

C# DASTURLASH TILIDA MASSIVLARGA KIRISH

Eshpo'latov Mansur O'tkir o'g'li
Xolturayeva Kamola Bahrom qizi
G'afforova Lobar Safar qizi

*Termiz davlat universiteti "Amaliy matematika va intellektual texnologiyalar"
fakulteti "Amaliy matematika" ta'lim yo'nalishi I bosqich talabalari.*

Annotatsiya: *Dasturlashga yangi kirib kelayotganlar uchun eng muhim vazifalardan biri bu dastur kodini qisqaroq va tushunarliroq ko'rinishda yozish. Buning uchun esa tajriba va salohiyat muhim o'rin tutadi. Dasturchi o'zi yozgan dastur kodini qisqa va foydalanuvchiga tushunarli bo'lishi uchun bir qancha yo'llardan foydalanadi. Biz siz bilan ko'rib o'tmoqchi bo'lgan yo'limiz massivlar orqali bir xil turdagi o'zgruvchilarni bir nom ostida birlashtirish.*

Kalit so'zlar: *massiv, massiv o'lchami, indeks, statik massiv, dinamik massiv, ko'p o'lchovli massiv, matritsa, pog'onali massiv.*

Massiv tarkibida elementlar mavjud bo'lib eng ko'pi bilan ketishi mumkin bo'lgan elementlar soni uning o'lchamini bildiradi. Massivning elementi turgan o'rni uning indeksi deyiladi va shu elementga uning indeksi orqali murojaat qilinadi. Massivning indeksi sifatida butun sonlar xizmat qiladi. Har bir massiv o'zining individual nomiga ega bo'lishi kerak, ya'ni bir xil nomdagi massivlar bo'lmaydi. Ularning nomi oldin e'lon qilingan oddiy o'zgaruvchi nomi bilan ustma-ust tushmasligi kerak. Massivlar o'lchamiga qarab ikki turga bo'linadi: bir o'lchovli massivlar va ko'p o'lchovli massivlar.

Massivlarni e'lon qilinishi ikki xil: statik va dinamik ko'rinishda bo'ladi. Bir o'lchovli massivning statik e'lon qilinishi quyidagicha:

```
<tip>[] <massiv_nomi> = {boshlang'ich_qiyamatlar};
```

Misol:

```
int[] a = {1, 2, 4, 8};
```

```
double[] b = {1.3, 4.6, 3.8};
```

Bir o'lchovli massivlarni dinamik e'lon qilish new operatori orqali amalga oshiriladi:

```
<tip>[] <massiv_nomi> = new <tip>[o'lcham];
```

Bu yerda new operatori o'lcham ga mos ravishda xotiradan joy ajratadi.

Misol:

```
int[] M1 = new int[10];
```

```
double[] M2 = new double[100];
```

```
n = Convert.ToInt32(Console.ReadLine());
```

```
double[] M3 = new double[n];
```

M1 va M2 nomli massivlarning elementlari uchun mos ravishda 10 va 100 ta joy ajratilgan. Ular ham dinamik ko‘rinishda e‘lon qilingan massiv hisoblanadi. M3 massiv uchun xotiradan qancha joy ajratish foydalanuvchining o‘ziga havola qilingan, ya’ni n o‘zgaruvchisi foydalanuvchi tomonidan kiritiladi, bu o‘zgaruvchi qiymati esa M3 massiv o‘lchami sifatida qabul qilinadi. M3 nomli massiv dinamik e‘lon qilishga yorqin misoldir.

Statik massivlarning elementlari oldindan aniqlanadi, xo‘sh dinamik massivlarga qanday qilib qiymat beramiz? Buning uchun sikl operatorlariga murojaat qilamiz. Masalan, quyidagi misolda dinamik massivga qiymat berish hamda uning elementlarini chop etish amalda ko‘rsatilgan.

Eslatma: dasturlash tillarida sanoq noldan boshlanadi, ya'ni: 0,1, 2, 3, ...

