

C++ DASTURLASH TILIDA MASSIVLAR USTIDA AMALLAR VA TENGLAMALAR SISTEMASINI GAUSS USULIDA YECHISH

Yunusova Dilrabo Akmaljon qizi

*Farg‘ona davlat universiteti Axborot tizimlari va
Texnologiyalari yo‘nalishi I kurs talabasi*

Annotatsiya: Ushbu maqolada C++ dasturlash tilida massivlar bilan ishlash va tenglamalar sistemasini Gauss usuli yordamida yechish masalasi yoritilgan. Massivlar bilan amallar bajarish, ularning turlari, xotirada joylashuvi va ulardan samarali foydalanish usullari haqida ma‘lumot beriladi. Shuningdek, Gauss usulining nazariy asosi, algoritmi va dastur kodini yozish bosqichlari tushuntiriladi. Maqolada algoritmning ishlash tamoyillari va natijalarning aniqligi tahlil qilinib, C++ tilida uning dasturiy ta‘minoti misollar bilan yoritilgan.

Kalit so‘zlar: C++ dasturlash tili, massivlar, massivlar ustida amallar, chiziqli tenglamalar sistemi, Gauss usuli, matematik modellashtirish, algoritmlar, dasturlash, ma‘lumotlar tuzilmasi, sonli usullar.

Аннотация: В статье рассматривается работа с массивами на языке программирования C++ и решение системы уравнений методом Гаусса. Приведена информация о выполнении операций с массивами, их типах, расположении в памяти и методах их эффективного использования. В нем также объясняются теоретические основы, алгоритм и этапы написания программного кода для метода Гаусса. В статье анализируются принципы работы алгоритма и точность получаемых результатов, а также иллюстрируется его реализация на языке C++ на примерах.

Ключевые слова: Язык программирования C++, массивы, операции над массивами, системы линейных уравнений, метод Гаусса, математическое моделирование, алгоритмы, программирование, структуры данных, численные методы.

Abstract: This article discusses the issue of working with arrays and solving a system of equations using the Gauss method in the C++ programming language. Information is provided about performing operations with arrays, their types, memory location, and methods for their effective use. The theoretical basis of the

Gauss method, the algorithm, and the stages of writing the program code are also explained. The article analyzes the principles of operation of the algorithm and the accuracy of the results, and its implementation in the C++ language is illustrated with examples.

Keywords: C++ programming language, arrays, operations on arrays, system of linear equations, Gaussian method, mathematical modeling, algorithms, programming, data structures, numerical methods.

KIRISH

Hozirgi kunda C++ dasturlash tili turli sohalarda keng qo‘llanilib, matematik va muhandislik masalalarini yechishda samarali vosita bo‘lib xizmat qilmoqda. Ayniqsa, massivlar bilan ishlash va tenglamalar sistemasini yechish kabi masalalarda C++ ning tezkorligi va samaradorligi katta ahamiyatga ega.

Ushbu maqolada C++ dasturlash tilida massivlar bilan ishlash tamoyillari, ularning turlari va ulardan foydalanish usullari batafsil yoritiladi. Shuningdek, chiziqli tenglamalar sistemasini Gauss usuli yordamida yechish jarayoni, uning algoritmik asosi va dasturiy ta‘minoti ko‘rib chiqiladi. Gauss usuli eng samarali sonli usullardan biri bo‘lib, uni dasturiy jihatdan amalga oshirish chiziqli algebra va hisoblash matematikasining muhim yo‘nalishlaridan biridir.

MASSIV TUSHUNCHASI

Massiv -o‘zida bir turga tegishli ma‘lumotlarni, tartiblangan ko‘rinishda saqlovchiligi sifatida qarasa bo‘ladi. Massivning xar bir elementiga uning adresi bo‘yicha murojaat qilish mumkin.

Massiv – bir turdagi ma‘lumotlarning tartiblangan ko‘rinishi.

C va C++ tillarida massivlar bilan ishlash va tuzilishi deyarli bir xil.

Massiv xossalari

Massivlar bilan ishlashda uning xossalariga alohida e‘tibor berishga to‘g‘ri keladi. Shularni hisobga olib quyida massivning asosiy xossalariga to‘xtalib o‘tamiz.

Massivda element deb nomlanuvchi alohida qiymatlar saqlanadi Massivning barcha elementlari bir xil turga tegishli bo‘lishi lozim

Massivning barcha elementlari xotirada ketma-ket joylashadi va birinchi element nolinch indeksga ega bo‘ladi

Massiv nomi o‘zgarmaydi, ya‘ni dastur o‘rinlanishi davomida oldindan ko‘rsatilgan nom bilan foydalaniladi.

