

## КӨПӨЛШЕМДІ МАССИВТЕР

### 9.1 Екі өлшемді массивтер

C# тілінде тек бір өлшемді массивтерді (векторлар) ғана емес, сонымен қатар элементтердің саны тек компьютер жадысының көлемімен шектелген екі өлшемді және көп өлшемді массивтерді анықтауға болады .

Үш өлшемді және көп өлшемді ақпараттар әдетте математикалық түрде сипатталып ұсынылады. Кәсіпорын иесінің 5 сауда орыны болсын. Бір жыл ішінде ай сайын 100 тауардың сатылымы жөнінде ақпарат келесі үш өлшемді массив арқылы көрсетіле алады:

```
int[ , ] tovar new int[5,12,100];,
```

мұнда

0..4 индексі тауарды сату орындарын анықтайды;

0..11 индексі – сатылым айы;

0..99 индексі – сатылатын тауар нөмері.

Тауарды сату туралы ақпаратты көрген кезде кәсіпкер оны монитор экранында кесте немесе матрица түрінде көруі керек. Мысалы, бір жыл ішінде бірінші сауда орынында тауардың сатылымы немесе ай сайын барлық тауарлардың сатылымы, т.б. Сонымен, көп өлшемді массивті екі өлшемді массивке айналдыру керек.

Көп өлшемді және екі өлшемді массивтермен жұмыс істеу технологиясы бірдей. Сондықтан екі өлшемді массивтерді қарастырамыз.

9.1-есеп. Минус 50-ден 100-ге дейінгі аралықтағы кездейсоқ бүтін сандардан тұратын 6×6 матрицасын құру керек. Барлық оң және теріс сандардың қосындысын есептеп, оны шығару керек. Матрицаның ең үлкен және ең кіші элементтерін тауып, экранға шығарыңыз.

Бағдарлама коды:

```
using System;
namespace ConsoleApplication1
{
    class Program
    {
        public static int sp, so;

        public static void sozd(int[,] ma)
        {
            Random rnd = new Random();
            Console.WriteLine("Matrisa kyrildi!!");
            for (int i = 0; i < 6; i++)
            {
                for (int j = 0; j < 6; j++)
                {
                    ma[i,j] = rnd.Next() % 101 - 50;
                    Console.Write(ma[i, j]+"\t");
                }
            }
        }
    }
}
```

```

    Console.WriteLine();
}
}
public static void polotr(int[,] ma)
{
    sp = 0; so = 0;
    for (int i = 0; i < 6; i++)
    for (int j = 0; j < 6; j++)
        if (ma[i,j] < 0)
            so = so + ma[i,j];
        else sp = sp + ma[i,j];
    Console.WriteLine("On sandardin kosindisi = {0}", sp);
    Console.WriteLine("Teris sandardin kosindisi = {0}", so);
}
public static void maxmin(int[,] ma)
{
    int maxi = -100, mini = 150;
    for (int i = 0; i < 6; i++)
    for (int j = 0; j < 6; j++)
    {
        if (maxi < ma[i,j]) maxi = ma[i,j];
        if (mini > ma[i,j]) mini = ma[i,j];
    }
    Console.WriteLine("Maksimal san = {0}", maxi);
    Console.WriteLine("Minimal san = {0}", mini);
}
static void Main()
{
    int[,] a = new int[6, 6];
    int k = 0;
    string buf;
    while (k < 4)
    {
        Console.WriteLine("1 - Matrisanin kuru jane shigaru 6x6");
        Console.WriteLine("2 - On jane teris sandardin kosindisin
tabu");
        Console.WriteLine("3 - Matrisanin maksimal, minimal
sandin tabu");
        Console.WriteLine("4 - Bagdarlamadan shigu");
        Console.WriteLine("Menu punktini tandaniz");
        buf = Console.ReadLine();
        k = Convert.ToInt32(buf);
        switch (k)
        {
            case 1: sozd(a); break;
            case 2: polotr(a); break;

```

```
case 3: maxmin(a); break;
default: break;
}
}
}
}
}
```

**Бағдарлама жұмысы:**

- 1 - Matrisanin kuru jane shigaru 6x6
- 2 - On jane teris sandardin kosindisin tabu
- 3 - Matrisanin maksimal, minimal sandarin tabu
- 4 - Bagdarlamadan shigu

Menu punktın tandaniz

1

Matrisa kyrildi!!

