



EASY

MEDIUM

HARD

CHALLENGING

SUAPC 2025 Summer Solution



Contents



SUAPC
의자 준비하기



OPENING
TIME



SWAP-C
SORT



17마리 낙타



삼시 몇 도예요?



TRI-TREE XOR



두 수열 만들기



위치 복원하기



돌 뒤집기 게임



최단 경로 쌍



숫자 놀이



PERMUTATION
SWAP



물고기와 퀴리



https://www.icpc-sinchon.io

A. SUAPC 의자 준비하기

math, arithmetic

출제진 의도 - **Easy**

- 제출 29번, 정답 23팀 (정답률 : 79.310%)
- 처음 푼 팀 : 저녁 뭐 먹지^{연세대학교}, 1분
- 출제자 : 강호현^{annieho}



https://www.icpc-sinchon.io



A. SUAPC 의자 준비하기

- $3N$ 과 M 을 비교하여 M 이 크거나 같으면 0을, 아닌 경우 $(3N - M)A + B$ 를 출력합니다.



https://www.icpc-sinchon.io

B. Opening Time

parametric search, depth first search, segment tree

출제진 의도 - **Challenging**

- 제출 2번, 정답 0팀 (정답률 : 0%)
- 처음 푼 팀 : 없음
- 출제자 : 양승민^{dadas08}



https://www.icpc-sinchon.io



B. Opening Time

- 최대 거리와 관련된 문제인 만큼, 지름을 생각할 수 있습니다.
- 지름의 중점을 M 이라 합시다. (간선 위에 존재할 수 있음)
- M 을 기준으로 왼쪽/오른쪽 트리로 분리됩니다. 왼쪽에 존재하는 x 에 대한 답을 구한다 생각합시다.



https://www.icpc-sinchon.io



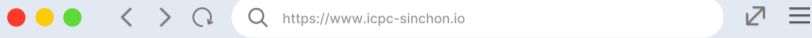
B. Opening Time

- 가맹점의 위치 y 의 후보를 줄여봅시다.
- y 는 지름의 중점에서 가까운 왼/오른쪽 정점 중 하나거나 오른쪽에 존재하는게 최적입니다.
- 왼쪽에 존재한다면, 왼쪽의 정점 중 M 에 가장 가까운 정점으로 바꾸면 이득입니다.
- 매개변수 탐색을 사용합니다. 최대 거리 D 을 고정합니다.
- 문제를 다음과 같이 생각할 수 있습니다.
- 각 정점 i 를 x 또는 y 에 대응시킨다. 대응된 정점과의 거리는 D 이하여야 한다.
- 이때, x 에 대응된 정점들을 " x 가 커버하는 정점"라 합시다.



B. Opening Time

- 지름의 양 끝을 L, R 이라 합시다. L 은 x 가 커버하고, R 은 y 가 커버하는 것이 최적입니다.
- 또한, y 는 지름 위의 정점 중 R 와 거리가 D 이하인 가장 왼쪽 정점에 두는게 최적입니다.
- 그런 y 를 이분 탐색으로 빠르게 찾을 수 있습니다.
- 이후는, 지름 위 적절한 정점 k 가 존재해 k 보다 왼쪽은 x 가, 오른쪽은 y 가 커버하는게 최적입니다.
- 이분 탐색으로 k 를 찾고, 판정은 세그먼트 트리 등의 자료구조를 이용해 가능합니다.
- 시간복잡도는 $O(N \log N \log 10^{14})$ 입니다.
- 여담으로, $O(N \log 10^{14})$ 풀이도 가능합니다.



C. SWAP-C Sort

constructive, permutation parity

출제진 의도 - **Hard**

- 제출 25 번, 정답 2 팀 (정답률 : 8%)
- 처음 푼 팀 : 저녁 뭐 먹지^{연세대학교}, 261 분
- 출제자 : 곽우석^{bubbler}



C. SWAP-C Sort

- 주어진 순열의 parity가 $\frac{n(n-1)}{2}$ 과 같다면 답이 존재하며, 그렇지 않다면 답은 존재하지 않습니다.
- $n = 2$ 의 경우 가능한 동작이 1개이므로 답은 자명합니다.
- $n = 3$ 의 경우 답이 존재하는 순열은 정확히 1개의 원소가 제자리에 있는 경우이며, 제자리에 있던 원소를 세 번 옮겨서 제자리에 다시 돌아오게 하면 올바른 구성이 됩니다.



