

1. Programavimo elementai

Susipažinsite su programavimo kalbos C# pagrindinėmis konstrukcijomis:

- pagrindiniu metodu `Main()`;
- baziniais duomenų tipais (`int`, `double`, `bool`, `char`);
- duomenų spausdinimu ekrane;
- duomenų įvedimu klaviatūra;
- sąlygos sakiniams;
- ciklo sakiniams;
- metodais, kaip programos struktūrizavimo priemone;
- simbolių eilutė `string`.

1.1. Ciklo sakiny

- Ciklo sakiny `for`.
- `Main()` metodas.
- Duomenų spausdinimas ekrane.
- `int` tipo kintamieji.

🕒Pirmas žingsnis.

- Sukurkite projektą Ciklas, kurį išsaugosite D diske. Į projektą įtraukite `.cs` failą, kurį pavadinkite Programa.
- Įveskite programos, išvedančios į ekraną skaičius nuo 1 iki 10 ir jų kvadratus, tekstą:

```
//-----  
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
//-----  
// Ciklas for  
//-----  
namespace ciklas  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            Console.WriteLine("Skaičiai nuo 1 iki 10 ir jų kvadratai:");  
            for (int i = 1; i < 11; i++)  
                Console.WriteLine(" {0,3:d}    {1,5:d}", i, i * i);  
        }  
    }  
}
```

```
//-----
```

- Suskaičiuokite laukiamus rezultatus.
- Kompiliuokite ir įvykdyskite programą.
- Patikrinkite, ar gauti rezultatai atitinka laukiamus rezultatus.

🕒Antras žingsnis.

- Analizuodami programos `Programa.cs` tekstą nustatykite, kuri programos vieta nusako skaičių intervalo kitimą. Šis sakiny vadinamas ciklo sakiniu.
- Pakeiskite programos tekstą taip, kad būtų išvedami skaičiai nuo 5 iki 15 ir jų kvadratai.
- Suskaičiuokite laukiamus rezultatus.
- Įvykdyskite pakeistą programą.

- Patikrinkite, ar gauti rezultatai atitinka laukiamus rezultatus.

☑ Trečias žingsnis.

- Pakeiskite programos tekstą taip, kad būtų išvedami skaičiai nuo 5 iki 15, jų kvadratai ir jų kubai.
- Suskaičiuokite laukiamus rezultatus.
- Įvykdysite pakeistą programą.
- Patikrinkite, ar gauti rezultatai atitinka laukiamus rezultatus.

☑ Ketvirtas žingsnis.

- Susipažindami su programavimo aplinka, panaudojote kintamuosius *a* ir *b* su pradinėmis reikšmėmis tokiame pavyzdyje:

```
//-----
// Dviejų kintamųjų reikšmių sumos radimas
//-----
namespace suma
{
    class Program
    {
        static void Main(string[] args)
        {
            int a=3;           // Pirmas kintamasis
            int b=5;           // Antras kintamasis
            int suma = a + b;   // Kintamasis sumai saugoti
            Console.Write(" {0,3:d} + {1,3:d} =", a, b);
            Console.WriteLine(" {0,3:d} \n", suma);
        }
    }
}
//-----
```

Pastaba. "{0} + {1} =" – nurodomas spausdinamas tekstas su specialiais pažymėjimais {0}, {1} įterpti kintamųjų reikšmėms. Po jo išvardinami kintamieji. Jie numeruojami nuo 0: {0} atitinka *a*, o {1} – *b*.

- Remdamiesi pavyzdžiu, programoje Programa.cs paskelbkite du kintamuosius, nurodančius ciklo kitimo režius. Panaudokite šiuos kintamuosius ciklo sakinyje.
- Suskaičiuokite laukiamus rezultatus.
- Įvykdysite pakeistą programą ir patikrinkite, ar gauti rezultatai atitinka laukiamus rezultatus.

☑ Penktas žingsnis.

- Į reikiamą programos vietą įterpkite eilutę, kuri visų rezultatų pabaigoje išvestų, kiek kartų buvo skaičiuota. Atspausdinkite ir paaiškinamąjį tekstą.
- Įvykdysite pakeistą programą ir patikrinkite rezultatus.

☑ Šeštasis žingsnis.

