Objective

Today, we are building on our knowledge of arrays by adding another dimension. Check out the Tutorial tab for learning materials and an instructional video.

Context

Given a 6×6 2D Array, A:

```
1 1 1 0 0 0
0 1 0 0 0 0
1 1 1 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0
```

We define an hourglass in A to be a subset of values with indices falling in this pattern in A's graphical representation:

```
a b c
  d
e f g
```

There are 16 hourglasses in A, and an hourglass sum is the sum of an hourglass' values.

Task

Calculate the hourglass sum for every hourglass in A, then print the maximum hourglass sum.

Example

In the array shown above, the maximum hourglass sum is 7 for the hourglass in the top left corner.

Input Format

There are 6 lines of input, where each line contains 6 space-separated integers that describe the 2D Array A.

Constraints

- $-9 \leq A[i][j] \leq 9$
- $0 \leq i, j \leq 5$

Output Format

Print the maximum hourglass sum in A.

Sample Input

```
1 1 1 0 0 0
0 1 0 0 0 0
1 1 1 0 0 0
0 0 2 4 4 0
0 0 0 2 0 0
0 0 1 2 4 0
```

Sample Output

```
19
```

Explanation

A contains the following hourglasses:

```
1 1 1   1 1 0   1 0 0   0 0 0
  1       0       0       0
```

Change Theme | Java 8

```
import java.io.*;
import java.math.*;
import java.security.*;
import java.text.*;
import java.util.*;
import java.util.concurrent.*;
import java.util.regex.*;

public class Solution {

    public static int sumHourGlass(int[][] arr){
        int sum = 0, maxHourGlassSum = -350;
        for (int i = 0; i <= 3; i++) {
            for (int j = 0; j <= 3; j++) {
                sum = arr[i][j] + arr[i][j + 1] + arr[i]
                          + arr[i + 1][j + 1 ] +
                      arr[i + 2][j] + arr[i + 2][j + 1]
                if (sum > maxHourGlassSum) maxHourGlassS
            }
        }
        return maxHourGlassSum;
    }


    private static final Scanner scanner = new Scanner(S

    public static void main(String[] args) {
        int[][] arr = new int[6][6];
```

Line: 12 Col: 45

Upload Code as File
Test against custom input

Run Code | Submit Code

You have earned 30.00 points!

You are now 3 challenges away from the 3rd star for your 30 days of code badge.

63% 12/15

30 Days of Code

Congratulations

You solved this challenge. Would you like to challenge your friends?

The next challenge in this tutorial will unlock in 12:55:03

Go to Dashboard | Try a Random Challenge

Test case 0
Test case 1
Test case 2
Test case 3
Test case 4

Compiler Message

Success

Input (stdin) Download

```
1 1 1 0 0 0
0 1 0 0 0 0
1 1 1 0 0 0
0 0 2 4 4 0
```