C. SWAP-C Sort

- $n \geq 4$ 부터는 $(1, n), (2, n), \dots, (n-1, n)$ 의 동작을 모두 사용하여 n 번째 자리에 n 이 오게 한 다음 작은 크기로 재귀하는 방법을 생각해볼 수 있습니다.
- n 이 제자리에 있지 않을 경우, $P_i = n$ 이라고 하면 모든 $1 \leq j < n, j \neq i$ 인 j 에 대해 (j, n) 의 동작을 모두 수행한 다음 (i, n) 을 수행하면 됩니다.
- n 이 이미 제자리에 있을 경우에는 이러한 방법은 존재하지 않습니다. 대신에 $n-1$ 번째 자리를 엮어서 둘을 동시에 제자리에 두는 방법을 찾아볼 수 있습니다.



C. SWAP-C Sort

- $P_{n-1} = i$ 라고 합시다. 가정에 의해 현재 $P_n = n$ 입니다.
- $i \neq n - 1$ 이면, 다음과 같은 방법이 존재합니다.
 - 먼저 (i, n) 을 수행하여 n 을 i 번째 자리로, $n - 1$ 을 n 번째 자리로 옮깁니다.
 - $j \notin \{i, n - 1, n\}$ 인 j 를 하나 골라 (j, n) 을 수행하여 $n - 1$ 을 j 번째 자리로 옮깁니다. $n \geq 4$ 이므로 이러한 j 는 반드시 존재합니다.
 - 이제 $k \notin \{i, j, n - 1, n\}$ 인 k 에 대해 (k, n) 동작을 모두 수행하고, $(i, n - 1)$ 과 $(n - 1, n)$ 을 순서대로 수행하여 n 을 제자리에 놓습니다.
 - 이제 $n - 1$ 을 제자리에 놓는 동작을 $(i, n - 1)$ 을 제외하고 수행하면 됩니다.



C. SWAP-C Sort

- $i = n - 1$ 이면, 다음과 같은 방법이 존재합니다.
 - 먼저 $(n - 1, n)$ 을 수행하여 $n - 1$ 과 n 의 자리를 바꿉니다.
 - $j \notin \{n - 2, n - 1, n\}$ 인 j 를 하나 골라 (j, n) 을 수행하여 $n - 1$ 을 j 번째 자리로 옮깁니다.
 - 이제 $k \notin \{j, n - 2, n - 1, n\}$ 인 k 에 대해 (k, n) 동작을 모두 수행하고, $(n - 2, n - 1)$ 과 $(n - 2, n)$ 을 순서대로 수행하여 n 을 제자리에 놓습니다.
 - 이제 $n - 1$ 을 제자리에 놓는 동작을 $(n - 2, n - 1)$ 을 제외하고 수행하면 됩니다.



https://www.icpc-sinchon.io



C. SWAP-C Sort

- 이 과정을 거치면 반드시 n 에서 $n - 1$ 또는 $n - 2$ 로 재귀하게 되므로, $n = 2$ 와 $n = 3$ 의 base case를 합치면 올바른 풀이가 됩니다.



https://www.icpc-sinchon.io

D. 17마리 낙타

math, arithmetic

출제진 의도 - **Easy**

- 제출 81 번, 정답 19 팀 (정답률 : 24.691%)
- 처음 푼 팀 : 공공면면공공연세대학교, 서강대학교, 숙명여자대학교, 12분
- 출제자 : 김도윤^{kdy40929}



https://www.icpc-sinchon.io



D. 17마리 낙타

- 삼형제가 가진 가축의 수는 각각 정수이고 정확히 N 이어야 합니다.
- 따라서 수식을 세우면 $\frac{N+K}{A} + \frac{N+K}{B} + \frac{N+K}{C} = N$ 이고, 각 항은 정수입니다.
- 위 식을 풀면 $K = N \times \left(\frac{ABC}{AB+BC+CA} - 1 \right)$ 로 유일하게 결정됩니다.
- 다만, 계산한 K 는 음이 아닌 정수여야 하고, 삼형제가 가지는 가축의 수가 각각 정수인지 확인해야 합니다.
- 따라서 계산한 K 의 값이 실제로 가능하면 K 를, 그렇지 않으면 -1 을 출력하면 됩니다.