- Modifikuokite programą taip, kad ciklo režijų reikšmės būtų įvedamos dialogu. Remkitės pavyzdžiu:

```
//-----
// Dviejų kintamųjų reikšmių sumos radimas
//-----
namespace suma_dialogas
{
    class Program
    {
        static void Main(string[] args)
        {
            int a;           // Pirmas kintamasis
            int b;           // Antras kintamasis
            int suma;        // Kintamasis sumai saugoti
            Console.Write("Įveskite sveikąją a reikšmę: ");
            a = int.Parse(Console.ReadLine());
            Console.Write("Įveskite sveikąją b reikšmę: ");
            b = int.Parse(Console.ReadLine());
            suma = a + b;
            Console.WriteLine("{0,3:d} + {1,3:d} = {2,4:d} \n", a, b, suma);
        }
    }
}
```

```

    }
}
//-----

```

Pastaba. {0, 3:d} – nurodyto numerio `int` tipo elemento spausdinimas, naudojant formatavimą: 3 – lauko ilgis, d – sveikas skaičius. Sveikų skaičių formatavimo bendra išraiška `n:d`, čia n – lauko ilgis.

Įvedimo lange spausdinkite paaiškinamąjį tekstą, iš kurio būtų aišku, ką reikia įvesti.

- Įvykdysite pakeistą programą ir patikrinkite rezultatus.

🔗 Septintas žingsnis.

- Baikite darbą su sistema.
- Katalogą **Ciklas** iš disko **D** perkeltkite į diską **Z**.

1.2. Sąlygos sakiny

- Sąlygos sakiny `if`.

🔗 Pirmas žingsnis.

- Sukurkite projektą `SalygosSakinys`.
- Įveskite programos, kuri spausdina žvaigždutes, tekstą:

```

//-----
// Sąlygos sakiny
// Simbolių paskirstymas į eilutes
//-----
namespace salygos_sakinys
{
    class Program
    {
        static void Main(string[] args)
        {
            for (int i = 1; i < 51; i++)
            {
                Console.Write('*');
                if (i % 5 == 0)
                    Console.WriteLine("");
            }
            Console.WriteLine("");
        }
    }
}
//-----

```

- Atkreipkite dėmesį, kad patikrinimo sąlyga „ar lygu“ užrašoma dviem simboliais `==`. Jei reiktų kai kurių kitų sąlygų, tai jos užrašomos: „ar nelygu“ – dviem simboliais `!=`, „ne“ – simboliu `!`, „ir“ – dviem simboliais `&&`, „arba“ – dviem simboliais `!!`.
- Taip pat atkreipkite dėmesį į rištinius skliaustus, kurie nurodo ciklo viduje esančius sakinius. Jeigu tokį sakinių bloką sudaro tik vienas sakiny, tuomet rištiniai skliaustai nebūtini.
- Apskaičiuokite laukiamus rezultatus, įvykdysite programą ir patikrinkite, ar gauti rezultatai sutampa su apskaičiuotais.

🔗 Antras žingsnis.

- Pakeiskite programą taip, kad būtų galima nurodyti spausdinamą simbolį. Taip pat pakeiskite sąlygos sakinį. Tai leis atsisakyti rištinių skliaustų:

```

//-----
// Sąlygos sakiny
// Simbolių paskirstymas į eilutes
//-----
namespace salygos_sakinys
{
    class Program
    {

```

```

static void Main(string[] args)
{
    char simbolis;
    Console.Write("Įveskite spausdinamą simbolį: ");
    simbolis = (char)Console.Read();
    Console.ReadLine(); // Būtinas tik tada, jei dar būtų skaitymų!!!
    for (int i = 1; i < 51; i++)
        if (i % 5 == 0)
            Console.WriteLine(simbolis);
        else
            Console.Write(simbolis);
    Console.WriteLine("");
}
}
//-----

```

Pastaba. Negalima už ciklo spausdinti kintamojo, jei jis prieš tai buvo priskirtas tik ciklo.

- Kompiliuokite ir įvykdysite programą. Įveskite jums patinkantį simbolį.
- Patikrinkite, ar gauti rezultatai atitinka laukiamus rezultatus.

🔗 Trečias žingsnis.

- Papildykite programą, kad būtų galima nurodyti bendrą spausdinamų simbolių kiekį ir vienos eilutės simbolių kiekį.
- Kompiliuokite ir įvykdysite programą.
- Patikrinkite, ar gauti rezultatai atitinka laukiamus rezultatus.

Savarankiško darbo užduotis.

