2D Arrays - DS

OJ `hackerrank` `datastructure`

> **Problem Information:**
> problem's link
> **AC date** : **2014-04-18**
> **Category** : data structure->array

Question

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.

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

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

Output Format

Print the largest (maximum) hourglass sum found 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
```

Answer

C++

```cpp
//head.h
using namespace std;

int main(){
    int sum = 0;
    int max = -100;
    vector< vector<int> > arr(6,vector<int>(6));

    for(int arr_i = 0;arr_i < 6;arr_i++){
       for(int arr_j = 0;arr_j < 6;arr_j++){
          cin >> arr[arr_i][arr_j];
       }
    }

    for(int i=0; i<4; i++){
        for(int j=0; j<4; j++){
            sum = arr[i][j]+arr[i][j+1]+arr[i][j+2]+arr[i+1][j+1]+a
rr[i+2][j]+arr[i+2][j+1]+arr[i+2][j+2];
            max = sum > max ? sum : max;
        }
    }
    cout << max << endl;
    return 0;
}
```