

부등호

baekjoon 2529

202021018 오윤주

문제

2 2529번

제출

맞힌 사람

숏코딩

재채점 결과

채점 현황

내 제출

🔗 난이도 기여

강의▼

질문 검색

부등호

성공



2 실버 II

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	17902	9743	6582	52.681%

문제

두 종류의 부등호 기호 '<'와 '>'가 k개 나열된 순서열 A가 있다. 우리는 이 부등호 기호 앞뒤에 서로 다른 한 자릿수 숫자를 넣어서 모든 부등호 관계를 만족시키려고 한다. 예를 들어, 제시된 부등호 순서열 A가 다음과 같다고 하자.

$$A \Rightarrow <<<><<><>$$

부등호 기호 앞뒤에 넣을 수 있는 숫자는 0부터 9까지의 정수이며 선택된 숫자는 모두 달라야 한다. 아래는 부등호 순서열 A를 만족시키는 한 예이다.

$$3 < 4 < 5 < 6 > 1 < 2 < 8 > 7 < 9 > 0$$

이 상황에서 부등호 기호를 제거한 뒤, 숫자를 모두 붙이면 하나의 수를 만들 수 있는데 이 수를 주어진 부등호 관계를 만족시키는 정수라고 한다. 그런데 주어진 부등호 관계를 만족하는 정수는 하나 이상 존재한다. 예를 들어 3456128790 뿐만 아니라 5689023174도 아래와 같이 부등호 관계 A를 만족시킨다.

$$5 < 6 < 8 < 9 > 0 < 2 < 3 > 1 < 7 > 4$$

여러분은 제시된 k개의 부등호 순서를 만족하는 (k+1)자리의 정수 중에서 최댓값과 최솟값을 찾아야 한다. 앞서 설명한 대로 각 부등호의 앞뒤에 들어가는 숫자는 { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 } 중에서 선택해야 하며 선택된 숫자는 모두 달라야 한다.

문제

```
K = int(input())  
B = input().split()
```

// 수의 개수
// 부등호

```
visit = [0] * 10  
max = ""  
min = ""
```

```
def check(i, j, k): // 부등호를 입력해서 수의 크기 비교 해주는 함수  
    if k == '>':  
        return i > j  
    else:  
        return i < j
```

문제

```
def backtracking(n, m):  
    global max, min  
  
    if(n == K+1):  
        if(len(min) == 0): // 맨처음 생기는 값이 최소이고  
            min = m        // 마지막에 나오는 값이 최대이다  
        else:  
            max = m  
        return  
  
    for i in range(10):  
        if(visit[i] == 0): // 가능한 곳에서  
            if(n == 0 or check(m[-1], str(i), B[n-1])): // 처음이거나 그 다음 수  
                visit[i] = True  
                backtracking(n + 1, m + str(i)) // 백트래킹으로 값 업데이트  
                visit[i] = False  
  
backtracking(0, "")  
print(max) // 결과 출력  
print(min)
```

감사합니다.