Convolution

จงเขียนฟังก์ชัน `convolute(M, K)` รับ `M` และ `K` เป็นลิสต์ซ้อนลิสต์ ที่แทนเมทริกซ์ของจำนวนเต็ม โดยที่ `M` และ `K` มีขนาดเท่ากัน ฟังก์ชันนี้คืนจำนวนเต็มที่เป็น**ผลรวมของผลคูณของจำนวนในเมทริกซ์ M และ K ที่ตำแหน่งเดียวกัน** เช่น

$$\underset{M}{\begin{bmatrix}1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9\end{bmatrix}} * \underset{K}{\begin{bmatrix}-1 & -2 & -1\\ 0 & 0 & 0\\ 1 & 2 & 1\end{bmatrix}}$$

$$= (1*-1) + (2*-2) + (3*-1) + (4*0) + (5*0) + (6*0) + (7*1) + (8*2) + (9*1)$$
$$= \quad -1 \quad -4 \quad -3 \quad +0 \quad +0 \quad +0 \quad +7 \quad +16 \quad +9$$
$$= 24$$

```
def convolute(M, K):



exec(input().strip()) #ต้องมีบรรทัดนี้เมื่อส่งไป grader
```

ข้อมูลนำเข้า

คำสั่งในการทดสอบฟังก์ชันที่เขียน

ข้อมูลส่งออก

ผลที่ได้จากคำสั่งที่ป้อนเป็นข้อมูลนำเข้า

ตัวอย่าง

Input	Output
`print(convolute([[1,2],[3,4]],[[1,1],[1,1]]))`	10
`print(convolute([[1,2,3],[4,5,6],[7,8,9]],[[-1,-2,-1],[0,0,0],[1,2,1]]))`	24
`print(convolute([[1,2],[3,4]],[[0,0],[0,0]]))`	0
`print(convolute([[1,2],[1,2]],[[-1,-1],[1,1]]))`	0