ad_hoc, binary_search

출제진 의도 - **Medium**

- 제출 75 번, 정답 18 팀 (정답률 : 24%)
- 처음 푼 팀 : 공공면면공공연세대학교, 서강대학교, 숙명여자대학교, 24분
- 출제자 : 서종환^{mujigae}



https://www.icpc-sinchon.io



E. 삼시 몇 도예요?

- 맨 처음 삼복이의 체온이 존재할 수 있는 범위는 $[0, 2025]$ 입니다.
- 첫 쿼리로 $\lfloor \frac{0+2025}{2} \rfloor$ 인 1012를 보내면 삼복이의 응답에 따라 범위를 $[\lfloor \frac{0+1012}{2} \rfloor, 1012 - 1]$ 또는 $[1012 + 1, \lfloor \frac{1012+2025}{2} \rfloor]$ 로 좁힐 수 있습니다.
- 즉, 하나의 쿼리에 범위를 4배씩 좁힐 수 있습니다.
- $2^{10} < 2025 < 2^{11}$ 이므로 최소 6번의 쿼리를 이용하면 삼복이의 체온과 샤워기의 온도를 동일하게 만들 수 있습니다.



https://www.icpc-sinchon.io

F. Tri-Tree XOR

trees, constructive

출제진 의도 - **Hard**

- 제출 60번, 정답 7팀 (정답률 : 11.667%)
- 처음 푼 팀 : **goraani never dies**^{서강대학교}, 75분
- 출제자 : 김도윤^{kdy40929}



F. Tri-Tree XOR

- XOR 연산은 임의의 서로 다른 두 정점 u 와 v 에 대해서 두 정점을 잇는 간선이 그래프 A 와 그래프 B 중 홀수 개에 존재할 때 그래프 C 에서 정점 u 와 v 를 연결하는 연산입니다.
- 따라서 간선의 수가 a 인 그래프 A 와 간선의 수가 b 인 그래프 B 를 XOR한 그래프 C 의 간선의 수를 c 라고 하면 $a + b \equiv c \pmod{2}$ 가 성립합니다.
- 그래프 A, B, C 가 모두 정점이 N 개인 트리이면 트리의 간선 수는 $N - 1$ 개이므로 $(N - 1) + (N - 1) \equiv N - 1 \pmod{2}$ 가 성립, 즉 N 이 홀수여야 합니다.
- 따라서 N 이 짝수일 때의 정답은 -1 입니다.



https://www.icpc-sinchon.io



F. Tri-Tree XOR

- 먼저, N 이 홀수인 경우 중 최소인 $N = 3$ 인 경우를 고려합니다.
- $1 - 2, 2 - 3$ 을 연결한 트리 A 와 $1 - 3, 2 - 3$ 을 연결한 트리 B 를 XOR하면 $1 - 2, 1 - 3$ 을 연결한 트리 C 가 형성됩니다.
- 일반적으로 N 이 큰 경우에도 정점을 공유하는 두 간선을 고르고, 이 두 간선을 $N = 3$ 과 같이 하나의 간선을 유지하고 인접하지 않던 두 정점을 연결하는 간선을 추가하는 과정을 반복하여 트리 B 를 구성할 수 있습니다.



https://www.icpc-sinchon.io



F. Tri-Tree XOR

- N 이 홀수이면 간선의 수 $N - 1$ 은 짝수이고, 모든 정점이 연결되어 있으므로 항상 트리를 정점을 공유하는 간선 쌍 $\frac{N-1}{2}$ 개로 분할할 수 있습니다.
- 임의로 루트를 정한 후, 리프 노드에서부터 간선을 2개씩 짝지어 새로운 트리의 간선을 구성하는 과정을 반복하면 $O(N)$ 에 문제를 해결할 수 있습니다.



https://www.icpc-sinchon.io

G. 두 수열 만들기

graph_traversal, knapsack

출제진 의도 - **Medium**

- 제출 45 번, 정답 13 팀 (정답률 : 28.889%)
- 처음 풀 팀 : 공공면면공공연세대학교, 서강대학교, 숙명여자대학교, 51 분
- 출제자 : 장우성^{saywoo}



https://www.icpc-sinchon.io



G. 두 수열 만들기

- $x \oplus y = 2^j$ ($j \geq 0$)을 만족하는 y 는 $x \oplus 2^j$ 입니다.
- 그리고 $x \oplus y = 2^j$ ($j \geq 0$)을 만족하는 (x, y) 쌍들이 있을 때, x, y 를 같은 수열에 배치해야 함을 알 수 있습니다.