(Buriyev, 2022)

1-listing

```
using System;
```

```
internal class ArrayDemo
```

```
{
```

```
private static void Main()
```

```
{
```

```
int[] A = new int[10];
```

```
int i;
```

```
for (i = 0; i < 10; i = i + 1)
```

```
A[i] = i;
```

```
for (i = 0; i < 10; i = i + 1)
```

```
Console.WriteLine("A[" + i + "]: " + A[i]);
```

```
}
```

```
}
```

Bu yerda A massivning elementlari sifatida i ning qiymatlari olinmoqda. Xuddi shunday massiv elementlarini foydalanuvchi tomonidan kiritiladigan qilish mumkin.

Misol:

2-listing

```
using System;
```

```
internal class ArrayDemo
```

```
{
```

```
private static void Main()
```

```
{
```

```
int[] A = new int[10];
```

```
int i;
```

```
for (i = 0; i < 10; i = i + 1)
```

```
A[i] = Convert.ToInt32(Console.ReadLine());
```

```
for (i = 0; i < 10; i = i + 1)
```

```
Console.WriteLine("A[" + i + "]: " + A[i]);
```

```
}  
}
```

Quyidagi mavzuga doir masala orqali massivlar ustida amallarni bajaramiz:

Sinfda N ta o'quvchi bor va ularning yoshlari berilgan. 1 yildan so'ng ularning yoshlaridagi o'zgarishlar ekranga chiqarilsin.

3-listing

```
using System;  
internal class ArrayDemo  
{  
private static void Main()  
{  
Console.Write("O'quvchilar soni: ");  
int n=Convert.ToInt32(Console.ReadLine());  
int[] oquvchilarning_yoshlari=new int[n];  
Console.WriteLine("O'quvchilarning yoshlarini kiriting: ");  
for(int i=0; i<n; i++)  
{  
oquvchilarning_yoshlari[i]=Convert.ToInt32(Console.ReadLine());  
}  
//o'quvchilarni yoshlariga 1 ni qo'shib chiqamiz  
for(int i=0; i<n; i++)  
{  
oquvchilarning_yoshlari[i]++;  
Console.WriteLine((i+1)+"-o'quvchining yoshi: "+oquvchilarning_yoshlari[i]);  
}  
}  
}
```

Shunaqa masalalar borki ularni bir o'lchovli massivlar orqali hal qila olmaymiz. Shunda bizga ko'p o'lchovli massivlar yordamga keladi. Ularning e'lon qilinishi quyidagicha;

```
<tip>[,,...] <massiv_nomi> = new <tip> [o'lcham_1, ... ,o'lcham_n];
```

n o'lchovli massiv uchun [] ichiga (n-1) ta (,) ishlatiladi.

Ikki o'lchovli massivning umumiy ko'rinishi quyidagicha:

```
<tip>[,] <massiv_nomi> = new <tip> [o'lcham_1, o'lcham_2];
```

Ikki o'lchovli massivlarni ishlatganimizda massiv elementlariga mos ravishda matsitsa hosil bo'ldi. Barchamiz bilamiz matritsa sonlardan iborat jadval, ya'ni satrlar va ustunlardan iborat. Masalan, 5x3 matritsaning 1-satrida ham, 2-satrida ham va hokazo 5-satrida ham 3 tadan element bo'lishi kerak.

Ikki o'lchovli massivga doir misol qaraymiz:

Sinfda N ta o'quvchi bor. Ularning yoshlari va yillik baholari berilgan. 1 yildan so'ng ularning baholari o'zgarishsiz qolsa yoshlari va baholarini ekranga chiqaring.

