



백준 1092문제

# 문제

## 문제

지민이는 항구에서 일한다. 그리고 화물을 배에 실어야 한다. 모든 화물은 박스에 안에 넣어져 있다. 항구에는 크레인이  $N$ 대 있고, 1분에 박스를 하나씩 배에 실을 수 있다. 모든 크레인은 동시에 움직인다.

각 크레인은 무게 제한이 있다. 이 무게 제한보다 무거운 박스는 크레인으로 움직일 수 없다. 모든 박스를 배로 옮기는데 드는 시간의 최소값을 구하는 프로그램을 작성하시오.

## 입력

첫째 줄에  $N$ 이 주어진다.  $N$ 은 50보다 작거나 같은 자연수이다. 둘째 줄에는 각 크레인의 무게 제한이 주어진다. 이 값은 1,000,000보다 작거나 같다. 셋째 줄에는 박스의 수  $M$ 이 주어진다.  $M$ 은 10,000보다 작거나 같은 자연수이다. 넷째 줄에는 각 박스의 무게가 주어진다. 이 값도 1,000,000보다 작거나 같은 자연수이다.

## 출력

첫째 줄에 모든 박스를 배로 옮기는데 드는 시간의 최소값을 출력한다. 만약 모든 박스를 배로 옮길 수 없으면 -1을 출력한다.

무게제한이 있는  $N$ 개의 크레인을 이용해서  $M$ 개의 화물을 최소시간으로 옮겨라

# 정렬

화물의 무게에 따라 옮길 수 있는 크레인이 제한됨

-> 화물과 크레인을 정렬한 뒤, 각 크레인이 들 수 있는 화물의 개수를 셈 (이전 크레인이 들 수 있는 화물이면 세지 않음) 범위에 따라 카운팅!

예제 입력 1 복사

3					
6	8	9			
5					
2	5	2	4	7	

① 정렬

$crain = [6, 8, 9]$

$box = [2, 2, 4, 5, 7]$

② counting

6 {2, 2, 4, 5}

8 {7}

9 { }

$count = [4, 1, 0]$

# 그리디

카운팅한 화물의 개수를 최대한 균등하게 나누어주어야함 (최소시간을 구하기위해)

뒤에 위치한 크레인의 화물은 앞에 위치한 크레인이 들 수 없으므로  
반복문을 돌면서 앞의 크레인부터 화물을 배분해줌

-현재 크레인(i)부터 마지막크레인(n)까지 들어야하는 화물의 개수 K개를 크레인  
개수로 나누어 (n-i) -> 요약하면 평균 화물의 개수를 구함  
현재 크레인이 몇개의 화물을 들어야 시간이 덜 걸릴 수 있는지를 구함

# 그리디

## 예제 입력 1 복사

```
3
6 8 9
5
2 5 2 4 7
```

counting  $\begin{matrix} 6 & 8 & 9 \\ 4 & 1 & 0 \end{matrix}$

(i=0) 크레인 6이 들 수 있는 화물의 수 = 4  
(남은 화물 5) 현재 화물의 무게 = (5 // 3) = 1  
 $[1, 4, 0]$   
→ 나머지 화물은 다음 크레인에게 넘김

(i=1) 크레인 8이 들 수 있는 화물의 수 = 4  
(남은 화물 4) 현재 화물의 무게 = (4 // 2) = 2  
 $[1, 2, 2]$

# 알고리즘

```
int main() {  
    cin >> N;  
    vector<int>crain(N);  
  
    for (int i = 0; i < N; i++)  
        cin >> crain[i];  
  
    cin >> M;  
    vector<int>box(M);  
  
    for (int i = 0; i < M; i++)  
        cin >> box[i];  
  
    sort(crain.begin(), crain.end());  
    sort(box.begin(), box.end());  
  
    cout << getCargoTime(crain, box) << "Wn";  
  
    return 0;  
}
```

# 알고리즘

```
int N, M;
int getCargoTime(vector<int>& crain, vector<int>& box) {
    if (crain.back() < box.back())
        return -1;

    vector<int> time(N, 0);

    // 범위에 따라 카운팅
    int j = 0;
    for (int i = 0; i < box.size(); i++) {
        while (crain[j] < box[i])
            j++;

        time[j]++;
    }

    int total = M;
    for (int i = 0; i < N - 1; i++) {
        int mid = (total / (N - i));
        if (time[i] > mid) {
            time[i + 1] = time[i + 1] + time[i] - mid;
            time[i] = mid;
        }

        total -= time[i];
    }

    int ans = 0;
    for (int i = 0; i < N; i++)
        ans = max(ans, time[i]);

    return ans;
}
```

크레인이 들 수 없는 무게면 -1

크레인이 들 수 있는 화물을 카운팅

남은 화물의 수 total로 화물의 평균을 구해줌 (반)

화물의 최대시간을 구하고 반환

**감사합니다**