https://www.icpc-sinchon.io



G. 두 수열 만들기

- x_1, x_2, \dots, x_{2N} 을 각각 정점으로 하는 그래프를 생각해봅시다.
- 어떤 x 에서 $x \oplus y = 2^j$ 를 만족하는 y 는 $x \oplus 2^0, x \oplus 2^1, \dots, x \oplus 2^{19}$ 로 총 20개이므로, 가능한 y 를 모두 찾아보면서 x 와 y 를 잇는 간선을 만들어줍니다.
- 그러면 연결 그래프가 $M(1 \leq M \leq N)$ 개 있는 그래프가 됩니다.



https://www.icpc-sinchon.io



G. 두 수열 만들기

- 같은 연결 그래프에 있는 수들은 모두 같은 수열에 배치해야 합니다.
- M 개의 연결 그래프에 있는 정점의 개수를 각각 c_1, c_2, \dots, c_M 개라고 할 때, c 의 부분 집합을 골라 합이 N 이 되게 하는 방법이 있는지 찾는 문제로 바꿀 수 있습니다.
- 해당 문제는 Knapsack DP로 $O(NM)$ 에 해결 가능합니다.
- 분리 집합을 사용하면 연결 그래프를 효율적으로 관리할 수 있지만, 나이브한 그래프 탐색으로도 시간 안에 문제를 해결할 수 있습니다.



https://www.icpc-sinchon.io

H. 위치 복원하기

ad hoc

출제진 의도 - **Hard**

- 제출 36 번, 정답 2 팀 (정답률 : 5.556%)
- 처음 풀 팀 : 저녁 뭐 먹지^{연세대학교}, 232분
- 출제자 : 박진한^{jinhan814}



https://www.icpc-sinchon.io



H. 위치 복원하기

- $(1, 2)$ 를 이용하면 $N = 2$ 일 때 1번의 쿼리로 정답을 구할 수 있습니다.
- $(1, 2), (1, 3), (2, 3)$ 을 이용하면 $N = 3$ 일 때 3번의 쿼리로 정답을 구할 수 있습니다.
- $(1, 2), (1, 3), (2, 3), (1, 4), (2, 4)$ 를 이용하면 $N = 4$ 일 때 5번의 쿼리로 정답을 구할 수 있습니다.
- 이제 N 이 5 이상인 경우를 살펴보겠습니다.



H. 위치 복원하기

- $3 < i < j$ 에 대해 $|x_1 - x_2|, |x_1 - x_3|, |x_2 - x_3|$ 중 적어도 하나는 $|x_i - x_j|$ 가 아닙니다.
- 일반성을 잃지 않고 $|x_1 - x_2| \neq |x_i - x_j|$ 라 하면 $(1, i), (i, j), (j, 2)$ 를 이용해 3번의 쿼리로 x_i, x_j 를 유일하게 결정할 수 있습니다.
- N 이 5 이상의 홀수일 때는 x_1, x_2, x_3 을 3번의 쿼리로 결정한 뒤 남은 $(x_4, x_5), \dots, (x_{N-1}, x_N)$ 을 $\frac{3(N-3)}{2}$ 번의 쿼리로 결정할 수 있습니다.
- N 이 6 이상의 짝수일 때는 x_1, x_2, x_3, x_4 를 5번의 쿼리로 결정한 뒤 남은 $(x_5, x_6), \dots, (x_{N-1}, x_N)$ 을 $\frac{3(N-4)}{2}$ 번의 쿼리로 결정할 수 있습니다.



https://www.icpc-sinchon.io

I. 돌 뒤집기 게임

ad hoc, game theory

출제진 의도 - **Medium**

- 제출 50번, 정답 17팀 (정답률 : 34%)
- 처음 풀 팀 : 권현우팬클럽연세대학교, 42분
- 출제자 : 유상혁golazcc83



https://www.icpc-sinchon.io



I. 돌 뒤집기 게임

- 앞면이 연속해서 나오는 뒤편을 길이 K 의 뒤편이라고 정의합니다.
- 관찰 1: 자신의 턴일 때, 상대방에게 턴을 넘기지 않으면서 모든 뒤편을 길이 1, 2의 뒤편으로 나눌 수 있습니다.
 - K 가 홀수인 경우 : HTHT..HTH
 - K 가 짝수인 경우 : HTHT..HTHH
- 위의 관찰대로 뒤편을 나눴다면, 길이 1의 뒤편의 개수에 따라 승패가 결정됩니다. 개수가 홀수라면 선공이, 짝수라면 후공이 승리합니다.



