

수열 축소

길이가 N인 수열이 주어지면 인접한 두 수의 차이를 이용해 길이가 N-1인 수열을 만듭니다. 만약 수열이 [5, 3, 7, 9, -2]라면 [(3-5), (7-3), (9-7), (-2-9)] => [-2, 4, 2, -11]로 수열의 길이를 줄일 수 있습니다. 이런 과정을 길이축소작업이라 하겠습니다.

N길이의 수열이 주어지면 M번의 길이축소작업을 한 결과를 구하는 프로그램을 작성하세요.

■ 입력설명

매개변수 nums에 $N(3 \leq N \leq 30)$ 길이의 수열이 주어지고, 매개변수 m에 $M(M < N)$ 이 주어집니다.

■ 출력설명

M번의 길이축소작업을 거친 수열을 반환합니다..

■ 매개변수 형식 1

[5, 3, 7, 9, -2], 1

■ 반환값 형식 1

[-2, 4, 2, -11]

■ 매개변수 형식 2

[5, 3, 7, 9, -2], 2

■ 반환값 형식 1

[6, -2, -13]

초기코드형식

```
function solution(nums, m){
    let answer;
    return answer;
}
console.log(solution([5, 3, 7, 9, -2], 1));
```

가장 높은 증가수열

길이가 N인 수열이 주어지면 이 수열에서 연속된 부분 증가수열을 찾습니다. 각 부분증가수열은 높이가 있습니다. 증가수열의 높이란 증가수열의 첫항과 마지막항의 차를 의미합니다. 수열이 주어지면 여러 증가수열 중 가장 높은 부분증가수열을 찾는 프로그램을 작성하세요. 만약 수열이 [5, 2, 4, 7, 7, 3, 9, 10, 11]이 주어지면 가장 높은 부분증가수열은 [3, 9, 10, 11]이고, 높이는 8입니다.

■ 입력설명

매개변수 nums에 $N(3 \leq N \leq 30)$ 길이의 수열이 주어집니다. 수열의 원소는 자연수입니다.

■ 출력설명

가장 높은 연속부분증가수열의 높이를 반환합니다.

■ 매개변수 형식 1

[5, 2, 4, 7, 7, 3, 9, 10, 11]

■ 반환값 형식 1

8

■ 매개변수 형식 2

[8, 12, 2, 3, 7, 6, 20, 3]

■ 반환값 형식 2

14

가장 긴 수열

길이가 N인 수열이 주어지면 이 수열에서 연속으로 증가하거나, 또는 연속으로 작아지는 부분 수열 중 가장 길이가 긴 수열을 찾는 프로그램을 작성하세요.

만약 [5, 3, 6, 7, 9, 8, 5, 3, 1, 2]이 주어지면 우리가 찾는 가장 긴 수열은 [9, 8, 5, 3, 1]입니다.

수열 [1, 2, 3, 3, 4, 5, 6]과 같이 같은 값이 연속으로 있는 것은 증가 또는 감소로 보지 않기 때문에 가장 긴 수열은 [3, 4, 5, 6]이 됩니다.

■ 입력설명

매개변수 nums에 $N(3 \leq N \leq 30)$ 길이의 수열이 주어집니다. 수열의 원소는 자연수입니다.

■ 출력설명

가장 긴 수열의 길이를 반환합니다.

■ 매개변수 형식 1

[5, 3, 6, 7, 9, 8, 5, 3, 1, 2]

■ 반환값 형식 1

5

■ 매개변수 형식 2

[5, 2, 4, 7, 6, 3, 9, 10, 11]

■ 반환값 형식 2

8

■ 매개변수 형식 3

[1, 2, 3, 3, 4, 5, 6, 7, 7]

■ 반환값 형식 3

5

바이토닉 수열

바이토닉 수열이란 수열이 증가했다가 감소하는 수열을 의미합니다.

길이가 N인 수열이 주어지면 이 수열이 바이토닉 수열인지 판별하는 프로그램을 작성하세요.

만약 [1, 2, 3, 4, 2, 1]이면 바이토닉 수열입니다. 하지만 [1, 2, 2, 3, 2, 1]과 같이 같은 값이 연속으로 있으면 바이토닉 수열이라 하지 않습니다.

■ 입력설명

매개변수 nums에 $N(3 \leq N \leq 30)$ 길이의 수열이 주어집니다. 수열의 원소는 자연수입니다.

■ 출력설명

바이토닉 수열이면 "YES", 아니면 "NO"를 반환합니다.