4-listing

```
using System;
```

```
internal class ArrayDemo
```

```
{
```

```
private static void Main()
```

```
{
```

```
Console.WriteLine("O'quvchilar soni: ");
```

```
int n=Convert.ToInt32(Console.ReadLine());
```

```
int a=2;
```

//bu yerda a o'quvchining nechta parametrini olishimizga qarab o'zgaradi. Bizga faqat yosh va baho kerak, demak a=2

```
int[,] oquvchilar=new int[a,n];
```

```
Console.WriteLine("O'quvchilarning yoshlarini kiriting: ");
```

```
for(int i=0; i<n; i++)
```

```
{
```

```
oquvchilar[0,i]=Convert.ToInt32(Console.ReadLine());
```

```
}
```

```
Console.WriteLine("O'quvchilarning baholarini kiriting: ");
```

```
for(int i=0; i<n; i++)
```

```
{
```

```
oquvchilar[1,i]=Convert.ToInt32(Console.ReadLine());
```

```
}
```

```
//o'quvchilarni yoshlariga 1 ni qo'shib chiqamiz
```

```
for(int i=0; i<n; i++)
```

```
{
```

```
oquvchilar[0,i]++;
```

```
Console.WriteLine((i+1)+"-o'quvchining yoshi "+oquvchilar[0,i]+" va bahosi "+oquvchilar[1,i]);
```

```
}}}
```

C# dasturlash tilida massivning maxsus tipi yaratilgan bo'lib, u pog'onali(zinali) massiv deb yuritiladi. Pog'onali massiv massiv ichida massivni joylashtirib, har satrida turlicha miqdordagi elementni joylashtirishi mumkin. Shunday ekan, pog'onali massiv asosan satr elementlari soni turlicha bo'lganda qo'llaniladi. Buni quyidagi jadvallarda ko'rishimiz mumkin:

Oddiy ikki o'lchovli massiv

1	6	2	2	1	6	1	5	2	2
2	7	0	3	2	5	2	4	3	1
3	8	0	2	3	4	3	3	4	2
4	9	0	1	5	3	4	2	3	3

5	1	0	2	4	2	5	1	3	4
---	---	---	---	---	---	---	---	---	---

Pog'onali massiv

1	6	2	2	1	6	1			
2	7	0	3	2					
3	8	0	2	3	4	3	3	4	
4	9	0	1	5	3				
5	1	0	2	4	2	5	1	3	4

(M. O'., 2024)

Pog'onali massivni e'lon qilishning umumiy ko'rinishi quyidagicha:

```
<tip> [] [] <massiv_nomi>=new <tip>[o'lcham_1] [];
```

Bu yerda o'lcham_1 - massiv satrlari soni. Ikkinchi [] ichi bo'sh bo'ladi. Dastur davomida har bir satr uchun alohida o'lcham kiritiladi. Xotiradan satrlar uchun elementlar soni alohida tashkil qilinadi, shuning uchun ham har bir satrning elementlari turlicha yoki bir xil bo'lishi mumkin. Masalan A nomli massivni pog'onali ko'rinishda e'lon qilaylik:

```
int[][] A = new int[3][];
```

```
A[0] = new int[4];
```

```
A[1] = new int[3];
```

```
A[2] = new int[5];
```

Misolda A nomli massivning uchta satri bor deb qabul qilingan. 0-satrida 4 ta, 1-satrida 3 ta, 2-satrida esa 5 ta element uchun xotiradan joy ajratilgan. Massivning jadval ko'rinishi quyidagi bo'ladi:

A[0][0]	A[0][1]	A[0][2]	A[0][3]	
A[1][0]	A[1][1]	A[1][2]		
A[2][0]	A[2][1]	A[2][2]	A[2][3]	A[2][4]

Endi pog'onali massivga doir masalalarni amalda bajaramiz.

Maktabdagi eski binoda 3 ta sinf bor. Ulardagi o'quvchilarning yoshlari va yillik baholari berilgan. 1 yildan so'ng ularning baholari o'zgarishsiz qolsa yoshlari va baholarini ekranga chiqaring.

