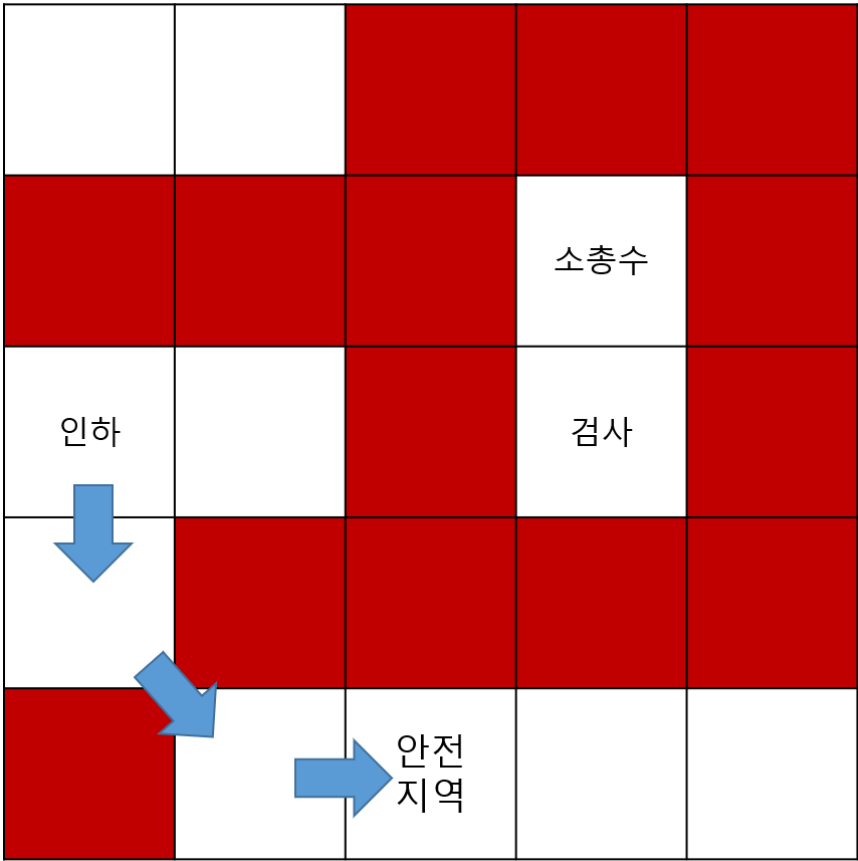


# 탈출

척후병 김인하는 현재  $N \times N$  크기의 좌표평면에 있다. 적군의 상황과 지형 등을 파악한 뒤, 아군이 있는 안전지역으로 이동하려고 한다. 한편, 근처에 적군의 검사 1명과 소총수 1명이 있다. 적군의 소총수와 검사는 이동하지 않지만, 검사는 인접한 8개의 방향으로 한 칸만 공격할 수 있고, 소총수는 검사와 같은 8개의 방향으로 거리 제한 없이 공격할 수 있다. 단, 아군끼리는 서로 공격하지 않으며, 소총수의 공격 방향에 검사가 존재한다면, 소총수는 검사의 건너편 방향을 공격할 수 없다. 인하는 인접한 8개의 방향으로 이동할 수 있다. 적군의 공격범위를 피해 안전지역으로 이동하기 위해, 인하의 최소 이동횟수를 알아보자.

예를 들어 아래의 그림과 같이,  $N = 5$ 일 때, 인하는 위치 (3, 1), 안전지역은 위치 (5, 3), 검사는 위치 (3, 4), 소총수는 위치 (2, 4)에 있다. (좌표는 행, 열 순서로 주어짐) 좌표평면에 빨간색으로 칠해진 위치들은 소총수와 검사의 공격범위로 인하가 갈 수 없는 지점이다. 참고로, 소총수는 검사의 건너편 위치인 (4, 4), (5, 4)는 공격할 수 없다. 이때 인하는 화살표 방향으로 움직이면, 최소 이동횟수인 3회만에 안전지역에 도착할 수 있다. (인하의 이동경로 중 (4,1)에서 (5,2)로 이동하는 과정에서 인접한 모서리를 통해서는 소총수의 공격을 받지 않는다)



[그림 1] 5 × 5 좌표평면에서의 예시

※ 프로그램의 실행 시간은 2 초, 메모리 사용량은 512MB 를 초과할 수 없다.

사용할 수 있는 언어는 C, C++로 제한한다. C++의 경우 main 함수 내의 시작 지점에 다음 내용을 추가함으로써 cin, cout 의 입출력 속도를 개선할 수 있다.

```
ios_base::sync_with_stdio(false);
cin.tie(NULL);
cout.tie(NULL);
```

단, 위의 내용을 추가할 경우 cin, cout 만 사용해야 하며, scanf, printf 등 C 입출력을 혼용해서 사용하면 안된다. C++의 std::endl 의 경우 출력 속도가 느리므로, cout<<endl; 대신 cout<<"\n";을 사용하는 것을 권장한다.

## 입력

첫 번째 줄에 테스트 케이스 수  $T$  ( $1 \leq T \leq 100$ )가 주어진다.

각 테스트 케이스의 구성은 다음과 같다.

- 첫 번째 줄에 좌표평면의 크기  $N$  ( $5 \leq N \leq 1,000$ )이 주어진다.
- 두 번째 줄에 인하의 위치  $a_x, a_y$ 와 안전지역의 위치  $b_x, b_y$ 가 공백으로 구분되어 주어진다. 이는 각각  $a_x$ 행  $a_y$ 열,  $b_x$ 행  $b_y$ 열을 의미한다( $1 \leq a_x, a_y, b_x, b_y \leq N$ ).
- 세 번째 줄에 검사의 위치  $c_x, c_y$ 와 소총수의 위치  $d_x, d_y$ 가 공백으로 구분되어 주어진다. 이는 각각  $c_x$ 행  $c_y$ 열,  $d_x$ 행  $d_y$ 열을 의미한다( $1 \leq c_x, c_y, d_x, d_y \leq N$ ).

단, 인하, 안전지역, 검사, 소총수의 위치는 모두 다르게 주어지며, 인하의 시작 위치와 안전지역의 위치는 적군의 공격범위 외에서 주어진다.

## 출력

테스트 케이스마다 인하가 안전지역에 도착하기 위한 최소 이동횟수를 각 줄에 출력한다. 만약 도착할 수 없으면, 0 을 출력한다.

### 예제 입출력

예제 입력	예제 출력
1 5 3 1 5 3 3 4 2 4	3