

Bölüm 7: Çokboyutlu Diziler (Multidimensional Arrays)



Motivasyon

Şimdiye kadar, doğrusal yapıları modellemek için tek boyutlu diziler kullanıldı. Bir matrisi veya tabloyu göstermek için iki boyutlu dizi kullanılabilir. Örneğin, şehirler arasındaki mesafeleri açıklayan aşağıdaki tablo iki boyutlu bir dizi kullanılarak oluşturulmuştur.

Distance Table (in miles)							
	Chicago	Boston	New York	Atlanta	Miami	Dallas	Houston
Chicago	0	983	787	714	1375	967	1087
Boston	983	0	214	1102	1763	1723	1842
New York	787	214	0	888	1549	1548	1627
Atlanta	714	1102	888	0	661	781	810
Miami	1375	1763	1549	661	0	1426	1187
Dallas	967	1723	1548	781	1426	0	239
Houston	1087	1842	1627	810	1187	239	0

Amaçlar

- ❑ İki boyutlu dizi kullanarak, verileri gösteren örnekler verme(§ 7.1).
- ❑ İki boyutlu diziler için değişken bildirmek, diziler oluşturmak ve satır ve sütun dizinlerini kullanarak iki boyutlu bir dizide dizi öğelerine erişmek(§ 7.2).
- ❑ İki boyutlu diziler için ortak işlemleri programlamak (dizileri göstermek, tüm öğeleri toplamak, min ve max elemanlarını bulmak ve rastgele karıştırmak)(§ 7.3).
- ❑ İki boyutlu dizilerin metodlara geçişi (§ 7.4).
- ❑ İki boyutlu dizileri kullanarak çoktan seçmeli soruları derecelendirmek için bir program yazmak(§ 7.5).
- ❑ İki boyutlu dizileri kullanarak en yakın çift problemini çözmek(§ 7.6).
- ❑ İki boyutlu dizileri kullanarak bir Sudoku çözümünü kontrol etmek
- ❑ Çokboyutlu dizileri kullanmak(§ 7.8).



İki boyutlu dizi bildirimi/oluşturma

```
//refvar adında dizi bildirimi  
dataType[][] refVar;
```

```
// Dizi oluşturma ve değişkene referansını atama  
refVar = new dataType[10][10];
```

```
// Tek ifadede zizi bildirim ve oluşturma  
dataType[][] refVar = new dataType[10][10];
```

```
// Alternatif syntax  
dataType refVar[][] = new dataType[10][10];
```



İki boyutlu dizinin değişkenlerinin bildirimi ve iki boyutlu dizi oluşturma

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

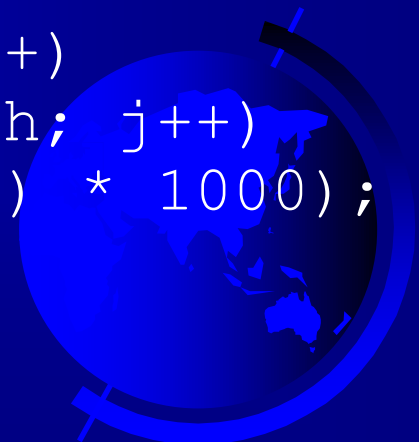
veya

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

```
matrix[0][0] = 3;
```

```
for (int i = 0; i < matrix.length; i++)  
    for (int j = 0; j < matrix[i].length; j++)  
        matrix[i][j] = (int) (Math.random() * 1000);
```

```
double[][] x;
```



İki boyutlu dizi örnekleri

	0	1	2	3	4
0					
1					
2					
3					
4					

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

	0	1	2	3	4
0					
1					
2		7			
3					
4					

```
matrix[2][1] = 7;
```

	0	1	2
0	1	2	3
1	4	5	6
2	7	8	9
3	10	11	12

```
int[][] array = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

matrix.length? 5

matrix[0].length? 5

array.length? 4

array[0].length? 3

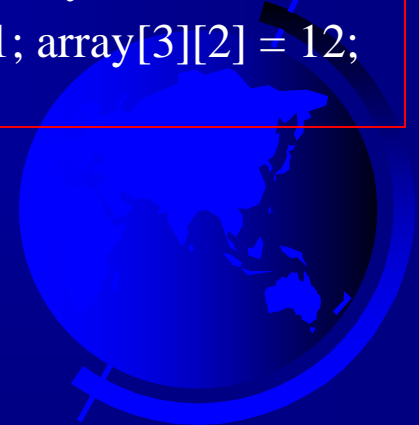
Kısayol Notasyonlarını Kullanarak Bildirim, Oluşturma ve Başlatma

İki boyutlu bir dizi bildirmek, oluşturmak ve başlatmak için bir dizi başlatıcısı da kullanabilirsiniz. Örneğin,

