Konspektas

C++ programavimas

7 pamoka

Loginiai duomenys

Programuojant operuojama ne tik sveikųjų bei realiųjų skaičių, bet ir loginio tipo duomenimis. Loginiai kintamieji gali įgyti tik dvi reikšmes: true (teisinga) ir false (neteisinga). Su jais galima atlikti logines operacijas: neigimą (inversiją), sudėtį (disjunkciją) ir daugybą (konjunkciją).

Loginiai dydžiai ir loginės išraiškos vartojami sąlyginėse valdymo struktūrose, kuriu sintaksė yra tokia:

if (<Loginė sąlyga>)

<Šaka TAIP>;

[else <Šaka NE>];

***

```
if (5*a-3*b == c)
cout << "Teisingai!" << endl;
else
cout << "Klaida..." << endl;
```

Sąlyginė valdymo struktūra nurodo, kad turi būti patikrinta joje įrašyta loginė sąlyga ir, priklausomai nuo patikrinimo rezultatu, parenkama vykdymui struktūros šaka TAIP arba NE. Kadangi tokia valdymo struktūra aprašo, kaip vykdymui parenkama viena iš dviejų galimų alternatyvų, ji dar yra vadinama sąlyginiu dvivariantiniu alternatyvu parinkimo operatoriumi.

Loginės sąlygos C++ kalboje apibrėžiamos taip:

== tikrinama ar kintamieji lygus

!= tikrinama ar kintamieji nelygūs

< tikrinama ar pirmas kintamasis mažesnis už antrą

> tikrinama ar pirmas kintamasis didesnis už antrą

<= tikrinama ar pirmas kintamasis mažesnis už antrą arba lygus jam

>= tikrinama ar pirmas kintamasis didesnis už antrą arba lygus jam

|| tikrinama ar bent vienas iš kintamųjų tinka pagal sąlygą (loginis ARBA)

&& tikrinama ar visi kintamieji tinka pagal sąlygą (loginis IR)

```
if                                                                          (k==0)
                        cout                    <<                  "lygu                   nuliui";
else
     cout << "nelygu nuliui";
```

Struktūroje šaka *else* gali būti praleidžiama, tuomet ji aprašo jau ne alternatyvu parinkimą, o vieno operatoriaus vykdymo sąlygas.

```
 if                                                                         (k==0)
     cout << "lygu nuliui";
```

Valdančiose struktūrose vartojama paprasčiausia programų struktūrizavimo priemonė - blokai, kurie dar vadinami sudėtiniais sakiniais. Tai tarp figūriniu skliaustu **{** ir **}** įrašyta programos sakiniu grupė. Su tokia grupe transliatorius elgiasi taip, lyg tai būtu vienas sakinys (operatorius), todėl figūriniai skliaustai dar vadinami operatoriniais skliaustais. Jei kiekvienoje pasirinkimo šakoje (*true* ir *false*) yra *daugiau nei viena vykdoma operacija*, būtina naudoti figūrinius skliaustus – priešingu atveju kompiliatorius įvykdys tik pirmą sakinį šakoje arba rodys klaidą.

```
 if                                                                          (k==0)
                                                                                 {
                        cout                    <<                  "kintamasis                 k";
                        cout                    <<                   "lygu                  nuliui";
                                                                                 }
 else
                                                                                 {
                        cout                    <<                  "kintamasis                 k";
                        cout                    <<                  "nelygu                 nuliui";
    }
```

Aprašant variantinį nauju reikšmių skaičiavimą, galima vartoti specialu sąlyginį operatorių ?, kurio sintaksė:

