

COS126 StdIn Activity - Exercise 1.5.1

Write a program `MaxMin.java` that reads in integers (as many as the user enters) from standard input and prints out the maximum and minimum values.

```
1  /*****
2  *  Compilation:  javac-introcs MaxMin.java
3  *  Execution:   java-introcs MaxMin
4  *              [input required from standard input      ]
5  *              [ use ctl-d (OS/X) or ctl-z (Windows) for EOF ]
6  *
7  *  Dependencies: StdIn.java, StdOut.java
8  *
9  *  Read in integers from standard input
10 *  and print out the maximum and minimum values read in.
11 *
12 *  % java-introcs MaxMin
13 *  23 45 17 56 32
14 *  89 10 53 32 34
15 *  16
16 *  ctl-d
17 *  maximum = 89, minimum = 10
18 *  (Ex. 1.5.1)
19 *****/
20
21 public class MaxMin {
22     public static void main(String[] args) {
23
24         // first value read initializes min and max
25         int max =          ;
26         int min =          ;
27
28         // read in the data, keep track of min and max
29         while (            ) {
30             int value = StdIn.readInt();
31
32
33         }
34
35         // output
36         StdOut.println("max = " + max + "    min = " + min);
37     }
38 }
```

Recommended Exercises: 1.5.3, 1.5.11, 1.5.13, 1.5.15

COS126 Parallel Array, StdIn Activity: 1.4, 1.5 (Booksite Web Ex.1.5.31)

```

1 /*****
2 * Compilation: javac-introcs Students.java
3 * Execution:   java-introcs Students < students.txt
4 *   data file http://www.cs.princeton.edu/introcs/15inout/students.txt
5 * Dependencies: StdIn.java StdOut.java
6 *
7 * Reads in the integer N from standard input, then a list
8 * of N student records, where each record consists of four
9 * fields, separated by whitespace:
10 *   - first name
11 *   - last name
12 *   - email address
13 *   - which section they're in
14 * Then, print a list of email address of students in sections 4 and 5.
15 * (Booksite Web Exercise 1.5.31)
16 *****/
. . .
48 public class Students {
49     public static void main(String[] args) {
50
51         // read the number of students
52         int N = _____
53
54         // declare and initialize four parallel arrays
55         String[] first   = new String[N];
56         _____ last   = _____
57         String[] _____ = _____
58         int[] section    = _____
59
60         // read in the data from standard input
61         for (_____ ; _____ ; _____) {
62             first[___]   = StdIn.readString();
63             last[___]    = _____
64             email[___]   = _____
65             section[___] = _____
66         }
67
68         // print email addresses of all students in section 4
69         StdOut.println("Section 4");
70         StdOut.println("-----");
71         for (int i = 0; i < N; i++) {
72             if (_____ ) {
73                 StdOut.println(_____);
74             }
75         }
. . .
87     }
88 }

```