**Title**

-------------------------------------------------------------------------------------------------------------

**Flipping the Matrix**

-------------------------------------------------------------------------------------------------------------

**Description**

-------------------------------------------------------------------------------------------------------------

Sean invented a game involving a 2n x 2n matrix where each cell of the matrix contains an integer. He can reverse any of its rows or columns any number of times. The goal of the game is to maximize the sum of the elements in the n x n

submatrix located in the upper-left quadrant of the matrix.

Given the initial configurations for q matrices, help Sean reverse the rows and columns of each matrix in thebest possible way so that the sum of the elements in the matrix's upper-left quadrant is maximal.

**Example**

matrix = [[1, 2], [3, 4]]

It is 2 x 2 and we want to maximize the top left quadrant, a 1 x 1 matrix.

Reverse row 1:

1 2

4 3

And now reverse column 0:

4 2

1 3

The maximal sum is 4

**Function Description**

Complete the flippingMatrix function in the editor below.

flippingMatrix has the following parameters:

int matrix[2n][2n]: a 2-dimensional array of integers.

**Returns**

int: the maximum sum possible.

**Input Format**

The first line contains an integer q, the number of queries.

The next q sets of lines are in the following format:

The first line of each query contains an integer,n.

Each of the next 2n lines contains 2n space-separated integers matrix[i][j]

in row i of the matrix.

**Constraints**

1 <= q <= 16

1 <= n <= 128

0 <= matrix[i][j] <= 4096, where 0 <= i,j < 2n

Sample Input

STDIN Function

----- --------

1 q = 1

2 n = 2

112 42 83 119 matrix = [[112, 42, 83, 119], [56, 125, 56, 49],

56 125 56 49 [15, 78, 101, 43], [62, 98, 114, 108]]

15 78 101 43

62 98 114 108

**Sample Output**

414

-------------------------------------------------------------------------------------------------------------

**Code**

-------------------------------------------------------------------------------------------------------------

package main

import(

    "fmt"

)

func main(){

    matrix := [][]int32{{112, 42, 83, 119}, {56, 125, 56, 49},{15, 78, 101, 43}, {62, 98, 114, 108}}

    var res int32

    res = flippingMatrix(matrix)

    fmt.Println(res)

}

func flippingMatrix(matrix [][]int32) int32{

    end := len(matrix)-1

    mid := len(matrix)/2

    var sum int32

    for i:= 0; i<mid;i++{

        for j:=0; j<mid; j++{

            slc := make([]int32,4)

            slc = append(slc,matrix[i][j],matrix[i][end-j],matrix[end-i][j],matrix[end-i][end-j])

            var max int32

            for \_, v := range slc{

                if v>max{

                    max = v

                }

            }

            sum += max

        }

    }

    return sum

}

-------------------------------------------------------------------------------------------------------------