

3. Objektų rinkinys

Susipažinsite su:

- masyvo struktūra ir masyvo tipo kintamuoju, kuriame bus saugomas objektų rinkinys;
- tekstiniais failais, kuriuose saugomi objektų duomenys ir į kuriuos surašomi skaičiavimų rezultatai;
- sumos, kiekio, aritmetinio vidurkio skaičiavimo algoritmais;
- didžiausios (mažiausios) reikšmės paieškos algoritmu;
- dviejų objektų rinkinių apjungimo į vieną algoritmu;

3.1. Pažintis su objektų rinkiniu

- Masyvas objektų rinkiniui saugoti.
- Sumos, kiekio ir aritmetinio vidurkio skaičiavimo algoritmų taikymas.
- Duomenų skaitymas iš tekstinio failo.

Užduotis. Dviratis.

Dviračių nuomos punkte yra įvairių modelių dviračiai. Žinomi kiekvieno dviračio pagaminimo metai ir kaina.

Parašykite programą, kuri:

- suskaičiuotų, kiek yra tinkamų naudoti dviračių (ne senesni kaip m metų) ir kokia jų piniginė vertė;
- kiek yra netinkamų naudoti dviračių ir kokia jų piniginė vertė;
- koks vidutinis tinkamų naudoti ir jau netinkamų naudoti dviračių amžius.

Pradiniai duomenys ir rezultatai.

Pradiniai duomenys	Paaškinimai
15	Kritinis dviračio naudojimo laikas
5	Dviračių skaičius
1935; 365,10;	Dviračio pagaminimo metai ir kaina
1988; 87,09;	Dviračio pagaminimo metai ir kaina
1995; 58,10;	Dviračio pagaminimo metai ir kaina
2008; 285,58;	Dviračio pagaminimo metai ir kaina
2012; 1040,42;	Dviračio pagaminimo metai ir kaina
Skaičiavimų rezultatai	
Tinkami naudoti: 2 1326,00	
Netinkami naudoti: 3 510,29	
Tinkamų naudoti dviračių vidutinis amžius: 5,00	
Netinkamų naudoti dviračių vidutinis amžius: 42,33	

Programos kūrimo eiga.

- Sukuriama klasė dviračio duomenims saugoti.
- Parašomi klasės metodai, skirti duomenims dėti ir imti.
- Paruošiamas duomenų failas.
- Paskelbiamas objektų rinkinys.
- Parašomas duomenų skaitymo iš failo metodas, kuris suteikia rinkinio objektams reikšmes.
- Parašomas nurodyto amžiaus intervalo dviračių kiekio ir jų piniginės vertės skaičiavimo metodas.
- Parašomas nurodyto amžiaus intervalo dviračių amžiaus vidurkio skaičiavimo metodas.

Pirmas žingsnis.

- Sukurkite klasę, skirtą dviračio duomenims saugoti. Jos aprašą rašykite virš pagrindinio metodo `Main()`.

```
//-----  
/** Klasė dviračio duomenims saugoti  
    @class Dviratis */  
class Dviratis  
{  
    private int metai;        // pagaminimo metai
```

```

private double kaina;    // piniginė vertė

//-----
/** Dviračio duomenys
@param metai - nauja metų reikšmė
@param kaina - nauja kainos reikšmė */
//-----
public Dviratis(int metai, double kaina)
{
    this.metai = metai;
    this.kaina = kaina;
}

/** grąžina metus */
public int ImtiMetus() { return metai; }

/** grąžina dviračio reikšmę */
public double ImtiKainą() { return kaina; }
}
//-----

```

- Patikrinkite, ar klasė aprašyta teisingai, ar teisingai dirba sukurti metodai. Tam pagrindiniame metode Main() parašykite darbo su klasės objektu imitaciją:

```

//-----
class Program
{
    static void Main(string[] args)
    {
        Dviratis d;
        d = new Dviratis(2012, 25.26);
        Console.WriteLine("{0} {1, 6:f2} \n", d.ImtiMetus(), d.ImtiKainą());
        Console.WriteLine("Programa baigė darbą!");
    }
}
//-----

```

- Įvykdysite programą. Ekrane turėtumėte matyti dviračio duomenis:

```

2012  25.26
Programa baigė darbą!
Press any key to continue . . .

```

Antras žingsnis.

- Paruoškite duomenų failą.
- Parašykite rinkinio (masyvo) maksimalų galimą dydį apibrėžiančią konstantą:

```
const int Cn = 100;
```

- Parašykite duomenų failo vardą, kuris nurodomas konstanta:

```
const string CFd = "...\\...\\Duom.txt";
```

- Pašalinkite darbo su objektu d veiksmus.
- Objektą d pakeiskite objektų rinkiniu D[Cn], kur Cn – konstanta, reiškianti didžiausią galimą turėti dviračių skaičių.
- Parašykite du sveikojo tipo kintamuosius, skirtus rinkinio duomenų skaičiui ir dviračio naudojimo kritinio amžiaus reikšmėms saugoti.
- Parašykite duomenų skaitymo iš failo metodą.

```

//-----
/** Skaityti duomenis iš failo
@param fv - failo vardas, kuris nurodomas konstanta CFd
@param D - objektų rinkinys dviračių duomenims saugoti
@param n - dviračių skaičius
@param m - kritinis dviračių naudojimo amžius */
//-----
static void Skaityti(string fv, Dviratis[] D, out int n, out int m)

```

```

int metai;
double kaina;
using (StreamReader reader = new StreamReader(fv))
{
    string line;
    line = reader.ReadLine();
    string[] parts;
    m = int.Parse(line);
    line = reader.ReadLine();
    n = int.Parse(line);
    for (int i = 0; i < n; i++)
    {
        line = reader.ReadLine();
        parts = line.Split(';');
        metai = int.Parse(parts[0]);
        kaina = double.Parse(parts[1]);
        D[i] = new Dviratis(metai, kaina);
    }
}
}

