

3

Pakartokime – aritmetiniai veiksmai

- Sudėties: +
- Atimties: -
- Daugybos: *
- Dalybos: /
- Dalybos liekana: %

```
pavad += 5;
```

```
pavad -= 5;
```

```
pavad *= 5;
```

```
pavad /= 5;
```

```
pavad %= 5;
```

```
pavad++; ++pavad;
```

```
pavad--; --pavad;
```

Pakartokime – if() ... else if() ... else.

```
if (salyga)
{
    //vykdoma jeigu salyga == true
} else if(salyga2)
{
    //vykdoma jeigu salyga == false ir salyga2 == true
} else
{
    //vykdoma jeigu salyga == false ir salyga 2 == false
}
```

Pakartokime – switch (case, break, default)

```
int skaicius = 5;

switch (skaicius) {
    case 1:
        // skaicius == 1
        break;
    case 2:
        // skaicius == 2
        break;
    case 3:
    case 4:
        // skaicius == 3 arba == 4
        break;
    default:
        // skaicus nelygus 1,2,3,4,5 arba 6
        break;
}
```

Pakartokime - loginiai operatoriai

Operatorius	Paaiškinimas
>	daugiau
<	mažiau
>=	daugiau arba lygu
<=	mažiau arba lygu
==	lygu
!=	nelygu

Operatorius	Paaiškinimas	Pavyzdys
&&	IR	$a > b \ \&\& a > c$
	ARBA	$a > b \ a > c$
!	NE	!a

Ciklai

loops

Apie ciklus

Ciklai naudojami kai norima atlikti tą patį veiksmą daug kartų.

Yra keli skirtini ciklai.

- for
- while
- do-while

(+ foreach ciklas kurį išmoksime vėliau)

Ciklas for

for loop

For sintakse

Šis ciklas dažniausiai naudojamas kai yra žinoma kiek tiksliai kartų reikia kartoti tą patį veiksmą.
(... arba, žinoma pradžia ir pabaiga)

```
vienkartine           daugkartine  
for (pradineOperacija; kartojimoSalyga; pabaigosOperacija)  
{  
    // ciklo kodas  
}
```

For pavyzdys 1

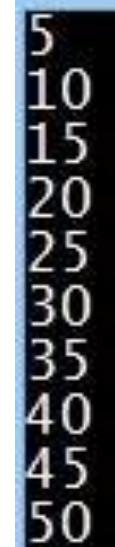
```
for (int i = 0; i < 10; i++)  
{  
    Console.WriteLine(i);  
}
```



0
1
2
3
4
5
6
7
8
9

For pavyzdys 2

```
for (int i = 5; i <= 50; i+=5)
{
    Console.WriteLine(i);
}
```



5
10
15
20
25
30
35
40
45
50

Užduotys

1 : Parašyti for ciklą, kuris išvestų kas trečią skaičių, pradedant nuo 0 iki 20.
(Išvedimo pavyzdys: 0, 3, 6, 9, 12, 15, 18)

2 : Išvesti į ekraną visų skaičių, kurie yra tarp vartotojo įvestu rėžiu kvadratus!

- Vartotojas turi įvesti rėžius, t.y. jų pradžią ir pabaigą. (2 int skaicius)
- If sąlyga tikrina ar pradžios skaičius yra didesnis už pabaigos skaičių, jei taip išveda į ekraną "bad limits". Jeigu viskas gerai (else dalyje) yra for ciklas, kuris pereina visus skaicius nuo žmogaus duotos pradžios, iki pabaigos.
- Į ekraną išvesti ciklo skaičių, tarpą, jo kvadratą ir tada naują eilutę.

3 : Rasti visų teigiamų skaičių, mažesnių už 1000 ir kurie dalijasi iš 3 arba 5, sumą.

```
for (int i = 0; i < 10; i++)  
{  
    ...  
}
```

A WIFE CALLS HER PROGRAMMER
HUSBAND AND TELLS HIM 'WHILE
YOU'RE OUT, BUY SOME MILK.'



HE NEVER RETURNS HOME

Ciklas while

while loop

Ciklas while

Šis ciklas dažniausiai naudojamas tuomet kai nėra žinoma kiek kartų reikės vykdyti ciklą:

```
while ( sąlyga )
{
    // vykdomas kodas
}
```

While pavyzdys

```
int skaicius = 0;

while (skaicius < 10)
{
    Console.WriteLine(skaicius);
    skaicius++;
}
```



0
1
2
3
4
5
6
7
8
9

Užduotis

- Išveskite menu su 3 pasirinkimais:
 - 1 Kava
 - 2 Arbata
 - 3 Vanduo
- Vartotojas ivedinėja skaičius tol kol įveda teisingą.
- Jei įveda neteisingai – atspausdinti klaidą. (ir kartoti įvedimą)
- Jei teisingai – išvesti pasirinkimą, baigti programą.

```
while (sąlyga)
{
    // vykdomas kodas
}
```

Ciklas do-while

do- while loop

Ciklas do-while

Šis ciklas dažniausiai naudojamas tuomet kai nėra žinoma kiek kartų reikės vykdyti ciklą... (tas pats kas while, tiesiog patikrinimas sąlygos gale.):

```
do {  
    // vykdomas kodas  
} while (sąlyga);
```

Užduotis

Turime lėktuvą su 8 sėdimomis vietomis, 4 prie lango.

