



by

KPSC × A.N.S.i × CTP × ZeroPage



Hosting Club





Executive Board

개최

✓ wapas 중앙대
✓ chan120714 국민대

집행

✓ Haru_101 국민대
✓ bluerini 아주대
✓ hjkim9911 인하대
✓ thepackour 중앙대



Staff

운영

✓ chan120714	국민대	✓ chlwnsgud7	인하대	✓ hibye1217
✓ Haru_101	국민대	✓ hjkim9911	인하대	✓ kckyoung2
✓ kangsm02	국민대	✓ plan_pt	인하대	✓ kevinlys00
✓ saywoo	국민대	✓ ANZ	인하대	✓ mon_de2738
✓ tapris	국민대	✓ hui	중앙대	✓ oh040411
✓ usb9245	국민대	✓ yumeaster	중앙대	✓ wider93
✓ akim9905	아주대	✓ thepackour	중앙대	✓ yyyy7089
✓ bluerini	아주대	✓ wapas	중앙대	✓ PositiveSmile
✓ js9028	아주대			
✓ pani	아주대			



Supporter

검수		출제		테스터
✓ usb9245	15	✓ wapas	9	✓ ANZ
✓ chlwnsgud7	14	✓ chan120714	2	✓ kckyoung2
✓ plan_pt	13	✓ hjkim9911	2	✓ yumeaster
✓ chan120714	9	✓ kangsm02	2	✓ mon_de2738
✓ tapris	7	✓ plan_pt	2	✓ sksms1375
✓ hjkim9911	6	✓ saywoo	2	
✓ kangsm02	6	✓ bluerini	1	
✓ oh040411	6	✓ tapris	1	
✓ saywoo	4	✓ yyyy7089	1	
✓ yyyy7089	4			
✓ kevinlys00	3			



Academic Sponsor



국민대학교

소프트웨어융합대학
SW중심대학사업단



인하대학교
INHA UNIVERSITY

컴퓨터공학과



중앙대학교 SW교육원

SW중심대학



Sponsors

개인 후원

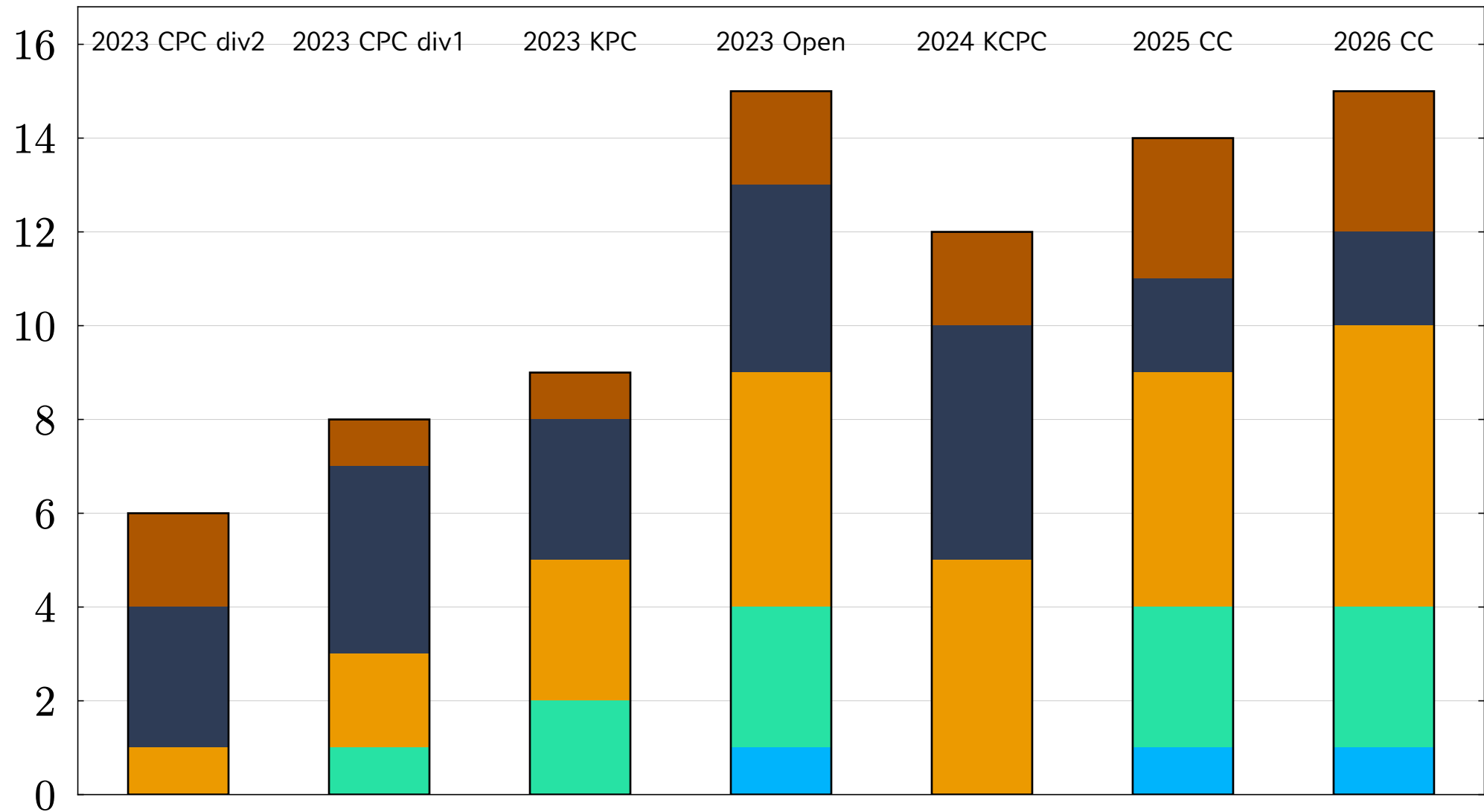
✓ kiwiyou		✓ hui	중앙대
✓ RyuZU		✓ usb9245	국민대
✓ haru_101	국민대	✓ kep1er07	
✓ cubic		✓ encore	인하대
✓ 류현우		✓ plan_pt	인하대
		✓ PositiveSmile	
		✓ wapas	중앙대