```

- Papildykite pagrindinį metodą Main() kreipiniu į parašytą skaitymo metodą.

```

//-----
const int Cn = 100;
const string CFd = "...\\...\\Dum.txt";

static void Main(string[] args)
{
    Dviratis [] D = new Dviratis[Cn]; // dviračių duomenys - objektai
    int n; // dviračių skaičius
    int am; // dviračio tinkamumo naudoti kritinis amžius
    Skaityti(CFd, D, out n, out am);
    Console.WriteLine("Dviračių kiekis {0}\\n", n); // kontrolinis spausdinimas
    Console.WriteLine("Programa baigė darbą!");
}
//-----

```

- Įvykdykite programą. Jeigu ekrane matote skaičių 5, tai reiškia, kad galbūt duomenys teisingai perskaityti. Norėdami įsitikinti, ar taip tikrai yra, tikslinga kontrolinį spausdinimą papildyti sakiniais, kurie spausdintų ekrane visus dviračių rinkinio duomenis:

```

Console.WriteLine("Pagaminimo metai      Kaina");
for (int i = 0; i < n; i++)
    Console.WriteLine("      {0}      {1, 7:f2}",
        D[i].ImtiMetus(), D[i].ImtiKaina());
Console.WriteLine();

```

- Įvykdykite programą. Vėliau kontrolinio duomenų spausdinimo sakiniai nebus reikalingi, todėl juos bus galima pašalinti.

Trečias žingsnis.

- Parašykite metodą, skaičiuojantį nurodyto amžiaus intervalo dviračių [amPr, amPb] kiekį ir jų piniginę vertę. Toks metodas universalus, nes galima atlikti veiksmus skirtingiems amžių intervalams.

```

//-----
/** Skaičiuoja tinkamo amžiaus dviračių kainų sumą ir kiekį
@param D - dviračių duomenys
@param n - dviračių skaičius
@param amPr - dviračių paieškos amžiaus intervalo pradžia
@param amPb - dviračių paieškos amžiaus intervalo pabaiga
@param metai - metai, kurių atžvilgiu skaičiuojamas amžius
@param kiek - dviračių skaičius duotame amžiaus intervale
@param suma - duoto amžiaus intervalo dviračių piniginė vertė */
//-----
static void Pinigai(Dviratis[] D, int n, int amPr, int amPb, int metai,
    out int kiek, out double suma)

```

```

{
    kiek = 0;
    suma = 0.0;
    int amžius;
    for (int i = 0; i < n; i++)
    {
        amžius = metai - D[i].ImtiMetus();
        if ((amPr <= amžius) && (amžius <= amPb))
        {
            suma = suma + D[i].ImtiKainą();
            kiek++;
        }
    }
}
}
//-----

```

- Papildykite pagrindinį metodą Main() kreipiniu į metodą Pinigai() ir gautų rezultatų spausdinimo sakiniu:

```

int kiekTinka;
double sumaTinka;
Pinigai(D, n, 0, am, 2015, out kiekTinka, out sumaTinka);
Console.WriteLine("Tinkami naudoti: {0,3:d} {1,7:f2}", kiekTinka, sumaTinka);

```

- Įvykdykite programą ir patikrinkite gautus rezultatus.
- Papildykite programą veiksmais dėl netinkamų naudoti dviračių:

```

int kiekNetinka;
double sumaNetinka;
// Netinkami naudoti dviračiai, kurių amžius didelis,
// t.y. intervale nuo m iki begalybės (pvz., 1000 metų)
Pinigai(D, n, am + 1, 1000, 2015, out kiekNetinka, out sumaNetinka);
Console.WriteLine("Netinkami naudoti: {0} {1,7:f2}\n", kiekNetinka,
    sumaNetinka);

```

- Įvykdykite programą ir patikrinkite gautus rezultatus.

Ketvirtas žingsnis.

- Parašykite metodą, skaičiuojantį nurodyto dviračių amžiaus intervalo vidurkį:

```

//-----
/** Skaiciuoja nurodyto dviračių amžiaus intervalo vidurkį
@param D - dviračių duomenys
@param n - dviračių skaičius
@param metai - metai, kurių atžvilgiu skaičiuojamas dviračio amžius
@param amPr - dviračių paieškos amžiaus intervalo pradžia
@param amPb - dviračių paieškos amžiaus intervalo pabaiga */
//-----
static double Vidurkis(Dviratis[] D, int n, int metai, int amPr, int amPb)
{
    int kiek = 0, suma = 0;
    int amžius;
    for (int i = 0; i < n; i++)
    {
        amžius = metai - D[i].ImtiMetus();
        if ((amPr <= amžius) && (amžius <= amPb))
        {
            suma = suma + amžius;
            kiek++;
        }
    }
    if (kiek > 0) return (double) suma / kiek;
    return 0.0;
}
//-----

```

- Papildykite pagrindinį metodą Main() likusiais skaičiavimais:

```

double vidurkisTinka = Vidurkis(D, n, 2015, 0, am);
if (vidurkisTinka > 0)
    Console.WriteLine("Tinkamų naudoti dviračių vidutinis amžius: {0,7:f2}" ,

```

```

        vidurkisTinka );
    else Console.WriteLine("Tinkamų naudoti dviračių nėra");

    double vidurkisNetinka = Vidurkis(D, n, 2015, am + 1, 1000);
    if (vidurkisNetinka > 0)
        Console.WriteLine("Netinkamų naudoti dviračių vidutinis amžius: {0,7:f2}\n",
            vidurkisNetinka);

    else Console.WriteLine("Netinkamų naudoti dviračių nėra\n");

```

- Pasitikrinkite gautus atsakymus:

```

Dviračių kiekis 5

Pagaminimo metai      Kaina
1935                  365,10
1988                  87,09
1995                  58,10
2008                  285,58
2012                  1040,42

Tinkami naudoti: 2    1326,00
Netinkami naudoti: 3    510,29

Tinkamų naudoti dviračių vidutinis amžius:      5,00
Netinkamų naudoti dviračių vidutinis amžius:    42,33

Programa baigė darbą!
Press any key to continue . . .

