## 백준 18353 병사 배치하기

# 003 코드 설명



## N (1<= N <= 2,000) 전투력 (10,000,000보다 작거나 같은 자연수)

N명의 병사가 무작위로 나열되어 있다. 각 병사는 특정한 값의 전투력을 보유하고 있으며, 병사를 배치할 때는 전<mark>투력이 높은 병사가 앞쪽에 오도록 내림차순으로 배치</mark>를 하고자 한다. 다시 말해 앞쪽에 있는 병사의 전투력 이 항상 뒤쪽에 있는 병사보다 높아야 한다.

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

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

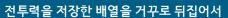
병사 번호	1	2	3	4	5	6	7
전투력	15	11	4	8	5	2	4

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

병사 번호	1	2	4	5	7
전투력	15	11	8	5	4

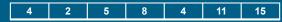
병사에 대한 정보가 주어졌을 때, 남아있는 병사의 수가 최대가 되도록 하기 위해서 열외해야 하는 병사의 수를 출력하는 프로그램을 작성하시오.

## # 아이디어



15	11	4	8	5	2	4

가장 긴 증가하는 수열 문제에서 사용한 알고리즘 사용



증가하는 수열의 길이 저장

dp 1 1 2 3 2 4 5

## # 코드 설명

```
import sys
N = int(sys.stdin.readline())
power = list(map(int, sys.stdin.readline().split()))
power.reverse()
              전투력저장한 배열 거꾸로 뒤집기
dp = [0] * N 0으로 초기화
for i in range(N): 배열처음~끝
   max_dp = 0 0으로 초기화
   for j in range(i): 0부터 i 전까지
      if power[i] > power[j]: 크기비교
          if max_dp < dp[j]:
             max_dp = dp[j] 최대값을 max dp에 저장
   dp[i] = max_dp + 1 dp 저장 (+1 = 본인 포함)
print(N-max(dp)) N - 가장 긴 증가하는 수열의 길이
```

감사합니다