

835. Image Overlap

[Description](#)[Hints](#)[Submissions](#)[Discuss](#)[Solution](#)

Two images A and B are given, represented as binary, square matrices of the same size. (A binary matrix has only 0s and 1s as values.)

We translate one image however we choose (sliding it left, right, up, or down any number of units), and place it on top of the other image. After, the *overlap* of this translation is the number of positions that have a 1 in both images.

(Note also that a translation does **not** include any kind of rotation.)

What is the largest possible overlap?

Example 1:

```
Input: A = [[1,1,0],
            [0,1,0],
            [0,1,0]]
        B = [[0,0,0],
            [0,1,1],
            [0,0,1]]
```

Output: 3

Explanation: We slide A to right by 1 unit and down by 1 unit.

Notes:

1. $1 \leq A.length = A[0].length = B.length = B[0].length \leq 30$
 2. $0 \leq A[i][j], B[i][j] \leq 1$
-

Seen this question in a real interview before?

-
- [Subscribe](#) to see which companies asked this question.

Related Topics Array

We do the brute force solution, for all dh, dv in $-N+1, N$, in which N is the size of row and col in A collect $A[i][j] * B[i-dh][j-dv]$

```

int largestOverlap(vector<vector<int>>& A, vector<vector<int>>& B)
{
    int m = A.size();
    int n = A[0].size(); int i,j;
    int ans = -1;
    for(int dh=-m+1;dh<m;dh++)
    {
        for(int dv=-n+1;dv<n;dv++)
        {
            int total = 0;
            for(i=max(0,dh);i<min(m,m+dh);i++)
            {
                for(j=max(0,dv);j<min(n,n+dv);j++)
                {
                    total+=A[i][j]*B[i-dh][j-dv];
                }
            }
            ans = max(total,ans);
        }
    }
    return ans;
}

```