

# 카드 합체 놀이

백준 15903

202021018 오윤주

# 문제

## 카드 합체 놀이 출처



2 실버 II

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초 (추가 시간 없음)	512 MB	5810	2354	1968	41.624%

## 문제

석환이는 아기다. 아기 석환이는 자연수가 쓰여져있는 카드를 갖고 다양한 놀이를 하며 노는 것을 좋아한다. 오늘 아기 석환이는 무슨 놀이를 하고 있을까? 바로 카드 합체 놀이이다!

아기 석환이는 자연수가 쓰여진 카드를  $n$ 장 갖고 있다. 처음에  $i$ 번 카드엔  $a_i$ 가 쓰여있다. 카드 합체 놀이는 이 카드들을 합체하며 노는 놀이이다. 카드 합체는 다음과 같은 과정으로 이루어진다.

1.  $x$ 번 카드와  $y$ 번 카드를 골라 그 두 장에 쓰여진 수를 더한 값을 계산한다. ( $x \neq y$ )
2. 계산한 값을  $x$ 번 카드와  $y$ 번 카드 두 장 모두에 덮어 쓴다.

이 카드 합체를 총  $m$ 번 하면 놀이가 끝난다.  $m$ 번의 합체를 모두 끝낸 뒤,  $n$ 장의 카드에 쓰여있는 수를 모두 더한 값이 이 놀이의 점수가 된다. 이 점수를 가장 작게 만드는 것이 놀이의 목표이다.

아기 석환이는 수학을 좋아하긴 하지만, 아직 아기이기 때문에 점수를 얼마나 작게 만들 수 있는지를 알 수는 없었다(어른 석환이는 당연히 쉽게 알 수 있다). 그래서 문제 해결 능력이 뛰어난 여러분에게 도움을 요청했다. 만들 수 있는 가장 작은 점수를 계산하는 프로그램을 만들어보자.

# 문제

예제 입력 2 복사

```
4 2
4 2 3 1
```

예제 출력 2 복사

```
19
```

카드 4장, 2번 더하기

$2 + 1 = 3$   
>> 4 3 3 3

$3 + 3 = 6$   
>> 4 6 6 3

$4 + 6 + 6 + 3 = 19$

# 구현

```
import heapq

n, m = map(int, input().split()) // 카드개수 n, 계산 횟수 m 입력받기

card = list(map(int, input().split())) // 카드 입력, 큐 생성
queue = []
```

# 구현

```
for i in range(n):  
    heapq.heappush(queue, card[i]) // 카드 힙에 넣어주기  
  
for i in range(m):  
    x = heapq.heappop(queue)  
    y = heapq.heappop(queue) // x, y 값 힙 pop (최솟값)  
  
    heapq.heappush(queue, x+y)  
    heapq.heappush(queue, x+y) // x+y 값 다시 힙에 push  
  
print(sum(queue)) // 전체 큐 값의 합
```

**감사합니다.**