# Unit 3—Arrays, Recursion, and Complexity

# Chapter 10— Introduction to Arrays

EXERCISE 10.1

1. An array can contain many items and still be treated as one thing. Thus, instead of having many variables for many items, an array requires just one variable and an operator to access each item within it.
2. To access an item in an array, the programmer uses the subscript [].



exercise 10.2

1. a. 23

b. 12

c. 156

1. Because each of these positions is outside of the range of allowable positions for this array, a range-bound error occurs, and Java throws an array index out of bounds exception.

EXERCISE 10.3

for (int i = 0; i < a.length; i++)

System.out.println(a[i]);

for (int i = a.length - 1; i >= 0; i--)

System.out.println(a[i]);

int pos = 0;

for (pos = 0; pos < a.length; pos++)

if (a[pos] < 0)

break;

1. a. Replaces all the numbers in the array with their absolute values.

b. Builds a new string with the items in the array. These items can be anything.

1. You can use the variable length when you want to visit all the items in an array. This reference does the job for any array, so its use makes the code more portable and maintainable.

EXERCISE 10.4

1. a. double[] doubleArray = new double[15];

b. String[] stringArray = new String[20];

1. An initializer list allows the program to specify the values that initially occupy positions in an array. An example of its use is int[] a = {2, 4, 6};
2. a. int[] scores = {100, 90, 75, 60, 88};

b. double[] rates = {0.12, 0.05, 0.15};

c. String[] names = {"Martin", "Osborne"};

1. It is better to use the form <type>[] <variable> when declaring an array so as not to confuse it with the form <type> variable when declaring ordinary variables.

EXERCISE 10.5

1. When a programmer tries to access an item in an array cell whose index is greater than or equal to the array’s logical size, this item is garbage (not currently part of the useful data in the program). The value returned could be an arbitrary number if we have an array of numbers or even null if we have an array of objects.
2. An application that inputs an arbitrary number of data items from a user would need an array that could accommodate all the items. If the number of items varies, so will the logical size of the array, and the array will not necessarily be full.

EXERCISE 10.6

1. Parallel arrays are two or more arrays that have the same length and use the same index values to establish logical connections among their items.
2. One might use parallel arrays to represent a phone book. The names occupy one array and the phone numbers occupy the other array.

String[] names = new String[50];

int[] ages = new int[50];

String[] ssn = new String[50];

for (int i = 0; i < names.length; i++)

System.out.printf("%-20s%3d", names[i], numbers[i]);

int expos = new int[10];

int powers = new int[10];

for (int i = 0; i < expos.length; i++){

expos[i] = i;

powers[i] = Math.pow(2, i);

}

EXERCISE 10.7

1. a. for (String str : abc)

System.out.println(str);

b. for (String str : abc)

if (target.equals(str)){

found = true;

break;

}

1. An enhanced for loop would not work when traversing an array from the last position to the first position. It would also not work when replacing an element at a given position in an array.

EXERCISE 10.8

1. The assignment operator makes the two variables refer to the same array object.
2. As with other objects, arrays are not copied when the assignment operator is used. Thus, one must instantiate a new array of the same type and length as the original, transfer the items to the new array, and return the new array.

double average(double[]a){

double sum = 0;

for (int i : a)

sum += i;

return sum / a.length;

}

int[] subArray(int[] a, int first, int last){

int[] result = new int[last – first + 1];

for (int i = first; i <= last; i++)

result[i – first] = a[i];

return result;

}

Point search(int[][] a, int target){

int row = 0;

int col = 0;

for (row = 0; row < a.length; row++)

for (col = 0; col < a[row].length; col++)

if (a[row][col] == target)

return new Point(row, col);

return new Point(row, col);

}

EXERCISE 10.9

1. Student getHighStudent(Student[] students){

Student highest = null;

for (int i = 0; i < studentCount; i++)

if (highest == null ||

students[i].getHighScore() > highest.getHighScore())

highest = students[i];

return highest;

}

1. The JVM will throw an array index out of bounds exception after the scores for the five students are displayed. The reason this happens is that the elements in the array after position 4 are null, and the array’s length is used in the output loop.

EXERCISE 10.10

1. private class AveScoreListener implements ActionListener{

public void actionPerformed(ActionEvent e){

if (model.size() == 0){

JOptionPane.showMessageDialog(TestScoresView.this,

"No student is available");

return;

}

int ave = model.getClassAverage();

JOptionPane.showMessageDialog(TestScoresView.this,

"The average score is " + ave);

}

}

2. The advantage of using a single listener class for all button events is that only a single instance needs to be created and attached to all the buttons. The disadvantages are that the actionPerformed method consists of a complex multiway if statement to handle all of the cases, and the responsibilities for handling many tasks are located in one object.

Review Questions

## Written Questions

1. a. Valid

b. Valid

c. Invalid

d. Invalid

e. Invalid

f. Valid

g. Valid

h. Invalid

i. Invalid

j. Valid

2.

a. Valid, 6

b. Valid, 3

c. Valid, 7

d. Invalid, index position out of range

e. Valid, 9

f. Invalid, index position must be an integer

3.

a. An array of int variable is declared, but is set to an array of double.

b. A double, 1.5, cannot be used to specify the length of the array.

c. A negative number, –10, cannot be used to specify the length of the array.

d. A one-dimensional array object cannot be assigned to a variable of a two-dimensional array type.

4.

int selectRandom(int [] a){

Random rand = new Random();

int index = rand.nextInt(a.length));

return a[index];

}

1. int[][] a = new int[5][4];
2. for (int i = 0; i < a.length; i++)

for (int j = 0; j < a[i].length; j++)

a[i][j] = 1 + (int) (math.random() \* 20);