

#include <iostream>

using namespace std;

class Proizvod {

private: char \*Naziv;

private: char \*Proizvodjac;

private: float Cijena;

private: int Kolicina;

public: void Unos() {

}

public: void Ispis() {

}

};

void main() {

//Unijeti podatke o tri proizvoda, te ih ispisati.

system("pause>0");

}

2.

#include <iostream>

using namespace std;

class Apartman {

private: char \*Naziv;

private: char \*Mjesto;

private: float Cijena;

private: int BrojKreveta;

public: void Unos() {

}

public: void Info() {

}

public: float GetCijena() {

//vraća cijenu apartmana

}

public: bool IsFromMostar() {

//funkcija koja vraća informaciju da li je apartman iz Mostara

}

};

float GetProsjecnaCijena(Apartman \*niz, int vel) {

//Funkcija koja računa prosječnu cijenu apartmana

}

int GetBrojApartmanaInMostar(Apartman \*niz, int vel) {

//Funkcija koja vraća koliko je apartmana iz Mostara

}

void GetNajjeftinijiApartman(Apartman niz[], int vel) {

}

void main() {

int n;

cout << "Koliko apartmana želite dodati? ";

cin >> n;

cin.ignore();

Apartman \*apartmani = new Apartman[n];

for (int i = 0; i < n; i++)

apartmani[i].Unos();

system("cls");

for (int i = 0; i < n; i++) {

apartmani[i].Info();

cout << "===========================================\n\n";

}

cout << "Prosječna cijena apartmana iznosi: " << GetProsjecnaCijena(apartmani, n) << " KM\n";

cout << "Broj apartmana u Mostaru: " << GetBrojApartmanaInMostar(apartmani, n) << endl;

GetNajjeftinijiApartman(apartmani, n);

system("pause>0");

}

3.

#include<iostream>

using namespace std;

class Pregled {

public: char \*Dijagnoza;

private: double cijenaPregleda;

public: void Unos() {

}

public: void Ispis() {

}

public: void Dealociraj() {

}

public: double GetCijenaPregleda() {

}

};

class Pacijent {

private: char \*imePrezime;

private: Pregled \* preglediPacijenata[10];

private: int brojPregleda;

public:void Unos() {

}

public: void Ispis() {

//Ispis osnovnih informacija o pacijentu, te o njegovim pregledima

}

public: void DodajPregled() {

//Dodaje novi pregled za pacijenta

//voditi racuna da pacijent moze imati maksimalno 10 pregleda

}

public: double GetUkupneTroskove() {

//Funkcija vraća ukupnu cijenu svih pregleda pacijenta

}

public:void Dealociraj() {

}

public: bool HasDijagnoza(char \*dijagnoza) {

}

};

void PretragaPoDijagnozi(Pacijent \*pacijenti, int vel) {

//Vrši pretragu pregleda svih pacijenata po dijagnozi

//Ispisuje podatke pacijenta koji ima unesenu dijagnozu

//Unutar funkcije tražiti od korisnika da unese dijagnozu koju pretražuje

//te prema tome ispisati podatke pacijenata koji imaju unesenu dijagnozu

}

void main() {

int vel;

cout << "Unesite broj pacijenata koje zelite dodati:";

cin >> vel;

cin.ignore();

Pacijent \*pacijenti = new Pacijent[vel];

for (int i = 0; i < vel; i++)

{

pacijenti[i].Unos();

for (int j = 0; j < 2; j++)//za svakog pacijenta se dodaju po dva pregleda

pacijenti[i].DodajPregled();

}

PretragaPoDijagnozi(pacijenti, vel);

cout << "Ukupni troškovi: \n";

for (int i = 0; i < vel; i++)

{

cout<<i+1<<". pacijent: "<<pacijenti[i].GetUkupneTroskove()<<endl;

}

for (int i = 0; i < vel; i++)

{

pacijenti[i].Dealociraj();

}

delete[]pacijenti;

pacijenti = NULL;

system("pause>0");

}