5-listing

```
using System;
```

```
internal class ArrayDemo
```

```
{
```

```
private static void Main()
```

```
{
```

```
Console.Write("1 - sinf o'quvchilari soni: ");
```

```
int n=Convert.ToInt32(Console.ReadLine());
```

```
Console.Write("2 - sinf o'quvchilari: ");
```

```
int m=Convert.ToInt32(Console.ReadLine());
```

```
Console.Write("3 - sinf o'quvchilari soni: ");
```

```
int l=Convert.ToInt32(Console.ReadLine());
```

```
int b=3; //sinflar soni
int a=2; //parametrlar soni, ya'ni yosh va baho
int[,] oquvchilar=new int[b][,];
oquvchilar[0]=new int[a,n];
oquvchilar[1]=new int[a,m];
oquvchilar[2]=new int[a,l];
Console.WriteLine("1-sif o'quvchilarning yoshlarini kiriting: ");
for(int i=0; i<n; i++)
{
oquvchilar[0][0,i]=Convert.ToInt32(Console.ReadLine());
}
Console.WriteLine("1-sinf o'quvchilarning baholarini kiriting: ");
for(int i=0; i<n; i++)
{
oquvchilar[0][1,i]=Convert.ToInt32(Console.ReadLine());
}
Console.WriteLine("2-sif o'quvchilarning yoshlarini kiriting: ");
for(int i=0; i<m; i++)
{
oquvchilar[1][0,i]=Convert.ToInt32(Console.ReadLine());
}
Console.WriteLine("2-sinf o'quvchilarning baholarini kiriting: ");
for(int i=0; i<m; i++)
{
oquvchilar[1][1,i]=Convert.ToInt32(Console.ReadLine());
}
Console.WriteLine("3-sif o'quvchilarning yoshlarini kiriting: ");
for(int i=0; i<l; i++)
{
oquvchilar[2][0,i]=Convert.ToInt32(Console.ReadLine());
}
Console.WriteLine("3-sinf o'quvchilarning baholarini kiriting: ");
for(int i=0; i<l; i++)
{
oquvchilar[2][1,i]=Convert.ToInt32(Console.ReadLine());
}

//o'quvchilarni yoshlariga 1 ni qo'shib chiqamiz
for(int i=0; i<n; i++)
{
```

```

oquvchilar[0][0,i]++;
Console.WriteLine("1-sinf "+(i+1)+"-o'quvchisining yoshi "+oquvchilar[0][0,i]+"
va bahosi "+oquvchilar[0][1,i]);
}for(int i=0; i<m; i++){oquvchilar[1][0,i]++;
Console.WriteLine("2-sinf "+(i+1)+"-o'quvchisining yoshi "+oquvchilar[1][0,i]+"
va bahosi "+oquvchilar[1][1,i]);
}for(int i=0; i<l; i++){
oquvchilar[2][0,i]++;
Console.WriteLine("3-sinf "+(i+1)+"-o'quvchisining yoshi "+oquvchilar[2][0,i]+"
va bahosi "+oquvchilar[2][1,i]);
}}}
```

Dasturlash tillarida dasturchining ishini osonlashtirish maqsadida turli xil metodlar yaratilgan. Shunday metodlardan massivlar uchun mo'ljallanganlari ham mavjud. Quyida dasturchilar tomonidan eng ko'p ishlatiladigan massivlarga doir metodlar keltirilgan.

Metod	Tavsif
BinarySearch()	Ikkilik qidiruv algoritmidan foydalangan holda qiymatni massiv ichidan qidiradi va mavjud bo'lsa element indeksini qaytaradi.
Clear()	Massivdagi elementlar oralig'ini har bir element turining standart qiymatiga o'rnatadi.
Clone()	Massivning sayoz nusxasini yaratadi.
ConvertAll()	Bir turdagi massivni boshqa turdagi massivga aylantiradi.
Copy()	Bir massivdagi bir qator elementlarni boshqa massivga nusxa ko'chiradi.
CopyTo()	Joriy bir o'lchovli massivning barcha elementlarini belgilangan bir o'lchovli massivga nusxalaydi.
Empty()	Bo'sh massivni qaytaradi.
Exists()	Belgilangan massivda belgilangan predikat tomonidan belgilangan shartlarga mos keladigan elementlar mavjudligini aniqlaydi va bool qiymat qaytaradi.
Find()	Belgilangan predikat tomonidan belgilangan shartlarga mos keladigan elementni qidiradi va butun massivdagi birinchi hodisani qaytaradi.

