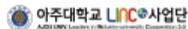


shake!23

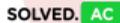
제 9회 경인지역 6개 대학 연합 프로그래밍 경시대회

주최 / 주관



경희대학교 컴퓨터공학부 학생회
성균관대학교 알고리즘 동아리 NPC
아주대학교 알고리즘 소학회 A.N.S.I.
인하대학교 컴퓨터공학 학술소모임 CTP
한국항공대학교 알고리즘 학회 koala
한양대학교 ERICA 알고리즘 연구학회 0&1

후원



Official Solutions

본 대회가 성공적으로 개최되기까지 도와주신 모든 분들께 감사의 말씀을 올립니다.

운영	출제	검수
✓ 김현빈	✓ 김민겸 ^{39d11} 인하대	✓ lky7674
✓ 이현우	✓ 김승환 ^{overnap} 경희대	✓ leo020630
	✓ 김현빈 ^{akim9905} 아주대	✓ nick832
	✓ 신정환 ^{shjohw12} 성균관대	✓ hibye1217
	✓ 이승재 ^{coxie} 성균관대	✓ wjdclgns12
	✓ 장민우 ^{pani} 아주대	✓ solvedac

문제 번호	문제 제목	의도한 난이도	출제자
A	별 보러 가자	Hard	김현빈 ^{akim9905}
B	실 전화기	Easy	장민우 ^{pani}
C	또 수열 문제야	Easy	김현빈 ^{akim9905}
D	모비스터디	Normal	김민겸 ^{39d11}
E	마카롱카마	Normal	장민우 ^{pani}
F	삼국지	Hard	신정환 ^{shjohw12}
G	관광 상품	Normal	김민겸 ^{39d11}
H	대역폭 관리	Hard	김승환 ^{overnap}
I	올라올라	Normal	이승재 ^{coxie}
J	볼록볼록	Hard	이승재 ^{coxie}

A. 별 보러 가자

dynamic_programming, math

출제진 의도 - Hard

- ✓ 제출 10번, 정답 0명 (정답률 0.000%)
- ✓ 처음 푼 사람: -, -분
- ✓ 출제자: 김현빈^{akim9905}

A. 별 보러 가자

- ✓ 두 좌표 $(x_1, y_1), (x_2, y_2)$ 간의 거리가 가지는 값은 다음 네 개의 값 중 max 값입니다.
 - $(x_1 + y_1) - (x_2 + y_2)$
 - $-(x_1 + y_1) + (x_2 + y_2)$
 - $(x_1 - y_1) - (x_2 - y_2)$
 - $-(x_1 - y_1) + (x_2 - y_2)$
- ✓ 편의상 $x + y$ 의 꼴을 1번 꼴, $x - y$ 의 꼴을 2번 꼴이라고 하겠습니다.
- ✓ 위의 결과를 통해 하나의 구간에서는 같은 꼴이 한번씩 더해지고 빼진다는 사실을 알 수 있습니다.

A. 별 보러 가자

- ✓ 위의 결과를 확장해 하나의 구간에는 오직 5개의 상태만 존재할 수 있음을 알 수 있습니다.
 1. 같은 꿀이 한번씩 더해지고 빼진 경우
 2. 1번 꿀이 한번 더해진 경우
 3. 1번 꿀이 한번 빼진 경우
 4. 2번 꿀이 한번 더해진 경우
 5. 2번 꿀이 한번 빼진 경우

A. 별 보러 가자

- ✓ 이제 dp를 정의할 수 있습니다.
- ✓ $dp(i, j, k) = i$ 번 별까지 j 개의 구간을 뽑았고 상태가 k 일때, 대푯값 합의 최댓값
- ✓ $k = 1$ 인 경우, 다음 여섯 개의 값 중 max 값입니다.
 - $dp(i - 1, j, 1)$
 - $dp(i - 1, j - 1, 1)$
 - $dp(i - 1, j - 1, 2) - (x_i + y_i)$
 - $dp(i - 1, j - 1, 3) + (x_i + y_i)$
 - $dp(i - 1, j - 1, 4) - (x_i - y_i)$
 - $dp(i - 1, j - 1, 5) + (x_i - y_i)$

A. 별 보러 가자

- ✓ $k = 2$ 인 경우, 다음 두 개의 값 중 max 값입니다.
 - $dp(i - 1, j, 2)$
 - $dp(i - 1, j, 1) + (x_i + y_i)$
- ✓ $k = 3$ 인 경우, 다음 두 개의 값 중 max 값입니다.
 - $dp(i - 1, j, 3)$
 - $dp(i - 1, j, 1) - (x_i + y_i)$
- ✓ $k = 4, 5$ 일 때도 같은 논리로 식을 작성할 수 있습니다.
- ✓ 시간 복잡도는 $\mathcal{O}(NM)$ 입니다.