```

Programos papildymas.

- Parašykite pagrindiniame metode `Main()` sakinius, kurie suskaičiuotų ir išspausdintų visų dviračių piniginę vertę ir visų dviračių vidutinį amžių.
- Parašykite pagrindiniame metode `Main()` sakinius, kurie suskaičiuotų ir išspausdintų dviračių, kurie buvo pagaminti nurodytais (pvz., 2012) metais, piniginę vertę ir visų dviračių vidutinį amžių.
- Parašykite pagrindiniame metode `Main()` sakinius, kurie suskaičiuotų ir išspausdintų dviračių, kurie buvo pagaminti 1000 metais (tokių nėra), piniginę vertę ir visų dviračių vidutinį amžių. Ar programa pateikia teisingus rezultatus?

Savarankiško darbo užduotis.

Žinoma, kiek kuris turistinės grupės narys turi pinigų (eurai, centai). Duomenys tekstiniame faile. Parašykite programą, kuri suskaičiuotų, kiek grupė turi pinigų iš viso ir kiek vidutiniškai jų tenka kiekvienam nariui. Kiekvienas narys bendroms grupės išlaidoms skiria ketvirtadalį turimų pinigų. Kiek pinigų bus iš viso surinkta bendroms grupės išlaidoms?

3.2. Du objektų rinkiniai

- Duomenys dviejuose tekstiniuose failuose.
- Veiksmai su dviem objektų rinkiniais.
- Objektų rinkinių charakteristikų radimas.

Užduotis. Dviratis.

Dviejuose dviračių nuomos punktuose yra įvairių modelių dviračiai. Žinomi kiekvieno dviračio pagaminimo metai ir kaina.

Parašykite programą, kuri suskaičiuotų:

- kiek yra tinkamų naudoti dviračių (ne senesni kaip m metų) ir kokia jų piniginė vertė;
- kiek yra netinkamų naudoti dviračių ir kokia jų piniginė vertė;
- vidutinį tinkamų naudoti ir jau netinkamų naudoti dviračių amžių.

Pradiniai duomenys.

Pirmojo nuomos punkto duomenys	Paiškinimai
Lėk su vėjeliu	Nuomos punkto pavadinimas
15	Kritinis dviračio naudojimo laikas
5	Dviračių skaičius
1935; 365,10;	Dviračio pagaminimo metai ir kaina

1988; 87,09;	Dviračio pagaminimo metai ir kaina
1995; 58,10;	Dviračio pagaminimo metai ir kaina
2008; 285,58;	Dviračio pagaminimo metai ir kaina
2012; 1040,42;	Dviračio pagaminimo metai ir kaina

Antrojo nuomos punkto duomenys	Paaiškinimai
Neskubėk ir būsi pirmas	Nuomos punkto pavadinimas
10	Kritinis dviračio naudojimo laikas
6	Dviračių skaičius
1955; 290,02;	Dviračio pagaminimo metai ir kaina
1990; 100,13;	Dviračio pagaminimo metai ir kaina
1995; 145,06;	Dviračio pagaminimo metai ir kaina
2010; 260,94;	Dviračio pagaminimo metai ir kaina
2012; 1040,42;	Dviračio pagaminimo metai ir kaina
2012; 435,07;	Dviračio pagaminimo metai ir kaina

Programos kūrimo eiga.

- Nukopijuojama ankstesnio pratimo programa.
- Paruošiamas antras duomenų failas.
- Programa papildoma dviem objektų rinkiniais ir veiksmams su jais.

Pirmas žingsnis.

- Nukopijuokite ankstesnio pratimo programą.
- Papildykite programą antru duomenų rinkiniu.
- Duomenų rinkinio pavadinimą D pakeiskite pavadinimu D1 bei kitus kintamųjų pavadinimus. Papildykite programą kintamaisiais, skirtais nuomos punkto pavadinimui saugoti ir antruoju objektų rinkiniu:

```
// Pirmojo dviračių nuomos punkto
Dviratis [] D1 = new Dviratis[Cn]; // dviračių duomenys
int n1; // dviračių skaičius
int am1; // dviračio tinkamumo naudoti kritinis amžius
string pav1; // nuomos punkto pavadinimas
int kiekTinka1, kiekNetinka1;
double sumaTinka1, sumaNetinka1;

// Antrojo dviračių nuomos punkto
Dviratis [] D2 = new Dviratis[Cn]; // dviračių duomenys
int n2; // dviračių skaičius
int am2; // dviračio tinkamumo naudoti kritinis amžius
string pav2; // nuomos punkto pavadinimas
int kiekTinka2, kiekNetinka2;
double sumaTinka2, sumaNetinka2;
```

