제2회 보라매컵 본선 해설

2023년 9월 30일

문제		의도한 난이도	출제자
A	병영외 급식	**Easy**	yooyou7
B	전역 임무	**Easy**	junhyung9985
C	차량 배치	**Medium**	junhyung9985
D	휴가 나가기	**Medium**	kybs0627
E	식당지원 차출	**Hard**	sangmin7b
F	참호전	**Hard**	asdarwin03
G	BX 내기	**Challenging**	junhyung9985
H	생활관 건설하기	**Challenging**	99asdfg

A. 병영외 급식

`math`, `arithmetic`, `ad hoc`, `constructive`
출제진 의도 – **Easy**

- 제출 477번, 정답 307명 (정답률 65.828%)
- 처음 푼 사람: **jihwan_0319**, 0분
- 출제자: yooyou7

A. 병영외 급식

- 행복하게 병영외 급식을 즐기는 방법이 있다고 가정해봅시다.
- 해당 방법의 모든 그룹을 하나로 합치면, 합친 그룹에 속한 병사의 수 또한 X 로 나누어떨어질 것입니다.
- 즉, 가능한 방법이 있다면 모든 생활관의 병사의 수의 합은 X 로 나누어떨어져야 합니다.
- 반대로 생각해보면, 모든 생활관의 병사의 수가 X 로 나누어떨어지지 않는다면 모든 그룹의 병사 수가 X 로 나누어떨어지는 경우는 없습니다.
- 모든 생활관을 하나의 그룹으로 묶는 것 또한 유효한 그룹이므로, 전체 병사의 수가 X 로 나누어떨어지는지에 따라 답을 출력해주면 됩니다.

B. 전역 임무

`greedy`, `sorting`
출제진 의도 – **Easy**

- 제출 957번, 정답 204명 (정답률 22.780%)
- 처음 푼 사람: **riroan**, 6분
- 출제자: `junhyung9985`

B. 전역 임무

- 임무를 완료하기 위해서는 어느 순서로 적군들을 격파하는게 제일 유리할까요?
- 당연히, 현재 격파가 가능한 적군들을 모두 격파하여 전투력을 가능한만큼 키우는게 제일 유리합니다.
- 그렇다면, 적군들은 제일 전투력이 낮은 적군들부터 순서대로 격파하는게 제일 유리합니다.
- 이는 각 기지별로 정렬하여 $\mathcal{O}\left(NM\log M\right)$ 내로 가능합니다.

B. 전역 임무

- 그렇다면, 아이템이 있는 층은 언제 들리는게 제일 유리할까요?
- 아이템의 역할은 현재까지 얻은 전투력 + 기본 전투력을 두 배로 늘려주는 역할을 합니다.
- 이는 현재까지 격파한 적군들로 부터 얻은 전투력을 두 배로 증폭시키는 역할도 함으로, 전투력이 낮은 순서대로 격파하다가 본인보다 전투력이 높은 적을 만나거나, 전부 격파하고 아이템 방만 남았을 경우에 들리는게 제일 유리합니다.
- 이는 정렬을 하면서 -1인 값들이 몇 개가 있었는지를 세고, 필요한 경우에 들려주는 식으로 구현하면 됩니다.

B. 전역 임무

- ✓ 따라서 총 시간복잡도는 $\mathcal{O}(NM \log M)$ 입니다.
- ✓ 다만 32-bit 정수형을 사용하는 경우에는 그냥 전투력을 계속 끝없이 올린다면 오버플로우가 날 수 있기 때문에, 32-bit 정수형을 사용할 때는 본인의 전투력이 적군의 최대 전투력인 10^9 이상이 되었을 때부터는 전투력을 그만 올리도록 구현해야 합니다.