https://www.icpc-sinchon.io



I. 돌 뒤집기 게임

- 관찰 2: 게임이 시작했을 때 $K = 3$ 인 묶음이 존재한다면, 선공은 아래의 전략대로 $K = 1$ 인 묶음의 홀짝성을 정하여 후공에게 넘겨줄 수 있으므로 항상 승리할 수 있습니다.
 1. $K = 3$ 인 묶음을 하나 선택하고, 선택한 묶음을 제외한 나머지 묶음을 관찰 1에 따라 나눈 다음, $K = 1$ 인 묶음의 개수를 셉니다.
 2. 길이 1의 묶음의 개수가 홀수라면 $K = 3$ 의 묶음을 HHH \rightarrow HTH \rightarrow HTT 순으로 뒤집어, 개수를 짝수로 만든 다음 상대에게 넘겨줍니다.
 3. 길이 1의 묶음이 개수가 짝수라면 $K = 3$ 의 묶음을 HHH \rightarrow HHT 순으로 뒤집어, 개수를 짝수로 만든 다음 상대에게 넘겨줍니다.



I. 돌 뒤집기 게임

- 관찰 3: $K = X \geq 5$ 인 묶음이 존재한다면, $K = N - 4, 3$ 인 묶음으로 나누고 자신의 턴을 유지할 수 있으므로 선공이 항상 승리합니다.
- 이제 $K = 1, 2, 4$ 의 묶음만 존재하는 경우에 대해서 생각해봅시다. 길이 4의 묶음을 HHHH라 한다면,
 - HHHT로 뒤집으면 $K = 3$ 묶음을 후공에게 넘겨주기 때문에 패배합니다.
 - 따라서 HHTH와 같이 뒤집어야 합니다. 이 때 자신의 턴은 유지하므로, $K = 4$ 묶음의 홀짝성은 $K = 1$ 의 홀짝성과 동일합니다.



https://www.icpc-sinchon.io



I. 돌 뒤집기 게임

- 따라서 최종 풀이는 아래와 같습니다.
 - $K = 3$ 또는 $K \geq 5$ 인 묶음이 존재한다면 선공이 승리합니다.
 - 그렇지 않다면, $K = 1$ 또는 $K = 4$ 의 묶음의 개수를 세어서, 홀수면 선공이 승리하고 짝수면 후공이 승리합니다.



https://www.icpc-sinchon.io

J. 최단 경로 쌍

graph theory

출제진 의도 - **Challenging**

- 제출 4 번, 정답 1 팀 (정답률 : 25%)
- 처음 푼 팀 : 오락실거주중인고양이들^{서강대학교}, 264 분
- 출제자 : 이승재^{Coxie}



https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 일단, 1번 노드에서부터의 최단경로에 포함되지 않는 모든 간선은 제거합니다.
- 즉, 간선 u, v, w 에 대하여, $dist[u] + w = dist[v]$ 를 만족하는 간선만을 남깁니다.
- 추가로, 1번 노드에서 도달할 수 없는 노드는 자명하게 0이므로 무시하겠습니다.
- 그러면 DAG 그래프가 됩니다. 또한, 남아있는 모든 경로는 1로부터의 최단 경로가 됩니다.



https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 노드 1 에서 x 로 이동할 때, 반드시 방문해야 하는 노드 $y (\notin \{1, x\})$ 가 있다고 해봅시다.
- 그럼, 자명하게 0을 출력해야 합니다.
- 이제 이러한 y 가 없다라고 가정해봅시다.
- 그러면, $\{1, x\}$ 를 제외한 나머지 노드들에 대하여, 그 노드를 방문하지 않는 경로가 존재합니다.

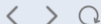


https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 이제, 그래프에서 1과 x 사이의 경로를 없애는 mincut(최소 간선 개수)을 생각해 봅시다.
- 이러한 mincut이 (u, v) 간선 하나만을 포함한다고 해봅시다.
- 그림 1에서 x 로의 경로는 반드시 v 를 포함해야 합니다.
- 이는 모든 노드에 대하여 방문하지 않는 경로가 있다는 사실에 모순입니다.