번호	문제	난이도	출제자
	진짜 해 구성하기	Easy	plan_pt
	승차권 검표	Easy	hjkim9911, wapas
	별점 테러	Medium	saywoo, wapas
	Alea jacta est!	Hard	kangsm02
	최소 넓이 직각 격자 다각형	Hard	plan_pt
	자릿수 합분해	Challenging	chan120714, wapas
	순열 LCA	Challenging	kangsm02, wapas
	발전소	Extreme	yyyy7089
	달빛 등대의 섬	Challenging	saywoo, wapas
	Max Mex Matrix	Hard	tapris
	센트컵	Hard	bluerini
	회의실 예약 시스템 저격	Hard	chan120714, wapas
	슬라이딩 퍼즐	Medium	wapas
	Tessellation	Medium	wapas
	제곱수^제곱수^제곱수	Easy	hjkim9911, wapas

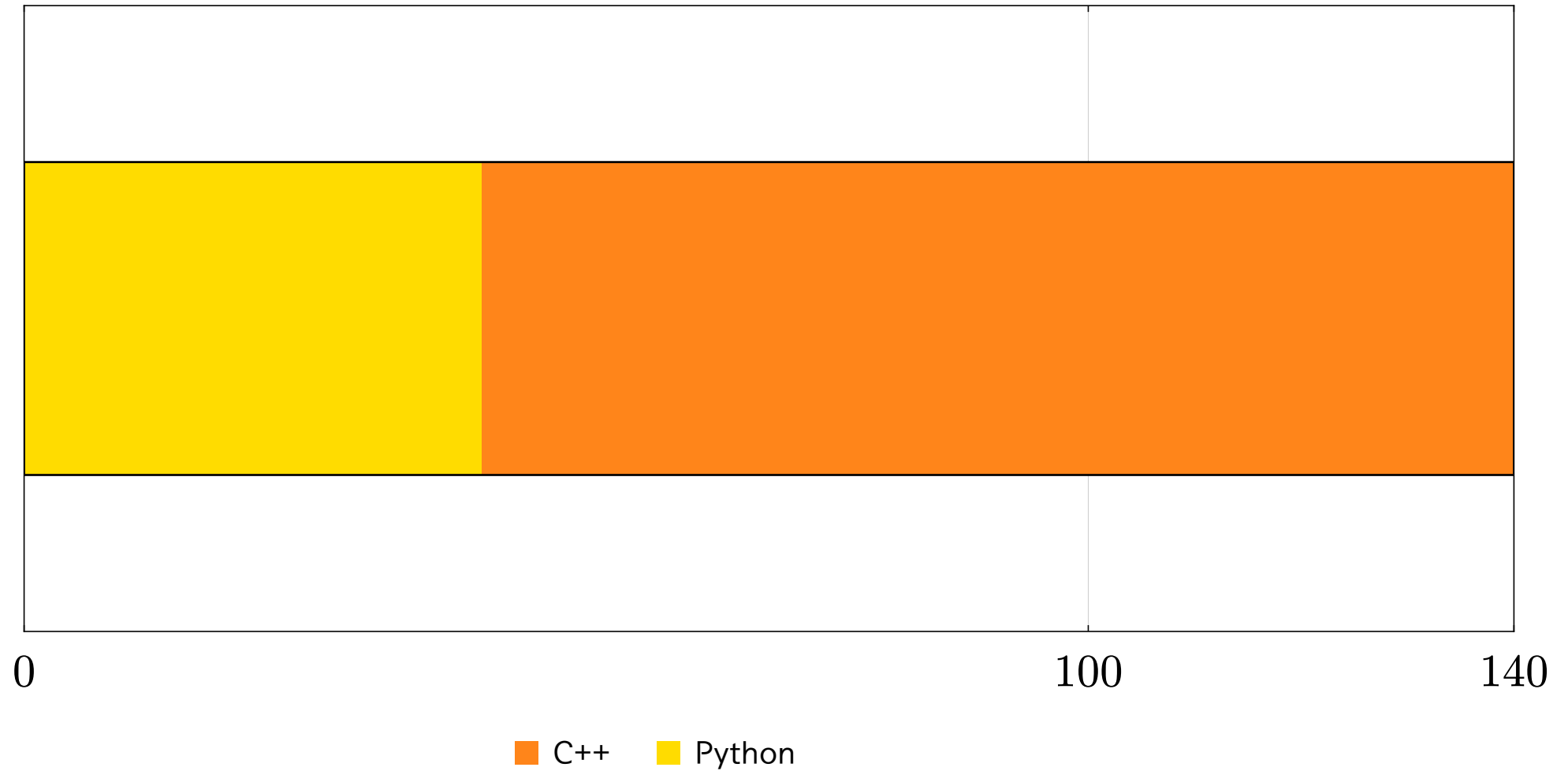


Difficulty History (Filter by 연합)