```
int[][] array = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

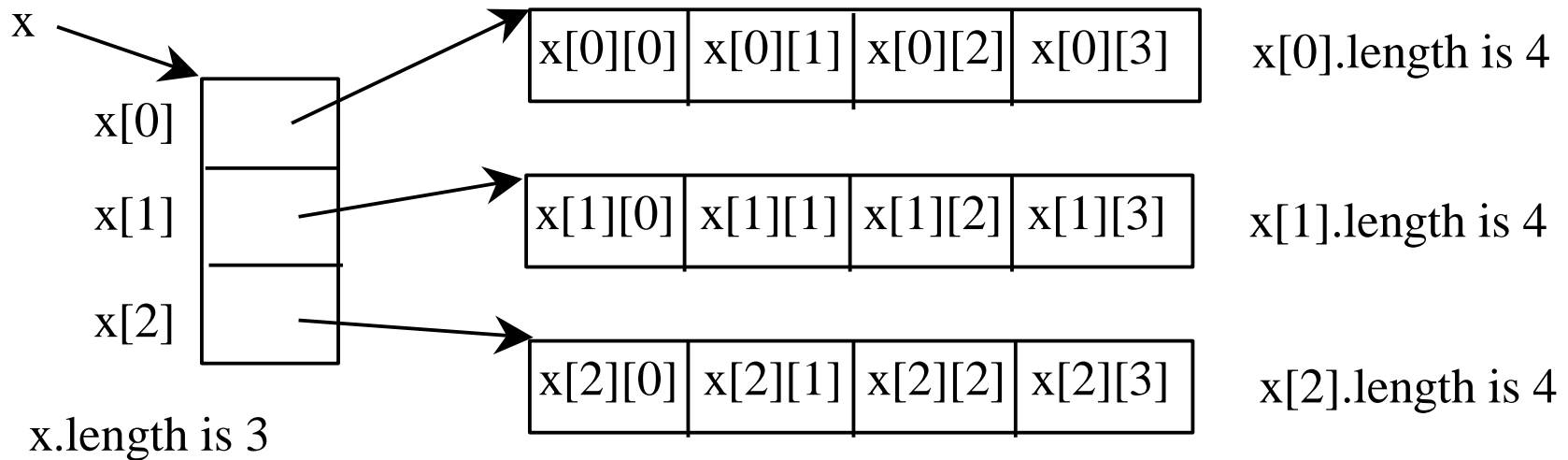
Aynı

```
int[][] array = new int[4][3];  
array[0][0] = 1; array[0][1] = 2; array[0][2] = 3;  
array[1][0] = 4; array[1][1] = 5; array[1][2] = 6;  
array[2][0] = 7; array[2][1] = 8; array[2][2] = 9;  
array[3][0] = 10; array[3][1] = 11; array[3][2] = 12;
```



İki boyutlu dizilerin uzunlukları

```
int[][] x = new int[3][4];
```



İki boyutlu dizilerin uzunlukları

```
int[][] array = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9},  
    {10, 11, 12}  
};
```

array.length

array[0].length

array[1].length

array[2].length

array[3].length

array[4].length

ArrayIndexOutOfBoundsException



Düzensiz Diziler

İki boyutlu bir dizideki her satırın kendisi bir dizidir.