Metod	Tavsif
FindAll()	Belgilangan predikat tomonidan belgilangan shartlarga mos keladigan barcha elementlarni oladi.
FindIndex()	Belgilangan predikat tomonidan belgilangan shartlarga mos keladigan elementni qidiradi va birinchi uchragan element indeksini qaytaradi. Agar bo'lmasa -1 ni qaytaradi.
FindLast()	Belgilangan predikat tomonidan belgilangan shartlarga mos keladigan elementni qidiradi va butun massivdagi oxirgi hodisani qaytaradi.
FindLastIndex()	Belgilangan predikat tomonidan belgilangan shartlarga mos keladigan elementni qidiradi va oxirgi uchragan element indeksini qaytaradi. Agar bo'lmasa -1 ni qaytaradi.
ForEach()	Belgilangan massivning har bir elementida belgilangan amalni bajaradi.
GetLength()	Massivning belgilangan o'lchamidagi elementlar sonini ifodalovchi 32 bitli butun sonni oladi.
GetLongLength()	Massivning belgilangan o'lchamidagi elementlar sonini ifodalovchi 64 bitli butun sonni oladi.
GetLowerBound()	Massivdagi belgilangan o'lchamning birinchi elementi indeksini oladi.
GetUpperBound()	Massivdagi belgilangan o'lchamning oxirgi elementi indeksini oladi.
GetValue()	Joriy Massivda belgilangan elementning qiymatini oladi.
IndexOf()	Belgilangan ob'ekttni qidiradi va uning bir o'lehovli massivda yoki massivdagi elementlar oralig'ida birinchi marta paydo bo'lish indeksini qaytaradi.
LastIndexOf()	Bir o'lehovli massivda yoki massivning bir qismida qiymatning oxirgi marta paydo bo'lishi indeksini qaytaradi.
MemberwiseClone()	Joriy Ob'ektning sayoz nusxasini yaratadi.
Resize()	Bir o'lehovli massiv elementlari sonini belgilangan yangi o'lchamga o'zgartiradi.

Metod	Tavsif
Reverse()	Bir o'lchovli massivdagi yoki massivning bir qismidagi elementlarning tartibini o'zgartiradi, ya'ni teskarilaydi.
SetValue()	Joriy Massivdagi belgilangan qiymatni belgilangan elementga o'rnatadi.
Sort()	Bir o'lchovli massivdagi elementlarni tartiblaydi.
ToString()	Joriy ob'ektni ifodalovchi qatorni qaytaradi. (Ob'ektdan meros qilib olingan)
TrueForAll()	Massivdagi har bir element belgilangan predikat tomonidan belgilangan shartlarga mos kelishini aniqlaydi.

FOYDALANILGAN ADABIYOTLAR:

1. Керов, Л. А. (2009). Массивы и коллекции в языке с. Компьютерные инструменты в образовании, (4), 39-52.
2. Керов, Л. А. (2009). Методы классы и конструкторы в языке с. Компьютерные инструменты в образовании, (5), 45-59.
3. Nosir o'g'li, B. J. (2022). O'ZBEKISTON RESPUBLIKASI AHOLI POPULYATSIYASINING MATEMATIK MODELI. O'ZBEKISTONDA FANLARARO INNOVATSIYALAR VA ILMIY TADQIQOTLAR JURNALI, 2(14), 676-681.
4. Mamadiyor o'g'li, J. B. (2024). GPA BALINI POG 'ONALI MASSIVLAR YORDAMIDA OSON HISOBLASH. O'ZBEKISTONDA FANLARARO INNOVATSIYALAR VA ILMIY TADQIQOTLAR JURNALI, 3(30), 235-237.
5. Begaliyevich, N. C., Shokirovich, T. D., Yuldashevich, J. I., Nosirovich, B. J., & Bakhtiyorovich, J. M. (2021). Definition of Different Schemes for Calculation of General Solutions. CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES, 2(12), 61-70.