-43 50 31 42 39 5

33 -9 40 25 -16 45

-43 19 26 -27 -17 -12

24 26 21 -13 -4 2

-11 15 1 -43 40 50

8 13 -41 -37 -21 13

- 1 - Matrisanin kuru jane shigaru 6x6
- 2 - On jane teris sandardin kosindisin tabu
- 3 - Matrisanin maksimal, minimal sandarin tabu
- 4 - Bagdarlamadan shigu

Menu punktın tandaniz

2

On sandardin kosindisi = 568

Teris sandardin kosindisi = -337

- 1 - Matrisanin kuru jane shigaru 6x6
- 2 - On jane teris sandardin kosindisin tabu
- 3 - Matrisanin maksimal, minimal sandarin tabu
- 4 - Bagdarlamadan shigu

Menu punktın tandaniz

3

Maksimal san = 50

Minimal san = -43

- 1 - Matrisanin kuru jane shigaru 6x6
- 2 - On jane teris sandardin kosindisin tabu
- 3 - Matrisanin maksimal, minimal sandarin tabu
- 4 - Bagdarlamadan shigu

Menu punktın tandaniz

Есепті шешу алгоритмін дайындау барысында есеп жеке бөліктерге бөлінді: матрицаны құру мен оны шығару процессі, барлық оң және теріс элементтердің қосындысын есептеу, матрицаның ең үлкен және ең кіші элементтерін табу. Аталған

процесстерді `static void Main()` әдісінде жеке статистикалық әдістер арқылы орындау бағдарламада менюді ұйымдастырады, бағдарлама кодының жеңіл және түсінікті болуына мүмкіндік береді.

Бағдарламада менюге арналған цикл меню тармағы таңдалғаннан кейін аяқталады (меню тармағы 4-ке тең).

Меню тармағының жұмысы `switch` операторы арқылы орындалады.

`switch` операторы мына пішімде жазылады:

`switch` (өрнек)

{

case таңдау тұрақтысы\_1 : [операторлар тізімі; ]

case таңдау тұрақтысы\_2 : [операторлар тізімі; ]

...

[ default операторлар тізімі; ]

}

`Switch` операторы екі бөлімнен тұрады. Оператордың бірінші бөлімінде `switch` қызметтік сөзі, ал одан кейін дөңгелек жақшаларда өрнек жазылады. Өрнек кез келген типте бола алады. Оны айқын емес түрде бүтін санды типтерге немесе `char` типіне түрлендіруге болады.

Екінші бөлімде `case` таңдау операторы арқылы бағдарламаны жалғастыру мүмкін нұсқалары анықталады. Егер `switch` сөзінен кейін өрнек 2-ге тең болса, онда бағдарламада тұрақтысы 2-ге тең `case` таңдау операторы ізделінеді және осы таңбадан (меткадан) кейін тұрған барлық операторлар орындалады.

Егер `case` таңдау операторының тұрақтысы мен `switch` операторынан кейін тұрған өрнектің мәні тең болмаса, онда бағдарлама `default` қызметтік сөзінен кейін тұрған операторларды немесе `switch` операторынан кейін орналасқан операторды орындауға көшеді.

«Операторлар тізімі» өту операторларымен аяқталуы тиіс, мысалы, `break`, `goto` немесе `return` операторлары.

Әдетте ең соңында `break` операторы қолданылады, ол `switch` операторын аяқтайды. Ол келесі таңдау операторының орындалуына жол бермейді.

Бірнеше таңдау операторларын ретті орындау үшін әдетте `goto` операторы қолданылады.

## 9.2 Көп өлшемді массивтер

Анықтама бойынша массивтер бір типтегі айнымалыларды және бір өлшемді массивтерді біріктіреді. Біріктірілген массивтер көп өлшемді массивтер деп аталады.

Егер біріктірілген массивтердің өлшемдері бірдей болса, онда ондай көп өлшемді массивтер «тіктөртбұрышты» массивтер деп аталады.

