



1-2. King of Chairs

의자왕은 궁녀를 선발하기 위하여 궁궐에서 면접을 보려고 한다. 면접을 보려고 하는 궁녀 후보생은 총 N 명이며 이들의 몸무게는 궁궐에 오는 순서대로 각각 $W[0], W[1], \dots, W[N-1]$ 이다. 의자왕은 사전 조사를 하여 $W[i]$ 의 값을 미리 알아놓았다. 궁녀의 몸무게는 변하지 않는다.

의자왕은 의자덕후답게 면접 대기실에 N 개의 다채로운 의자를 궁궐에 들여놓았다. 각 의자의 강도는 $C[0], C[1], \dots, C[N-1]$ 로 j 번 의자에는 몸무게가 $C[j]$ 이하인 사람 한 명만 앉을 수 있다. 만약 몸무게가 $C[j]$ 보다 무거운 사람이 의자에 앉아서 의자가 부서지면 의자왕이 격노할 것이다. 왕은 $N, W[i], C[j]$ 를 모두 알고 있다.

의자왕은 면접장에서 직접 면접을 하러 앉아있기 때문에 면접 대기실에는 궁녀 후보생들과 신하들밖에 없다. 면접 대기실에 궁녀 후보생이 들어오는 대로 신하들은 이 후보생을 앉힐 지 서있게 할 지 결정하여 실행한다. 궁녀 후보생은 모든 후보생이 도착할 때까지 계속 같은 의자에 앉아있거나 계속 서있어야 한다. 의자왕은 궁녀 후보생들을 배려하는 차원에서 최대한 많은 궁녀들이 의자에 앉을 수 있게 하라고 신하들에게 명령을 내렸다. 모든 정보를 알고 있는 왕이 최대 q 명의 궁녀 후보생을 앉힐 수 있다는 것을 확인했다면, 신하들은 q 명의 궁녀를 앉혀야 한다.

그러나 신하들은 오직 $N, C[j]$ 만 알고 있고 궁녀 후보생들의 몸무게를 모른다. 물론 면접 대기실에 체중계가 있어서 궁녀 후보생이 오면 몸무게를 확인할 수 있으나 궁녀 후보생이 궁궐에 오기 전까지는 몸무게를 알 방법이 없다. 문제를 타개하기 위해 의자왕은 쪽지로 숫자를 보내 신하들이 앞으로 올 궁녀들의 정보를 대충 파악할 수 있도록 하려고 한다.

영의정인 당신은 의자왕과 면접 대기실에 있는 신하들을 보좌하여 숫자를 만드는 규칙과 숫자를 해석하는 방법, 그리고 궁녀 후보생들을 앉히는 방법을 알려줘야 한다. 서쪽에서 들어온 신문물 컴퓨터를 활용하여 의자왕과 신하들을 도와주자.

요구 사항

당신은 파일 두 개를 제출해야 한다.

첫 번째 파일은 `king.cpp`이다. 이 파일에서 당신은 다음 함수를 구현해야 한다. 이 파일은 `king.h`와 같이 컴파일된다.

```
int64 SendInfo(int[] W, int[] C)
```

- W : 길이 N 의 배열. $W[i]$ ($0 \leq i \leq N-1$)는 i 번 궁녀 후보생의 몸무게이다.
- C : 길이 N 의 배열. $C[j]$ ($0 \leq j \leq N-1$)는 j 번 의자의 강도이다.
- 이 함수는 신하에게 보낼 수를 반환한다. 반환값은 음이 아닌 정수여야 한다.
- 이 함수는 한 번 호출된다.

두 번째 파일은 `vassal.cpp`이다. 이 파일에서 당신은 다음 함수를 구현해야 한다. 이 파일은 `vassal.h`와 같이 컴파일된다.

```
void Init(int64 B, int[] C)
```

- B : 왕이 보낸 수. 이 값은 `SendInfo` 함수의 반환값과 같다.
- C : 길이 N 의 배열. 이 값은 `SendInfo` 함수의 인자 C 와 같다.
- 이 함수는 한 번 호출된다.

```
int Maid(int w)
```

- w : 면접장에 들어온 궁녀 후보생의 무게.
- 이 함수는 궁녀 후보생을 앉힐 의자의 번호 (0 이상 $N - 1$ 이하)를 반환한다. 만약 궁녀 후보생을 앉히지 않는다면 -1 을 반환한다.
- 함수의 반환값이 이미 궁녀 후보생이 배정된 의자의 번호이면 안 된다.
- 이 함수는 `Init` 함수가 끝나고 나서 N 번 호출된다.
- 의자에 앉은 궁녀의 수가 최대가 되어야 한다.

위에 언급된 조건을 만족하지 않거나 궁녀를 최대한 앉히지 못했다면, 당신의 프로그램은 **Wrong Answer**으로 채점된다. 나머지 경우에는 당신의 프로그램은 **Accepted**으로 채점된다.