<i1> ? <i2> : <i3>

Šiame operatoriuje išraiška i1 aprašo tikrinamą loginę sąlygą, o išraiškos i2 ir i3 - rezultatui parenkamas reikšmes. Jei išraiškos i1 reikšmė nelygi nuliui (true), operatoriui suteikiama išraiškos i2 reikšmė, o kitu atveju - i3 reikšmė. Pavyzdžiui, sąlyginį operatorių

```
if                                                                          (k==0)
                        cout                    <<                  "lygu                   nuliui";
else
     cout << "nelygu nuliui";
```

galima pakeisti į tokį sakinį

```
 (k==0)? cout << "lygu nuliui":cout <<"nelygu nuliui";
```

Kai parenkamu alternatyvu skaičius didesnis už 2, galima vartoti struktūrų if grandines, tačiau vaizdesnės ir patogesnės yra daugiavariantiniu alternatyvu aprašymo struktūros *switch*. Jų sintaksė:

switch (<Raktas>)

{

case <l-ji rakto reikšmė>: <l-ji alternatyva>; break;

case <2-ji rakto reikšmė>: <2-ji alternatyva>; break;

case <n-ji rakto reikšmė>: <n-ji alternatyva>; break;

[default: <Alternatyva visoms kitoms rakto reikšmėms>; break;]

} // Struktūros pabaigos žymė

Raktas - tai diskretinio tipo kintamasis, kurio reikšmės valdo alternatyvu parinkimą. Kiekvienos altern atyvos aprašymo pabaigą žymi nutraukimo operatorius break, kuris nurodo, kad likusiu operatoriaus switch šakų nagrinėti nebereikia.

```
cin >> z;
switch (z)
{
case 1: cout<<"vienetas"; break;
case 2: cout<<"dvejetas"; break;
case 3: cout<<"trejetas"; break;
case 4: cout<<"ketvertas"; break;
case 5: cout<<"penketas"; break;
default: cout <<"daugiau uz penkis"; break;
}
```

Žemiau pateikta programa konvertuoja įvestus metrus į milimetrus, centimetrus, decimetrus arba kilometrus:

```
#include <iostream.h>
main()
{
 char s;
 float k, sk;
 cout <<"Ilgio vienetu kovertavimo programa"<< '\n';
 cout << "Kiek metru?" << '\n' << "l=";
 cin >> k;
 cout << '\n';
 cout <<"Metrus paversti i:"<< '\n';
 cout << "'m'-milimetrus" << '\n';
 cout << "'c'-centimetrus" << '\n';
 cout << "'d'-decimetrus" << '\n' ;
 cout << "'k'-kilometrus" << '\n';
 cout << "--";
 cin >> s;
 switch (s)
 {
 case 'm':
           sk=k*1000;
           cout <<k<<" metru" <<" turi " << sk <<" milimetru" <<'\n';
           break;
 case 'c':
           sk=k*100;
           cout <<k<<" metru" <<" turi " << sk <<" centimetru" <<'\n';
           break;
 case 'd':
           sk=k*10;
           cout <<k<<" metru" <<" turi " << sk <<" decimetru" <<'\n';
           break;
 case 'k':
           sk=k/1000;
           cout <<k<<" metru" <<" turi " << sk <<" kilometru" <<'\n';
           break;
 default:
           cout <<"Tokio pasirinkimo nera!" << '\n';
           break;
 }
 cout << '\n';
 cout << "noredami iseiti iveskite bet koki simboli" << "\n";
 cin >> k;
}
```

Paimta iš http://www.technologijos.lt/n/technologijos/it/straipsnis?name=straipsnis-2645
http://www.angelfire.com/hero/mythas/lab2_element.html

http://p4i.armandas.lt/manual.php/read

http://www.doc.ic.ac.uk/~wjk/C++Intro/RobMillerE5.html

http://www.coderland.lt/

Pavyzdys

C ++

****

#include <iostream>

using namespace std;

int main()

{

```
int a, b, c;
cout << "Iveskite du skaicius skaicius:" << endl;
cin >> a >> b;
cout << "Kiek bus 5a-3b? ";
cin >> c;
if (5*a-3*b == c)
cout << "Teisingai!" << endl;
else
cout << "Klaida..." << endl;
}
```