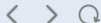


https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 따라서, mincut의 크기는 2 이상입니다.
- 이제, mincut max flow theorem에 따라 두개의 경로가 생겼습니다.
- 결국, 반드시 방문해야 하는 노드가 1번 노드 하나라면, 1을 출력하면 됩니다.



https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 이러한 개념을 dominator라고 부릅니다. 즉, immediate dominator가 1번 노드라면 1을 출력하면 됩니다.
- 팀노트를 사용할수도 있겠지만, 우리의 팀노트에는 이런 이상한 알고리즘이 없을 수 있습니다.
- 우리에게 주어진 그래프가 DAG라는 점을 이용하여 좀 더 쉽게 해결해 봅시다.



https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 각 노드 u 에 대하여, 1에서 u 로 이동하는 데, 반드시 방문해야 하는 노드들 중 u 와 가장 가까운 노드를 p_u 라고 합니다.
- u 의 부모 노드를 p_u 로 하는 가상의 트리를 생각하겠습니다.
- 그러면, u 로 이동할 때 반드시 방문해야 하는 노드들의 집합은 $\{p_u, p_{p_u}, p_{p_{p_u}} \dots\}$ 가 됩니다.
- 따라서, 각 노드 v 에 대하여, v 로 들어오는 모든 간선 (u_i, v) 를 생각해본다면,
- p_v 는, u_i 들의 LCA가 됩니다.



https://www.icpc-sinchon.io



J. 최단 경로 쌍

- 최종적으로 시간복잡도는 $O(m \log m)$ 이 됩니다.



https://www.icpc-sinchon.io

K. 숫자 놀이

math, number theory

출제진 의도 - **Medium**

- 제출 52번, 정답 17팀 (정답률 : 32.692%)
- 처음 푼 팀 : 오락거주중인고양이들^{서강대학교}, 22분
- 출제자 : 곽우석^{bubbler}



K. 숫자 놀이

- 주어진 구간의 모든 수를 $\text{mod } c$ 로 분류하여 각각의 집합 내의 원소들을 오름차순으로 나열했다고 가정해 봅시다.
- 이때 A_i 와 B_i 는 같은 집합 내에서 서로 이웃한 수가 됩니다.
- c 는 $0 \text{ mod } c$ 의 집합 내에서 항상 가장 작은 원소이기 때문에, c 의 값을 고정하였을 때 c 를 빼고 각 집합의 원소의 개수가 모두 짝수 개이면 문제의 조건을 충족하는 수열 A 와 B 가 존재하게 되고, 그렇지 않다면 존재하지 않습니다.



https://www.icpc-sinchon.io



K. 숫자 놀이

- 주어진 범위의 특징으로 인해 $i \bmod c$ ($i \neq 0$)인 수의 개수는 모두 서로 같아야 하고, $0 \bmod c$ 인 수의 개수는 그보다 하나 적거나 많아야 함을 알 수 있습니다.
- 하나 적은 경우는 $x \equiv 1 \pmod{c}$, $x + 2n \equiv -1 \pmod{c}$ 이 성립하고, 하나 많은 경우는 $x \equiv x + 2n \equiv 0 \pmod{c}$ 가 성립합니다.



https://www.icpc-sinchon.io



K. 숫자 놀이

- 전자의 경우는 다음과 같이 분석할 수 있습니다.
 - $x \leq c$ 이므로 $x \equiv 1 \pmod{c}$ 가 되려면 $x = 1$ 이어야 합니다.
 - 각 집합의 원소의 개수가 짝수 개가 되기 위해서는 $1 + 2n \equiv -1 \pmod{2c}$ 가 되어야 하고, $n + 1$ 이 c 의 배수이면 됩니다.
 - 따라서 이 경우의 답은 $n + 1$ 의 약수의 개수와 같습니다.



https://www.icpc-sinchon.io



K. 숫자 놀이

- 후자의 경우는 다음과 같이 분석할 수 있습니다.
 - $x \equiv 0 \pmod{c}$ 가 되려면 $x = c$ 이어야 합니다.
 - 이때는 $2n \equiv 0 \pmod{2c}$ 여야 하므로, $x \neq 1$ 일 경우 n 이 x 의 배수이면 답은 1, 아니면 0이 됩니다.