Kai kuriais atvejais tą patį simbolių spausdinimo rezultatą galima pasiekti nesinaudojant sąlygos sakiniu, bet panaudojant vienas į kitą įdėtus 2 ciklo sakinius (ciklą ciklo). Parenkite tokią programą. Ciklą kintamieji turi būti skirtingi.

1.3. Matematinis reiškiny

- Sąlygos sakiny.
- Matematinės funkcijos.

Užduotis.

Suskaičiuoti funkcijos reikšmę: $f(x) = \sqrt{z-1}$

- D diske sukurkite projektą.
- Parašykite programą:

```

//-----
// Funkcijos reikšmės skaičiavimas
//-----
namespace funkcijos_reiksmes
{
    class Program
    {
        static void Main(string[] args)
        {
            double fx;
            int z;
            Console.Write("Įveskite z reikšmę: ");
            z = int.Parse(Console.ReadLine());
            if (z - 1 >= 0) {
                fx = Math.Pow(z - 1, 0.5); // Matematinė funkcijų klasė, Pow - kėlimas laipsniu
                Console.WriteLine(" z = {0} f(x) = {1,8:f3}", z, fx);
            }
            else
                Console.WriteLine(" z = {0} f-ja neegzistuoja", z);
        }
    }
}

```

}

//-----

Pastaba 1. {1, 8:f3} – nurodyto numerio elemento spausdinimas, naudojant formatavimą: 8 – visas lauko ilgis, f – realus skaičius, 3 – trupmeninės dalies ilgis. Realių skaičių formatavimo bendra išraiška **n:fm**, čia n – visas lauko ilgis, m – trupmeninės dalies ilgis. Praleidus m, spausdinami 2 skaitmenys po kablelio.

Pastaba 2. Atkreipkite dėmesį, kad sąlygos sakinyje panaudojome rištinius skliaustus. Jie būtini, nes sąlygos sakinio šakoje yra daugiau kaip vienas programos sakiny.

- Įvykdysite programą ir patikrinkite rezultatus.

Užduotis savarankiškam darbui.

Suskaičiuoti funkcijos reikšmę: $f(x, y) = \frac{y^2 - 2yx + x^2}{x^3 - y}$.

1.4. Sudėtingesnis matematinis reiškiny

- Sąlygos sakiny.
- Matematinės funkcijos.

Užduotis.

Suskaičiuoti funkcijos reikšmę: $f(x) = \begin{cases} ae^{-x}, & x \leq 0, \\ 5ax - 7, & 0 < x < 1, \\ \sqrt{x+1}. \end{cases}$

//-----

// Sudėtingas if sakiny

//-----

namespace sudetingas_if

{

class Program

{

static void Main(string[] args)

{

double fx;

int a;

double x;

Console.WriteLine(" Įveskite a reikšmę (tik sveikas skaičius): ");

a = int.Parse(Console.ReadLine());

Console.WriteLine(" Įveskite x reikšmę: ");

x = double.Parse(Console.ReadLine());

Console.Clear();

// Išvalo langą

Console.SetCursorPosition(5, 6);

// Nustato pradinę žymeklio padėtį

if (x <= 0)

fx = a * Math.Exp(-x);

else

if (x < 1)

fx = 5 * a * x - 7;

else

fx = Math.Pow(x + 1, 0.5);

Console.WriteLine(" Reikšmė a = {0,3:d}, reikšmė x = {1,6:f2}, fx = {2,8:f3}",
a, x, fx);

}

}

}

//-----

- Įvykdysite programą ir patikrinkite rezultatus.
- Atkreipkite dėmesį į `Console.Clear()` ir `Console.SetCursorPosition(x, y)` metodus.

Savarankiško darbo užduotis.

Suskaičiuoti funkcijos reikšmę: $f(x) = \begin{cases} f_1(x), & \text{kai 1 sąlyga} \\ f_2(x), & \text{kai 2 sąlyga} \\ f_3(x), & \text{kitais atvejais} \end{cases}$

Funkcijos variantą pasirinkite pagal kompiuterio numerį. Kompiuterio numerį dalinkite iš 12, ir gauta liekana rodo varianto numerį.