C. 차량 배치

`graph_traversal`, `graphs`, `bfs`, `combinatorics`
출제진 의도 – **Medium**

- 제출 191번, 정답 74명 (정답률 40.314%)
- 처음 푼 사람: **dabbler1**, 10분
- 출제자: `junhyung9985`

C. 차량 배치

- 차량들을 배치하기 전에, 규칙에 따라서 차량들은 1 번 지점을 향해서 최단 거리로 이동합니다.
- 각 지점을 노드, 도로를 거리가 1 인 간선으로 보았을 때 차량들이 이동하는 경로는 1 번 지점을 기준으로 각 모든 지점들을 향해 최단거리로 움직이는 경로들이 됩니다.
- 경로들의 비용이 1 로 동일함에 따라서 최단 거리 경로들을 그릴때, 사이클이 생길 수가 없습니다. 따라서 결과는 1 번 노드가 루트인 트리가 그려집니다.
- 이는 BFS를 활용해서 쉽게 찾을 수 있습니다.

C. 차량 배치

- 이렇게 경로들이 트리 형태로 나타났다는 점을 알면서 배차를 시작할 때, 같은 시간에 두 대 이상의 차량들이 같은 지점에 도달하는 것은 어떤 경우인지 생각해봅시다.
- 일단, 임의의 두 노드가 1 번 노드로 향해서 계속 움직일 때 경로가 겹치는 구간은 두 노드의 최소 공통 조상부터 겹칩니다.
- 최소 공통 조상에서 두 노드가 동시에 만난다는 것은 최소 공통 조상에서부터 두 노드의 깊이가 동일하다는 뜻이됩니다.
- 편의상 최소 공통 조상으로부터의 두 노드의 깊이가 각각 p, q라 하고, 1 번 노드로부터의 최소 공통 조상의 깊이를 r이라고 합시다.
- 그렇다면 $p = q$를 만족하므로, $p + r = q + r$도 만족합니다. 따라서, 두 노드가 1 번 노드로부터의 깊이가 동일하면 교통사고가 납니다.

C. 차량 배치

- 결국에는 배차할 때, 1 번 노드로부터의 같은 깊이의 노드들을 고르지 않는 경우의 수가 몇 개인지를 계산하면 됩니다.
- N 개의 노드들이 있고, 1 번 노드의 깊이가 1 일 때, d_h 를 깊이가 h 인 노드들의 개수라고 한다면, 아래와 같은 식으로 표현할 수 있습니다.
- $d_1 + d_2 + \cdots + d_N + d_1 \times d_2 + d_1 \times d_3 + \cdots + d_1 \times d_2 \times \cdots \times d_N = (d_1 + 1) \times (d_2 + 1) \times \cdots \times (d_N + 1) - 1$
- 따라서, 각 노드들의 1 번 노드로부터의 깊이를 구하면서 깊이 별의 노드들의 개수들도 구하고 위 식을 적용하여 풀면 됩니다.
- 총 시간복잡도는 BFS의 시간복잡도인 $\mathcal{O}(N + M)$ 입니다.

D. 휴가 나가기

`graph, dp`
출제진 의도 – **Medium**

- 제출 96번, 정답 27명 (정답률 28.125%)
- 처음 푼 사람: **hyperbolic**, 23분
- 출제자: kybs0627

D. 휴가 나가기

- 우선 모든 업무가 선행 업무를 가지고 있지 않다고 가정 해 봅시다.
- w_i 의 중요도를 가지고 t_i 의 처리 시간을 가지는 업무들을 가지고 중요도의 합이 S 이상이 되게 하면 됩니다.
- 이는 전형적인 배낭 문제이므로 아래와 같이 Dp 식을 세울 수 있습니다.
- $Dp[i]$: 처리한 문서들의 중요도 합이 i일 때 필요한 시간의 최솟값
- 위 Dp 식을 토대로 배낭 문제를 해결하면 $O(SN)$ 안에 문제를 해결할 수 있습니다.