B. 실 전화기

ad_hoc, number_theory

출제진 의도 - **Easy**

- ✓ 제출 133번, 정답 28명 (정답률 21.805%)
- ✓ 처음 푼 사람: **최민선**, 8분
- ✓ 출제자: 장민우^{pani}

B. 실 전화기

- ✓ N 이 K 의 배수라면, 실 전화기들은 교점이 없는 볼록다각형 형태가 됩니다. 따라서 답은 0입니다.
- ✓ N 과 K 가 서로소인 경우, 주어진 시행에서 모든 참가자가 한 번씩 선택됩니다.
 - 이때 $K < \frac{N}{2}$ 인 경우, 1번 이름표를 가진 사람과 $K + 1$ 번 이름표를 가진 사람을 잇는 실 전화기와 교차하게 되는 실 전화기의 수를 세봅시다.
 - $1 < i < K + 1$ 인 i 에 대해서 i 번 이름표를 가진 사람은 $((i - 1 - K) \bmod N) + 1$ 번 사람과 연결되고, $(i + K)$ 번 사람과 연결됩니다.
 - 이는 둘 다 1번 이름표를 가진 사람과 $K + 1$ 번 이름표를 가진 사람을 잇는 실 전화기와 교차하는 실 전화기입니다.

B. 실 전화기

- 즉, $1 < i < K + 1$ 인 모든 i 번 이름표를 가진 사람에 대해 2개의 교차하는 쌍이 생기고, 총 실 전화기의 수는 N 개 이므로 답은 $N \times (K - 1)$ 이 됩니다.
 - $K > \frac{N}{2}$ 인 경우, K 를 $N - K$ 로 바꾸어 문제를 해결할 수 있습니다.
- ✓ N 과 K 가 서로소가 아닌 경우, $d = \gcd(N, K)$ 에 대해서 $\frac{N}{d}$ 를 새로운 N , $\frac{K}{d}$ 를 새로운 K 로 놓고 해결할 수 있습니다.

C. 또 수열 문제야

ad_hoc, construction

출제진 의도 - **Easy**

- ✓ 제출 94번, 정답 43명 (정답률 45.745%)
- ✓ 처음 푼 사람: **오준원**, 4분
- ✓ 출제자: 김현빈^{akim9905}

c. 또 수열 문제야

- ✓ A 의 모든 항이 홀수라면 $A_i + A_j$ 는 짝수, $A_i \times A_j$ 는 홀수이므로 어떤 경우에도 $A_i + A_j$ 가 $A_i \times A_j$ 를 나눌 수 없습니다.
- ✓ 따라서 $A_i = 2i - 1$ 이면 주어진 조건을 만족하는 수열을 구성할 수 있습니다.
- ✓ 이외에도 랜덤 등 수많은 풀이가 존재합니다.

D. 모비스터디

dijkstra

출제진 의도 - Normal

- ✓ 제출 228번, 정답 21명 (정답률 9.211%)
- ✓ 처음 푼 사람: **최준형**, 20분
- ✓ 출제자: 김민겸^{39d11}

D. 모비스터디

- ✓ A 번 도시에서 B 번 도시까지 가는 최단 경로 위에 존재하는 도시들의 공통점을 찾을 수 있습니다.
- ✓ 어떤 도시에서 A 번 도시까지의 거리와 B 번 도시까지의 거리의 합이 A 번 도시부터 B 번 도시까지의 최단 거리와 같다면, 그 도시는 정답이 될 수 있습니다.

D. 모비스터디

- ✓ A 번 도시에서 다른 도시까지의 최단 거리를 구합니다. $d_A(u)$ 라고 표기하겠습니다.
- ✓ B 번 도시에서 다른 도시까지의 최단 거리를 구합니다. $d_B(u)$ 라고 표기하겠습니다.
- ✓ $d_A(u) + d_B(u) = d_A(B)$ 를 만족하는 도시 u 를 찾습니다.
- ✓ 다익스트라 알고리즘을 이용해서 최단 거리를 빠르게 구할 수 있습니다.
- ✓ 총 시간 복잡도는 $\mathcal{O}(M \log N)$ 입니다.