Nr.	$f_1(x)$	$f_2(x)$	$f_3(x)$	1 sąlyga	2 sąlyga
0.	e^{-x}	$2x^2+1$	$1/x^2$	$-2 \leq x \leq 0$	$0 < x < 2$
1.	e^{-x}	$5x-7$	$\sqrt{x+1}$	$-3 \leq x \leq 0$	$0 < x \leq 1$
2.	$\cos x$	$1/(x+5)^3$	$\sqrt{x^2+1}$	$-4 \leq x \leq 0$	$0 < x \leq 4$
3.	$x+2x^2$	x^2+4	$\sqrt{2x^2+3}$	$-5 \leq x \leq 0$	$0 < x < 3$
4.	$\cos x^2$	$4x^2+3$	$\sqrt{x^2+x+4}$	$-1 \leq x \leq 0$	$0 < x < 1$
5.	$1/(x+5)$	$x+1$	$\sqrt{x^2+x+1}$	$-1 \leq x < 0$	$0 \leq x < 1$
6.	$\sin^2 x$	$(x-1)^2$	x^2+x+1	$-1 \leq x < 0$	$0 \leq x < 1$
7.	$3.2x^2$	$\sin^2(x+1)$	$3x^2-1$	$-2 \leq x < 0$	$0 \leq x \leq 1$
8.	$1/(x-5)$	$(x+3)^2$	$\sqrt{x+5}$	$-3 \leq x \leq 0$	$0 < x \leq 5$
9.	$1/x$	$\cos x$	$2x+4$	$-4 \leq x < -2$	$-2 \leq x \leq 2$
10.	$1/x^2$	$\sin(x+1)$	$\sqrt{2x}$	$-5 \leq x < 0$	$0 \leq x \leq 2$
11.	$1/(x-3)$	x^2+6	$2x+3$	$-6 \leq x \leq -1$	$-1 < x \leq 3$

Pastaba. \cos , \sin funkcijų argumentas yra išreikštas radianais.

Savarankiško darbo užduotis.

Sukurkite mini skaičiuotuvą darbui su realiais skaičiais. Skaičiuotuvą atlieka 4 operacijas: sudėtis (+), atimtis (-), daugyba (*), dalyba (/). Operacijas žymėkite atitinkamais simboliais. Įveskite porą skaičių. Įveskite operacijos simbolį. Galima įvesti bet kokią simbolį. Jei įvestas simbolis neatitinka nei vieno operacijos ženklo, turi būti rodomas pranešimas ERROR (arba KLAIDA). Jei simbolis atitinka kurią nors operaciją, suskaičiuokite bei išveskite visus įvestus duomenis ir operacijos rezultatą. Sąlygos sakinyje operacijos ženklą nurodykite tarp apostrofų (viengubų kabučių).

1.5. Vartotojo kuriami metodai

- Metodas, negrąžinantis reikšmės.
- Metodas, grąžinantis reikšmę.

Pirmas žingsnis. Metodas, negrąžinantis reikšmės

- 1.2 pratimo antrojo žingsnio programą pertvarkykite taip, kad skaičiavimai būtų atliekami metode:

```
//-----
// Metodas, negrąžinantis reikšmės
//-----
namespace metodai
{
    class Program
    {
        static void Main(string[] args)
        {
            char simbolis;
            Console.WriteLine("Įveskite spausdinamą simbolį: ");
            simbolis = (char)Console.Read();
            Console.Clear();
            Spausdinti(simbolis);
        }
    }
}
//-----
// Ekrane spausdinamas 51 simbolis po 5 eilutėje
```

```
// ch - parametro reikšmė yra spausdinamas simbolis
static void Spausdinti(char ch)
{
    for (int i = 1; i < 51; i++)
        if (i % 5 == 0)
            Console.WriteLine(ch);
        else
            Console.Write(ch);
    Console.WriteLine("");
}
}
//-----
```

- Metodo parametrai ir jų reikšmės susiejami pagal vietą parametų sąrašė, bet ne pagal vardą: metodo antraštėje parametras pavadintas `ch`, o kreipinyje – simbolis.
- Įvykdykite programą ir patikrinkite rezultatus.