- Papildykite duomenų skaitymo metodą parametrais, skirtais failo vardui perduoti ir grąžinti perskaitytam nuomos punkto pavadinimui:

```
//-----
/** Skaity duomenis iš failo
@param fv - failo vardas
@param D - objektų rinkinys dviračių duomenims saugoti
@param n - dviračių skaičius
@param m - kritinis dviračių naudojimo amžius
@param pav - nuomos punkto pavadinimas */
//-----
static void Skaityti(string fv, Dviratis[] D, out int n, out int m, out string pav)
{
    int metai;
    double kaina;
    using (StreamReader reader = new StreamReader(fv))
    {
        string line;
        line = reader.ReadLine();
        string[] parts;
        pav = line;
    }
}
```

```

        line = reader.ReadLine();
        m = int.Parse(line);
        line = reader.ReadLine();
        n = int.Parse(line);
        for (int i = 0; i < n; i++)
        {
            line = reader.ReadLine();
            parts = line.Split(';');
            metai = int.Parse(parts[0]);
            kaina = double.Parse(parts[1]);
            D[i] = new Dviratis(metai, kaina);
        }
    }
}
//-----

```

- Pakeiskite ir papildykite failų vardų konstantų sąrašą:

```

const string CFd1 = "...\\...\\D uom1.txt";
const string CFd2 = "...\\...\\D uom2.txt";

```

- Parašykite kreipinį į duomenų skaitymo metodą pirmo nuomos punkto duomenims skaityti:
Skaityti(CFd1, D1, out n1, out am1, out pav1);

- Patikrinkite, ar pirmojo dviračių nuomos punkto duomenys perskaityti teisingai.

Antras žingsnis.

- Papildykite programą sakiniais, skirtais antrojo nuomos punkto duomenims skaityti. Patikrinkite, ar antrojo dviračių nuomos punkto duomenys perskaityti teisingai.

Trečias žingsnis.

- Papildykite programą sakiniais, skaičiuojančiais, kiek abiejuose nuomos punktuose yra tinkamų naudoti dviračių ir kokia jų piniginė vertė bei vidutinis amžius. Spausdinkite gautus rezultatus.
- Patikrinkite, ar gauti rezultatai teisingi.
- Papildykite programą sakiniais, skaičiuojančiais, kiek abiejuose nuomos punktuose yra netinkamų naudoti dviračių ir kokia jų piniginė vertė bei vidutinis amžius. Spausdinkite gautus rezultatus.
- Patikrinkite, ar gauti rezultatai teisingi.

Ketvirtas žingsnis.

- Parašykite sakinius, kuriais nustatoma, kuriame punkte yra daugiau tinkamų naudoti dviračių.
- Parašykite sakinius, kuriais nustatoma, kurio punkto turimų dviračių piniginė vertė yra didesnė.

Savarankiško darbo užduotis.

Barboros turimos valiutos surašytos faile B.txt, o Anupro – faile A.txt. Vienoje eilutėje – vienos šalies valiutos duomenys: pinigai (pvz., doleriai ir centai) ir kursas, perskaičiuojant į eurus ir centus. Parašykite programą, kuri suskaičiuotų, kiek pinigų eurai ir centais turi Barbora ir kiek Anupras atskirai ir kiek abu bendrai.

3.3. Struktūriniai duomenys.

- Didžiausios (mažiausios) reikšmės paieška.
- Naujo objektų rinkinio formavimas.

Užduotis. Dviratis.

Dviejuose dviračių nuomos punktuose yra įvairių modelių dviračiai. Žinomas kiekvieno dviračio modelis, kiekis, pagaminimo metai ir kaina.

Parašykite programą, kuri:

- surastų, kuriame dviračių nuomos punkte yra seniausias dviratis;
- surastų, kuriame dviračių nuomos punkte yra brangiausias dviratis ir kokia jo piniginė vertė;
- suformuotų dviračių modelių, kurie yra tuo pačiu metu abiejuose nuomos punktuose, rinkinį.

Pradiniai duomenys.

Pirmojo nuomos punkto duomenys	Paiškinimai
Lėk su vėjeliu 5 ORION; 2; 1935; 365,1; POLARIS; 1; 1988; 87,09; INDUSTRIERAD; 10; 1995; 58,10; Bauer; 25; 2008; 285,58; HERREN; 15; 2012; 1040,42;	Nuomos punkto pavadinimas Įrašų skaičius Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina

Antrojo nuomos punkto duomenys	Paiškinimai
Neskubėk ir būsi pirmas 6 VOLTA; 1; 1955; 290,02; POLARIS; 2; 1990; 100,13; BANANA; 3; 1995; 145,06; INDUSTRIERAD; 10; 2010; 260,94; ORION; 15; 2012; 1040,42; POLARIS; 20; 2012; 945,07;	Nuomos punkto pavadinimas Įrašų skaičius Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina Dviračio modelis, kiekis, pagaminimo metai ir kaina

Programos kūrimo eiga.

- Ankstesnio pratimo programos teksto kopija modifikuojama darbui su dviračių rinkiniais, kai žinomi dviračių modelių pavadinimai.
- Parašomas objektų rinkinio duomenims spausdinti failė metodas.
- Parašomas metodas seniausiam dviračiui rasti nurodytame objektų rinkinyje.
- Parašomas metodas brangiausiam dviračiui rasti nurodytame objektų rinkinyje.
- Parašomas metodas, skirtas rasti nurodyto modelio dviračiui nurodytame objektų rinkinyje.
- Parašomi metodai, skirti formuoti bendram objektų sąrašui iš turimų dviejų.

Pirmas žingsnis.

