

ROBOTK

Cho một cái sân hình chữ nhật gồm N hàng và M cột. Các hàng được đánh số từ 1 tới N từ trên xuống dưới và các cột được đánh số từ 1 tới M từ trái sang phải. Gọi ô ở giao điểm của hàng thứ i và cột thứ j là ô (i, j) . Có một con robot xuất phát ở ô $(1, 1)$. Con robot này có thể di chuyển sang các ô **kề cạnh** của nó, miễn sao không đi ra khỏi phạm vi của sân. Cụ thể hơn, khi con robot ở ô (i, j) , nó có thể di chuyển đến một trong các ô $(i + 1, j)$, $(i, j + 1)$, $(i - 1, j)$, $(i, j - 1)$ nếu ô đó không nằm ngoài sân. Tại mỗi ô sẽ có một số điểm. Khi robot đi từ ô hiện tại sang ô tiếp theo thì sẽ được cộng thêm số điểm của ô vừa tới. Để đảm bảo cho robot không đi lạc quá lâu, robot chỉ được phép đi lên trên hoặc sang trái **tối đa tổng cộng K lần**. Hãy điều khiển robot di chuyển sao cho robot dừng lại ở ô (N, M) và đạt được tổng số điểm là lớn nhất.

Lưu ý: Nếu như robot đi tới một ô nhiều lần thì số điểm sẽ được cộng dồn theo số lần tương ứng.

Dữ liệu vào: File **ROBOTK.INP** gồm:

- Dòng đầu tiên gồm ba số nguyên N, M, K ($1 \leq N, M \leq 50, 0 \leq K \leq 50$) lần lượt là số hàng, số cột của sân chơi và số lần tối đa robot được đi lên trên hoặc sang trái.
- N dòng tiếp theo, mỗi dòng gồm M số nguyên là số điểm của các ô trên sân. Số thứ j của dòng i là số điểm của ô (i, j) . Các số điểm của các ô có giá trị tuyệt đối không vượt quá 1000.

Dữ liệu ra: File **ROBOTK.OUT** gồm một số nguyên duy nhất là tổng điểm lớn nhất đạt được.

- Giới hạn:**
- 40% số test có $1 \leq N, M \leq 5, K = 0$.
 - 30% số test tiếp theo có $1 \leq N, M \leq 50, K = 0$.
 - 30% số test còn lại không có ràng buộc gì thêm.

ROBOTK . INP	ROBOTK . OUT
3 4 1 1 2 -1 -1 1 -1 -1 -1 1 1 1 1	8

Giải thích:

1	2	-1	-1
1	-1	-1	-1
1	1	1	1

Robot sẽ di chuyển như sau: $(1, 1) \rightarrow (1, 2)$ (+2 điểm) $\rightarrow (1, 1)$ (+1 điểm) $\rightarrow (2, 1)$ (+1 điểm) $\rightarrow (3, 1)$ (+1 điểm) $\rightarrow (3, 2)$ (+1 điểm) $\rightarrow (3, 3)$ (+1 điểm) $\rightarrow (3, 4)$ (+1 điểm)