📌 Antras žingsnis. Metodas, grąžinantis reikšmę

- 1.1 pratimo ketvirtą žingsnio programą pertvarkykite taip, kad skaičiavimus atliktumėte metode:

```
//-----
// Metodas, grąžinantis reikšmę
//-----
namespace metodas
{
    class Program
    {
        static void Main(string[] args)
        {
            int a;           // Pirmas kintamasis
            int b;           // Antras kintamasis
            Console.Write("Įveskite sveikąją a reikšmę: ");
            a = int.Parse(Console.ReadLine());
            Console.Write("Įveskite sveikąją b reikšmę: ");
            b = int.Parse(Console.ReadLine());
            Console.WriteLine("{0,3:d} + {1,3:d} = {2,4:d}", a, b, Suma(a,b));
        }
        //-----
        // metodas grąžina dviejų sveikų skaičių sk1 ir sk2 sumą
        //-----
        static int Suma(int sk1, int sk2)
        {
            int suma = sk1 + sk2; ;
            return suma;
        }
    }
}
//-----
```

Metodo antraštę sudaro:

- Raktinis žodis `static`;
- Grąžinamos reikšmės tipas;
- Metodo vardas ir parametrai.

Jei metodas nieko negrąžina, grąžinamos reikšmės tipas – `void`. Jei metodas grąžina reikšmę, metodo kamienas turi būti bent vienas `return` sakinytis. Metodo realizacija užrašoma po pagrindine programa (pagrindinis metodas `Main()`). Pagrindinėje programoje (pagrindiniame metode) turi būti kreipinys į vartotojo sukurtą metodą.

- Metodas gali būti ir toks:

```
//-----
static int Suma(int sk1, int sk2)
{
    int suma = sk1 + sk2; ;
    return suma;
}
```

```

    }
//-----

```

☛ Trečias žingsnis. Metodo apibrėžtumas ir universalumas

- 1.3 programą pertvarkykite taip, kad skaičiavimai būtų atliekami metode:

```

//-----
// Metodas suteikia universalumo
//-----
namespace universalus_metodas
{
    class Program
    {
        static void Main(string[] args)
        {
            double fx;
            int z;
            Console.Write("Įveskite z reikšmę: ");
            z = int.Parse(Console.ReadLine());
            if (z - 1 >= 0)
                Console.WriteLine(" z = {0,3:d} f(x) = {1,8:f3}", z, Reiksme(z, 1, 0.5));
            else
                Console.WriteLine(" z = {0} f-ja neegzistuoja", z);
        }
        //-----
        // Gražina parametų sk1 ir sk2 skirtumą, pakeltą laipsniu laipsnis
        //-----
        static double Reiksme(int sk1, int sk2, double laipsnis)
        {
            return Math.Pow(sk1 - sk2, laipsnis);
        }
    }
}
//-----

```

Metodas visada privalo atlikti vieną apibrėžtą veiksmą. Į vieną metodą negalima jungti reikšmės skaičiavimo ir reikšmės spausdinimo. Jei visos naudojamos konstantos perkeliamos į parametų sąrašą, metodas tampa universalesnis.

Savarankiško darbo užduotis.

Pertvarkykite visas anksčiau sukurtas programas taip, kad skaičiavimai būtų atliekami metoduose.

1.6. Simboliai ir simbolių eilutės

- Simbolinis duomenų tipas `char`.
- Simbolių eilutė `string`.

☛ Pirmas žingsnis. Simbolinis duomenų tipas `char`.

- Parašykite programą, kuri atspausdintų mažąsias lotyniškos abėcėlės raides.

```

//-----
// Mažosios raidės
//-----
namespace simboliai
{
    class Program
    {
        static void Main(string[] args)
        {
            for (char ch = 'a'; ch <= 'z'; ch++)
                Console.Write("{0} ", ch);
            Console.WriteLine("");
        }
    }
}
//-----

```

- Įvykdysite programą ir patikrinkite rezultatus.

U Antras žingsnis. Raidės ir jų kodai.

- Papildykite ankstesnę programą taip, kad būtų spausdinamos raidės ir jų kodai.

```
//-----
// Mažosios raidės ir jų kodai
//-----
namespace mazosios_raides
{
    class Program
    {
        static void Main(string[] args)
        {
            for (char ch = 'a'; ch <= 'z'; ch++)
                Console.WriteLine("{0} - {1}", ch, (int)ch);
        }
    }
}
//-----
```

- Ivykdykite programą ir patikrinkite rezultatus.

