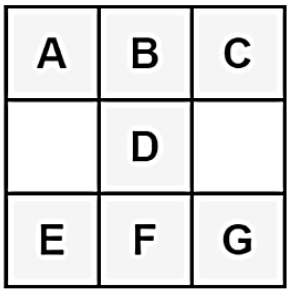
PROBLEM

Maximum sum of hour glass

Medium Accuracy: 54.82% Submissions: 33K+ Points: 4

Given two integers **n**, **m** and a 2D matrix **mat** of dimensions **nxm**, the task is to find the **maximum sum** of an **hourglass**.

An **hourglass** is defined as a part of the matrix with the following form:



Return **-1** if any hourglass is **not found**.

Example 1:

Input:
n = 3, m = 3
mat = [[1, 2, 3],
[4, 5, 6],
[7, 8, 9]]
Output:
35
Explanation:
There is only one hour glass which is
1 2 3
5
7 8 9 and its sum is 35.

Example 2:

Input:
n = 2, m = 3
mat = [[1, 2, 3],
[4, 5, 6]]
Output:
-1
Explanation:
There are no hour glasses in this matrix.

Your Task:
You don't need to read input or print anything. Your task is to complete the function **findMaxSum()** which takes the two integers **n**, **m**, and the 2D matrix **mat** as input parameters and returns the maximum sum of an hourglass in the matrix. If there are no hourglasses, it returns -1.

Expected Time Complexity: O(n*m)
Expected Auxillary Space: O(1)

Constraints:
1 <= n <= 150
3 <= m <= 150
0 <= mat[i][j] <= 10^6

CODE

```
#User function Template for python3


class Solution:
   def findMaxSum(self,n,m,mat):
      res = -1
      for i in range(n-2):
         for j in range(m-2):
            tem = (mat[i][j] + mat[i][j + 1] + mat[i][j + 2]) + (mat[i + 1][j
+ 1]) + (mat[i + 2][j] + mat[i + 2][j + 1] + mat[i + 2][j + 2])
            res = max(res,tem)
      return res
      #code here
```