Parašykite programą kuri klausia keleivių ar jie nori sėdėti prie lango, ar ne.

Keleiviai atsako "taip", arba "ne".

Sustoti klausinėti kai 4 keleiviai sutinka sėdēti prie lango, arba kai visi 8 keleiviai duoda savo atsakymus(bet kokius).

- Patikrinkite ar keleivių įvedimas teisingas, jei ne – kartokite įvedimą.
- Pamatykit ką galima išsaugoti kaip konstantas. (4 konstantas galima sukurti.)
- Praplėskite įvedimo variantus: "t", "n", "1", "0", "Taip", "Ne"..

```
do {  
    // vykdomas kodas  
} while (salyga);
```

continue, break...

```
for (int i = 0; i < 10; i++)
{
    if(arPraleisti) {
        continue;
        // peršokti į gala, vykdyti sekanti cikla.
    }
    // vykdyti jei nereikia praleisti..
    if(arNutraukti) {
        break;
        // sustabdyti cikla.
    }
}
```

Užduotis

- 1) Parašyti for ciklą kuris sumuoja visus skaičius nuo 1 iki 1000;
 - 2) Papildyti užduotį: jei skaičius dalijasi iš 5 – praleisti. (nesumuoti)
 - 3) Papildyti užduotį: jei skaičius didesnis už 123 – nutraukti ciklą.
-
- 4) Pakeisti 1 užduoties ciklą į:

```
for(;;){  
    ...  
}
```

```
for (int i = 0; i < 10; i++)  
{  
    continue;  
    ...  
    break;  
    ...  
}
```

...sukurkite trūkstamus kintamuosius ir trūkstamą kodą - kad rezultatas būtų tas pats.

Papildoma užduotis - for

- Parašyti for ciklą, kuris leistu vartotojui įvesti kiek Fibonačio skaičių atspausdinti į ekraną.
- Fibonačio skaičių seka – sekantis skaičius yra dviejų paskutinių veiksmų rezultatų suma.
 - 0
 - 1
 - $0 + 1 = 1$
 - $1 + 1 = 2$
 - $1 + 2 = 3$
 - $2 + 3 = 5$
 - $3 + 5 = 8$
 - $5 + 8 = 13$
 - ...

Papildoma užduotis - while

- Kompiuteris sugalvoja atsitiktinį skaičių nuo 0 iki 100.
- int skaicius = new Random().Next(0, 100);
- vartotojas spėja skaičius, tol kol atspéja (while loop)
- - jei neatspēja – kompiuteris pasako ar spėjimas mažesnis ar didesnis.
- - jei atspéja – žaidėjas laimi, kompiuteris pasako kiek kartų žaidėjui reikėjo spėti.
- Sukurkite bool tipo kintamąjį, saugoti ar reikia ciklą testi ar ne.

```
while (salyga)
{
    // vykdomas kodas
}
```

Papildoma užduotis - do..while

- 1) Sukurti skaičiuotuvą, kuris leistų vartotojui įvesti veiksmą (+, -, *, /, %), tada du skaičius.
- 2) Pagal atitinkamą veiksmą vykdykite atitinkamus skaičiavimus ir juos atvaizduokite ekrane.
- 3) Kartoti operaciją tol kol vartotojas, nutaria baigti programą. Pvz – įveda ‘n’, arba ‘0’.
- 4) Jei neteisingas veiksmas, išveskite klaidą.
- 5) Pabandykite veiksmą įvesti su.
`char keyPress = Console.ReadKey(true).KeyChar;`
- 6) patikrinkite dalyba iš nulio... (išveskite klaidą.)
- 7) vykdykite tol, kol vartotojas paspausi ESC. (vietoj veiksmo). Escape char: (`char`)²⁷

```
do {  
    // vykdomas koda  
} while (sąlyga);
```

Papildoma medžiaga

<http://www.tutorialsteacher.com/csharp/csharp-for-loop>

<http://www.tutorialsteacher.com/csharp/csharp-while-loop>

<http://www.tutorialsteacher.com/csharp/csharp-do-while-loop>

https://www.tutorialspoint.com/csharp/csharp_loops.htm

<https://www.codingame.com/training/easy/horse-racing-duals>