Massivlarni e‘lon qilish. Massivlarni ta‘riflash (e‘lon qilish)o‘zgarivchilarni e‘lon qilishga o‘xshab ketadi. Farqi massiv nomidan keyin kvadrat qavslar ichida, massiv hajmini ko‘rsativchio‘zgarmas ifoda beriladi. Masalan

```
int butun_mas[15]; // 15 ta butun sondan iborat massiv
```

```
char simral_mas [10]; // 10 ta simvoldan iborat massiv
```

O‘zgarmas miqdorlardan foydalanishning ahamiyati shundaki, massivda mavjud bo‘lmagan elementlarga murojat qilishdan kelib chiqadigan xatoliklarni chetlabo‘tish imkoniyatini beradi.

Massiv elementlariga qiymat berish va murojat qilish. C/C++ tillarida massivlarga qiymat boshqa dasturlash tillaridagidek bir nechta usullari mavjud. Shuning bilan birga ayrim o‘zgachiliklarga ham ega. Jumladan Turbo Pascal tilida massiv elementlariga hech qanday qiymat bermasdan elementlar qiymatini ekranga chiqarib ko‘radigan bo‘lsak , massiv turiga mos har hil qiymatlarni chiqarishi mumkin. C++ tilida esa statik va umumiy (global) masalalarda bu xollar kuzatilmaydi, ya'ni tinch xolatida daslabki elemenlari uchun nol qiymati qabul qilinadi.

Dastur o‘rinlanishi natijasida ekranda massiv elementlariga tinch xolatida qabul qilingan qiymatlari , nollar xosil bo‘ladi . Qiymat qabul qilishning yana bir,o‘zgarivchilariga qiymat qabul qilgan kabi, aniq qiymat qabul qilish quyidagicha bo‘ladi.

```
Int butun_mas [ 5 ] = {10,-3,0,4,1};
```

```
Float haqiqiy_mas [ 3 ] = {3.1417, 2.7 , 0.25 };
```

```
static int butun_hato [ 2 ] = {9,8,7,6,5,4,3,2,1};
```

Bizga ma'lum , massivga murojat qilishda massiv elementining tartib nomerlarin ifodalovchi indeksni ko‘rsatish lozim . Bu yerda qat'iy e‘tibor berishimiz kerak bo‘lgan narsa, massivning birinchi elementining indeksi hamma vaqt noldan boshlanadi. Boshlavchi programmistlar ko‘pincha birinchi element indeksi 1ga teng deb xatoga yo‘l qo‘yishadi.

Endi quyidagi dasturga to‘xtalib o‘taylik. Massiv elementlarining ichida eng katta va eng kichik elementlarini topish uchun dastur tuzing.

```
# include <>
```

```
# define mas_max 10
main ( )
{int mas [mas_max ] = { 4,-3,0,9,7,10,2,-1,15,5 }
int min qmas [0] , max q mas [0] ;
int index ; for (index = 1 ; index < mas_max ; index ++ )
{if (mas [index] < min) min = mas [index] ;
if (mas [index] > max) max = mas [index] ; }
cout <<< max << " ga teng\n" ;
cout <<< min << " ga teng\n";}
```

Bir necha indeksli massivlar. Massivlar bir necha indeksga ega bo‘lishlari mumkin. C++ kompilyatorlari eng kamida 12 ta indeks bilan ishlashlari mumkin. Masalan, matematikadagi $m \times n$ kattalikdagi matritsani ikkita indeksli massiv yordamida berisak bo‘ladi. `int matritsa [4][10];`

Yuqorida to‘rt satrlik, 10 ustunlik matritsani e‘lon qildik. Bir indeksli massivlar kabi ko‘p indeksli massivlarni initsializatsiya ro‘yhati bilan birga e‘lon qilish mumkin. Masalan:

```
char c[3][4] = { { 2, 3,9, 5}, // birinchi satr qiymatlari
                {-10, 77,5, 1}, // ikkinchi " "
                {90,233,3,-3} }; // uchinchi " "
```

```
int m[2][2] = {56,77,8,-3}; //oldin birinchi satrga qiymatlar beriladi, keyin
ikkinci satrga
```

```
double d[4][3][6] = {2.55, -46,0988}; // birinchi satrning dastlabki ikkita elementi
qiymat
```

//oladi, massivning qolgan elementlari esa nolga tenglashtiriladi

Massivning har bir indeksi alohida [] qavslar ichiga olinishi kerak. Yuqoridagi `c[][]` massivining ikkinchi satr, birinchi ustunidagi elementi qiymatini birga oshirish uchun

```
++c[1][0]; // yoki c[1][0]++; // c[1][0] += 1; // c[1][0] = c[1][0] + 1;
```

deb yozishimiz mumkin. Massiv indeksleri 0 dan boshlanishini unutmaslik zarur.

FUNKTSIYALAR. QISM DASTURLAR.

C/C++ dasturlash tillari dastur kodining asosiy qismini funktsiyalar tashkil etadi. Ular dasturni bir necha bloklarga bo‘lish imkoniyatini beradi. Bizga ma‘lumki bu

tillardagi ixtiyoriy dastur `main( )` funksiyasini o‘zida mujassamlashtiradi. Funksiyalarni yaxshi tayorlashi, dasturning effektiv va ishonchli ishlashini ta'minlaydi.

