

2217 로프 문제 풀이

완탐은 시간복잡도가 2^N 이라 최적화를 해야한다.

문제 설명은 쉽다. 유의깊게 읽어야할 부분은

1. 각 로프는 N개가 아니라 k개를 선택할 수 있음 -> k개 1부터 다 따져야할수도..?
2. 각 로프는 w/k 의 부하가 걸린다.

직관적으로 일단 가장 높은 무게를 들 수 있는 로프는 우선적으로 쟁여놔야 한다.

로프는 또 k개 자유롭게 고를 수 있다. 이 중에서 최대의 무게를 골라야 한다.

=> 인덱스를 요구하지 않으니 정렬해도 상관없는 문제유형이란걸 조금 의식적으로 보기!!

정렬 후 -1번째 인덱스부터 순차적으로 k개를 고를 때 최대중량을 `max()`로 선별하면 된다.

이때 포인트는 로프 세트들이 들 수 있는 무게는 로프 세트 중 가장 낮은 중량을 들 수 있는 로프, 즉 N-k번째 로프의 중량에 달려있다.

따라서 로프 세트들이 들 수 있는 무게 = `arr[N-k] * k`이다. (부하중량이 넘겨버리면 안되니까 가장 작은 로프의 부하중량을 key로 봐야 했다)

```
int main(){
    // n개의 로프 중 k개를 사용해 중량이 w인 물체를 들어올린다'
    // 이때 각 로프는 w/k의 중량이 걸릴 때 가능한 물체의 최대 중량을 구하라
    cin>>N;
    for(int i=0;i<N;i++) cin>>arr[i];
    sort(arr, arr + N);
    // 로프를 정렬 후 k개를 뒤에서 고를 때 그 로프가 들 수 있는 중량은
    // k개 중 가장 낮은 중량의 로프의 무게 * k이다
    for(int i=1; i<=N; i++){
        int tmp = arr[N-i]*i;
        ans = max(ans, tmp);
    }
    cout<<ans;
}
```