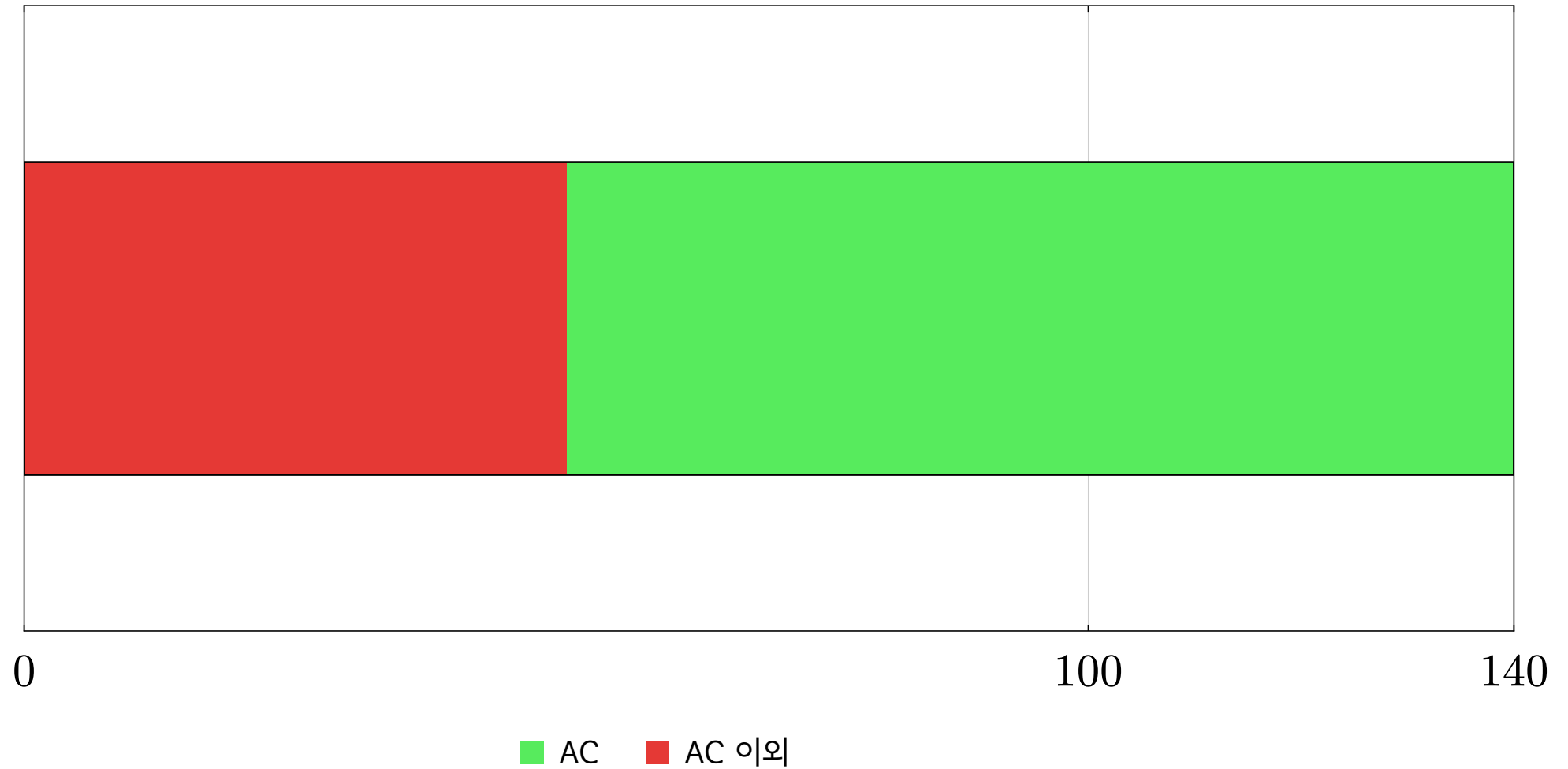
언어별 제출 통계



Open Contest · 총 140 회



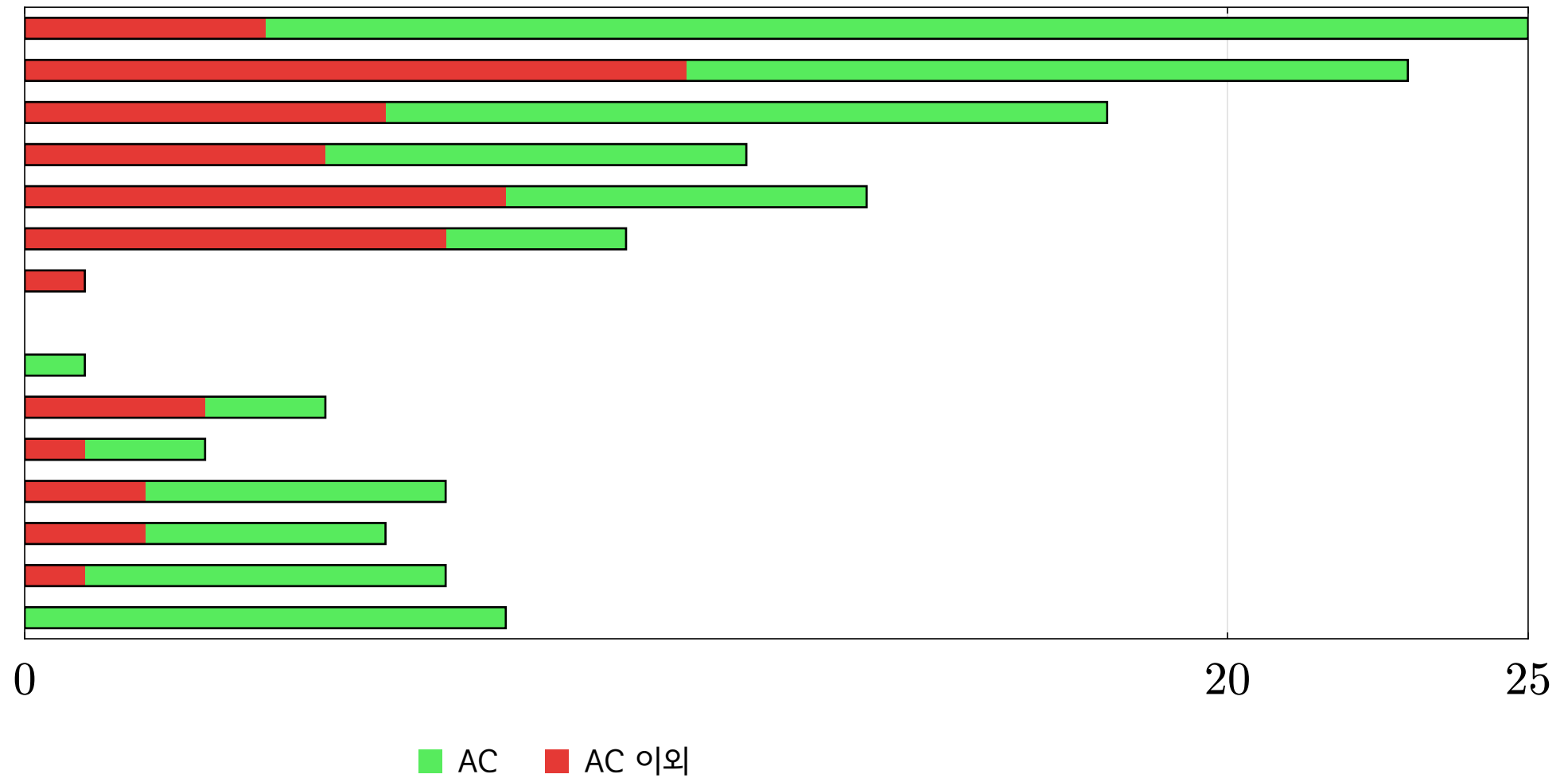
전체 제출 통계



Open Contest · 총 140 회 · AC 89 회 · AC 이외 51 회



문제별 제출 통계



Open Contest · 전체 시간의 AC 및 AC 이외 제출을 표시합니다.



☀. 진짜 해 구성하기

#constructive, #implementation, #string

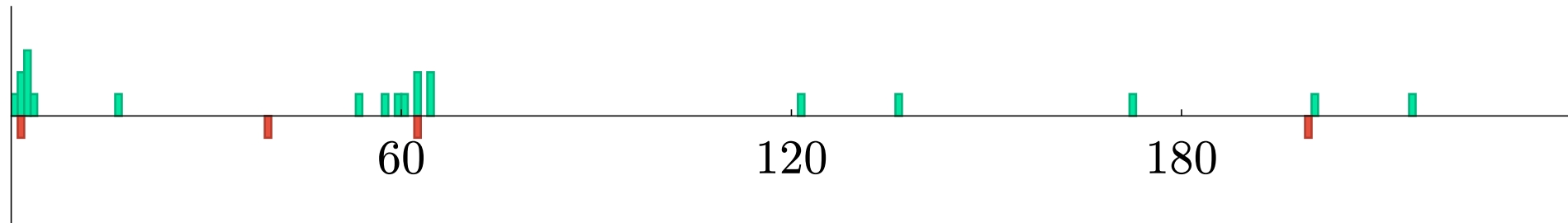
출제진 의도 - Easy

✓ 출제자: plan_pt

✓ 정답: 21 명 · 정답률 84.000%

✓ 처음 푼 참가자 dadasO8

0 분



초록: AC · 빨강: AC 이외 판정



진짜 해 구성하기

- ✓ 간단하게 진짜 해를 만들어 봅시다.
- ✓ 태양 중심부에서는 약 1,500만 K의 온도와 높은 밀도, 압력에서 핵융합이 일어납니다.
- ✓ 이 부분은 구현을 생략하겠습니다.



진짜 해 구성하기

- ✓ 그런데 해를 만들기만 해서는 표준 출력이 비어 있으므로 WA입니다.
- ✓ 성분비는 지문에 이미 주어져 있으므로, 직접 성분을 분석할 필요도 없습니다.
- ✓ 문제 그대로 작성하면 됩니다!
- ✓ 예를 들어 h 74개, h 24개, a 2개를 공백 없이 이어서 printf 등의 함수로 표준 출력에 출력하면 됩니다.