■ 매개변수 형식 1

[1, 2, 3, 4, 5, 3, 1]

■ 반환값 형식 1

YES

■ 매개변수 형식 2

[1, 3, 4, 5, 5, 6, 4, 3]

■ 반환값 형식 2

NO

■ 매개변수 형식 2

[1, 2, 3, 4, 5]

■ 반환값 형식 2

NO

거리 두기

현수는 영화관에 도착했습니다. 영화상영 시간보다 약간 늦은 현수는 남은 좌석을 빨리 선택하고 영화를 보려고 합니다.

현수에게 일렬로된 좌석정보가 주어지면, 이미 앉아 있는 사람들 중 가장 가까운 사람과 최대한 멀리 떨어져 앉을 자석을 선택해야 합니다. 여러분이 도와주세요.

■ 입력설명

매개변수 `nums`에 길이가 $N(3 \leq N \leq 100)$ 인 수열을 통해 좌석의 정보가 주어집니다. 좌석정보는 1은 이미 사람이 앉은 좌석이고 0의 빈 좌석입니다.

■ 출력설명

현수가 이미 앉은 사람과 최대한 멀리 앉을 수 있는 거리를 반환합니다.

■ 입력예제 1

[1, 0, 0, 0, 1, 0, 0, 1, 0, 1]

■ 출력예제 1

2

출력설명

총 10개의 좌석이 왼쪽부터 0번 좌석으로 해서 9번 좌석까지 존재한다면 현수가 2번 좌석에 앉으면 가장 가까운 사람과의 거리가 2가 됩니다. 만약 6번 좌석에 앉으면 가장 가까운 사람과의 거리는 1입니다.

2차원 배열 1의 개수

0과 1로 구성된 2차원 배열이 주어지면 각 행의 1의 개수를 세어 개수가 가장 작은 행번호부터 출력하는 프로그램을 작성하세요. 1의 개수가 같은 행은 여러개이면 행번호가 작은 것부터 출력합니다. 이차원배열의 행크기가 N이면 행번호는 0번부터 N-1번까지입니다.

■ 입력설명

매개변수 nums에 $N \times N$ ($2 \leq N \leq 100$) 크기의 이차원 배열이 주어집니다.

■ 출력설명

1의 개수에 의하여 정렬된 행 번호를 반환합니다.

■ 매개변수 형식 1

[[1, 0, 0, 1], [0, 0, 0, 1], [1, 1, 0, 1], [0, 1, 0, 1]]

■ 반환값 형식 1

[1, 0, 3, 2]

행과열의 최솟값

자연수로 채워져 있는 2차원 배열이 주어지면 행과열의 최솟값들을 구하는 프로그램을 작성하세요. 행과열의 최솟값이란 2차원 배열의 숫자가 자신이 속한 행과 자신이 속한 열에서 모두 가장 작은 숫자를 의미합니다.

2차원 배열에 존재하는 모든 행과열의 최솟값들을 찾아주는 프로그램을 작성하세요.

■ 입력설명

매개변수 nums에 $N \times N$ ($3 \leq N \leq 100$) 크기의 이차원 배열이 주어집니다. 2차원 배열의 원소들은 모두 다릅니다.

■ 출력설명

행과열의 최솟값들을 오름차순 정렬하여 반환합니다.

■ 매개변수 형식 1

[[4, 6, 22, 1], [9, 3, 10, 12], [30, 7, 20, 2], [15, 8, 5, 13]]

■ 반환값 형식 1

[1, 3, 5]

봉우리

지도 정보가 $N \times N$ 격자판에 주어집니다. 각 격자에는 그 지역의 높이가 쓰여있습니다. 각 격자판의 숫자 중 자신의 상하좌우 숫자보다 큰 숫자는 봉우리 지역입니다. 봉우리 지역이 몇 개 있는 지 알아내는 프로그램을 작성하세요.

격자의 가장자리는 0으로 초기화 되었다고 가정한다.

만약 $N=5$ 이고, 격자판의 숫자가 다음과 같다면 봉우리의 개수는 10개입니다.

0	0	0	0	0	0	0
0	5	3	7	2	3	0
0	3	7	1	6	1	0
0	7	2	5	3	4	0
0	4	3	6	4	1	0
0	8	7	3	5	2	0
0	0	0	0	0	0	0

■ 입력설명

매개변수 nums에 $N \times N$ ($2 \leq N \leq 100$) 크기의 격자판 정보가 주어집니다. 각 자연수는 100을 넘지 않는다.

■ 출력설명

봉우리의 개수를 반환하세요.

■ 매개변수 형식

[[5, 3, 7, 2, 3], [3, 7, 1, 6, 1], [7, 2, 5, 3, 4], [4, 3, 6, 4, 1], [8, 7, 3, 5, 2]]

■ 반환값 형식

10