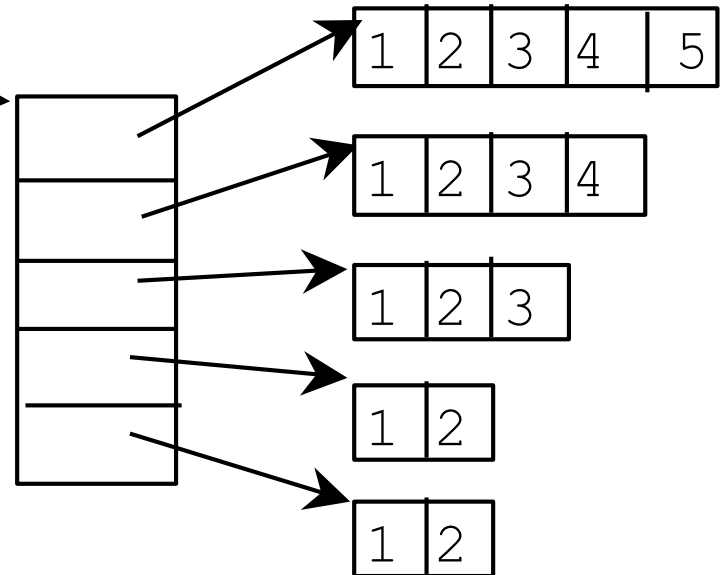
Böylece, satırların farklı uzunlukları olabilir. Böyle bir dizi düzensiz bir dizi olarak bilinir. Örneğin,

```
int[][] matrix = {  
    {1, 2, 3, 4, 5},  
    {2, 3, 4, 5},  
    {3, 4, 5},  
    {4, 5},  
    {5}  
};
```

```
matrix.length is 5  
matrix[0].length is 5  
matrix[1].length is 4  
matrix[2].length is 3  
matrix[3].length is 2  
matrix[4].length is 1
```

Düzensiz Diziler

```
int[][] triangleArray = {  
    {1, 2, 3, 4, 5},  
    {2, 3, 4, 5},  
    {3, 4, 5},  
    {4, 5},  
    {5}  
};
```



İki Boyutlu Dizileri İşleme

Metindeki örneklere bakalım.

1. (Girdi değerlerine sahip dizileri başlatma)
2. (Dizileri yazdırma)
3. (Elemanları toplama)
4. (Sütun elemanlarını toplama)
5. (Hangi satırda en büyük toplam olduğunu bulma)
6. (En büyük elemanın en küçük indisini bulmak)
7. (Rastgele karıştırma (*Random shuffling*))



Girdi değerlerine sahip dizileri başlatma

```
java.util.Scanner input = new Scanner(System.in);
System.out.println("Enter " + matrix.length + " rows and " +
    matrix[0].length + " columns: ");
for (int row = 0; row < matrix.length; row++) {
    for (int column = 0; column < matrix[row].length; column++) {
        matrix[row][column] = input.nextInt();
    }
}
```



Rastgele değerler ile dizi başlatma

```
for (int row = 0; row < matrix.length; row++) {  
    for (int column = 0; column < matrix[row].length; column++) {  
        matrix[row][column] = (int)(Math.random() * 100);  
    }  
}
```



Dizileri yazdırma

```
for (int row = 0; row < matrix.length; row++) {  
    for (int column = 0; column < matrix[row].length; column++) {  
        System.out.print(matrix[row][column] + " ");  
    }  
  
    System.out.println();  
}
```



Tüm elemanları toplama

```
int total = 0;
for (int row = 0; row < matrix.length; row++) {
    for (int column = 0; column < matrix[row].length; column++) {
        total += matrix[row][column];
    }
}
```



Sütun elemanlarını toplama

```
for (int column = 0; column < matrix[0].length; column++) {  
    int total = 0;  
    for (int row = 0; row < matrix.length; row++)  
        total += matrix[row][column];  
    System.out.println("Sum for column " + column + " is "  
        + total);  
}
```



Rasgele karıştırma

```
for (int i = 0; i < matrix.length; i++) {  
    for (int j = 0; j < matrix[i].length; j++) {  
        int i1 = (int)(Math.random() * matrix.length);  
        int j1 = (int)(Math.random() * matrix[i].length);  
        // Swap matrix[i][j] with matrix[i1][j1]  
        int temp = matrix[i][j];  
        matrix[i][j] = matrix[i1][j1];  
        matrix[i1][j1] = temp;  
    }  
}
```

Metodlara iki boyutlu dizi geçişi

PassTwoDimensionalArray

Run

```
public class PassTwoDimensionalArray {  
public static void main(String[] args) {  
Scanner input = new Scanner(System.in); // Enter array values  
    int[][] m = new int[3][4];  
    System.out.println("Enter " + m.length + " rows and " + m[0].length + " columns: ");  
    for (int i = 0; i < m.length; i++)  
        for (int j = 0; j < m[i].length; j++)  
            m[i][j] = input.nextInt();  
    // Display result  
    System.out.println("\nSum of all elements is " + sum(m));  
}  
public static int sum(int[][] m) {  
    int total = 0;  
    for (int row = 0; row < m.length; row++) {  
        for (int column = 0; column < m[row].length; column++) {  
            total += m[row][column];  
        }  
    }  
    return total;  
}}
```