. 승차권 검표

#constructive, #math, #number_theory

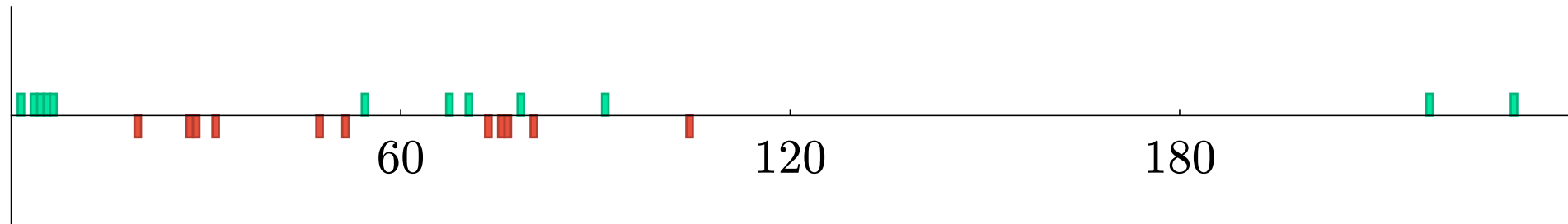
출제진 의도 - Easy

✓ 출제자: hjkim9911, wapas

✓ 정답: 12 명 · 정답률 52.174%

✓ 처음 푼 참가자 dadasO8

1 분



초록: AC · 빨강: AC 이외 판정



. 승차권 검표

- ✓ 주어진 조건들을 순서대로 확인해봅시다.
- ✓ 첫 번째 조건은 항상 참입니다.
 - ✓ 두 양의 정수 a, b 에 대해, $k = \gcd(a, b)$ 라고 합시다.
 - ✓ $a = a'k, b = b'k$ (a', b' 은 양의 정수)로 나타낼 수 있으며,
 - ✓ $a + b = (a' + b')k$ 이므로 $a + b$ 는 k 의 배수입니다.
- ✓ 두 번째 조건도 항상 참입니다.
 - ✓ 위와 유사한 방식으로 보일 수 있습니다.
- ✓ 임의의 서로 다른 양의 정수 $a_1, a_2, \dots, a_N$ 대해 세 번째 조건 또한 항상 참입니다.
 - ✓
$$\sum_{i=1}^N a_i \geq \sum_{i=1}^N i = \frac{N(N+1)}{2}$$
- ✓ 따라서 10^9 이하의 서로 다른 양의 정수 N 개를 아무거나 출력하면 됩니다.
 - ✓ 여담으로 $N = 10^5$ 에서 양의 정수 N 개를 무작위로 출력했을 때 틀릴 확률은 약 99.33% 입니다.



★. 별점 테러

#constructive, #math

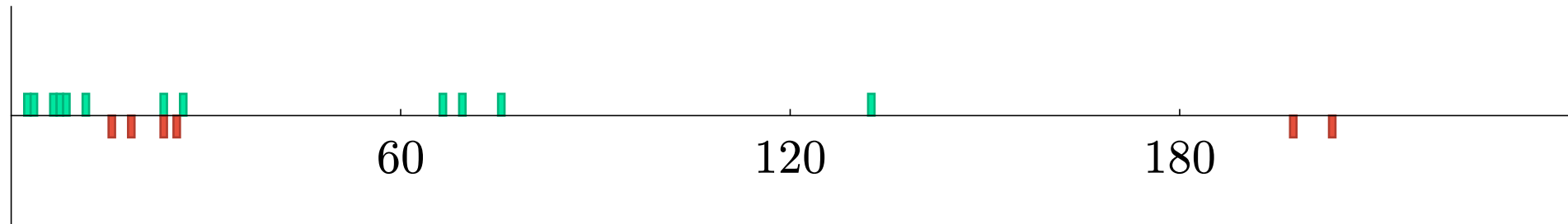
출제진 의도 - Medium

✓ 출제자: saywoo, wapas

✓ 정답: 12 명 · 정답률 66.667%

✓ 처음 푼 참가자 jkrt

2 분



초록: AC · 빨강: AC 이외 판정



★. 별점 테러

✓ N 개의 리뷰의 합을 S 라고 할 때, 평점이 X 가 되려면 다음이 성립해야 합니다.

$$\frac{S}{N} = X, \quad S = N \times X$$

✓ 여기서 S 는 항상 정수여야 하므로, $N \times X$ 의 값이 정수일 때만 리뷰를 작성하는 방법이 존재합니다.

✓ $N \times X$ 의 값이 정수인지 판별할 때 double과 같은 실수 자료형을 사용하면, 부동소수점 오차로 인해 올바르게 판별되지 않을 수 있습니다.

✓ 그래서 정수 자료형을 사용하여 X 에 100을 곱한 후, $N \times 100X$ 가 100으로 나눠지는 지 확인하는 방법을 사용해야 합니다.

✓ 리뷰를 작성하는 방법이 존재한다면, 1부터 5사이의 수 N 개의 합이 S 가 되게 하는 방법을 구성하는 문제로 생각할 수 있습니다.



★. 별점 테러

- ✓ 구성하는 방법은 여러 가지 존재하지만, 그중 한 가지 방법은 아래와 같습니다.
- ✓ $i(1 \leq i \leq N)$ 번째 리뷰를 작성할 때, 남은 리뷰 합을 S_i 라고 하겠습니다.
- ✓ 뒤에 남은 리뷰의 개수가 $N - i$ 개이므로, i 번째 리뷰를 작성하고 나서 남은 최소 합이 $N - i$ 이상 되어야 합니다.
- ✓ 그러므로 i 번째 리뷰에서 작성할 수 있는 최댓값은 $S_i - (N - i)$ 가 됩니다.
- ✓ 각 리뷰는 최대 5점이므로, i 번째 리뷰마다 $\min(5, S_i - (N - i))$ 점을 작성하면 됩니다.



. Alea jacta est!