Funksiyalar tuzilishining umumiy ko‘rinishi.

Funktsiya dasturining nom berilgan shunday qismiki, unga dasturning boshqa Funksiyalarning tuzilish usuliga to‘xtalib o‘tishdan oldin funktsiya tushinchasiga bir bo‘limidan qancha talab qilinsa, shuncha murojat qilish mumkin.

ANSI C standartiga mos holda hozirgi funktsiyaning argument turlari va chiqarivchi tur nomi oldindan (asosiy blokdan oldin) elon qilinishi lozim. Quyida funktsiyaning umumiy holda e‘lon qilamiz:

Natija _turi funktsiya_nomi (argument_turi shart_bo‘lmagan_argument_nomi [...]);

Funktsiya `void`, `int`, `float`, `char`, va h.k turdagi qiymatlardan birini berishi mumkin (qaytarishi). Ushbu ko‘rinish `main()` funksiyasi (asosiy dastur) dan oldin yoziladi.

Funktsiya kodi `main ( )` funksiyasidan avval yoki keyin yoziladi:

Natija_turi funktsiya_nomi (argument_turi argument_nomi [...])

{.....funktsiya_tanasi }.....

Proceduraga murojat qilganda ‘=’ kerak emas. Procedura tashkil qilishda agar natijalar birne4ta bo‘lsa ko‘rsatkichlardan foydalaniladi. Ko‘rsatkich – bu biroro‘zgaruvchining adresini o‘zida saqlovchi kattalik. Ko‘rsatkichni e‘lon qilishda dan foydalaniladi. Ko‘rsatkichlardan foydalanilganda ularni osonlashtirish uchun ‘adresni ol’ (&) belgisi orqali ham amalgam oshirsa bo‘ladi. Lekin bu amal faqato‘zgaruvchilarga qo‘llanadi, songa qo‘llash mumkin emas. Proceduralar hosil qilishda * va & amallaridan yoki to‘g‘ridan to‘g‘ri & operatsiyasi orqali ishlatishimiz mumkin.

Misol:

```
Void top(float a, float b, float *s,float *p)
```

```
{*s=a*b; *p=2*(a+b);}
```

```
main()
```

```
{ —//—
```

```
top (23,4,&s1,&p1); cout<<s1<<<s1<<p1;}
```

```
yoki
```

```
Void top(float a, float b, float &s, float &p)
{ s=a*b; p=2*(a+b); }
main()
{ —//— top (23,4,s1,p1); cout<<s1<<p1; }
```

Xulosa: Ushbu maqolada C++ dasturlash tilida massivlar bilan ishlash va ularning Gauss usuli orqali chiziqli tenglamalar sistemasini yechish jarayonidagi ahamiyati yoritildi. Massivlar ma'lumotlarni saqlash va ularga samarali ishlov berish vositasi sifatida muhim rolni o'ynaydi. Ayniqsa, Gauss usulida asosiy hisob-kitoblar massivlar orqali amalga oshiriladi, bu esa tezkorlik va aniqlikni ta'minlaydi.

Ish davomida massivlarning tuzilishi, ularning xotirada joylashuvi, ustida bajariladigan amallar va Gauss eliminatsiya jarayoni uchun optimallashtirilgan yondashuvlar ko'rib chiqildi. Dasturiy ta'minotda massivlardan samarali foydalanish algoritmining tezkor ishlashi va resurslardan oqilona foydalanish imkoniyatini oshiradi.

Natijalarga ko'ra, massivlar yordamida Gauss usulini dasturiy amalga oshirish chiziqli algebra masalalarini yechishda ishonchli va samarali yechim bo'lib xizmat qiladi. Shu bois massivlardan optimal foydalanish dastur tezligi va samaradorligini oshirish uchun muhim omillardan biri hisoblanadi.

FOYDALANILGAN ADABIYOTLAR.

1. Jess Liberti, "Освой самостоятельно C++ за 21 день", Sankt Peterburg 2000, 815 с.
2. Liberti D. Освой самостоятельно C++: 10 минут на урок. Ingl. Tarjima. Vilyams, 374 стр, 2004 г.
3. Sayfiev J. F., «C++ tiliga kirish», Buxoro 2004 y.
4. Shmidskiy Ya.K. Программирование на языке C++: Самоучитель. Учебное пособие. Диалектика. 361 стр, 2004 г.
5. Kimmel P., «Borland C++5» . - СПб.: BHV, 1997.
6. Bryan Straustrap. Введение в язык C++ <http://www.infocity.kiev.ua/>
7. Kris Pappas, Uilyam Myurrey. Программирование на C и C++ "Ирина", BHV, Kiev 2000 y.
8. Fridman Aleksandr L'vovich, Язык программирование C++