

9 PASKAITA

Turinys: Masyvai. Daugiamačiai masyvai. Kai kurie algoritmai masyvams. Kai kurios formatavimo galimybės.

MASYVAI

Masyvas – vidinis C++ konteineris to paties tipo duomenų grupavimui: seka, turinti vieną vardą ir daug reikšmių, iš kurių norima atrenkama kreipiantis sekos vardu ir nurodant sekos nario indeksų reikšmes.

Masyvai – vienmačiai, dvimačiai,

Galima talpinti vidinių kalbos tipų ir vartotojo apibrėžtų tipų duomenis.

Indeksai pradedami skaičiuoti nuo nulio.

Gyvavimo ciklas: skelbimas ir apibrėžimas (skiriama atmintis), inicializavimas (suformuojamos reikšmės), darbas su masyvu, masyvo sunaikinimas (baigiantis programai, išlaisvinama masyvo užimta atmintis).

Skelbimas ir apibrėžimas:

tipas masyvoVardas [matai];

Inicializavimas: pavyzdžiai:

```
int x[ 5 ] = { 10, 5, 3, 7, 0 };  
int y[ ] = { 10, 5, 3, 7, 0 };
```

Jei inicializuojant matai buvo nurodyti, o reikšmių pateikta mažiau nei nurodyta – likę reikšmės bus nulinės; jei reikšmių per daug – bus deklaruota kompiliavimo klaida.

Pavyzdžiai.

1. Vienmačio masyvo elementų sumos skaičiavimas. Elementai įvedami iš klaviatūros.

```
#include <iostream>  
using namespace std;  
int main( ){  
    float x[ 5 ];  
    float s = 0.f;  
    for( int i = 0; i < 5; i++ ){  
        cout<<"Iveskite "<<i+1<<" –aji masyvo elementa";  
        cin>>x[ i ];  
        cout<<"Ivestas elementas "<<x[ i ]<<endl;  
        s += x[ i ];  
    }  
    cout<<"Elementu suma yra "<<s<<endl;  
    return 0;  
}
```

2. Teigiamų masyvo elementų vidurkio skaičiavimas.

```
#include <iostream>
using namespace std;
int main( ){
    const int N = 5; // rekomenduojamas stilius
    float x[ N ];
    float s = 0.f;
    int k = 0; // teigiamu elementu kiekis
    // Ivedimas
    for( int i = 0; i < N; i++ ){
        cout<<"Iveskite "<<i+1<<" –aji masyvo elementa";
        cin>>x[ i ];
        cout<<"Ivestas elementas "<<x[ i ]<<endl;
        s += x[ i ];
    }
    // Skaiciavimas ir isvedimas
    for( i = 0; i < N; i++ ){
        if( x[ i ] > 0.f ){
            k ++;
            s += x[ i ];
        }
    }
    if( k > 0 ){
        cout<<"Teigiamu elementu vidurkis yra "<<s/k<<endl;
    } else {
        cout<<"Teigiamu elementu nera\n";
    }
    return 0;
}
```

Dvimačiai masyvai. Kai kurios formatavimo galimybės

Tai – vienmatis masyvas, kurio elementai savo ruožtu yra vienmačiai masyvai.

Analogiškai apibrėžiami ir daugiau matų turintys masyvai.

Atmintyje dvimačiai masyvai išdėstomi kaip vienmatė seka eilutė po eilutės.

Skelbimas ir apibrėžimas: pavyzdžiui:

```
int x[ 3 ] [ 3 ];
```

Inicializavimas:

```
int x[ 3 ] [ 3 ] = { {1, 2, 3}, {11, 12, 13}, {21, 22, 23} };
```

Pavyzdys: dvimačio masyvo eilučių vidurkių skaičiavimas. Išvedimas – formatinis.

```
#include <iostream>
#include <iomanip> //formatavimo manipulatoriams
using namespace std;
int main( ){
    const int N = 10, M = 10;
    float x[ N ][ M ];
        // Ivedimas
    cout<<"Iveskite masyvo elementus eilutemis:\n";
    for( int i = 0; i < N; i++ ){
        for( int j = 0; j < M; j++ ){
            cout<<"Iveskite elementa "<<i+1<<j+1<<" :";
            cin>>x[ i ][ j ];
            cout<<"Ivestas elementas "<<i+1<<j+1<<x[ i ][ j ]<<endl;
        }
    }
        // Skaiciavimas ir isvedimas
    for( i = 0; i < N; i++ ){
        float s = 0.f;
        for( int j = 0; j < M; j++ )
            s += x[ i ][ j ];
        s /= M;
        cout<<setiosflags( ios::fixed )<< //pastaba
        setiosflags( ios::showpoint )<<
        setprecision( 2 )<<
        setw( 10 )<<
        s<<endl;
    }
    return 0;
}
```

Pastaba: *ios* – formatavimo konstantų klasė. Konstantos teikiamos tokiu būdu kaip parodyta ir reiškia: *fixed* – išvestis ne eksponentinė; *showpoint* – išvestyje visada rodyti dešimtainį tašką. Manipulatorius *setprecision* nustato skaitmenų po kablelio kiekį, o *setw* – išvesties lauko plotį. Prireikus, konstantas galima atšaukti manipulatoriumi *resetiosflags*.