#constructive, #bitmask

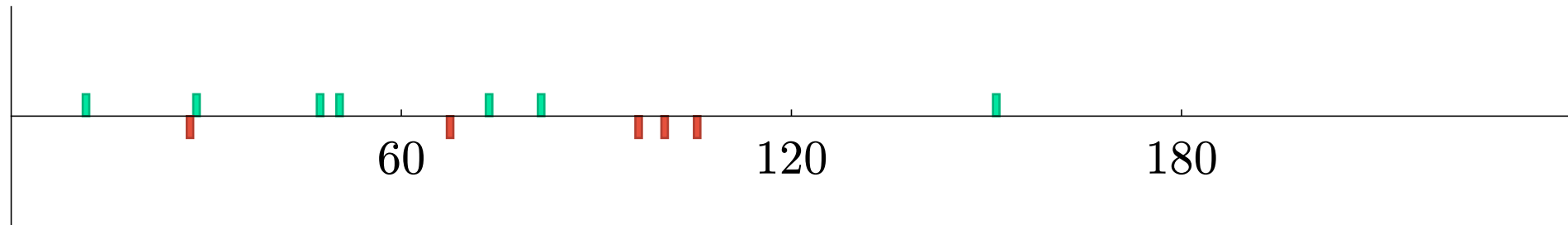
출제진 의도 - Hard

✓ 출제자: kangsm02

✓ 정답: 7 명 · 정답률 58.333%

✓ 처음 푼 참가자 cubious

11 분



초록: AC · 빨강: AC 이외 판정



🎲. Alea jacta est!





 . Alea jacta est!

- ✓ 주사위 각 면은 S 의 subset이어야 합니다.
- ✓ S 에 포함되지 않은 비트가 어떤 주사위의 면에 존재하는 경우, OR연산으로 S 를 만들 수 없게 됩니다.
- ✓ 반대로, S 가 만들어지기 위해서는 S 의 모든 켜진 비트가 최소 하나 이상의 주사위에 의해 보장되어야 합니다.
- ✓ i 번 주사위의 모든 면에 공통으로 들어간 비트들을 a_i 라고 정의합니다.
- ✓ 조건을 만족하기 위해서는 모든 주사위의 a_i 를 OR한 값이 S 가 되도록 구성하면 됩니다.



 . Alea jacta est!

- ✓ a_i 에 대해 켜질 수 있는 최대 비트 수를 B_i 라고 정의합시다.
- ✓ 주사위의 면을 N 개의 서로 다른 수로 채우려면, 다르게 조합할 수 있는 비트가 최소 $\lceil \log_2 N \rceil$ 개 필요합니다.
- ✓ S 에 켜져 있는 전체 비트의 개수를 B_S 라고 하면, $B_i = B_S - \lceil \log_2 N \rceil$ 가 됩니다.
- ✓ 우리가 구하려는 최소 주사위 개수를 K 라고 할 때, K 개의 주사위가 각각 최대 B_i 개의 비트를 보장하여 총 B_S 개의 비트를 모두 만들어야 합니다.
- ✓ 따라서 $K \times B_i \geq B_S$ 를 만족해야 하므로, 최소 $\lceil \frac{B_S}{B_i} \rceil$ 개의 주사위가 필요합니다.
- ✓ 단, $B_i \leq 0$ 인 경우, 주사위를 구성하는 것이 불가능하므로 -1 을 출력해야 합니다.



 . Alea jacta est!

- ✓ $L = \lceil \log_2 N \rceil$ 이라고 합시다.
- ✓ S 의 켜진 비트들을 최대 B_i 개씩 묶어, $K = \lceil \frac{B_S}{B_i} \rceil$ 개의 묶음으로 나눕니다.
- ✓ 각 묶음마다 주사위를 하나씩 만듭니다.
- ✓ i 번 묶음의 비트들을 OR한 값을 a_i 로 두고, 모든 면에 공통으로 넣습니다.
- ✓ 묶음에 포함되지 않은 S 의 켜진 비트는 최소 $B_S - B_i = L$ 개이므로, 그중 L 개를 고릅니다.
- ✓ 0부터 $N - 1$ 까지의 정수를 이진수로 나타내고, 각 자리를 선택한 비트에 대응시켜 면마다 추가할 비트를 정합니다.
- ✓ $N \leq 2^L$ 이므로 각 주사위의 면을 N 개의 서로 다른 수로 채울 수 있습니다.
- ✓ 각 묶음이 비어 있지 않아 모든 면의 값은 1 이상입니다.
- ✓ 모든 면의 값은 S 의 subset이므로 10^9 이하입니다.



 . Alea jacta est!

- ✓ S 의 모든 켜진 비트는 자신이 속한 묶음의 주사위가 보장하므로, 모든 a_i 를 OR한 값은 S 입니다.
- ✓ 따라서 K 개의 주사위로 조건을 만족시킬 수 있으며, 이것이 필요한 주사위의 최소 개수입니다.