Егер біріктірілген массивтердің өлшемдері бірдей болмаса, онда ондай көп өлшемді массивтер «сынық» (`jagged`) массивтер деп аталады.

Тіктөртбұрышты массивтермен жұмыс жасауға арналған кейбір алгоритмдер алдыңғы есепте қарастырылды.

«Сынық» массивтер элементтерін құру және өңдеу процестерін қарастырайық.

9.2-есеп. Кәсіпорын иесінің 5 сауда орыны бар. Әрбір сауда орынында түрлі тауарлар саны 10 мен 30 арасындағы кездейсоқ сан болады. Әрбір тауардың құны 10 мен 70 ш.б. аралығындағы кездейсоқ сан болады. 5 сауда орынында өткізілетін тауар санын және олардың жалпы құнын табу керек.

Бағдарлама коды:

```
using System;
namespace ConsoleApplication1
{
    class Program
    {
        static void Main()
        {
            int[][] mtovar = new int[5][];
            int k = 0;
            int j, kol, cto;
            string buf;
            Random rnd = new Random();
            //Сынық массивті жариялау
            for (int i = 0; i < 5; i++)
            {
                j = rnd.Next() % 21 + 10;
                mtovar[i] = new int[j];
            }
            Console.WriteLine("Matrisa kurildi!!");
            kol = 0; cto = 0;
            // Сынық массивті құру және экранға шығару
            for (int i = 0; i < 5; i++)
            {
                kol = kol + mtovar[i].Length;
                for ( j = 0; j < mtovar[i].Length; j++)
                {
                    mtovar[i][j] = rnd.Next() % 61 + 10;
                    cto = cto + mtovar[i][j];
                    Console.Write(mtovar[i][j]+" ");
                }
                Console.WriteLine();
            }
            Console.WriteLine();
            Console.WriteLine("Tayar sani = {0}", kol);
            Console.WriteLine("Olardin bagasi = {0}", cto);
            Console.WriteLine("Enter pernesin basiniz");
            Console.ReadLine();
        }
    }
}
```

Бағдарлама жұмысы:

```
Matrisa kurildi!!
```

```
16 26 20 29 15 25 67 68 12 12 34 61 14 36 52 58 24 50 26
38 40 58 34 20 54 64 27
57 18
27 36 14 29 36 18 52 50 23 41 35 65 24 70 46 32 18 13 41
59
40 33 33 19 39 60 29 60 68 46 19 69 20 60 56 47
27 42 49 28 68 18 11 16 62 44 23 59 59 59 43 58 57 69 46
62 10 66
61 44 57 63 53 45 32 37 52 32 21 67 12 47 35 20
```

```
Tayar sani = 103
```

```
Olardin bagasi = 4136
```

```
Enter pernesin basiniz
```

Сынық массивті жариялаған кезде екінші индекс өлшемі көрсетілмейді, индекстер квадрат жақшаға алынады (үтір қолданылмайды), мысалы:

```
int[][] mtovar = new int[5][];
```

«Бағаналар» саны әр сынық массив «жолдарды» үшін динамикалық түрде анықталады:

```
j = rnd.Next() % 21 + 10;
```

```
mtovar[i] = new int[j];
```

Сынық массив элементтеріне әрбір индексті квадрат жақшаға орналастыру керек, мысалы:

```
mtovar[i][j] = rnd.Next() % 61 + 10;
```

### 9.3 Array массивтер класы

Айнымалыларды массивке біріктіру түрлі алгоритмдерде және деректерді өңдеу жүйелерінде қолданылады, мысалы, реляциялық деректер базасында немесе тізімдермен берілген массивтерде сұрыптау мен іздеу алгоритмдерінде. Сондықтан көптеген визуалды бағдарламалау орталарында массивтермен жұмыс істеуге арналған арнайы кластар бар

C# тілінде массивтермен жұмыс жасау үшін `System.Array` класы қолданылады. Осы класта массивтермен жұмыс істеуге арналған статикалық қасиеттер мен әдістер бар. Ең жиі қолданылатын қасиеттердің бірі - массивте элементтер санын есептейтін `Length` типіндегі қасиет. Мысалы, `masi` массивінің элементтер саны `masi.Length` арқылы анықталады. Бүтін типтегі `Rank` қасиет – массив өлшемін анықтауға мүмкіндік береді. `System.Array` класының ең жиі қолданылатын әдістері 9.1-кестеде көрсетілген.