Problem:Çoktan seçmeli test notu

- Amaç: çoktan seçmeli test sonuçlarını puanlayan program

Students' Answers to the Questions:

0 1 2 3 4 5 6 7 8 9

Student 0	A	B	A	C	C	D	E	E	A	D
Student 1	D	B	A	B	C	A	E	E	A	D
Student 2	E	D	D	A	C	B	E	E	A	D
Student 3	C	B	A	E	D	C	E	E	A	D
Student 4	A	B	D	C	C	D	E	E	A	D
Student 5	B	B	E	C	C	D	E	E	A	D
Student 6	B	B	A	C	C	D	E	E	A	D
Student 7	E	B	E	C	C	D	E	E	A	D

Key

Key to the Questions:

0 1 2 3 4 5 6 7 8 9

D B D C C D A E A D

GradeExam

Run

```

public class GradeExam {
    /** Main method */
    public static void main(String args[]) {
        // Students' answers to the questions
        char[][] answers = {
            {'A', 'B', 'A', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},
            {'D', 'B', 'A', 'B', 'C', 'A', 'E', 'E', 'A', 'D'},
            {'E', 'D', 'D', 'A', 'C', 'B', 'E', 'E', 'A', 'D'},
            {'C', 'B', 'A', 'E', 'D', 'C', 'E', 'E', 'A', 'D'},
            {'A', 'B', 'D', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},
            {'B', 'B', 'E', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},
            {'B', 'B', 'A', 'C', 'C', 'D', 'E', 'E', 'A', 'D'},
            {'E', 'B', 'E', 'C', 'C', 'D', 'E', 'E', 'A', 'D'}};

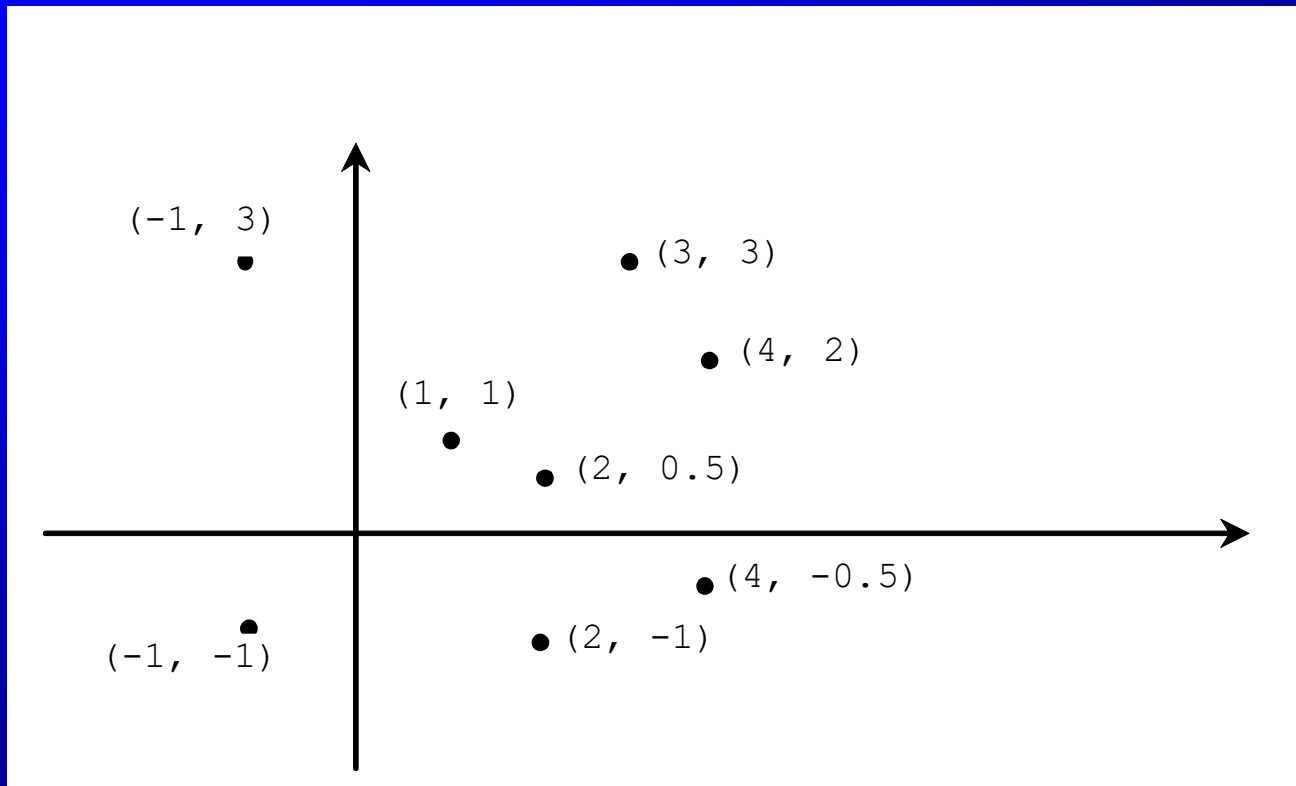
        // Key to the questions
        char[] keys = {'D', 'B', 'D', 'C', 'C', 'D', 'A', 'E', 'A', 'D'};

        // Grade all answers
        for (int i = 0; i < answers.length; i++) {
            // Grade one student
            int correctCount = 0;
            for (int j = 0; j < answers[i].length; j++) {
                if (answers[i][j] == keys[j])
                    correctCount++;
            }

            System.out.println("Student " + i + "'s correct count is " +
                               correctCount);
        }
    }
}