https://www.icpc-sinchon.io

L. Permutation Swap

case work, sweeping, segment tree

출제진 의도 - **Challenging**

- 제출 10번, 정답 0팀 (정답률 : 0%)
- 처음 푼 팀 : 없음
- 출제자 : 박진한^{jinhan814}



https://www.icpc-sinchon.io



L. Permutation Swap

- P_i, P_j 를 교환할 때 영향을 받는 pair는 4개이고, 이때 정답은 최대 2만큼 증가합니다.
- $j = i + 1 \vee j = i + 2 \vee i = 1 \vee j = N$ 인 (i, j) 는 $\mathcal{O}(N)$ 에 직접 확인할 수 있습니다.
- 이제 $1 < i < j < N$ 이면서 $j - i > 2$ 인 (i, j) 중 정답이 될 수 있는 pair를 고려해 보겠습니다.



L. Permutation Swap

- $1 < i < j < N$ 이면서 $j - i > 2$ 인 (i, j) 중 i, j 와 인접한 4개 pair의 변화량의 합이 양수인 경우는 대칭을 제외할 때 다음의 5가지 case가 있습니다.
 - $(1, 0, 0, 1)$ ($\Delta = 2$)
 - $(1, 0, 0, 0), (0, 1, 0, 0), (1, 0, -1, 1), (0, 1, 1, -1)$ ($\Delta = 1$)
- case 별 표기는 $(i - 1, i), (i, i + 1), (j - 1, j), (j, j + 1)$ 의 증가 쌍 여부의 변화량을 의미합니다.
- 각 case에 해당하는 (i, j) 가 존재하는지 여부는 i, j 와 인접한 6개 칸 간의 대소 관계를 부등식으로 표현한 뒤 두 3차원 좌표에 대한 스위핑 문제를 풀며 $\mathcal{O}(N \log N)$ 에 구할 수 있습니다.



https://www.icpc-sinchon.io

M. 물고기와 퀴리

brute_force

출제진 의도 - **Medium**

- 제출 27번, 정답 6팀 (정답률 : 22.222%)
- 처음 푼 팀 : **Retros**^{서강대학교}, 142분
- 출제자 : 송주현^{lunarlity}



https://www.icpc-sinchon.io



M. 물고기와 쿼리

- 이 문제는 다른 쿼리 문제들과 달리, 물고기 개수를 구하는 쿼리가 전체 범위를 요구합니다.
- 세그먼트 트리를 쓰면 그냥 구간 쿼리도 될 것 같은데... 왜 그럴까요?
- 전체 범위일 경우 세그먼트 트리가 굳이 필요 없기 때문입니다.



https://www.icpc-sinchon.io



M. 물고기와 퀴리

- 편의상 i 번째 물고기 조각을 F_i 라고 하겠습니다.
- 이 문제에선 1번 퀴리에 대한 관찰이 중요합니다.
- 물고기 문자열에서 $><>$ 혹은 $<><$ 모양은, $F_i \neq F_{i+1} \wedge F_{i+1} \neq F_{i+2}$ 라는 것과 동치입니다.
- 하지만 1번 퀴리 실행 시, 인접한 두 물고기 조각의 동일 여부는 내부 구간에서는 변하지 않고, 변화가 생기는 부분은 퀴리의 양 끝뿐입니다.
- 따라서 1번 퀴리 한 번으로 변하는 물고기 조각의 개수는 최대 4개입니다.



https://www.icpc-sinchon.io



M. 물고기와 쿼리

- 이를 활용해 $1 \leq i \leq N - 1$ 인 모든 정수 i 에 대해 F_i 와 F_{i+1} 의 동일 여부를 저장하는 배열 G 를 만들고, 1번 쿼리가 들어올 때마다 해당 구간의 양 끝만 갱신하며 변화량을 계산하면 됩니다.
- 맨 처음의 경우에는 직접 배열 전체를 순회하며 물고기 문자열 개수를 세줍니다.
- 2번 쿼리는 이렇게 관리해둔 물고기 문자열 개수를 그대로 출력하면 됩니다.
- 따라서 시간복잡도는 $\mathcal{O}(N + Q)$ 이며, 세그먼트 트리를 사용 시 $\mathcal{O}(Q \log N)$ 에 가능합니다. 이 문제의 난이도를 고려하여 세그먼트 트리가 통과하게 제한을 잡았습니다.