E. 마카롱카마

greedy

출제진 의도 - Normal

- ✓ 제출 40번, 정답 4명 (정답률 10.000%)
- ✓ 처음 푼 사람: **최민선**, 108분
- ✓ 출제자: 장민우^{pani}

E. 마카롱카마

- ✓ i 번째 빨간색 코크의 크기를 a_i , i 번째 파란색 코크의 크기를 b_i , 롤루가 얻을 수 있는 최대 점수의 i 번째 숫자를 c_i 라고 합시다.
- ✓ 파란색 코크의 위치를 코크의 크기가 큰 것부터 순서대로 결정해 봅시다.
- ✓ b_i 의 위치를 결정할 때, 다음과 같이 케이스를 분류하여 해결할 수 있습니다.
- ✓ Case 1. i 번 위치에 파란색 코크가 아직 배치되지 않은 경우
 - 앞으로 배치해 줄 파란색 코크의 크기는 모두 현재 배치하고자 하는 코크의 크기보다 작거나 같습니다.
 - 따라서 c_i 와 c_{N+1-i} 의 값은 b_i 가 되어야 합니다.

E. 마카롱카마

- 이때, 다음의 두 경우로 나누어 생각할 수 있습니다.
- a_i 와 a_{N+1-i} 의 값이 다른 경우
 - ▶ b_i 를 $N + 1 - i$ 번 위치에 배치합니다.
- a_i 와 a_{N+1-i} 의 값이 같은 경우
 - ▶ i 번 위치와 $N + 1 - i$ 번 위치에 b_i 를 배치할 필요가 없으므로 아직 위치가 결정되지 않은 위치중 자릿수가 높은 칸과 해당 칸에 대칭되는 위치에 b_i 와 b_{N+1-i} 를 배치합니다.

E. 마카롱카마

- ✓ Case 2. i 번 위치에 이미 다른 파란색 코크가 배치된 경우
 - b_i 는 다른 위치에 배치되어야 합니다.
 - b_i 와 같은 크기를 가지는 코크 중 이와 같이 다른 위치에 배치되어야 하는 코크 수를 P 개라고 합시다.
 - 이전에 다른 칸을 채움으로써 j 번 칸을 비워둔 b_j 의 수를 Q 개라고 합시다.
 - 이때, $Q \geq P$ 임은 자명합니다.
 - P 가 2 이상인 경우 $P < 2$ 가 될 때 까지 현재 코크가 배치되지 않은 칸 중 가장 자릿수가 높은 칸과 해당 칸에 대칭되는 위치에 코크들을 배치합니다.
 - 남은 파란색 코크들의 경우 비어있는 Q 개의 칸 중 아무 곳이나 배치하면 됩니다.

F. 삼국지

tree, dfs, set/map

출제진 의도 - **Hard**

- ✓ 제출 2번, 정답 0명 (정답률 0.000%)
- ✓ 처음 푼 사람: -, -분
- ✓ 출제자: 신정환^{shjohw12}

F. 삼국지

- ✓ 편의상 도시를 정점, 도로를 간선으로 생각하고, 1번 정점을 루트로 하는 rooted tree를 생각해 봅시다.
- ✓ 한 번의 DFS로 i 번 정점을 루트로 하는 서브 트리의 모든 정점의 가중치의 합 $SubSum_i$ 를 전처리합니다.
- ✓ 어떤 정점과 그 부모 정점을 잇는 간선을 부모 간선이라 합시다.
- ✓ 서로 다른 두 정점을 택하고 정점들의 부모 간선을 자르면 트리는 세 부분으로 나뉩니다.
- ✓ 다시 DFS 하여, 현재 방문한 정점의 부모 간선을 잘랐을 때 최적해를 구하기 위해 다른 부모 간선 중 어떤 것을 자를 것인지 효율적으로 판단하면 됩니다. 현재 정점을 cur 라 합시다.
- ✓ 이때 두 케이스가 존재합니다.

F. 삼국지

Case 1. 한 정점이 서브 트리의 루트이고, 다른 정점이 서브 트리에 속하는 경우

- ✓ cur 를 루트로 하는 서브 트리의 정점들을 관리하는 것은 어려우므로 cur 와 루트를 잇는 경로상의 정점들을 관리합니다.
- ✓ 부모 간선을 자르는 다른 정점을 x 라 하면 각 부분의 전투력은 $\{SubSum_1 - SubSum_x, SubSum_x - SubSum_{cur}, SubSum_{cur}\}$ 입니다.
- ✓ 이때 최적해는 $SubSum_1 - SubSum_x$ 와 $SubSum_x - SubSum_{cur}$ 의 차가 최소일 때 발생합니다.
- ✓ 정점들의 $SubSum$ 을 `std::multiset` 으로 관리하여 `std::lower_bound` 를 적절히 활용하면 최적해를 효율적으로 구할 수 있습니다.

