

코딩 테스트 모의고사 (B형)

< 문제지 >

난이도	중	문제 풀이시간	3시간
문제 개수	3문제	합격 커트라인	2~3문제

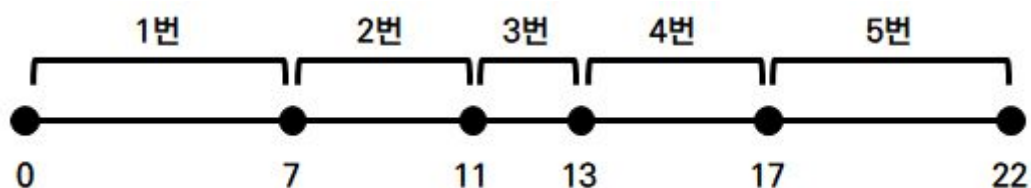
코딩 테스트 모의고사 문제는 인터넷 검색이 불가능한 상황에서 제한 시간안에 풀어주세요. 소스코드 작성 및 프로그램 실행 결과를 계산할 수 있는 Python 3.7 개발 환경은 제공된다고 가정합니다.

문제 1. 왕복

(시간 제한: 1초, 메모리 제한: 256MB)

왕복 달리기 선수는 N 개의 이어진 일직선상의 코스들을 모두 지나 끝까지 도달한 뒤에, 다시 출발 지점으로 돌아와야 한다. 전체 코스들을 지나고 있는 상황에서 이동 거리가 K 일 때, 현재 지나고 있는 코스의 번호를 출력하는 프로그램을 작성하시오. 단, 이동 거리가 K 가 두 코스 사이에 위치한 경우에는 '지나야 할' 코스의 번호를 출력한다.

예를 들어 $N = 5$ 일 때, 각 코스의 길이가 차례대로 7, 4, 2, 4, 5 라고 가정하자. 출발 지점을 0이라고 하면, 전체 코스가 구성된 형태를 다음과 같이 그릴 수 있다.



- 1) $K = 0$ 일 때, 1 번 코스를 지나고 있으므로 1 을 출력한다.
- 2) $K = 7$ 일 때, 2 번 코스를 지나고 있으므로 2 를 출력한다.
- 3) $K = 9$ 일 때, 2 번 코스를 지나고 있으므로 2 를 출력한다.
- 4) $K = 12$ 일 때, 3 번 코스를 지나고 있으므로 3 을 출력한다.
- 5) $K = 28$ 일 때, 이는 끝까지 도달한 뒤에 시작 위치로 돌아오는 과정으로 볼 수 있다. 4 번 코스를 지나고 있으므로 4 를 출력한다.

또한 K 는 항상 왕복 거리보다 작은 양의 정수 혹은 0으로 주어진다. 예를 들어 위와 같이 전체 코스의 길이가 합을 22라고 하면, $0 \leq K \leq 43$ 이다.

입력 조건

첫째 줄에 정수 N, K 가 공백을 기준으로 구분되어 주어진다. ($1 \leq N \leq 100,000$) 단, K 는 항상 왕복 거리보다 작은 양의 정수 혹은 0으로 주어진다.

둘째 줄에 1번부터 N 번까지 각 코스의 길이가 공백을 기준으로 구분되어 차례대로 주어진다. 각 코스의 길이는 50,000보다 작거나 같은 자연수다.

출력 조건

첫째 줄에 이동 거리가 K 일 때, 지나고 있는 코스의 번호를 출력한다.

입력 예시

```
5 28
7 4 2 4 5
```

출력 예시

```
4
```

문제 2. 병사 배치하기

(시간 제한: 1초, 메모리 제한: 256MB)

N 명의 병사가 무작위로 나열되어 있다. 각 병사는 특정한 값의 전투력을 보유하고 있으며, 병사를 배치할 때는 전투력이 높은 병사가 앞쪽에 오도록 내림차순으로 배치를 하고자 한다. 다시 말해 앞쪽에 있는 병사의 전투력이 항상 뒤쪽에 있는 병사보다 높아야 한다.

또한 배치 과정에서는 특정한 위치에 있는 병사를 옆이시키는 방법을 이용한다. 그러면서도 남아있는 병사의 수가 최대가 되도록 하고 싶다.

예를 들어, $N = 7$ 일 때 나열된 병사들의 전투력이 다음과 같다고 가정하자.