최소 넓이 직각 격자 다각형

#constructive, #arithmetic, #geometry

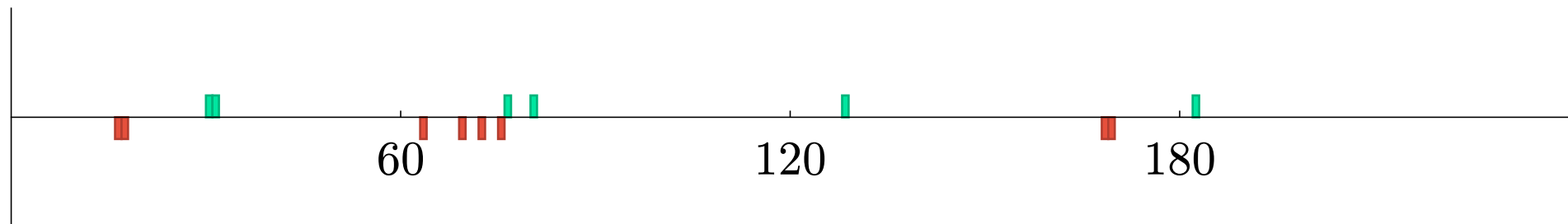
출제진 의도 - Hard

✓ 출제자: plan_pt

✓ 정답: 6 명 · 정답률 42.857%

✓ 처음 푼 참가자 cubious

30 분



초록: AC · 빨강: AC 이외 판정



. 최소 넓이 직각 격자 다각형

- ✓ 직각 다각형의 외곽선을 따라가면 가로 방향의 변과 세로 방향의 변이 번갈아 나타납니다. 따라서 꼭짓점의 개수 K 는 4 이상의 짝수여야 합니다.
- ✓ $K < 4$ 이거나 K 가 홀수이면 -1을 출력합니다.
- ✓ 이제 도형의 넓이를 A , 둘레를 P , 변을 공유하는 채운 칸 쌍의 개수를 E 라고 합시다. 각 칸의 넓이가 1이므로 A 는 채운 칸의 개수와 같습니다.
- ✓ 채운 칸 하나는 둘레에 4를 기여하지만, 두 채운 칸이 공유하는 변은 둘레에서 두 번 빠집니다. 따라서 $P = 4A - 2E$ 입니다.
- ✓ 채운 칸을 정점으로, 변을 공유하는 두 칸을 간선으로 연결한 그래프를 생각합시다. 모든 채운 칸이 연결되어 있으므로 $E \geq A - 1$ 이고, 따라서 $P \leq 2A + 2$ 입니다.
- ✓ 다각형의 변은 K 개이고 각 변의 길이가 1 이상이므로 $K \leq P$ 입니다. 이를 합치면 $K \leq 2A + 2$, 즉 $A \geq \frac{K}{2} - 1$ 을 얻습니다.
- ✓ 따라서 어떤 도형을 만들더라도 최소한 $\frac{K}{2} - 1$ 개의 칸이 필요합니다.



. 최소 넓이 직각 격자 다각형

- ✓ 정확히 $\frac{K}{2} - 1$ 개의 칸만 사용하는 경우를 생각해 봅시다. 앞의 부등식에 모두 등호가 성립해야 하므로, 외곽선의 모든 변의 길이는 1이고 채운 칸의 인접 그래프는 트리어야 합니다.
- ✓ 격자를 체스판처럼 두 색으로 칠합시다. 외곽선과 변을 공유하는 채운 칸을 경계 칸, 나머지 채운 칸을 내부 칸이라고 부르겠습니다.
- ✓ 외곽선을 따라 각 변의 안쪽 칸을 살펴보면, 볼록 꼭짓점에서는 같은 칸을 유지하고 오목 꼭짓점에서는 대각선에 있는 칸으로 넘어갑니다. 두 경우 모두 색이 바뀌지 않으므로 모든 경계 칸의 색이 같습니다.
- ✓ 따라서 경계 칸끼리는 인접할 수 없습니다. 인접한 두 칸은 항상 서로 다른 색이기 때문입니다.



. 최소 넓이 직각 격자 다각형

- ✓ 내부 칸끼리도 인접할 수 없습니다. 내부 칸은 네 방향의 이웃이 모두 채워져 있으므로, 내부 칸 두 개가 인접하면 채워진 2×2 정사각형이 생깁니다. 이는 인접 그래프에 사이클을 만들어 트리라는 조건에 모순됩니다.
- ✓ 결국 모든 간선은 경계 칸과 내부 칸을 연결합니다. 내부 칸의 개수를 X 라고 하면 각 내부 칸의 차수가 4이므로 $E = 4X$ 입니다.
- ✓ 트리어므로 $A = E + 1 = 4X + 1$ 이고, $K = 2A + 2 = 8X + 4$ 입니다.
- ✓ 즉, K 를 8로 나눈 나머지가 4일 때만 넓이 $\frac{K}{2} - 1$ 이 가능합니다. 나머지 경우에는 넓이가 정수이므로 적어도 $\frac{K}{2}$ 개의 칸이 필요합니다.



. 최소 넓이 직각 격자 다각형

- ✓ 이제 앞에서 구한 하한을 달성하는 도형을 구성합니다. K 를 8로 나눈 몫을 t 라고 하겠습니다.
- ✓ 첫 번째 행을 $.#.$ 으로 만든 뒤, $###$ 과 $.#.$ 두 행을 t 번 반복해서 붙입니다. 예를 들어 $t = 2$ 이면 다음과 같습니다.

$.#.$
 $###$
 $.#.$
 $###$
 $.#.$

- ✓ 한 칸짜리 행이 $t + 1$ 개, 세 칸짜리 행이 t 개이므로 넓이는 $4t + 1$ 입니다. 이웃한 두 행 사이에서 꼭짓점이 4개씩, 도형의 위아래 끝에서 2개씩 생기므로 꼭짓점은 $8t + 4$ 개입니다.
- ✓ 기본 도형의 맨 아래 부분을 바꾸어 네 가지 경우를 처리합니다.
 - ✓ $K = 8t + 4$ 이면 그대로 사용합니다. 넓이는 $4t + 1 = \frac{K}{2} - 1$ 입니다.