9.1-кестесі – `System.Array` класының әдістері

| Әдіс       | Сипаттама   |
|------------|---|
| Static int | Екілік іздеу әдісі. Сұрыпталған бір өлшемді Array |

|  |   |
|--|---|
| BinarySearch (Array, object, IComparer);   | массивінде IComparer интерфейсі арқылы object элементін іздейді және элемент индексін қайтарады, егер элемент табылмаса теріс санды қайтарады.  |
| public static void CopyTo (Array, Index);  | Ағымдағы бір өлшемді массивтен барлық элементтерді Array массивіне Index индексінен бастап көшіреміз.   |
| public static Array CreateInstance(Type ElementsType, int[] Lengths, int[] LowerBounds); | Әрбір өлшемі бойынша элементтер саны Lengths және Индекстердің төменгі шегі LowerBounds болатын, ElementsType типіндегі элементтерден тұратын көп өлшемді массивті құрайды. Қайта жүктелетін әдістер индекстері 0-ден басталады бір өлшемді және екі өлшемді массивтерді құруға мүмкіндік береді. |
| public int GetLowerBound (Dimension);  | Dimension өлшемі бойынша индекстің ең кіші мәнін қайтарады.   |
| public int GetUpperBound (Dimension);  | Dimension өлшемінде индекстің ең үлен мәнін қайтарады.  |
| public static void Reverse (Array);  | Бір өлшемді Array массивінің элементтер тәртібін кері орналастырады.  |
| public static void Sort (Array);   | Бір өлшемді Array массивін сұрыптайды   |
| public static void Clear (Array, Index, Length);   | Массивті тазарту. Бір өлшемді Array массивінде Length элементтерін орналастырамыз. Элементтер типіне қарай Index элементінен бастап мәндері 0, false немесе null болады.  |
| public int GetLength (Dimension);  | Dimension өлшемі бойынша массив элементтерінің санын қайтарады.   |

Есепті шығару үшін System.Array класын қолданып, бағдарламаны жазу процесін қарастырайық. Сонымен қатар класс әдістерінің массиві құру процестері және осы массив пен әдеттегі жолмен құрылған массив элементтерін өңдеу үшін класс әдістерін қолдану қарастырылады.

9.3-есеп. Минус 40 пен 40 аралығындағы кездейсоқ бүтін сандардан тұратын A массивін құру. Массивті шығару. Массивтің барлық элементтерін сұрыптау. Құрылған массив элементтерін өлшемі жағынан бірдей басқа массивке көшіру. Онда диалог режимінде берілген кілт бойынша элементтің бинарлық іздестірілуін орындау. Бағдарламаны жазғанда System.Array класының әдістерін қолдану.

Бағдарлама коды:

```
using System;
namespace ConsoleApplication1
{
    class Program
    {
```

```

public static int[] ctl = { 15 };
public static int[] niz = { 1 };
public static Array masi =
Array.CreateInstance(typeof(int), ctl, niz);
public static void sozd()
{
Random rnd = new Random();
Console.WriteLine("Massiv kurildi!!: ");
for (int i = masi.GetLowerBound(0); i <= masi.Length; i++)
masi.SetValue((rnd.Next(81) - 40), i);
for (int i = masi.GetLowerBound(0); i <= masi.Length; i++)
Console.Write(masi.GetValue(i) + " ");
Console.WriteLine();
}
static void Main()
{
int[] a = new int[15];
int k = 0;
string buf;

sozd();
Console.WriteLine("Massivti sorttaimiz jane shigaramiz: ");
Array.Sort(masi);
foreach (int i in masi)
Console.Write(i + " ");
Console.WriteLine();
Console.WriteLine("Massivti bytin sandardan tyratin
massivke koshiremiz: ");
Array.Copy(masi, a, masi.Length);
foreach (int i in a)
Console.Write(i + " ");
Console.WriteLine();

//массивті реверстейміз
Array.Reverse(a);
Console.WriteLine("Shana massivti reverstey jane shigary:
");
foreach (int i in a)
Console.Write(i + " ");
Console.WriteLine();
//санды кілт бойынша бинарлы іздеу
Console.WriteLine("Kilt boinsha massivte sandi binarli
izdey");
Console.WriteLine("Izdey kiltin engiziniz");
buf = Console.ReadLine();
k = Convert.ToInt32(buf);