U Trečias žingsnis. Nurodyto intervalo raidės ir jų kodai

- Papildykite programą taip, kad būtų spausdinamos nurodyto simbolių intervalo [pr, pb] raidės ir jų kodai.

```
//-----
// Nurodyto intervalo raidės ir jų kodai
//-----
namespace raidziu_intervalas
{
    class Program
    {
        static void Main(string[] args)
        {
            char pr, pb;
            Console.Write("Įveskite raidę raidžių intervalo pradžiai: ");
            pr = Console.ReadLine()[0];
            Console.Write("Įveskite raidę raidžių intervalo pabaigai: ");
            pb = (char)Console.Read();
            for (char ch = pr; ch <= pb; ch++)
                Console.WriteLine("{0} - {1}", ch, (int)ch);
        }
    }
}
//-----
```

- Ivykdykite programą ir patikrinkite rezultatus.

U Ketvirtas žingsnis. Simbolių eilutė string.

- Parašykite programą, kuri pasisveikina su prisistačiusiu vartotoju.

```
//-----
// Duomenų tipas string
//-----
namespace simboliai_string
{
    class Program
    {
        static void Main(string[] args)
        {
            string vardas;
            Console.Write("Koks jūsų vardas: ");
            vardas = Console.ReadLine();
            Console.WriteLine(" Sveika(s), {0}! ", vardas);
        }
    }
}
//-----
```

- Įvykdysite programą ir patikrinkite rezultatus.

Penktas žingsnis. Savaitės dienos.

- Parašysite programą, kuri atpažįsta ir apibūdina savaitės dienas.

```
//-----
// Savaitės dienos
//-----
namespace simboliai_dienos
{
    class Program
    {
        static void Main(string[] args)
        {
            string diena;
            Console.WriteLine("Kokia šiandien savaitės diena (5verkite mažosiomis raidėmis)? ");
            diena = Console.ReadLine();
            if (diena == "pirmadienis")
                Console.WriteLine("Pirmadienis - sudėtingiausia savaitės diena.");
            else
            {
                if (diena == "antradienis")
                    Console.WriteLine("Antradienis - aktyvių veiksmų, Marso diena.");
                else
                {
                    if (diena == "trečiadienis")
                        Console.WriteLine("Trečiadienis - sandoriams sudaryti " +
                            "tinkamiausia diena.");
                    else
                    {
                        if (diena == "ketvirtadienis")
                            Console.WriteLine("Ketvirtadienį reikėtų imtis " +
                                "visuomeninių darbų.");
                        else
                        {
                            if (diena == "penktadienis")
                                Console.WriteLine("Penktadienį lengvai gimsta šedevrai," +
                                    "susitinka mylimieji.");
                            else
                            {
                                if (diena == "šeštadienis")
                                    Console.WriteLine("Šeštadienis - savo problemų " +
                                        "sprendimo diena.");
                                else
                                {
                                    if (diena == "sekmadienis")
                                        Console.WriteLine("Sekmadienį reikėtų pradėti " +
                                            "naujus darbus.");
                                    else
                                        Console.WriteLine("Tokios savaitės dienos " +
                                            "pas mus nebūna.");
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
//-----
```

- Įvykdysite programą ir patikrinkite rezultatus.

Šeštasis žingsnis. Savaitės dienų atpažinimas.

- Savaitės dienų atpažinimo programą papildysime, kad savaitės dienos būtų atpažįstamos net ir tuo atveju, jei dienos pavadinimas buvo įvestas didžiosiomis ir mažosiomis raidėmis (pvz., Pirmadienis). Tam panaudosime string klasės metodą ToLower().
- Kai sąlygos operatorius turi daug šakų, jį skaityti žymiai sunkiau. Tokiais atvejais, jei tik leidžia lyginimo sąlygos ir reikšmių tipas, sąlygos operatorius if keičiamas varianto operatoriumi switch. C# kalba yra viena kalbų, kur galima varianto operatoriuje lyginti eilutes, todėl panaudosime operatorių switch.

```
//-----
// Savaitės dienos su switch operatoriumi
//-----
namespace simboliai_su_switch
{
    class Program
```

```

{
    static void Main(string[] args)
    {
        string diena;
        Console.WriteLine("Kokia šiandien savaitės diena? ");
        diena = Console.ReadLine().ToLower();
        switch (diena)
        {
            case "pirmadienis":
                Console.WriteLine("Pirmadienis - sudėtingiausia savaitės diena.");
                break;
            case "antradienis":
                Console.WriteLine("Antradienis - aktyvių veiksmų, Marso diena.");
                break;
            case "trečiadienis":
                Console.WriteLine("Trečiadienis - sandoriams sudaryti " +
                                "tinkamiausia diena.");
                break;
            case "ketvirtadienis":
                Console.WriteLine("Ketvirtadienį reikėtų imtis visuomeninių darbų.");
                break;
            case "penktadienis":
                Console.WriteLine("Penktadienį lengvai gimsta šedevrai, " +
                                "susitinka mylimieji.");
                break;
            case "šeštadienis":
                Console.WriteLine("Šeštadienis - savo problemų sprendimo diena.");
                break;
            case "sekmadienis":
                Console.WriteLine("Sekmadienį reikėtų pradėti naujus darbus, " +
                                "kurti planus.");
                break;
            default:
                Console.WriteLine("Tokios savaitės dienos pas mus nebūna.");
                break;
        }
    }
}