F. 삼국지

Case 2. 그 외의 경우

- ✓ 이미 방문한 정점과 cur 를 루트로 하는 서브 트리의 정점들을 제외한 정점들을 관리하면 됩니다.
- ✓ 부모 간선을 자르는 다른 정점을 x 라 하면 각 부분의 전투력은 $\{SubSum_1 - SubSum_{cur} - SubSum_x, SubSum_{cur}, SubSum_x\}$ 입니다.
- ✓ 이고, 최적해는 Case 1.과 동일한 방법으로 구할 수 있습니다.
- ✓ 시간 복잡도는 $\mathcal{O}(N \log N)$ 입니다.

G. 관광 상품

ad_hoc, binary_search

출제진 의도 - Normal

- ✓ 제출 5번, 정답 0명 (정답률 0.000%)
- ✓ 처음 푼 사람: -, -분
- ✓ 출제자: 김민겸^{39d11}

G. 관광 상품

Solution 1. 관찰

- ✓ 답이 되는 부분 수열의 길이를 N , 이때의 답을 k 라고 합시다.
- ✓ 이때 k 이상의 수는 $\left\lfloor \frac{N}{2} \right\rfloor + 1$ 개 이상 존재하고, 이는 부분 수열의 길이의 절반보다 큽니다.
- ✓ 이 때문에 길이가 3인 연속 부분 배열 중 k 이상의 수가 2개 이상인 배열이 항상 존재합니다.
- ✓ 따라서 길이가 3인 연속 부분 배열의 중앙값 중에 답이 항상 존재함을 알 수 있고, 길이가 3인 연속 부분 배열의 중앙값 중 최댓값이 문제의 답입니다.
- ✓ $N = 2$ 인 경우는 두 수 중 작은 수를 출력하도록 예외 처리를 해야 합니다.
- ✓ 총 시간 복잡도는 $\mathcal{O}(N)$ 입니다.

Solution 2. 이분 탐색

- ✓ 구하고자 하는 답이 특정 값보다 크거나 같은지 판별할 수 있다면 이분 탐색을 통해 문제를 해결할 수 있습니다.
- ✓ 구하고자 하는 답을 k 로 고정하고 결정 문제로 바꾸어 매개 변수 탐색을 진행합니다.

G. 관광 상품

- ✓ 답이 k 보다 크거나 같은지 판별하는 방법을 알아봅시다.
- ✓ 수열에서 k 보다 작은 수를 0, k 이상인 수를 1로 치환합니다.
- ✓ 1이 0보다 많이 있는 길이가 2 이상인 연속 부분 배열을 찾을 수 있다면 답이 k 보다 크거나 같습니다.

G. 관광 상품

- ✓ $d(i) =$ "첫 번째 수부터 i 번째 수까지 존재하는 1의 개수 - 0의 개수" 라고 합시다.
- ✓ 이때 $d(i) - d(j)$ 가 0보다 큰 i, j 가 존재하는지 확인하면 됩니다. ($i > j + 1$)
- ✓ $d(j)$ 의 최솟값을 들고 다니면서 배열을 순차적으로 탐색하면 됩니다.
- ✓ 실제 구현할 때에는 i 와 j 가 1만큼 차이나는 경우를 조심해야 합니다.

H. 트리와 대역폭

tree, LCA, prefix_sum, parametric_search

출제진 의도 - **Hard**

- ✓ 제출 17번, 정답 3명 (정답률 17.647%)
- ✓ 처음 푼 사람: **오준원**, 142분
- ✓ 출제자: 김승환^{overnap}

H. 트리와 대역폭

- ✓ 예약을 순서대로만 받을 수 있으므로 파라메트릭 서치를 고려할 수 있습니다.
- ✓ 이제 승인할 예약의 개수가 주어졌을 때 서버가 다운될지 결정하는 문제를 풀어봅시다.
- ✓ 그 전에, 각 예약의 LCA를 $\mathcal{O}((N + M) \log N)$ 정도로 전처리 합니다.