```



```

int idx = Array.BinarySearch(masi, k);
Console.WriteLine("Sannin indeksi = " + idx);
Console.WriteLine("jok kiltti engiziniz");
buf = Console.ReadLine();
k = Convert.ToInt32(buf);
idx = Array.BinarySearch(masi, k);
Console.WriteLine("Sannin indeksi = " + idx);
Console.ReadLine();
}
}
}

```

### Бағдарлама жұмысы:

Massiv kurildi!!!:

-15 -31 -2 -31 20 13 -30 12 19 25 -24 -32 -22 3 19

Massivti sorttaimiz jane shigaramiz:

-32 -31 -31 -30 -24 -22 -15 -2 3 12 13 19 19 20 25

Massivti bytin sandardan tyratin massivke koshiremiz:

-32 -31 -31 -30 -24 -22 -15 -2 3 12 13 19 19 20 25

Shana massivti reverstey jane shigary:

25 20 19 19 13 12 3 -2 -15 -22 -24 -30 -31 -31 -32

Kilt boinsha massivte sandi binarli izdey

Izdey kiltin engiziniz

12

Sannin indeksi = 10

jok kiltti engiziniz

18

Sannin indeksi = -13

`Array.CreateInstance(typeof( int),ctl,niz);` әдісі арқылы 15 элементтен тұратын бір өлшемді `masi` массивін құраймыз (`Lengths` қасиеті `ctl` айнымалысына тең), бірінші `LowerBounds` индексі 1-ге тең (индекс мәні 1-ден үлкен бола алады, бірақ 0-ге тең емес).

For циклінде `masi.GetLowerBound(0)` қолданылады. Айнымалының бастапқы мәні 1-ге, ал соңғы мәні – `masi.Length` 15-ке тең.

```

for (int i = masi.GetLowerBound(0); i <= masi.Length;
i++)
masi.SetValue((rnd.Next(81) - 40), i);

```

Массив элементтерін қолдану `System.Array` класының әдістері арқылы ғана мүмкін: жазуда – `masi.SetValue` және оқуды орындағанда – `masi.GetValue`.

Массивті сұрыптауды `Array.Sort(masi);` статикалық әдісімен орындаймыз және оны `foreach` циклі арқылы монитор экранына шығарамыз:

```

foreach (int i in masi)
Console.Write(i + " ");
Console.WriteLine();

```

`System.Array` класының әдістері арқылы құрылған массивті `a` массивіне көшіреміз:

```
Array.Copy(masi, a, masi.Length);
```

Ары қарай `a` массиві үшін `System.Array` класының реверс әдісі мен бинарлық іздестіруді қолдану қарастырылған.

9.4-есебі. 0 мен 20 аралығындағы кездейсоқ бүтін сандардан тұратын 5x5 матрицасын құрып, оны шығару керек. Матрицаны 25 элементтен тұратын массивке қайта жазып, осы массивтің барлық элементтерін сұрыптау керек. Массивті матрицаға кері жазуды орындаңыз, сұрыпталған матрицаны шығарыңыз.