```

Problem: Birbirine En Yakın İki Nokta Bulma



	x	y
0	-1	3
1	-1	-1
2	1	1
3	2	0.5
4	2	-1
5	3	3
6	4	2
7	4	-0.5

FindNearestPoints

Run

Sudoku nedir?

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



Her satır 1'den 9'a kadar sayı içerir

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

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

Her sütun 1'den 9'a kadar sayı içerir

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

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

Her 3×3 'lük kutucuk 1'den 9'a kadar sayı içerir

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

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



Çözümün doğruluğunu kontrol etme

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

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

CheckSudokuSolution

Run



Çok boyutlu diziler

Bazen, n boyutlu veri yapılarını göstermek gerekebilir. Java'da, n (bir tamsayı) boyutlu diziler oluşturabilirsiniz.

İki boyutlu dizi değişkenleri bildirme ve iki boyutlu diziler oluşturma yolu, n -boyutlu dizi değişkenlerini bildirmek ve $n \geq 3$ için n boyutlu diziler oluşturmak için genelleştirilebilir. Örneğin, aşağıdaki sözdizimi, skores üç boyutlu bir dizi değişkeni bildiren bir dizi oluşturur ve skorları referans olarak atar.

```
double[][][] scores = new double[10][5][2];
```



Problem: Toplam Puan Hesaplama

- ❑ Amaç: Bir sınıftaki öğrencilerin toplam puanını hesaplayan bir program yazmak.
- ❑ Puanların 3-boyutlu dizilerde tutulduğunu varsayalım. **scores** adı verilen 3 boyutlu dizide birinci indis öğrenciyi, ikinci indis bir sınavı ve üçüncüsü ise sınavın bir kısmını ifade eder. 7 öğrenci, 5 sınav olduğunu ve her sınavın iki bölümden oluştuğunu varsayalım - çoktan seçmeli bölüm ve programlama bölümü Bu yüzden, scores[i][j][0] i. öğrencinin j. sınavındaki çoktan seçmeli bölümündeki puanını göstermektedir. Program her bir öğrenci için toplam puanı göstermelidir.

TotalScore

Run

```

public class TotalScore {
public static void main(String args[]) {
    double[][][] scores = {
        {{7.5, 20.5}, {9.0, 22.5}, {15, 33.5}, {13, 21.5}, {15, 2.5}},
        {{4.5, 21.5}, {9.0, 22.5}, {15, 34.5}, {12, 20.5}, {14, 9.5}},
        {{6.5, 30.5}, {9.4, 10.5}, {11, 33.5}, {11, 23.5}, {10, 2.5}},
        {{6.5, 23.5}, {9.4, 32.5}, {13, 34.5}, {11, 20.5}, {16, 7.5}},
        {{8.5, 26.5}, {9.4, 52.5}, {13, 36.5}, {13, 24.5}, {16, 2.5}},
        {{9.5, 20.5}, {9.4, 42.5}, {13, 31.5}, {12, 20.5}, {16, 6.5}},
        {{1.5, 29.5}, {6.4, 22.5}, {14, 30.5}, {10, 30.5}, {16, 6.0}}};
    for (int i = 0; i < scores.length; i++) {
        double totalScore = 0;
        for (int j = 0; j < scores[i].length; j++)
            for (int k = 0; k < scores[i][j].length; k++)
                totalScore += scores[i][j][k];
        System.out.println("Student " + i + "'s score is " + totalScore);
    }
}

```