D. 휴가 나가기

- 이제 업무들이 선행 업무가 있다고 생각 해 봅시다.
- 서로 선후행 관계에 있는 업무들로 이루어진 업무들의 집합을 **업무 묶음**이라고 합시다.
- 각 업무들의 선행 업무가 모두 다르므로 모든 업무 묶음들은 서로소 관계에 있습니다.
- 따라서 서로 다른 업무 묶음들에 있는 업무들은 독립적이므로 각 업무 묶음들을 따로 처리해 줄 수 있습니다.

D. 휴가 나가기

- 이제 각 업무 묶음들에 있는 업무들을 어떻게 처리할지 생각해 봅시다.
- 업무 묶음에 있는 업무들을 처리해야 하는 순서대로 배치했을 때 각 업무들을 $l_1, l_2, \cdots, l_k$ 라고 합시다.
- 이 업무 묶음을 처리하는 방법은 l_1 만 처리하기, l_1, l_2 만 처리하기, $\cdots$, $l_1, l_2, \cdots, l_k$ 까지 모두 처리하는 방법까지 총 k 개가 있습니다.
- 따라서 각각의 방법에서 얻는 중요도와 처리하는 데 필요한 시간을 나열하면 다음과 같습니다.
- $(w_1, t_1), (w_1 + w_2, t_1 + t_2), \cdots, (w_1 + w_2 + \cdots + w_k, t_1 + t_2 + \cdots + t_k)$

D. 휴가 나가기

- 따라서 선행 업무가 없는 경우에는 현재 순회하고 있는 업무의 경우의 수만 확인해 주면 되지만, 업무 묶음들을 순회하는 경우에는 해당 업무 묶음에서 가능한 k가지의 경우의 수를 모두 고려해 주어야 합니다.
- 즉 선행 업무가 없는 경우와 동일하게 Dp 테이블을 정의하되, 업무 묶음들을 순회해주며 현재 업무 묶음에서 가능한 모든 경우의 수를 Dp 테이블에 채워주면 선행 업무가 없는 경우와 동일하게 $O(NS)$ 만에 답을 구할 수 있습니다.

E. 식당지원차출

`offline_query`, `simulation`, `math`
출제진 의도 – **Hard**

- 제출 103번, 정답 17명 (정답률 16.505%)
- 처음 푼 사람: **hyperbolic**, 38분
- 출제자: `sangmin7b`

E. 식당지원차출

- 가장 쉬운 접근법은 식당지원 차출 프로그램을 만든 뒤 시뮬레이션을 실행하면서 쿼리를 처리하는 것입니다.
- 쿼리를 식당지원 차출 프로그램 가동 일수(d, 이하 가동 일수)에 대해 오름차순으로 정렬합니다.
- 우선순위 큐를 만들어서 차출 기준에 따라 병사 3명을 차출한 뒤, 해당 가동 일수에 해당하는 쿼리를 처리해줍니다. 쿼리를 처리한 뒤에는 차출한 병사의 차출 횟수를 증가시켜 다시 우선순위 큐에 넣습니다.

E. 식당지원차출

- 하지만 쿼리에서 물어보는 가동 일수가 최대 10^9 로 매우 크기 때문에 10^9 번의 차출을 시뮬레이션하면 시간제한을 초과하게 됩니다.
- 병사들의 무작위 차출 횟수 a_i 가 30 이하로 매우 작다는 것과 ($3 < N$ 일 때) 시뮬레이션을 진행함에 따라 병사들의 카운트 차이가 감소한다는 것을 관찰할 수 있습니다.
- ($3 < N$ 일 때) $30N$ 일의 차출이 이루어지고 난 뒤에는 어떤 두 병사의 차출 횟수 차이도 1 을 초과하지 않으며, 군번이 큰 병사의 차출 횟수는 군번이 작은 병사의 차출 횟수보다 작거나 같게 됩니다.
- 따라서 $30N$ 일 이후에는 군번이 작은 순서로 차출되는 것이 반복됩니다.

