2D Array - DS ☆

124.77 more points to get your gold badge!

Rank: **105032** | Points: **725.23/850**

Problem Solving ★★★★

Your 2D Array - DS submission got 10.00 points. Share Tweet

You are now 124.77 points away from the gold level for your problem solving badge.

Problem Submissions Leaderboard Editorial

RATE THIS CHALLENGE

☆ ☆ ☆ ☆ ☆

Given a 6×6 2D Array, arr:

```
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 arr's graphical representation:

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

There are 16 hourglasses in arr, and an hourglass sum is the sum of an hourglass' values. Calculate the hourglass sum for every hourglass in arr, then print the maximum hourglass sum.

For example, given the 2D array:

```
-9 -9 -9  1 1 1
 0 -9  0  4 3 2
-9 -9 -9  1 2 3
 0  0  8  6 6 0
 0  0  0 -2 0 0
 0  0  1  2 4 0
```

We calculate the following 16 hourglass values:

```
-63, -34, -9, 12,
-10, 0, 28, 23,
-27, -11, -2, 10,
9, 17, 25, 18
```

Our highest hourglass value is 28 from the hourglass:

```
0 4 3
  1
8 6 6
```

Note: If you have already solved the Java domain's Java 2D Array challenge, you may wish to skip this challenge.

Function Description

Complete the function hourglassSum in the editor below. It should return an integer, the maximum hourglass sum in the array.

hourglassSum has the following parameter(s):

- arr: an array of integers

Input Format

Each of the 6 lines of inputs $arr[i]$ contains 6 space-separated integers $arr[i][j]$.

Constraints

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

Output Format

Print the largest (maximum) hourglass sum found in arr.

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

arr contains the following hourglasses:

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

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

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

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

The hourglass with the maximum sum (19) is:

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

Go

```go
package main

import (
    "bufio"
    "fmt"
    "io"
    "os"
    "strconv"
    "strings"
)

// Complete the hourglassSum function below.
func hourglassSum(arr [][]int32) int32 {
    var max int32
    rows := len(arr)
    cols := len(arr[0])
    for i := 0; i < rows - 2; i++ {
        for j := 0; j < cols - 2; j++ {
            var sum int32
            sum = (arr[i][j] + arr[i][j+1] + arr[i][j+2]) + (arr[i+1][j+1]) + (arr[i+2][j] + arr[i+2][j+1] + arr[i+2][j+2])
            if sum > max {
                max = sum
            }
        }
    }
    return max
```

Line: 14 Col: 18

Upload Code as File Test against custom input Run Code Submit Code

2/9 test cases failed :(

Ask your friends for help:

Test case 3 ✗

Test case 7 ✗

Test case 0 ✓

Test case 1 ✓

Test case 2 ✓

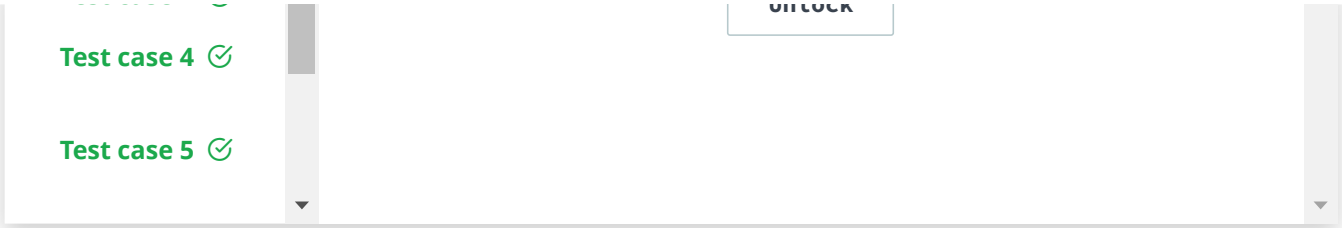
Compiler Message

Wrong Answer

Hidden Test Case

Unlock this testcase for 5 hackos.

Unlock


Test case 4
Test case 5