

# COMPTE RENDU LAB6

## TestArrays

### Code

```
13 public class TestArray {
14
15     /**
16      * Print an array of integers
17      * on the standard output.
18      * Format: [value0, value1, ..., valueN]
19      *
20      * @param arrayOfInt the specified array of integers
21      */
22     public static void print(int [] arrayOfInt) {
23         for(int i = 0; i < arrayOfInt.length; ++i)
24             System.out.print(arrayOfInt[i]+", ");
25         System.out.println();
26     }
27
28     /**
29      * Returns the maximum value
30      * of the specified array of integers
31      *
32      * @param arrayOfInt the specified array of integers
33      */
34     public static int getMaxValue(int[] arrayOfInt) {
35         int m = Integer.MIN_VALUE;
36         for(int i = 0; i < arrayOfInt.length; ++i)
37             if(arrayOfInt[i]>m) m = arrayOfInt[i];
38         return m;
39     }
40
41     /**
42      * Returns the minimum value
43      * of the specified array of integers
44      *
45      * @param arrayOfInt the specified array of integers
46      */
47     public static int getMinValue(int[] arrayOfInt) {
48         int m = Integer.MAX_VALUE;
49         for(int i = 0; i < arrayOfInt.length; ++i)
50             if(arrayOfInt[i]<m) m = arrayOfInt[i];
51         return m;
52     }
53 }
```

```
52  /**
53  *
54  * Sorts the specified array of objects
55  * into ascending order
56  *
57  * @param arrayOfInt the specified array of integer
58  */
59  public static void sortAscending(int[] arrayOfInt){
60      boolean sorted = false;
61      int temp;
62      while(!sorted) {
63          sorted = true;
64          for (int i = 0; i < arrayOfInt.length - 1; i++) {
65              if (arrayOfInt[i] > arrayOfInt[i+1]) {
66                  temp = arrayOfInt[i];
67                  arrayOfInt[i] = arrayOfInt[i+1];
68                  arrayOfInt[i+1] = temp;
69                  sorted = false;
70              }
71          }
72      }
73  }

74  /**
75  *
76  * Sorts the specified array of objects
77  * into descending order
78  *
79  * @param arrayOfInt the specified array of integer
80  */
81  public static void sortDescending(int[] arrayOfInt){
82      boolean sorted = false;
83      int temp;
84      while(!sorted) {
85          sorted = true;
86          for (int i = 0; i < arrayOfInt.length - 1; i++) {
87              if (arrayOfInt[i] < arrayOfInt[i+1]) {
88                  temp = arrayOfInt[i];
89                  arrayOfInt[i] = arrayOfInt[i+1];
90                  arrayOfInt[i+1] = temp;
91                  sorted = false;
92              }
93          }
94      }
95  }
```

```
100 public static void main(String[] args) {
101     int [] arrayOfInt = new int[10];
102     System.out.println(arrayOfInt.length);
103     System.out.println(arrayOfInt);
104     System.out.println(arrayOfInt[6]);
105     try {
106         System.out.println(arrayOfInt[10]);
107     } catch (ArrayIndexOutOfBoundsException error) {
108         System.out.println("ArrayIndexOutOfBoundsException cached");
109         error.printStackTrace();
110     }
111
112     Random randomInt = new Random();
113     for(int i = 0; i < arrayOfInt.length; ++i)
114         arrayOfInt[i] = randomInt.nextInt(100);
115     print(arrayOfInt);
116
117     System.out.println("Max: "+getMaxValue(arrayOfInt));
118     System.out.println("Min: "+getMinValue(arrayOfInt));
119     System.out.println("Sorted in ascending order:");
120     sortAscending(arrayOfInt);
121     print(arrayOfInt);
122     System.out.println("Sorted in descending order:");
123     sortDescending(arrayOfInt);
124     print(arrayOfInt);
125     System.out.println("...Using java methods...");
126     System.out.println("Max: "+Arrays.stream(arrayOfInt).max().getAsInt());
127     System.out.println("Min: "+Arrays.stream(arrayOfInt).min().getAsInt());
128     System.out.println("Sorted in ascending order:");
129     Arrays.sort(arrayOfInt);
130     print(arrayOfInt);
131     System.out.println("Sorted in descending order:");
132     for (int i = 0; i < arrayOfInt.length / 2; i++) {
133         int temp = arrayOfInt[i];
134         arrayOfInt[i] = arrayOfInt[arrayOfInt.length - 1 - i];
135         arrayOfInt[arrayOfInt.length - 1 - i] = temp;
136     }
137     print(arrayOfInt);
138 }
139 }
```

## Output

```
10
[I@4dc63996
0
ArrayIndexOutOfBoundsException cached
java.lang.ArrayIndexOutOfBoundsException: Index 10 out of bounds for length 10
    at gse4.labs.java.TestArray.main(TestArray.java:106)
61, 94, 85, 37, 93, 54, 2, 1, 1, 45,
Max: 94
Min: 1
Sorted in ascending order:
1, 1, 2, 37, 45, 54, 61, 85, 93, 94,
Sorted in descending order:
94, 93, 85, 61, 54, 45, 37, 2, 1, 1,
...Using java methods...
Max: 94
Min: 1
Sorted in ascending order:
1, 1, 2, 37, 45, 54, 61, 85, 93, 94,
Sorted in descending order:
94, 93, 85, 61, 54, 45, 37, 2, 1, 1,
```

On remarque que l'utilisation des méthodes Java donne les même résultats. Pour simplifier le code et accélérer le développement, on va donc préférer utiliser ces méthodes.

## TwoDimArrayDemo

### Code

```
12 public class TwoDimArrayDemo {
13
14     /**
15      * Prints a 2-dimensional array of integers (matrix)
16      * on the standard output.
17      * Format:
18      * [[value00, value01, ..., value0N],
19      * [value10, value11, ..., value1N],
20      * .....
21      * [valueM0, valueM1, ..., valueMN]]
22      *
23      * @param matrix2D the specified 2-dimensional array of integers
24      *
25      */
26     public static void printMatrix2D(int[][] matrix2D){
27         System.out.print("[");
28         for(int i = 0; i < matrix2D.length; ++i) {
29             System.out.print("[");
30             for(int j = 0; j < matrix2D[i].length; ++j)
31                 System.out.print(matrix2D[i][j]+", ");
32             System.out.println("],");
33         }
34         System.out.println("]");
35     }
36
37     /**
38      * @param args
39      */
40     public static void main(String[] args) {
41         int [][] matrix2D = new int[3][4];
42         System.out.println("Width "+matrix2D.length);
43         System.out.println("Height "+matrix2D[0].length);
44
45         Random randInt = new Random();
46         for(int i = 0; i < matrix2D.length; ++i)
47             for(int j = 0; j < matrix2D[i].length; ++j)
48                 matrix2D[i][j] = randInt.nextInt(5);
49         printMatrix2D(matrix2D);
50     }
51 }
```

## Output

```
Width 3  
Height 4  
[[4, 4, 2, 4, ],  
 [0, 4, 1, 1, ],  
 [4, 2, 1, 0, ],  
 ]
```