E. 식당지원차출

- $30N$ 일의 차출을 시뮬레이션한 뒤에는 가동 일수와 병사의 군번을 이용해 해당 병사가 몇 번 더 차출되는지 $O(1)$ 에 구하는 것이 가능합니다.
- 마지막 차출이 일어난 가동 일수를 D, 그때 차출된 마지막 병사의 군번을 idx 라고 하면 쿼리 (d_i, x_i) 에 대해 D 부터 d_i 까지 병사 x_i 가 차출된 횟수는 다음과 같습니다.
- N 명을 차출할때마다 1번씩, $\lfloor 3(d_i - D)/N \rfloor$ 번 차출되고
- idx 에서 x_i 까지의 거리만큼 더 차출하면 즉, $(x_i + N - idx - 1)\%N < 3(d_i - D)\%N$ 이면 한번 더 차출됩니다.
- 시뮬레이션에 $(90NlogN) = O(NlogN)$ 의 시간, 쿼리 정렬에 $O(QlogQ)$ 의 시간이 필요하여 전체 시간 복잡도는 $O(NlogN + QlogQ)$ 입니다.
- $N = 3$ 인 경우에는 아무리 많이 차출을 하더라도 병사들의 식당 지원 차출 횟수 차이가 줄어들지 않습니다. 모든 쿼리의 답이 d_i 와 동일합니다.

F. 참호전

`data_structures`, `sweeping`, `two_pointers`
출제진 의도 – **Hard**

- 제출 42번, 정답 13명 (정답률 33.333%)
- 처음 푼 사람: **hyperbolic**, 52분
- 출제자: `asdarwin03`

F. 참호전

- 폭탄 투하 위치와 폭파 범위, 각 아군 참호에 배치할 국군의 수를 적절히 설정하여 폭탄을 통해 승리한 참호전 개수와 아군 참호에 국군을 배치해 승리한 참호전 개수의 합을 최대화하는 문제입니다.
- 폭탄의 폭파 범위를 $[l, r](1 \leq l \leq r \leq N)$라 하겠습니다.
- $[l, r]$에서 임의의 l에 대해 최적의 r 값은 $a_l + a_{l+1} + \cdots + a_r \leq K$을 만족하는 가장 큰 r임은 자명합니다.
- 따라서 각 l에 대응하는 최적의 폭탄 범위 $[l, r]$에 대해, $a_l, \cdots, a_r$을 제외한 참호들 중 적은 인원을 필요로 하는 아군 참호 순서대로 국군을 최대한 배치하면 각 l에 대해 폭탄과 국군 배치로 승리할 수 있는 최대 참호전 개수를 구할 수 있습니다.
- 투 포인터를 활용해 모든 최적의 폭탄 범위를 탐색할 수 있으며, 이 중 참호전 개수의 최댓값을 출력하면 문제를 해결할 수 있습니다.

F. 참호전

- 하지만 이를 naive하게 구현하면 $\mathcal{O}\left(N^2\right)$ 로 시간초과를 받습니다.
- 어떤 폭파 범위 $[l, r]$ 에서 국군을 배치한 아군 참호 집합과 폭파 범위 $[l+1, r]$ 또는 $[l, r+1]$ 에서 국군을 배치한 아군 참호 집합과의 차이가 얼마나 클지를 생각해봅시다.
- 사실, 두 집합은 최대 원소 하나 차이밖에 나지 않습니다.
- 이 점에서 착안하여 multiset을 활용해 국군을 배치할 아군 참호를 관리하면 시간복잡도를 줄일 수 있습니다.