- Paruoškite duomenų failus, suteikdami jiems vardus Nuoma1.txt ir Nuoma2.txt.
- Papildykite duomenų failus duomenimis pagal pateiktą pavyzdį.
- Nukopijuokite ankstesnio pratimo programos tekstą ir išsaugokite nauju vardu.
- Pašalinkite visus metodus, išskyrus duomenų skaitymo iš failo. Iš pagrindinio metodo Main() pašalinkite kreipinius į pašalintus metodus.
- Papildykite klasę Dviratis kintamaisiais, skirtais modelio pavadinimui ir dviračių kiekiui saugoti:

```
private int kiek;  
private string pav;
```

- Klasės konstruktorių Dviratis() papildykite parametrais, skirtais dviračio modelio vardui ir kiekiui dėti. Konstruktoriaus antraštė:

```
public Dviratis(string pav, int kiek, int metai, double kaina );
```

- Papildykite klasę metodais, grąžinančiais dviračio modelio pavadinimą ir jų kiekį:

```
public string ImtiPavadinimą() { return pav; }  
public int ImtiKiekį() { return kiek; }
```

- Įvykdykite programą. Jeigu nėra klaidų, programa sėkmingai baigia darbą nieko neskaičiuodama.

Antras žingsnis.

- Modifikuokite duomenų skaitymo iš failo metodą:

```
//-----  
/** Skaitymo duomenis iš failo  
@param Fd - failo vardas  
@param D - objektų rinkinys dviračių duomenims saugoti  
@param n - dviračių skaičius  
@param pav - nuomos punkto pavadinimas */  
//-----  
static void Skaityti(string Fd, Dviratis[] D, out int n, out string pav)  
{
```



```

using (StreamReader reader = new StreamReader(Fd))
{
    string eil; int kiekn; int metain; double kainan;
    string line;
    line = reader.ReadLine();
    string[] parts;
    pav = line;
    line = reader.ReadLine();
    n = int.Parse(line);
    for (int i = 0; i < n; i++)
    {
        line = reader.ReadLine();
        parts = line.Split(';');
        eil = parts[0];
        kiekn = int.Parse(parts[1]);
        metain = int.Parse(parts[2]);
        kainan = double.Parse(parts[3]);
        D[i] = new Dviratis(eil, kiekn, metain, kainan);
    }
}
}
//-----

```

- Parašykite pagrindinio metodo Main() duomenų skaitymo sakinius, patikrinkite, ar gerai skaitomi duomenys.

```

//-----
const string CFd1 = "...\\...\\Nuoma1.txt";
const string CFd2 = "...\\...\\Nuoma2.txt";
//-----
static void Main(string[] args)
{
    // Pirmojo dviračių nuomos punkto
    Dviratis [] D1 = new Dviratis[Cn]; // dviračių duomenys
    int n1; // dviračių skaičius
    string pav1; // nuomos punkto pavadinimas

    // Antrojo dviračių nuomos punkto
    Dviratis [] D2 = new Dviratis[Cn]; // dviračių duomenys
    int n2; // dviračių skaičius
    string pav2; // nuomos punkto pavadinimas

    Skaityti(CFd1, D1, out n1, out pav1);
    Skaityti(CFd2, D2, out n2, out pav2);

    // -----
    // I nuomos punktas, kontrolinis spausdinimas
    Console.WriteLine("Pirmas nuomos punktas: {0}", pav1);
    Console.WriteLine("Dviračių kiekis {0}\n", n1);
    Console.WriteLine("Pavadinimas Kiekis Pagaminimo metai Kaina");
    for (int i = 0; i < n1; i++)
        Console.WriteLine("{0,-12} {1,4:d} {2,3:d} {3, 7:f2}",
            D1[i].ImtiPavadinimą(), D1[i].ImtiKiekį(),
            D1[i].ImtiMetus(), D1[i].ImtiKainą());
    Console.WriteLine();

    Console.WriteLine("Programa baigė darbą!");
}
//-----

```

Trečias žingsnis.

- Parašykite objektų rinkinio spausdinimo faile metodą:

```

//-----
/** Spausdina duomenis į failą
@param fv - rezultatų failo vardas
@param D - objektų rinkinys dviračių duomenims saugoti
@param kiek - dviračių skaičius
@param pav - nuomos firmos pavadinimas */

```

```
//-----
static void SpausdintiDuomenis(string fv, Dviratis[] D, int nkiek, string pav)
{
    const string virsus =
        + "-----|-----|-----|-----|\r\n"
        + "      Pavadinimas      Kiekis      Pagaminimo      Kaina      \r\n"
        + "      metai      (eurų)      \r\n"
        + "-----|-----|-----|-----|";
    using (var fr = File.AppendText(fv))
    {
        fr.WriteLine("Nuomos firma: {0}", pav);
        fr.WriteLine(virsus);
        Dviratis tarp;
        for (int i = 0; i < nkiek; i++)
        {
            tarp = D[i];
            fr.WriteLine("| {0,-15} | {1,8} | {2,5:d} | {3,7:F2} |",
                tarp.ImtiPavadinimą(), tarp.ImtiKieki(),
                tarp.ImtiMetus(), tarp.ImtiKainą());
        }
        fr.WriteLine("-----");
    }
}
//-----
```

- Papildykite failų vardų konstantų sąrašą rezultatų failo vardo CFrez konstanta.
- Parašykite pagrindiniame metode Main() egzistuojančio rezultatų failo išmetimo sakinį:

```
if (File.Exists(CFrez))
    File.Delete(CFrez);
```

- Papildykite pagrindinį metodą Main() krepiniais į metodą Spausdinti(), kuriais būtų išspausdinami nuomos punktų duomenys. Patikrinkite, ar rezultatų faile gavote tokias lenteles:

Nuomos firma: Lėk su vėjeliu

Pavadinimas	Kiekis	Pagaminimo metai	Kaina (eurų)
ORION	2	1935	365,10
POLARIS	1	1988	87,09
INDUSTRIERAD	10	1995	58,10
Bauer	25	2008	285,58
HERREN	15	2012	1040,42

Nuomos firma: Nesukubėk ir būsi pirmas

Pavadinimas	Kiekis	Pagaminimo metai	Kaina (eurų)
VOLTA	1	1955	290,00
POLARIS	2	1990	100,00
BANANA	3	1995	145,00
INDUSTRIERAD	10	2010	260,00
ORION	15	2012	1040,00
POLARIS	20	2012	945,00

Ketvirtas žingsnis.