Бағдарлама коды:

```
using System;
namespace ConsoleApplication1
{
    class Program
    {
        public static Array masi =
Array.CreateInstance(typeof(int), 5, 5);
        public static void sozd()
        {
            int kk;
            Random rnd = new Random();
            Console.WriteLine("Matrizani kyrildi: ");
            for (int i = 0; i <= 4; i++)
            for (int j = 0; j <= 4; j++)
            {
                kk = rnd.Next(21);
                masi.SetValue(kk, i, j);
            }
            for (int i = 0; i <= 4; i++)
            {
                for (int j = 0; j <= 4; j++)
                    Console.Write(masi.GetValue(i, j) + "\t");
                Console.WriteLine();
            }
        }
        static void Main()
        {
            int[] a = new int[25];
            int k = 0;
            string buf;
            sozd();
            Console.WriteLine("Matrizani massivke koshiry, ekranga
shigary: ");
            int j = 0;
            foreach (int i in masi)
            {
```

```

Console.Write(i + " "); a[j] = i; j++;
}
Console.WriteLine();
Console.WriteLine("Massivti syriptay, ekranga shigary: ");
Array.Sort(a);
foreach (int i in a)
Console.Write(i + " ");
Console.WriteLine();
Console.WriteLine("Massivti matrizaga koshiry: ");
k = 0;
for (int i = 0; i <= 4; i++)
{
for (j = 0; j <= 4; j++)
{
masi.SetValue(a[k], i, j); k++;
Console.Write(masi.GetValue(i, j) + "\t");
}
Console.WriteLine();
}
Console.ReadLine();
}
}
}

```

**Бағдарлама жұмысы:**

Matrizani kырildi:

2 9 3 17 8

4 14 12 17 11

9 18 8 7 5

16 7 15 6 12

4 2 15 6 16

Matrizani massivke koshiry, ekranga shigary:

2 9 3 17 8 4 14 12 17 11 9 18 8 7 5 16 7 15 6 12 4 2 15

6 16

Massivti syriptay, ekranga shigary:

2 2 3 4 4 5 6 6 7 7 8 8 9 9 11 12 12 14 15 15 16 16 17

17 18

Massivti matrizaga koshiry:

2 2 3 4 4

5 6 6 7 7

8 8 9 9 11

12 12 14 15 15

16 16 17 17 18

Екі өлшемді массив элементтерін құру және өңдеу үшін бағдарламада System.Array класының әдістері қолданылған. System.Array класының кейбір әдістерін тек бір өлшемді массивтер элементтеріне ғана қолдануға болады, мысалы,

элементтерді сұрыптау. Сондықтан бағдарламада кең таралған жолы қарастырылады: өңдеуді таңдау әдісімен орындау, екі өлшемді массив элементтерін бір өлшемді массивке қайтадан жазу және керісінше бір өлшемді массив элементтерін екі өлшемді массивке жазу.

## 9.4 Өзін-өзі тексеру сұрақтары

1 Switch операторы не үшін қолданылады?

2 Switch<өрнек> операторында <өрнек> не үшін қолданылады?

3 Here Switch операторында case таңбасының (метка) әрекеті break операторымен аяқталуы керек?

4 a матрицасына неше сан жазуға болады, егер `int[,] a = new int[3,4];`?

5 Бағдарламаның келесі үзіндісі нені орындайды?

```
for (int i = 0; i < 6; i++)
{
    for (int j = 0; j < 6; j++)
        Console.Write(ma[i, j]+"\t");
    Console.WriteLine();
}
```

6 Бағдарламаның келесі үзіндісі нені орындайды?

```
for ( i = 0; i < 9; i++)
for ( j = i+1; j < 10; j++)
{ b = a[i,j]; a[i,j] = a[j,i]; a[j,i] = b; }
```

7 Бағдарламаның келесі үзіндісі нені орындайды?

```
n = 0;
for (j = 0; j <= 10; j++)
for (I = 0; I <= 10; I++)
if (A[I, j] >= 0) n++;
```

8 Бағдарламаның келесі үзіндісі нені орындайды?

```
for (I = 0; I <= 10; I++)
if (A[I,10-I] > 0) A[I,10-I] = 0;
```

9 Бағдарламаның келесі үзіндісі нені орындайды?

```
for (I = 0; I <= 10; I++)
for (j = 0; j < 10; j++)
for (k = j +1; k <= 10; k++)
if (a[I,j] > a[I,k])
{ b = a[I,j]; a[I,j] = a[I,k]; a[I,k] = b; }
```

10 Бағдарламаның келесі үзіндісі нені орындайды?

```
for (I = 0; I <= 10; I++)
if (A[I,I] > 0) A[I,I] = 0;
```