F. 참호전

- multiset 2개를 만들어 각 폭파 범위에 대해 다음과 같은 상태가 되도록 관리해주면 됩니다.
  1. ms1: 폭파 범위 $[l, r]$에 해당하지 않는 참호들 중 국군을 배치한 아군 참호 집합. 이때 아군 참호들에 배치한 총 국군의 수는 M 이하.
  2. ms2: 폭파 범위 $[l, r]$에 해당하지 않으면서 ms1에도 포함되지 않은 아군 참호 집합.
- 이때 ms1에는 이전 언급했던 최적의 국군 배치 방식을 이용하여 아군 참호를 채워 넣습니다.
- 폭파 범위를 탐색하는 투 포인터가 이동할 때마다 두 multiset을 활용해 국군 배치로 승리할 수 있는 최대 참호전 개수를 관리해주면 $\mathcal{O}(NlogN)$에 문제를 해결할 수 있습니다.
- 해당 풀이가 정말로 $\mathcal{O}(NlogN)$인지 증명하겠습니다.

F. 참호전

- 어떤 폭파 범위 $[l, r]$에서 다음 범위구간 $[l, r+1]$로 이동할 때 a_{r+1}에 대해 일어날 수 있는 경우는 2가지입니다.
  1. a_{r+1}이 ms1에 포함된 경우
     - ms2에 있는 원소는 a_{r+1}보다 크거나 같음이 자명하므로, 따라서 ms1에서 a_{r+1}를 삭제하는 연산이 1번, ms2의 최솟값이 ms1로 이동하는 연산이 최대 1번 발생할 수 있습니다.
  2. a_{r+1}이 ms2에 포함된 경우
     - 국군을 배치하는 아군 참호에는 변동이 없으므로 ms1은 그대로 두고, ms2에서 a_{r+1}를 삭제하는 연산이 1번 발생합니다.
- 폭파 범위 $[l, r]$에서 $[l+1, r]$로 이동하는 경우에도 비슷한 방식으로 이해할 수 있고, 따라서 폭파 범위 변화에 따른 모든 연산이 $\mathcal{O}(logN)$에 가능합니다.

F. 참호전

- 별해로 좌표압축을 이용한 $\mathcal{O}\left(NlogN+N\right)$ 풀이가 있으며, 일반적으로 정해보다 빠르게 동작합니다.

G. BX 내기

`trie`, `binary_search`, `tree_set`
출제진 의도 – **Challenging**

- 제출 21번, 정답 8명 (정답률 38.095%)
- 처음 푼 사람: **hjroh0315**, 52분
- 출제자: `junhyung9985`

G. BX 내기

- 두 대표수를 더하는 규칙을 바꿈에 따라서 일반적인 덧셈과는 달리 각 자릿수들이 독립적으로 계산이 됩니다.
- 즉, 현재 대표수와 더했을 때의 제일 높은 점수가 나오는 수들의 순위는 아래와 같이 결정이 됩니다.
- 10^{N-1} 자리에서 더했을 때 제일 높은 값이 나오는 경우 $> 10^{N-2}$ 자리에서 더했을 때 제일 높은 값이 나오는 경우 $> \cdots > 1$ 자리에서 더했을 때 제일 높은 값이 나오는 경우
- 이렇게 덧셈을 할 순서들이 정해집니다. 높은 자릿수에서 덧셈 결과가 같다면, 더 낮은 자릿수에서 더 높은 값이 나오는 걸 찾으면 됩니다.

G. BX 내기

- 이는 Trie에서 높은 자리수를 루트에 가까이 두는 방식으로 관리하면서 쉽게 풀 수가 있습니다.
- 다른 방법으로는 set에 대표수들을 관리하고 이분탐색을 계속하여 현재 생성한 대표수와 더하여 제일 높은 점수가 나오는 값들을 찾는 방법도 있습니다.
- Trie로 푸는 방법의 총 시간복잡도는 $\mathcal{O}(NM)$, 이분탐색 + set으로 푸는 방법의 총 시간복잡도는 $\mathcal{O}(NM \log M)$ 입니다.
- 다만 구현할 때, i 번째 턴에 준형이는 $b_1, b_2, \cdots, b_{i-1}$ 들 중에서 더할 수를 찾아야 하고, 상민이는 $a_1, a_2, \cdots, a_i$ 들 중에서만 더할 수를 찾을 수 있다는 것을 주의해야 합니다.