- Sukurkite metodą, kuris duotame objektų rinkinyje surastų seniausio dviračio indeksą:

```
//-----
/** Gražina dviračio, kurio metų skaičius yra mažiausias, indeksą
@param D - objektų rinkinys
@param n - objektų skaičius rinkinyje */
```

```
//-----
static int Seniausias(Dviratis [] D, int n)
{
    int k = 0;
    for (int i = 0; i < n; i++)
        if (D[i].ImtiMetus() < D[k].ImtiMetus())
            k = i;
    return k;
}
//-----
```

- Papildykite pagrindinį metodą Main() veiksmams, kuriais būtų surandamas nuomos punktas, kuriame yra seniausias dviratis:

```
using (var fr = File.AppendText(CFrez))
{
    if (D1[Seniausias(D1, n1)].ImtiMetus() < D2[Seniausias(D2, n2)].ImtiMetus())
        fr.WriteLine("Seniausias dviratis yra nuomos punkte {0} ", pav1);
    else fr.WriteLine("Seniausias dviratis yra nuomos punkte {0} ", pav2);
}
```

Galima parašyti kitaip:

```
int ind1 = Seniausias(D1, n1);
int ind2 = Seniausias(D2, n2);
using (var fr = File.AppendText(CFrez))
{
    if (D1[ind1].ImtiMetus() < D2[ind2].ImtiMetus())
        fr.WriteLine("Seniausias dviratis yra nuomos punkte {0} ", pav1);
    else fr.WriteLine("Seniausias dviratis yra nuomos punkte {0} ", pav2);
}
```

- Patikrinkite, ar programa dirba teisingai.
- Pakeiskite duomenis taip, kad seniausias dviratis būtų kitame nuomos punkte. Patikrinkite, ar programa dirba teisingai.
- Pakeiskite duomenis taip, kad abiejuose punktuose yra vienodi seniausi dviračiai. Kurio punkto pavadinimą matote? Patikslinkite programą taip, kad šiuo atveju programa ekrane rodytų abiejų punktų pavadinimus.
- Patikrinkite, kaip programa dirba, kai punkte / punktuose yra keli seniausi dviračiai.

Penktas žingsnis.

- Sukurkite metodą, kuris nustatytų, ar objektų sąrašas yra nurodyto modelio dviratis:

```
//-----
/** Gražina surasto dviračio masyve indeksą arba -1, jeigu dviračio masyve nėra
@param D - objektų rinkinys
@param n - objektų skaičius rinkinyje
@param pav - ieškomo modelio pavadinimas */
//-----
static int YraModelis(Dviratis [] D, int n, string pav, int metai)
{
    for (int i = 0; i < n; i++)
        if (D[i].ImtiPavadinimą() == pav && D[i].ImtiMetus() == metai) return i;
    return -1;
}
//-----
```

- Papildykite klasę metodu

```
public void PapildytiKiekį(int k) { kiek = kiek + k; }
```

- Sukurkite metodą, kuris dviračių modelių sąrašą papildytų dviračiais iš duoto objektų rinkinio. Įrašymo metu atsižvelgiama į pagaminimo metus:

```
//-----
/** Objektų rinkinį papildo duomenimis iš kito objektų rinkinio
// Jeigu objektų rinkinio D tokio pat modelio dviratis yra objektų rinkinyje Dr,
// tuomet didinamas kiekis, kitaip - papildomas nauju objektu
@param n - objektų skaičius rinkinyje D
@param D - objektų rinkinys, iš kurio pildo
```

```

@param nr - objektų skaičius rinkinyje Dr
@param Dr - objektų rinkinys, kurį pildo */
//-----
static void Formuoti(Dviratis [] D, int n, Dviratis [] Dr, ref int nr)
{
    int k;
    for (int i = 0; i < n; i++)
    {
        k = YraModelis(Dr, nr, D[i].ImtiPavadinimą(), D[i].ImtiMetus());
        if (k >= 0)
            Dr[k].PapildytiKiekį(D[i].ImtiKiekį()); // didinamas kiekis

        else
        {
            Dr[nr] = D[i]; // papildomas rinkinys
            nr++;
        }
    }
}
//-----

```

- Papildykite pagrindinį metodą Main() veiksmiais, kuriais formuojamas abiejuose nuomos punktuose esančių dviračių modelių rinkinys.

```

Dviratis[] Dr = new Dviratis[Cn];
int nr;
nr = 0;
Formuoti(D1, n1, Dr, ref nr);
Formuoti(D2, n2, Dr, ref nr);
SpausdintiDuomenis(CFrez, Dr, nr, "Modelių sąrašas");

```

- Patikrinkite, ar rezultatų faile išspausdinta teisinga modelių lentelė:

Modelių sąrašas

Pavadinimas	Kiekis	Pagaminimo metai	Kaina (eurų)
ORION	2	1935	365,10
POLARIS	1	1988	87,09
INDUSTRIERAD	10	1995	58,10
Bauer	25	2008	285,58
HERREN	15	2012	1040,42
VOLTA	1	1955	290,00
POLARIS	2	1990	100,00
BANANA	3	1995	145,00
INDUSTRIERAD	10	2010	260,00
ORION	15	2012	1040,00
POLARIS	20	2012	945,00

3.4. Simbolių eilutės keli žodžiai.

- Objektų rinkinys.
- Reikšmių įvedimas iš failo.
- Reikšmių spausdinimas į failą lentele.
- Metodai.