H. 트리와 대역폭

- ✓ 어떤 예약 하나를 처리하는 과정을 봅시다.
- ✓ 누적합 배열 $psum$, 예약 $x y w$ 와 $l = \text{LCA}(x, y)$ 과 l 의 부모 l_{par} 에 대해,

$$psum(x) := psum(x) + w$$

$$psum(y) := psum(y) + w$$

$$psum(l) := psum(l) - w$$

$$psum(l_{par}) := psum(l_{par}) - w$$

을 하고 리프 노드부터 부모로 누적시켜봅시다.

H. 트리와 대역폭

- ✓ 이렇게 하면 x 와 y 를 잇는 단순 경로에 포함되는 모든 노드에 w 만큼 더해집니다.
- ✓ 모든 예약을 $psum$ 에 기록하고 한 번만 누적시키면 되므로 결정 문제의 시간복잡도는 $\mathcal{O}(N + M)$ 입니다.
- ✓ LCA 전처리와 파라메트릭 서치에 의해 전체 시간복잡도는 $\mathcal{O}((N + M) \log NM)$ 입니다.
- ✓ 동일한 누적합 아이디어와 ETT, 세그먼트/펜윅 트리를 사용해 실시간 쿼리로 바꾸어 $\mathcal{O}((N + M) \log N \log M)$ 에 풀 수도 있습니다.
- ✓ 이분 탐색 대신 HLD와 \min, \max 레이지 세그를 사용해 동일한 $\mathcal{O}((N + M) \log^2 N)$ 에 풀어낼 수도 있습니다.

I. 올라올라

stack

출제진 의도 - Normal

- ✓ 제출 169번, 정답 40명 (정답률 24.260%)
- ✓ 처음 푼 사람: **박재현**, 14분
- ✓ 출제자: 이승재^{coxie}

I. 올라올라

- ✓ A 배열 $\{3, 1, 4, 2, 5\}$ 를 생각해 봅시다. 이때 B 배열은 $\{3, 4, 4, 5\}$ 가 되어야 합니다.
- ✓ $\{3, 4, 5\}$ 는 앞에서부터 읽을 때 최댓값이 갱신되는 수입니다.
- ✓ 즉, 최댓값이 갱신된다면, B_i 의 값이 변경됩니다.
- ✓ 3 과 4 사이의 i 에 대하여 $B_i \geq 4$ 가,
4 와 5 사이의 i 에 대하여 $B_i \geq 5$ 가,
5 이후의 i 에 대하여 $B_i = 5$ 가 됩니다.

I. 올라올라

- ✓ 따라서, 최댓값이 갱신되는 인덱스의 차이 중 최댓값이 답입니다.
- ✓ 여기서 갱신되는 위치는 1, 3, 5, 6입니다. (편의상 $N + 1$ 을 추가했습니다.)
- ✓ 차이의 최댓값은 2입니다.
- ✓ 이때, 갱신된다는 의미는 $\max(A_1, A_2, \dots, A_{i-1}) \leq A_i$ 입니다.
- ✓ 시간복잡도는 $\mathcal{O}(N)$ 입니다.

J. 볼록볼록

geometry, two_pointer

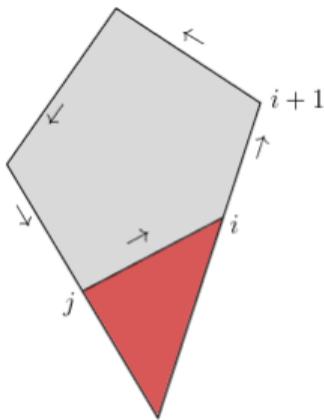
출제진 의도 - **Hard**

- ✓ 제출 58번, 정답 4명 (정답률 6.897%)
- ✓ 처음 푼 사람: **박재현**, 91분
- ✓ 출제자: 이승재^{coxie}

- ✓ P_i, P_{i+1}, \dots, P_j 가 볼록 다각형일 때, 점의 개수가 충분하다면 P_{i+1}, \dots, P_j 또한 볼록 다각형이 됩니다.
- ✓ 위 성질을 이용하여 투포인터를 사용합니다.

J. 볼록볼록

- ✓ 이제 P_i, \dots, P_j 가 볼록 다각형이라 가정 했을 때, P_i, \dots, P_j, P_{j+1} 이 다각형이 되기 위한 조건을 생각해 봅시다.



- ✓ 이 그림에서 빨간색 삼각형 내부에 P_{j+1} 이 들어가야 합니다. 이는 ccw를 통하여 확인 가능합니다.