```

//-----

Raidės simbolių lentelėje yra surašytos abėcėliška. Pirmiausiai – visos didžiosios, o toliau visos mažosios. Galima nenaudoti `ToUpper()` metodo. Mažąsias raides pakeisti didžiosiomis galima taip:

//-----

```

// Visas žodžio raides verčia didžiosiomis
//-----
static void ToUpper1(string str1, out string str)
{
    int y;
    str = str1;
    int n = str.Length;
    for (int pos = 0; pos < n; pos++)
        if (str[pos] >= 'a' && str[pos] <= 'z')
        {
            y = (int)str[pos] + (int)'A' - (int)'a';
            char z = (char)y;
            string z1 = z.ToString();
            str=str.Remove(pos, 1);
            str=str.Insert(pos,z1) ;
        }
}
//-----

```

Kontroliniai klausimai

1. Kokia bus kintamojo `i` reikšmė, atlikus šiuos veiksmus:

```
for (int i = 5; i < 3; i++)  
    i = i - 2;
```

- a. 5
- b. 3
- c. 0
- d. Neapibrėžta

2. Kokia bus kintamojo `i` reikšmė, atlikus šiuos veiksmus:

```
for (int i = 5; i < 8; i = i + 3)  
    i = i - 2;
```

- a. 7
- b. 8
- c. 3
- d. Neapibrėžta

3. Kokia bus kintamojo `i` reikšmė, atlikus šiuos veiksmus:

```
for (int i = 4; i < 10; i = i + 2)  
    if (i % 2 == 1)  
        i = i + 3;  
    else  
        i = i - 1;
```

- a. 8
- b. 10
- c. 5
- d. Neapibrėžta

4. Kokia bus kintamojo `k` reikšmė, atlikus šiuos veiksmus:

```
int i = 5678;  
int k = ((i / 1000) * 10 + i % 10) * 10 + (i / 10) + (i % 10);
```

- a. 575
- b. 1155
- c. 580
- d. 5678

5. Kokia bus kintamojo `k` reikšmė, atlikus šiuos veiksmus:

```
static int Skaičius(int a, int b)  
{  
    return a + b;  
}  
int i = 57;  
int k = Skaičius(i/3, i % 3) - Skaičius(i / 2, 12);
```

- a. 19
- b. 21
- c. -21
- d. 40

6. Kokia bus kintamojo `k` reikšmė, atlikus šiuos veiksmus:

```
static int Skaičius(int a, int b)
{
    if (a > b) return a;
    return b;
}
int i = 57;
int k = Skaičius(i/3, i % 3) - Skaičius(i / 4, 10);
```

- a. 29
- b. 33
- c. 5
- d. Metodas klaidingas

7. Kokia bus kintamojo `k` reikšmė, atlikus šiuos veiksmus:

```
static int Skaičius(int a, int b)
{
    if (a > b) return a - b;
    return b - a;
}
. . .
int k = Skaičius(Skaičius(13, 8), Skaičius(57 / 4, 20));
```

- a. 1
- b. 20
- c. `k` skaičiavimo reiškinyje klaidingas
- d. Metodas klaidingas

8. Kas bus matoma ekrane, atlikus šiuos veiksmus:

```
static int Eil(string a, string b)
{
    if (a > b) return 1;
    if (a == b) return 0;
    return 2;
}
. . .
Console.WriteLine("{0}{1}", Eil("Namas", "Nama"), Eil(" Du", "Du"));
```

- a. 12
- b. 10
- c. 20
- d. 21
- e. Metodas klaidingas

9. Kaip nustatyti, ar du `double` tipo skaičiai lygūs?

10. Kokie veiksmai galimi su simboliais ir simbolių eilutėmis?