

14002 가장 긴 증가하는 바이토닉 수열

2021년 2월 5일 금요일 오후 5:15

가장 긴 증가하는 부분 수열 4

4 Gold IV

난이도 제공: solved.ac

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	256 MB	10245	4023	3068	40.464%

문제

수열 A가 주어졌을 때, 가장 긴 증가하는 부분 수열을 구하는 프로그램을 작성하시오.

예를 들어, 수열 A = {10, 20, 10, 30, 20, 50} 인 경우에 가장 긴 증가하는 부분 수열은 A = {10, 20, 10, 30, 20, 50} 이고, 길이는 4이다.

입력

첫째 줄에 수열 A의 크기 N ($1 \leq N \leq 1,000$)이 주어진다.

둘째 줄에는 수열 A를 이루고 있는 A_i 가 주어진다. ($1 \leq A_i \leq 1,000$)

출력

첫째 줄에 수열 A의 가장 긴 증가하는 부분 수열의 길이를 출력한다.

둘째 줄에는 가장 긴 증가하는 부분 수열을 출력한다. 그러한 수열이 여러가지인 경우 아무거나 출력한다.

예제 입력 1

```
6
10 20 10 30 20 50
```

10 20 10 30 20 50

```
N = int(input())
arr = list(map(int, input().split()))
# 입력받은 N만큼 배열에 삽입
dp = [1]*N
# N까지 반복
for i in range(1,N):
    for j in range(i):
        # 앞 수가 더 클 경우 i와 j의 당므번째 수를 하여 넣어준다
        if arr[i] > arr[j]:
            dp[i] = max(dp[i], dp[j]+1)
print(max(dp))
order = max(dp)
lst = []
for i in range(N-1, -1, -1):
    if dp[i] == order:
        lst.append(arr[i])
        order -= 1
lst.reverse()
for i in lst:
    print(i, end = ' ')
```

1 2 3 4 5 6

4

dp = 4

1 2 1 3 2 4

10 20 10 30 20 50

예제 출력 1

```
4
10 20 30 50
```

```
N = int(input())
# 배열에 삽입
arr = list(map(int, input().split()))
# N번 1을 삽입한 list 생성
dp = [1] * N
for i in range(N):
    for j in range(i):
        # arr[i]의 기준에서 arr[j]보다 클 경우 쌓아놓은 dp[j]의 위치보다 +1일 것이다
        # 왜냐하면 해당 위치 다음에 올 것이기 때문에
        if arr[i] > arr[j]:
            dp[i] = max(dp[i], dp[j]+1)
print(max(dp))
# 뒤에서부터 반복문을 돌릴 것
order = max(dp)
result = []
for i in range(N-1, -1, -1):
    if order == dp[i]:
        result.append(arr[i])
        order -= 1
result.reverse()
for i in result:
    print(i, end=" ")
```

3

10 20 10 30 20 50

1 2 1 3 2 4