병사 번호	1	2	3	4	5	6	7
전투력	15	11	4	8	5	2	4

이 때 3 번 병사와 6 번 병사를 열외시키면, 다음과 같이 남아있는 병사의 수가 내림차순의 형태가 되며 5 명이 된다. 이는 남아있는 병사의 수가 최대가 되도록 하는 방법이다.

병사 번호	1	2	4	5	7
전투력	15	11	8	5	4

병사에 대한 정보가 주어졌을 때, 남아있는 병사의 수가 최대가 되도록 하기 위해서 열외해야 하는 병사의 수를 출력하는 프로그램을 작성하시오.

입력 조건

첫째 줄에 N 이 주어진다. ($1 \leq N \leq 2,000$)

둘째 줄에 각 병사의 전투력이 공백을 기준으로 구분되어 차례대로 주어진다. 각 병사의 전투력은 10,000,000 보다 작거나 같은 자연수다.

출력 조건

첫째 줄에 남아있는 병사의 수가 최대가 되도록 하기 위해서 열외해야 하는 병사의 수를 출력한다.

입력 예시

```
7
15 11 4 8 5 2 4
```

출력 예시

```
2
```

문제 3. 경쟁적 전염

(시간 제한: 1초, 메모리 제한: 256MB)

$N \times N$ 크기의 시험관이 있다. 시험관은 1×1 크기의 칸으로 나누어지며, 특정한 위치에는 바이러스가 존재할 수 있다. 모든 바이러스는 1 번부터 K 번까지의 바이러스 종류 중 하나에 속한다.

시험관에 존재하는 모든 바이러스는 1 초마다 상, 하, 좌, 우의 방향으로 증식해 나간다. 단, 매 초마다 번호가 낮은 종류의 바이러스부터 먼저 증식한다. 또한 증식 과정에서 특정한 칸에 이미 어떠한 바이러스가 존재한다면, 그 곳에는 다른 바이러스가 들어갈 수 없다.

시험관의 크기와 바이러스의 위치 정보가 주어졌을 때, S 초가 지난 후에 (X, Y) 에 존재하는 바이러스의 종류를 출력하는 프로그램을 작성하시오. 만약 S 초가 지난 후에 해당 위치에 바이러스가 존재하지 않는다면, 0을 출력한다. 이 때 X 와 Y 는 각각 행과 열의 위치를 의미하며, 시험관의 가장 왼쪽 위에 해당하는 곳은 $(1, 1)$ 에 해당한다.

예를 들어 다음과 같이 3×3 크기의 시험관이 있다고 하자. 서로 다른 1 번, 2 번, 3 번 바이러스가 각각 $(1, 1)$, $(1, 3)$, $(3, 1)$ 에 위치해 있다. 이 때 2 초가 지난 뒤에 $(3, 2)$ 에 존재하는 바이러스의 종류를 계산해보자.

1		2
3		

1 초가 지난 후에 시험관의 상태는 다음과 같다.

1	1	2
1		2
3	3	

2 초가 지난 후에 시험관의 상태는 다음과 같다.

1	1	2
1	1	2
3	3	2

결과적으로 2 초가 지난 뒤에 (3,2)에 존재하는 바이러스의 종류는 3 번 바이러스다. 따라서 3 을 출력하면 정답이다.

입력 조건

첫째 줄에 자연수 N, K 가 공백을 기준으로 구분되어 주어진다. ($1 \leq N \leq 200, 1 \leq K \leq 1,000$)

둘째 줄부터 N 개의 줄에 걸쳐서 시험관의 정보가 주어진다. 각 행은 N 개의 원소로 구성되며, 해당 위치에 존재하는 바이러스의 번호가 공백을 기준으로 구분되어 주어진다. 단, 해당 위치에 바이러스가 존재하지 않는 경우 0 이 주어진다. 또한 모든 바이러스의 번호는 K 이하의 자연수로만 주어진다.

$N+2$ 번째 줄에는 S, X, Y 가 공백을 기준으로 구분되어 주어진다. ($0 \leq S \leq 10,000, 1 \leq X, Y \leq N$)

출력 조건

S 초 뒤에 (X, Y) 에 존재하는 바이러스의 종류를 출력한다. 만약 S 초 뒤에 해당 위치에 바이러스가 존재하지 않는다면, 0 을 출력한다.

입력 예시

```
3 3
1 0 2
0 0 0
3 0 0
2 3 2
```

출력 예시

```
3
```