2D Array - DS

Given a 6×6 2D array, arr, an hourglass is a subset of values with indices falling in the following pattern:

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

There are 16 hourglasses in a 6×6 array. The $hourglass\ sum$ is the sum of the values in an hourglass. Calculate the hourglass sum for every hourglass in arr, then print the $maximum$ hourglass sum.

Example

$arr =$

```
-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
```

The 16 hourglass sums are:

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

The highest hourglass sum is 28 from the hourglass beginning at row 1, column 2:

```
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 $hourglassSum$ with the following parameter(s):

- $int\ arr[6][6]$: a 2-D array of integers

Returns

- int: the maximum hourglass sum

Input Format

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

Constraints

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

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
```

Change Theme Language Python 3

```
#!/bin/python3

import math
import os
import random
import re
import sys

#
# Complete the 'hourglassSum' function below.
#
# The function is expected to return an INTEGER.
# The function accepts 2D_INTEGER_ARRAY arr as parameter.
#
```

```
15
16  def hourglassSum(arr):
17      # Write your code here
18      res = -math.inf
19      for r in range(1, len(arr)-1):
20          for c in range(1, len(arr[0])-1):
21              local_sum = sum(arr[r-1][c-1:c+2]) + arr[r][c] + sum(arr[r+1][c-1:c+2])
22              res = max(res,local_sum)
23      return res
24
25  if __name__ == '__main__':
26      fptr = open(os.environ['OUTPUT_PATH'], 'w')
27
28      arr = []
29
30      for _ in range(6):
31          arr.append(list(map(int, input().rstrip().split())))
32
33      result = hourglassSum(arr)
34
```

EMACS Line: 38 Col: 1

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

You have earned 15.00 points!

You are now 44 points away from the 2nd star for your problem solving badge.

37% 56/100

Congratulations

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

Next Challenge

Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

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

Expected Output Download

```
19
```