

All Domains > Data Structures > Arrays > 2D Array - DS

2D Array - DS



Problem Submissions Leaderboard Discussions Editorial

Context

Given a $\mathbf{6} \times \mathbf{6}$ 2D Array, \mathbf{A} :

```
111000
0 1 0 0 0 0
1 1 1 0 0 0
000000
000000
000000
```

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.

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

Input Format

There are 6 lines of input, where each line contains 6 space-separated integers describing 2D Array A; every value in A will be in the inclusive range of -9 to 9.

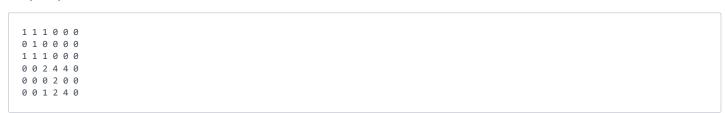
Constraints

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

Output Format

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

Sample Input



Sample Output

```
19
```

Explanation

 \boldsymbol{A} contains the following hourglasses:

```
111 110 100
 1
      0
            0
1 1 1
    1 1 0
          100
                0 0 0
    100
          000
                0 0 0
0 1 0
            0
                  0
      1
                4 4 0
0 0 2 0 2 4 2 4 4
     110 100
                000
1 1 1
      2
           4
                 4
000
    002
          0 2 0
                200
0 0 2
    0 2 4 2 4 4
     0
           2
                0
0
    0 1 2 1 2 4
```

The hourglass with the maximum sum (19) is:

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

in 💆 f

Submissions: 26968

Max Score: 15
Difficulty: Easy

More

```
Current Buffer (saved locally, editable) &
                                                                                            Java 7
 1 ▼ import java.io.*;
 2 import java.util.*;
 3 import java.text.*;
 4 import java.math.*;
   import java.util.regex.*;
 5
 6
 7 ▼ public class Solution {
8
9 ₹
        public static void main(String[] args) {
10
            Scanner in = new Scanner(System.in);
11
            int arr[][] = new int[6][6];
12 🔻
            for(int arr_i=0; arr_i < 6; arr_i++){</pre>
13 ▼
                 for(int arr_j=0; arr_j < 6; arr_j++){</pre>
14
                     arr[arr_i][arr_j] = in.nextInt();
15
                 }
            }
16
17
18
                                  Integer.MIN_VALUE;
                                                            //possibility of sum being negative
19
            int sum_temp
                                           //temporary sum
            int temp_i;
20
                                      //temporary i
21
                                      //temporary j
            int temp_j;
22
23
            for(int i=0; i<4;i++)
24
25 ▼
            {
26
                 for(int j=0;j<4;j++)</pre>
27 🔻
                 {
28
                         sum\_temp = arr[i][j] + arr[i][j+1] + arr[i][j+2]
```

```
29
                                               +arr[i+1][j+1]+
                                     arr[i+2][j]+arr[i+2][j+1]+arr[i+2][j+2];
30
31
                     if(sum_temp>sum)
32
                         sum = sum_temp;
                       System.out.println("i: "+i);
33
    //
                       System.out.println("j: "+j);
34
    //
35
                 }
36
             }
37
38
             System.out.println(sum);
39
40
41
         }
42
    }
                                                                                                                      Line: 1 Col: 1
                        ☐ Test against custom input
1 Upload Code as File
                                                                                                         Run Code
                                                                                                                      Submit Code
```

Copyright © 2016 HackerRank. All Rights Reserved

Join us on IRC at #hackerrank on freenode for hugs or bugs. Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature