

## 8 tema. Valdymo sakinyis IF

Temos uždaviniai. Išmokti taisyklingai sudaryti paprastą IF ir sudėtinius IF-ELSE sakinius.

Teoriniai pastebėjimai

Programa yra komandų seka. Visos anksčiau parašytos C++ programos vykdo komandas nuosekliai. Bet gali tekti vykdyti atskirą komandų grupę, esant patenkintai tam tikrai sąlygai, ir priešingai - kai sąlyga nepatenkinta, reikia vykdyti kitą komandų grupę. Kitaip tariant reikia, kad programa darytų sprendimus ir atitinkamai į juos reaguotų.

- Programos naudoja operatorių *if – else* vienos operatorių grupės vykdymui, jei sąlyga patenkinta, ir kitos, jeigu sąlyga nepatenkinta.
- Kombinuojant kelis operatorius *if – else*, galima tikrinti kelias sąlygas.
- Naudojant loginius C++ operatorius IR ir ARBA, galima tikrinti kelias sudėtines sąlygas.

### 8.1 Operatorius IF (vienas veiksmas)

Struktūra

```
if (salyga)
veiksmas;
```

**Pavyzdys.** Kintamajam a priskiriame programos viduje reikšmę 150, kadangi priskirta reikšmė lygi 150, tai ekrane išvedamas įrašas „a lygu 150“.

```
#include<iostream>
using namespace std;
int main(void)
{
    int a=150;
    if(a == 150)
        cout << "a lygu 150";
    return 0;
}
```

```
a lygu 150
Process returned 0 (0x0)   execution time : 0.005 s
Press any key to continue.
```

## 8.2 Operatorius IF (keli veiksmai)

Jeigu reikia atlikti daugiau nei vieną veiksmą, naudojami skliaustai {}

### Struktūra

```
if (salyga)
{
veiksmas1;
.....
veiksmas n;
}
```

**Pavyzdys.** Kintamajam a priskiriame programos viduje reikšmę lygią 150, kadangi priskirta reikšmė lygi 150, tai ekrane išvedamas įrašas „a lygu 150“.

```
#include<iostream>
using namespace std;
int main(void)
{
    int a=150;
    if (a == 150)
    {
        cout << "a lygu ";
        cout << a;
    }
    return 0;
}
```

```
a lygu 150
Process returned 0 (0x0)   execution time : 0.005 s
Press any key to continue.
```

## 8.3 Operatorius IF-ELSE

Junginio IF-ELSE pagalba yra aprašomi du veiksmų blokai – vienas iškart po operatoriaus IF, kitas iškart po operatoriaus ELSE.

### IF-ELSE (vienas veiksmas)

#### Struktūra

```
if (salyga)
veiksmas;

else
veiksmas;
```

**Pavyzdys .** Programa patikrina ar kintamojo x reikšmė lygi 110, jei taip ekrane išveda įrašą „x lygi 110“, priešingu atveju – įrašą „x nelygu 110“.

```
#include<iostream>
using namespace std;
int main(void)
{
    int a=150;
    if (a == 110)
        cout << "x lygu 110";
    else
        cout << "x nelygu 110";
    return 0;
}
```

```
x nelygu 110
Process returned 0 (0x0)   execution time : 0.015 s
Press any key to continue.
```

## 8.4 Sąlygos operatorius (?)

Sąlygą galima užrašyti ir sutrumpintu variantu:

(salyga) ? veiksmas (Taip (true) dalis) : veiksmas (Ne (false) dalis)

**Pavyzdys.** Programa palygina du skaičius ir išveda į ekraną didesnį.

```
#include<iostream>
using namespace std;
int main(void)
{
    int a,b,c;
    a=-11;
    b=10;
    c = (a>b) ? a : b;
    cout << c;
    return 0;
}
```

```
10
Process returned 0 (0x0)   execution time : 0.014 s
Press any key to continue.
```

### Pasipraktikuokite

```
7==5 ? 4 : 3
7==5+2 ? 4 : 3
5>3 ? a : b
a>b ? a : b
```

## 8.5 IF-ELSE (keli veiksmai)

### Struktūra

```
if (sąlyga) {  
    veiksmai;  
}  
else {  
    veiksmai;  
}
```

**Pavyzdys.** Programa randa didžiausią skaičių tarp dviejų įvestų klaviatūra.

```
#include<iostream>  
using namespace std;  
  
int main(void)  
{  
    int n,m;  
    cout << "Iveskite du skaicius" << endl;  
    cin >> n >> m;  
    if (n>m)  
    {  
        cout << "Didesnis skaicius:";  
        cout << n;  
    }  
    else  
    {  
        cout << "Didesnis skaicius:";  
        cout << m;  
    }  
    return 0;  
}
```

```
Iveskite du skaicius  
10  
15  
Didesnis skaicius:15  
Process returned 0 (0x0)   execution time : 2.361 s  
Press any key to continue.
```

## 8.6 Persipynę IF-ELSE sakiniai

Vienas IF operatorius leidžia daryti tik vieną sprendimą. Tačiau daugelyje atvejų programa turi padaryti eilę sprendimų. Tuo tikslu gali būti naudojami persipynę IF-ELSE operatoriai.

**Pavyzdys.** Programa tarp trijų skaičių randa didžiausią.

```
#include<iostream>
using namespace std;

int main(void)
{
    int n,m,l;
    cout << "Iveskite tris skaicius" << endl;
    cin >> n >> m >> l;
    if (n>m && n>l)
    {
        cout << "Didziausias skaicius: ";
        cout << n;
    }
    else if (m>n && m>l)
    {
        cout << "Didziausias skaicius: ";
        cout << m;
    }
    else
    {
        cout << "Didziausias skaicius: ";
        cout << l;
    }
    return 0;
}
```

```
Iveskite tris skaicius
1 5 8
Didziausias skaicius: 8
Process returned 0 (0x0)   execution time : 2.038 s
Press any key to continue.
```

## Pasipraktikuokite

1. Parenkite programą, kuri ekrane parodytų užrašą “Jūs atspėjote skaičių, jeigu įvestas skaičius sutapo su priskirtu programoje”, užrašą “Skaičius mažesnis”, jei įvestas skaičius mažesnis, už priskirtą programoje, užrašą “Skaičius didesnis”, jei įvestas skaičius didesnis už priskirtą programoje.
2. \*Ūkininkas nusprendė virve pažymėti stačiakampio plotą, kuriame sodins ankstyvasias bulves. Virvės ilgis lygus  $m$  metrų (sveikasis skaičius). Kokį didžiausią plotą  $s$  galės pažymėti ūkininkas? Rezultatą pateikite sveikuoju skaičiumi.  
Kai  $m=22$ , turi spausdinti  $s=30$ .