Užduotis. Firmos automobiliai.

Failo Duomenys.txt pirmoje eilutėje yra vienos UAB automobilių kiekis, kitose eilutėse – automobilių duomenys: pavadinimas, degalų tipas, degalų sąnaudos 100 km.

Sukurkite klasę automobilio duomenims saugoti. Į šios klasės objektų masyvą perskaitykite duomenis iš failo Duomenys.txt. Masyvo objektų reikšmės spausdinkite lentelę į failą Rezultatai.txt.

Pradiniai duomenys ir rezultatai.

Pradiniai duomenys faile Duomenys.txt		
8 Škoda Fabia;benzinas; 7,1; Opel Meriva;benzinas; 7,5; Volkswagen Golf;dyzelinas; 6,3; Opel Astra;dyzelinas; 6,1; Toyota Corolla;benzinas; 6,8; Toyota Yaris;benzinas; 5,9; Peugeot 207;dyzelinas; 5,5; Škoda Octavia;dyzelinas; 7,3;		
Rezultatai faile Rezultatai.txt		
Automobilių kiekis 8 Automobilių sąrašas:		
-----	-----	-----
Pavadinimas	Degalai	Sąnaudos (l/100 km)
-----	-----	-----
Škoda Fabia	benzinas	7,10
Opel Meriva	benzinas	7,50
Volkswagen Golf	dyzelinas	6,30
Opel Astra	dyzelinas	6,10
Toyota Corolla	benzinas	6,80
Toyota Yaris	benzinas	5,90
Peugeot 207	dyzelinas	5,50
Škoda Octavia	dyzelinas	7,30
-----	-----	-----

Programos kūrimo eiga.

- Sukuriama klasė vieno automobilio duomenims saugoti. Parašomi klasės sąsajos metodai.
- Pagrindiniame metode Main() skelbiamas automobilio klasės objektų masyvas.
- Parašomas metodas duomenims iš failo įvesti į objektų masyvą.
- Paruošiamas duomenų failas "Duomenys.txt".
- Parašomas metodas objektų masyvo reikšmėms spausdinti lentelę į failą.
- Pagrindinis metodas papildomas kreipiniais į parašytus metodus.

Pirmas žingsnis.

- Sukurkite klasę Auto ir jos sąsajos metodus:

```
//-----  
/** Klasė automobilio duomenims saugoti  
@class Auto */  
class Auto  
{  
    private string pav,           // automobilio pavadinimas  
                degalai;         // degalų tipas  
    private double sąnaudos;      // kuro sąnaudos 100 km  
  
    //-----  
    /** Automobilio duomenys  
    @param pav - nauja pavadinimo reikšmė  
    @param degalai - nauja degalų tipo reikšmė  
    @param sąnaudos - nauja sąnaudų reikšmė */  
    //-----  
    public Auto(string pav, string degalai, double sąnaudos)  
    {
```

```

        this.pav = pav;           // automobilio pavadinimas
        this.degalai = degalai;   // automobilio pavadinimas
        this.snaudos = snaudos;   // kuro snaudos 100 km
    }

    /** grąžina automobilio pavadinimas */
    public string ImtiPav() { return pav; }

    /** grąžina degalų tipą */
    public string ImtiDegalus() { return degalai; }

    /** grąžina degalų snaudą */
    public double ImtiSnaudos() { return snaudos; }
}
//-----

```

- Pagrindiniame metode `Main()` aprašykite klasės `Auto` objektų rinkinį `A` ir kintamąjį, kuriame bus saugomas UAB automobilių kiekis.

```

//-----
static void Main(string[] args)
{
    Auto [] A = new Auto[100];   // automobilių duomenys
    int na;                       // automobilių kiekis

    Console.WriteLine("Programa baigė darbą!");
}
//-----

```

Antras žingsnis.

- Parašykite duomenų įvedimo metodą.

```

//-----
/** Skaityti duomenis iš failo
@param Fd      - failo vardas
@param A       - objektų rinkinys automobilių duomenims saugoti
@param kiek    - dviračių skaičius */
//-----
static void Skaityti(string Fd, Auto[] A, out int kiek)
{
    using (StreamReader reader = new StreamReader(Fd))
    {
        string pav, degalai; double snaudos;
        string line;
        line = reader.ReadLine();
        string[] parts;
        kiek = int.Parse(line);
        for (int i = 0; i < kiek; i++)
        {
            line = reader.ReadLine();
            parts = line.Split(';');
            pav = parts[0];
            degalai = parts[1];
            snaudos = double.Parse(parts[2]);
            A[i] = new Auto(pav, degalai, snaudos);
        }
    }
}
//-----

```

Naudodamiesi pavyzdžiu paruoškite duomenų failą `Duomenys.txt`.

Trečias žingsnis.

- Parašykite objektų masyvo reikšmių spausdinimo metodą.

```

//-----
/** Spausdina duomenis į failą
@param fv      - rezultatu failo vardas
@param A       - objektų rinkinys automobilių duomenims saugoti
@param nkiek   - automobilių skaičius */

```

```
//-----
static void Spausdinti(string fv, Auto[] A, int nkiek)
{
    const string virsus =
        + "-----|-----|-----|\r\n"
        + "Pavadinimas    Degalai    Sąnaudos (l/100 km)|\r\n"
        + "-----|-----|-----|\r\n"
        + "-----|-----|-----|";
    using (var fr = File.AppendText(fv))
    {
        fr.WriteLine("Automobilių kiekis {0}", nkiek);
        fr.WriteLine("Automobilių sąrašas:");
        fr.WriteLine(virsus);
        Auto tarp;
        for (int i = 0; i < nkiek; i++)
        {
            tarp = A[i];
            fr.WriteLine("| {0,-15} | {1,-9} | {2,8:f2} |",
                tarp.ImtiPav(), tarp.ImtiDegalus(),
                tarp.ImtiSąnaudas());
        }
        fr.WriteLine("-----");
    }
}
//-----
```

Ketvirtas žingsnis.

