**Objective**   
Today, we're building on our knowledge of *Arrays* by adding another dimension. Check out the [Tutorial](https://www.hackerrank.com/challenges/30-2d-arrays/tutorial) tab for learning materials and an instructional video!

**Context**   
Given a  *2D Array*, :

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  to be a subset of values with indices falling in this pattern in 's graphical representation:

a b c

d

e f g

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

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

**Input Format**

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

**Constraints**

**Output Format**

Print the largest (maximum) hourglass sum found in .

**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**

 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

#!/bin/python

import sys

arr = []

for arr\_i in xrange(6):

arr\_temp = map(int,raw\_input().strip().split(' '))

arr.append(arr\_temp)