1.

코니와 브라운은 광활한 들판에서 '나 잡아 봐라' 게임을 하려고 합니다.

이 게임은 브라운이 코니를 잡거나, 코니가 너무 멀리 달아나면 끝납니다.

게임이 끝나는데 걸리는 최소 시간을 return하는 solution 함수를 작성해주세요.

제한사항

- ▶ 코니의 처음위치 C, 브라운의 처음위치 B가 주어집니다.
- ▼ 코니는 처음 위치 C에서 1초 후 1만큼 움직이고,
 이후에는 가속이 붙어 매 초마다 이전 이동 거리 +1 만큼 움직입니다.
 즉, 시간에 따른 코니의 위치는 C, C + 1, C + 3, C + 6, ... 입니다.
- ▶ 브라운은 현재 위치 B에서 다음 순간 B-1,B+1,B*2 중 하나로 움직일 수 있습니다.
- ▶ 코니와 브라운의 위치는 0보다 크거나 같고 200,000보다 작거나 같습니다.
- ▶ 브라운은 범위를 벗어나는 위치로는 이동할 수 없으며,만약 코니가 범위를 벗어나면 게임이 종료되며 -1을 return 합니다.

С	В	return
11	2	5
10	3	3
51	50	8
550	500	28

2.

남영이는 주변을 살펴보던 중 체스판과 말을 이용해서 새로운 게임을 만들기로 했습니다.

새로운 게임은 크기가 N×N인 체스판에서 진행되고, 사용하는 말의 개수는 K개입니다.

말은 원판모양이고, 하나의 말 위에 다른 말을 올릴 수 있습니다.

체스판의 각 칸은 흰색, 빨간색, 파란색 중 하나로 색칠되어있습니다.

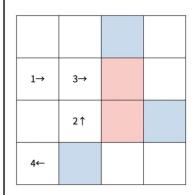
게임은 체스판 위에 말 K개를 놓고 시작합니다. 말은 1번부터 K번까지 번호가 매겨져 있고, 이동 방향도 미리 정해져 있습니다. 이동 방향은 위, 아래, 왼쪽, 오른쪽 4가지 중 하나입니다.

턴 한 번은 1번 말부터 K번 말까지 순서대로 이동시키는 것입니다. 한 말이 이동할 때 위에 올려져 있는 말도 함께 이동합니다. 말의 이동 방향에 있는 칸에 따라서 말의 이동이 다르며 아래와 같습니다. 턴이 진행되던 중에 말이 4개 이상 쌓이는 순간 게임이 종료됩니다.

A번 말이 이동하려는 칸의 색에 따라 말의 이동이 바뀝니다. 조건은 아래와 같습니다.

- 1) 흰색인 경우에는 그 칸으로 이동한다. 이동하려는 칸에 말이 이미 있는 경우에는 가장 위에 A 번 말을 올려놓는다.
- A번 말의 위에 다른 말이 있는 경우에는 A번 말과 위에 있는 모든 말이 이동한다.
- 예를 들어, A, B, C로 쌓여있고, 이동하려는 칸에 D, E가 있는 경우에는 A번 말이 이동한 후에는 D, E, A, B, C가 된다.
- 2) 빨간색인 경우에는 이동한 후에 A번 말과 그 위에 있는 모든 말의 쌓여있는 순서를 반대로 바꾼다.
- A, B, C가 이동하고, 이동하려는 칸에 말이 없는 경우에는 C, B, A가 된다.
- A, D, F, G가 이동하고, 이동하려는 칸에 말이 E, C, B로 있는 경우에는 E, C, B, G, F, D, A가 된다.
- 3) 파란색인 경우에는 A번 말의 이동 방향을 반대로 하고 한 칸 이동한다. 방향을 반대로 바꾼 후에 이동하려는 칸이 파란색인 경우에는 이동하지 않고 가만히 있는다.
- 4) 체스판을 벗어나는 경우에는 파란색과 같은 경우이다.

다음은 크기가 4×4인 체스판 위에 말이 4개 있는 경우입니다.



> 첫 번째 턴은 다음과 같이 진행됩니다.

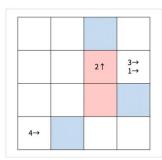
	1→ 3→	
	2↑	
4←		

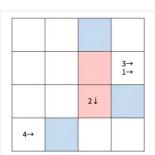
	2↑ 1→ 3→	
4←		

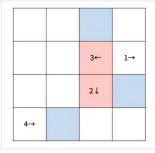
	3→ 1→ 2↑	
4←		

	3→ 1→ 2↑	
4→		

> 두 번째 턴은 다음과 같이 진행됩니다.







	3←	1→
	2↓	
4←		

체스판의 크기와 말의 위치, 이동 방향이 모두 주어졌을 때, 게임이 종료되는 턴의 번호를 반환하는 solution 함수를 작성해주세요.

제한사항

- ▶ chess_map 의 각 정수는 칸의 색을 의미합니다. 0은 흰색, 1은 빨간색, 2는 파란색입니다.
- ▶ 말의 개수를 뜻하는 K와 현재 말의 정보를 담은 배열 start_horse가 주어집니다.
- ▶ 말의 정보는 세 개의 정수로 이루어져 있고, 순서대로 행, 열의 인덱스, 이동 방향을 뜻합니다.
- ▶ 행과 열의 번호는 0부터 시작하며, 이동 방향은 0,1,2,3 입니다.
 - 각각 순서대로 0:→/1:←/2:↑/3:↓ 뜻합니다.
- ▶ 턴의 번호가 1,000보다 크거나 절대로 게임이 종료되지 않는 경우에는 -1을 return 합니다.

입출력 예

K	chess_map	start_horse	return
	[[0, 0, 0, 0],	[[0, 0, 0],	
4	[0, 0, 0, 0],	[0, 1, 0],	2
4	[0, 0, 0, 0],	[0, 2, 0],	۷
	[0, 0, 0, 0]]	[2, 2, 2]]	
	[[0, 0, 2, 0],	[[1, 0, 0],	
4	[0, 0, 1, 0],	[2, 1, 2],	-1
4	[0, 0, 1, 2],	[1, 1, 0],	l
	[0, 2, 0, 0]]	[3, 0, 1]]	
	[[0, 0, 0, 0],	[[0,1,0],	
4	[0, 0, 0, 0],	[1, 1, 0],	9
4	[0, 0, 0, 0],	[0, 2, 0],	פ
	[0, 0, 0, 0]]	[2, 2, 2]]	
	[[0, 0, 0, 0],	[[0,1,0],	
4	[0, 0, 0, 0],	[0, 1, 1],	3
4	[0, 0, 0, 0],	[0, 1, 0],	3
	[0, 0, 0, 0]]	[2, 1, 2]]	