- Papildykite pagrindinį metodą Main() esančio rezultatų failo ištrynimo veiksmams.

```
if (File.Exists(CFrez))
    File.Delete(CFrez);
```

- Papildykite pagrindinį metodą Main() kreipiniais į įvedimo ir spausdinimo metodus.

```
Skaityti(CFd, A, out na);
Spausdinti(CFrez, A, na);
```

- Įvykdysite programą. Rezultatų faile "Rezultatai.txt" matysite:

```
Automobilių kiekis 8
Automobilių sąrašas:
```

Pavadinimas	Degalai	Sąnaudos (l/100 km)
Škoda Fabia	benzinas	7,10
Opel Meriva	benzinas	7,50
Volkswagen Golf	dyzelinas	6,30
Opel Astra	dyzelinas	6,10
Toyota Corolla	benzinas	6,80
Toyota Yaris	benzinas	5,90
Peugeot 207	dyzelinas	5,50
Škoda Octavia	dyzelinas	7,30

Programos papildymas.

Pakeiskite programą taip, kad automobilių sąrašo lentelėje būtų spausdinami eilučių numeriai.

3.5. Veiksmai su objektais rinkinyje.

- Sumos, sandaugos, kiekio skaičiavimas.

Užduotis. Firmos automobilių degalų sąnaudos.

Failo "Duomenys.txt" pirmoje eilutėje yra vienos UAB automobilių kiekis, kitose eilutėse yra automobilių duomenys: pavadinimas, degalų tipas, degalų sąnaudos 100 km.

Sukurkite klasę automobilio duomenims saugoti. Į šios klasės objektų masyvą perskaitykite duomenis iš failo "Duomenys.txt".

Kiek degalų 100 km vidutiniškai sunaudoja vienas automobilis?

Duomenis ir rezultatus spausdinkite į failą "Rezultatai.txt".

Pradiniai duomenys ir rezultatai.

Pradiniai duomenys faile "Duomenys.txt"	
8	
Škoda Fabia;benzinas; 7,1;	
Opel Meriva;benzinas; 7,5;	
Volkswagen Golf;dyzelinas; 6,3;	
Opel Astra;dyzelinas; 6,1;	
Toyota Corolla;benzinas; 6,8;	
Toyota Yaris;benzinas; 5,9;	
Peugeot 207;dyzelinas; 5,5;	
Škoda Octavia;dyzelinas; 7,3;	

Rezultatai faile "Rezultatai.txt"		
Automobilių kiekis 8		
Automobilių sąrašas:		
Pavadinimas	Degalai	Sąnaudos (l/100 km)
Škoda Fabia	benzinas	7,10
Opel Meriva	benzinas	7,50
Volkswagen Golf	dyzelinas	6,30
Opel Astra	dyzelinas	6,10
Toyota Corolla	benzinas	6,80
Toyota Yaris	benzinas	5,90
Peugeot 207	dyzelinas	5,50
Škoda Octavia	dyzelinas	7,30
Vidutinės degalų sąnaudos: 6,56 litro/100 km		

Programos kūrimo eiga.

- Ankstesnio pratimo programa kopijuojama į atskirą katalogą.
- Parašomas metodas klasės Auto objektų masyvo lauko sąnaudos reikšmių vidurkiui skaičiuoti.
- Pagrindinis metodas Main() papildomas parašyto metodo grąžintos reikšmės spausdinimu.

Pirmas žingsnis.

- Nukopijuokite ankstesnio pratimo programą kartu su klase Auto ir duomenų failu Duomenys.txt.

Antras žingsnis.

- Parašykite vidurkio skaičiavimo metodą.

```
//-----  
/** Skaičiuoja ir grąžina objektų masyvo lauko sąnaudos vidutinę reikšmę  
@param A - objektų rinkinys automobilių duomenims saugoti
```

```

@param kiek - automobilių skaičius */
//-----
static double VidSąnaudos(Auto [] A, int kiek)
{
    double sum = 0;
    for (int i=0; i<kiek; i++)
    {
        sum = sum + A[i].ImtiSąnaudas();
    }
    return sum / kiek;
}
//-----

```

Trečias žingsnis.

- Papildykite pagrindinį metodą Main() parašyto metodo grąžintos reikšmės spausdinimu.

```

using (var fr = File.AppendText(CFrez))
{
    fr.WriteLine("Vidutinės degalų sąnaudos: {0,7:f2} litro/100 km" ,
        VidSąnaudos(A, na));
}

```

- Ivykdysite programą. Rezultatų faile "Rezultatai.txt" matysite:

Automobilių kiekis 8

Automobilių sąrašas:

Pavadinimas	Degalai	Sąnaudos (l/100 km)
Škoda Fabia	benzinas	7,10
Opel Meriva	benzinas	7,50
Volkswagen Golf	dyzelinas	6,30
Opel Astra	dyzelinas	6,10
Toyota Corolla	benzinas	6,80
Toyota Yaris	benzinas	5,90
Peugeot 207	dyzelinas	5,50
Škoda Octavia	dyzelinas	7,30

Vidutinės degalų sąnaudos: 6,56 litro/100 km

Programos papildymas.

Parašykite metodą dyzelinių automobilių kiekiui rasti. Papildykite programą šio kiekio radimu ir spausdinimu į rezultatų failą.