

BAEKJOON - 16401

과자 나눠 주기

202021020 우 나 현

16401 - 과자 나눠 주기

문제

과자 나눠주기

성공출처3 실버 III

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	256 MB	1652	684	536	41.550%

문제

명절이 되면, 홍익이 집에는 조카들이 놀러 온다. 떼를 쓰는 조카들을 달래기 위해 홍익이는 막대 과자를 하나씩 나눠준다.

조카들이 과자를 먹는 동안은 떼를 쓰지 않기 때문에, 홍익이는 조카들에게 최대한 긴 과자를 나눠주려고 한다.

그런데 나눠준 과자의 길이가 하나라도 다르면 조카끼리 싸움이 일어난다. 따라서 반드시 모든 조카에게 같은 길이의 막대 과자를 나눠주어야 한다.

M명의 조카가 있고 N개의 과자가 있을 때, 조카 1명에게 줄 수 있는 막대 과자의 최대 길이를 구하라.

단, 막대 과자는 길이와 상관없이 여러 조각으로 나눠질 수 있지만, 과자를 하나로 합칠 수는 없다. 단, 막대 과자의 길이는 양의 정수여야 한다.

문제

여러 개의 막대과자들을 같은 크기 중에 제일 크게
조카들의 수에 맞춰서 잘라내야 하는 문제.

예를 들어 조카는 4 명 과자는 3개
과자가 10 10 15 라면
과자를 7로 쪼개야 가장 크게 조카들에게
나누어 줄 수 있다.

접근

따라서 우선 과자들 크기가 들어있는 배열을
Sort함수로 오름차순 정렬 후,

제일 큰 15를 반으로 나눈 값을 기준(mid)으로 삼아서
반복문을 통해 과자를 쪼개 본다.

여기서 변수 $left=1, right=A[N-1], mid = (left+right)/2$
그리고 만약 쪼개서 나오는 과자의 수와 조카 수를 비교하여
변수 $left$ 와 $right$ 에 변화를 주는 방식으로
이진 탐색을 사용한다.

코드

```
#include<iostream>
#include<algorithm>
using namespace std;
int main() {
    long long M, N;
    cin >> M >> N;

    long long* L = new long long[N];
    for (int i = 0; i < N; i++) {
        cin >> L[i];
    }
    sort(L, L + N);

    long long mid = 0;
    long long left = 1;
    long long right = L[N - 1];
    long long cut = 0;
```

정렬을 위해 algorithm 선언

입력 받을 변수들을 선언 해 준 후
차례 대로 입력 받아준다.

밑에 mid는 이진탐색을 위한 중간값
Left를 1로 초기화 하여 0으로 나눌 시
나타날 런타임 에러를 방지해 준다.
Right 는 비교할 변수들 중 제일 큰 수.

Cut은 잘라진 과자들 개수를 저장.

코드

```
while (left <= right) {
    cut = 0;
    mid = (left + right) / 2;
    for (int i = 0; i < N; i++) {
        cut = (L[i] / mid) + cut;
    }
    if (cut < M)
        right = mid - 1;
    else
        left = mid + 1;
}
cout << right;

return 0;
```

이진탐색 이므로 left가 right를 넘거나
같은 경우엔 반복문을 탈출.

과자 수를 0으로 계속 초기화
Mid는 left와right의 중간 값.

Cut에 자르지 않은 과자들을 mid로 나
눠 준 값들을 넣어준다. (몇 개의 과자
가 나오는지)

그리고 과자 수가 조카 수보다 작으면
right = mid - 1 크거나 같으면 left =
mid + 1

그리고 right값을 출력해준다.

감 사 합 니 다