채점 과정

채점은 아래 절차대로 진행된다. 당신의 프로그램이 **Wrong Answer**임이 확실하다면 프로그램이 중간에 끝날 수 있다.

1. `SendInfo` 함수가 한 번 호출된다.
2. `Init` 함수가 한 번 호출된다.
3. `Maid` 함수가 N 번 호출된다. $(k + 1)$ 번째 호출에 대해서 ($0 \leq k \leq N - 1$), 인자 w 의 값은 $W[k]$ 이다. `Maid` 함수의 반환값을 통해, 의자에 앉은 궁녀의 수 q 를 계산한다.
4. q 가 최적해라면 당신의 프로그램은 **Accepted**으로 채점된다.

중요 공지사항

- 실제 채점 시에는 제출한 두 프로그램이 독립적으로 컴파일 및 실행된다.
- 첫 번째 프로그램이 비정상적으로 끝났으면 (**Runtime Error**, **Time/Memory Limit Exceeded**), 두 번째 프로그램을 실행하지 않고 바로 **Wrong Answer**로 채점된다.
- 시간 및 메모리 사용량은 각 프로그램의 사용량 합으로 계산된다.
- 작성한 프로그램에서 표준 입력과 표준 출력을 사용하면 안 되고, 어떠한 파일에도 접근하면 안 된다. 이를 어길 시 당신의 프로그램이 **Wrong Answer**로 채점될 수 있다.

제한

- $1 \leq N \leq 100\,000$
- $1 \leq W[i] \leq 1\,000\,000$ ($0 \leq i \leq N - 1$)
- $1 \leq C[j] \leq 1\,000\,000$ ($0 \leq j \leq N - 1$)

부분문제

1. (17점) $1 \leq N \leq 10$.
2. (38점) W 는 비내림차순으로 정렬되어 있다. 즉, $W[0] \leq W[1] \leq \dots \leq W[N - 1]$.
3. (45점) 추가 제약 조건 없음.

당신의 프로그램이 **Accepted**으로 채점되었다면, 당신의 점수 P 는 왕이 보낸 정수 B 의 값에 따라 결정된다. 점수를 계산하는 규칙은 다음과 같다.

- 부분문제 1. 만약 $0 \leq B \leq 2^{60}$ 이면, $P = 17$. 그렇지 않다면, $P = 0$.
- 부분문제 2. 만약 $0 \leq B \leq 2^{20}$ 이면, $P = 38$. 그렇지 않다면, $P = 0$.
- 부분문제 3.
 - $B \geq 2^{60}$ 이면, $P = 0$.
 - $2^{20} \leq B < 2^{60}$ 이면, $P = \lfloor 40 \times 0.97^{\lceil \log_2(B+1) \rceil - 20} \rfloor$.
 - $B = 2\,147\,483\,647$ 이면, $P = 28$.
 - $X < B < 2^{20}$ 이면, $P = 40$.
 - $B \leq X$ 이면, $P = 45$.
 - X 의 값은 대회에서 공개하지 않는다.

당신의 점수는 가장 낮은 점수를 받은 테스트 케이스의 점수이다.

예제

king.cpp에서 발생한 다음 호출을 고려해보자.

```
SendInfo([3, 6, 5], [3, 4, 5])
```

SendInfo 함수가 365를 반환했다고 가정하자.

vassal.cpp에서 다음 호출이 발생한다.

```
Init(365, [3, 4, 5])
Maid(3)
Maid(6)
Maid(5)
```

당신의 프로그램이 **Accepted**으로 채점되려면, 세 Maid 호출의 반환값은 0, -1, 2 또는 1, -1, 2

여야 한다.

샘플 인터페이스

문제 페이지에서 샘플 코드를 다운로드받을 수 있다. 만약 Visual Studio나 Eclipse, Code::Blocks와 같은 IDE 툴을 사용한다면 `king.cpp`, `king.h`, `vassal.cpp`, `vassal.h`, `grader.cpp`를 한 프로젝트에 넣어서 컴파일하면 된다.

터미널에서 코드를 컴파일한다면 아래 컴파일 명령어를 이용하면 된다.

```
g++-7 -Wall -lm -static -DEVAL -o chairs -O2 grader.cpp king.cpp vassal.cpp -std=c++17
```

답안을 제출할 때에는 `king.cpp`와 `vassal.cpp`를 제출하면 된다.

Input format

- line 1: N
- line 2: $W[0] \ W[1] \ \dots \ W[N-1]$.
- line 3: $C[0] \ C[1] \ \dots \ C[N-1]$.

Output format

당신의 프로그램이 오류 없이 궁녀 후보생을 앉혔다면, 샘플 그레이더는 첫 번째 줄에 **Correct**를 출력하고 두 번째 줄에 앉힌 궁녀 후보생의 수 q 를 출력한다. q 가 실제 정답이 아닐 수 있음에 유의하여라.

당신의 프로그램이 **Wrong Answer**으로 채점되었다면, 샘플 그레이더는 첫 번째 줄에 에러 메시지를 출력한다.