Placement Exam for Computer Science: Connect Four*

Department of Computer Science, Carleton College

Last updated: July 26, 2023

You will be given a two-dimensional grid of numbers, with m rows and n columns, in the following format:

674	20	305	-921	912	779	25	525	331	181	660	-123	-620	-88	269
650	640	613	575	617	-883	-202	159	-201	874	184	-893	-863	-655	550
30	888	649	959	-339	668	471	519	538	-45	905	-427	350	607	-275
-140	362	-397	230	-638	-658	-499	-74	710	840	-856	-553	726	14	547
753	288	-318	-432	-302	584	94	47	752	-901	-384	-14	-209	-435	443
-662	-943	59	-601	-650	934	-509	757	-323	-679	57	-780	-924	-853	207
823	-582	21	168	729	1	839	-873	334	151	252	919	733	-481	-565
-385	562	-661	298	-646	-834	592	-753	277	97	-179	64	993	-168	421
-540	-803	917	-781	989	-789	-158	243	775	-394	-130	-19	655	318	420
44	-18	112	-533	-886	699	-884	-607	-904	-17	971	-224	-956	134	110

Let's call four numbers that appear "in a row" a *quadsequence*. A quadsequence can be horizontal (e.g., starting in the upper-left corner, `674 20 305 -921`), vertical (`674 650 30 -140`), down-diagonal (`674 640 649 230`), or up-diagonal (the upper-left-most example is `-140 888 613 -921`).

The problem. Write a program that finds the largest value produced by multiplying together the four numbers in any quadsequence in a given grid. For example, the product of the first example quadsequence above—the horizontal sequence of the first four numbers in the first row—is $674 \cdot 20 \cdot 305 \cdot -921 = -3{,}786{,}599{,}400$. The product of the up-diagonal example is $-140 \cdot 888 \cdot 613 \cdot -921 = 70{,}187{,}715{,}360$. The latter is bigger. You seek the largest such product among all quadsequences in the grid.

Use Python or Java to solve this problem. (If you don't know Python or Java but want to do this problem in another language, please get in touch.) The input data for your program—that is, the grid of numbers—will be given to you via a URL, specified as a command-line argument to your program. (See below for help handling command-line parameters and URLs; don't worry if you haven't used them before!) The format of the grid located at the given URL will be as in the example above. You may assume that each line contains the same number of entries (e.g., 15 entries per row in the example grid above), that all entries are properly formatted integers (possibly negative), and that all entries are separated by one or more spaces. The grid in the file may not have the same dimensions as this example, and the number of rows might differ from the number of columns.

A test case. The above grid is at `https://cs.carleton.edu/placement/grid.txt`. The product of the largest quadsequence in this grid is 568,764,139,559. Before submitting your solution, make sure your program correctly computes the largest quadsequence of this grid. We will also run your submitted code on several other test cases.

Hint: If you write your solution in Java, use `Long` variables instead of `Integer` variables; some of the numbers you compute may be large.

*Thanks to one of the problems from Project Euler, `https://projecteuler.net`, for the inspiration for this question.

How your submission will be tested. We will run your program with the test URL as a command-line parameter. As output, your program should print one and only one line of output, which should be precisely the specified product (that is, the product of the largest quadsequence). Here are complete successful executions of this program in Python and Java using the sample grid:

```
$ python3 placement.py https://cs.carleton.edu/placement/grid.txt
568764139559
```

```
$ javac Placement.java
$ java Placement https://cs.carleton.edu/placement/grid.txt
568764139559
```

Creating your own test cases You may, of course, use the command lines shown above to do your testing on the test case `grid.txt`. However, once you submit your code, we will run it on several additional test cases that explore the boundaries of this problem. We encourage you to test your code on your own sample grids to ensure that your program works on more than just the provided test case.

For example, you could create a file named `grid1.txt` on your Desktop containing:

```
1 4 1 2 3
1 1 4 2 3
1 1 2 4 3
1 1 2 3 4
1 1 1 1 1
```

to help make sure your code works properly when the maximum-product quadsequence is a down-diagonal. To run your program with `grid1.txt` as input, you can use a `file:` URL rather than an `https:` URL.

For a Python program executed in Command Prompt on Windows, that would look like this:

```
> python3 placement.py file:///C:/Users/your-user-name/Desktop/grid1.txt
568764139559
```

In Terminal on macOS (look in /Applications/Utilities to find Terminal.app), it would look like this:

```
$ python3 placement.py file:///Users/your-user-name/Desktop/grid1.txt
568764139559
```

Help! I don't know how to open a webpage or use command-line parameters. Here are some skeleton pieces of code to help you get started. The following programs simply open a URL and print out the lines of that file, one by one. (You don't have to start from these skeletons, but they might be helpful.)

Python skeleton code

```
import sys
import urllib.request

command_line_parameter = sys.argv[1]
f = urllib.request.urlopen(command_line_parameter)
for line in f:
    print(line)
f.close()
```

Java skeleton code

```
import java.net.*;
import java.util.*;

public class Sample {
   public static void main(String[] args) throws Exception {
      String commandLineParameter = args[0];
      URL url = new URL(commandLineParameter);
      Scanner scanner = new Scanner(url.openStream());
      while (scanner.hasNextLine()) {
         System.out.println(scanner.nextLine());
      }
      scanner.close();
   }
}
```