H. 생활관 건설하기

`math`, `prefix_sum`, `parametric_search`, `data_structures`, `segtree`
출제진 의도 – **Challenging**

- 제출 12번, 정답 5명 (정답률 41.667%)
- 처음 푼 사람: **flappybird**, 88분
- 출제자: 99asdfg

H. 생활관 건설하기

- 구간 $[l, r]$ 의 모든 땅의 높이를 x 로 만들기 위해 필요한 병사의 수는 구간의 길이 $r - l + 1 = W$, i 번째 땅의 높이를 h_i 라고 하면 다음과 같이 정리할 수 있습니다.
- $(x - h_l)^2 + (x - h_{l+1})^2 + \cdots + (x - h_r)^2$
- 이때, 위 식은 다음과 같이 x 에 대한 이차함수 꼴로 나타낼 수 있습니다.
- $Wx^2 - 2(h_l + h_{l+1} + \cdots + h_r)x + (h_l^2 + h_{l+1}^2 + \cdots + h_r^2)$
- $W(x - \frac{(h_l + h_{l+1} + \cdots + h_r)}{W})^2 + (h_l^2 + h_{l+1}^2 + \cdots + h_r^2)W - \frac{(h_l + h_{l+1} + \cdots + h_r)^2}{W}$
- 따라서, 위 식은 $x = \frac{(h_l + h_{l+1} + \cdots + h_r)}{W}$ 인 경우에 최솟값 $(h_l^2 + h_{l+1}^2 + \cdots + h_r^2)W - \frac{(h_l + h_{l+1} + \cdots + h_r)^2}{W}$ 를 가집니다.

H. 생활관 건설하기

- 단, 문제 조건에 의해 x는 반드시 정수여야 하므로, 해당 값을 반올림한 값을 사용해야 합니다.
- $h_l + h_{l+1} + \cdots + h_r$과 $h_l^2 + h_{l+1}^2 + \cdots + h_r^2$는 부분 합 알고리즘을 활용하여 계산해 주면, 임의의 구간 $[l, r]$의 모든 땅의 높이를 동일하게 만드는 데 필요한 병사의 수는 상수 시간 내에 계산할 수 있습니다.
- 또한, 구간 $[l, r]$에서 동일한 l 값에 대해 r 값이 증가하면 반드시 필요한 병사의 수는 증가한다는 사실은 자명합니다.
- 따라서, 투 포인터 알고리즘을 활용하면 모든 구간 시작 지점 l에 대해 최적의 r 값 및 건설 가능한 건물의 길이 x_l를 $\mathcal{O}(N)$으로 전처리해둘 수 있습니다.

H. 생활관 건설하기

- 쿼리로 입력되는 건설 가능 구간 $[l, r]$이 주어졌을 때, 실제로 구해야 하는 값은 $l \leq i \leq r$에 대해 $\min(x_i, r - i + 1)$입니다.
- 이때, $i + x_i$의 값은 항상 단조 증가함은 자명하므로, $x_k > r - k + 1$이 되는 가장 작은 k에 대해 구간 $[l, k)$에서 x_i의 최댓값과 $r - k + 1$중 최댓값을 출력해 주면 됩니다.
- 구간 $[l, k)$에서 x_i의 최댓값은 segment tree등의 자료구조를 활용하여 구할 수 있고, $x_k > r - k + 1$이 되는 최소의 k는 이미 전처리해 둔 값을 바탕으로 이분 탐색으로 $\mathcal{O}(\log N)$으로 구할 수 있습니다.
- 따라서, 전처리 및 쿼리 처리에 대해 총 시간복잡도 $\mathcal{O}(N \log N + Q \log N)$으로 전체 문제를 해결할 수 있습니다.