. 최소 넓이 직각 격자 다각형

- ✓ $K = 8t$ 이면 마지막 .#. 행을 제거합니다. 돌출된 한 칸이 사라지면서 꼭짓점이 4개 줄어들고, 넓이는 $4t = \frac{K}{2}$ 가 됩니다.
- ✓ $K = 8t + 2$ 이면 마지막 .#. 행을 #..으로 바꿉니다. 돌출된 칸을 왼쪽으로 옮기면 꼭짓점이 2개 줄어들고, 넓이는 $4t + 1 = \frac{K}{2}$ 로 유지됩니다.
- ✓ $K = 8t + 6$ 이면 마지막 행 아래에 .## 행을 붙입니다. 오른쪽으로 꺾인 두 칸이 추가되면서 꼭짓점이 2개 늘어나고, 넓이는 $4t + 3 = \frac{K}{2}$ 가 됩니다.
- ✓ 모든 경우 각 행의 채운 칸은 연속하고, 이웃한 두 행의 채운 부분이 겹치므로 모든 채운 칸이 연결됩니다.
- ✓ 마지막으로 상하좌우에 빈 테두리를 한 겹 추가합니다. 각 행의 빈 칸은 왼쪽이나 오른쪽 테두리까지 이동할 수 있으므로 모든 빈 칸도 연결됩니다.
- ✓ 각 구성은 꼭짓점이 정확히 K 개이고, 넓이가 앞서 증명한 하한과 일치하므로 정답입니다.
- ✓ 행을 문자열로 직접 구성하면 시간 복잡도와 공간 복잡도는 모두 $\mathcal{O}(K)$ 입니다. 빈 테두리를 포함해 너비는 5, 높이는 최대 752이므로 출력 크기 제한도 만족합니다.



+ . 자릿수 합분해

#constructive, #math, #greedy, #implementation

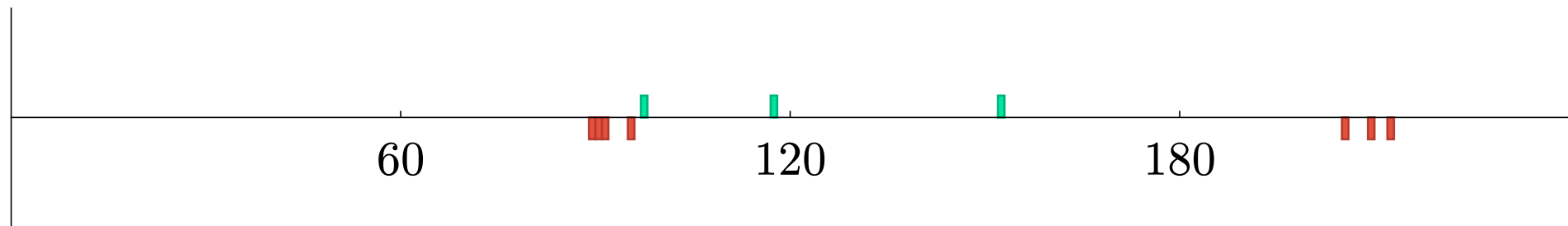
출제진 의도 - Challenging

✓ 출제자: chan120714, wapas

✓ 정답: 3 명 · 정답률 30.000%

✓ 처음 푼 참가자 hyperbolic

97 분



초록: AC · 빨강: AC 이외 판정



+ . 자릿수 합분해

- ✓ 양의 정수 N 개의 합이 M 이어야 하므로 $N \leq M$ 입니다.
- ✓ 또한, 각 양의 정수의 자릿수 합은 최소 1이므로 $K \geq N$ 이어야 합니다.
- ✓ M 의 i 번째 자리 숫자를 m_i , 그 자리에서 다음 자리로 넘어가는 받아 올림을 c_{i+1} , N 개 수의 i 번째 자리 숫자 합을 A_i 라고 해봅시다.
- ✓ 그러면 $A_i + c_i = m_i + 10c_{i+1}$ 이므로 $A_i = m_i + 10c_{i+1} - c_i$ 가 됩니다.
- ✓ $s(M)$ 을 M 의 자릿수 합이라고 둡시다.
- ✓ 각 자리에는 N 개의 숫자가 있으므로, 실제로 만들 수 있으려면 $0 \leq A_i \leq 9N$ 이어야 합니다.
- ✓ 모든 자리에 대해 위의 식을 더하면 $K = s(M) + 9\sum c_i$ 가 됩니다.
- ✓ 따라서 받아올림의 총합은 반드시 $C = \frac{K-s(M)}{9}$ 이어야 합니다.
- ✓ 그러므로 $K \geq s(M)$ 이고, $k \equiv s(M) \pmod{9}$ 이어야 합니다.
- ✓ 처음에는 받아올림이 하나도 없으므로 모든 $c_i = 0$, $A_i = m_i$ 로 초기화합니다.



+ . 자릿수 합분해

- ✓ 이제 총 $C = \frac{K-s(M)}{9}$ 번의 받아내림을 수행하면 됩니다.
- ✓ i 번째 자리에서 한 번 받아내리면 A_i 가 1 감소하고 A_{i-1} 이 10 증가합니다.
- ✓ 따라서 $A_i > 0$ 이고 $A_{i-1} + 10 \leq 9N$ 이면 받아내릴 수 있습니다.
- ✓ 가능한 가장 낮은 자리부터 받아내리면 그 자리의 값이 감소하므로 이후 더 높은 자리에서 받아내릴 공간을 만들 수 있습니다.
- ✓ 더 이상 받아내릴 수 없는데 C 가 남아있다면 불가능합니다.
- ✓ 이제 구한 A_i 들을 실제 N 개의 수에 분배해봅시다.
- ✓ 먼저 A_i 에서 1씩 꺼내 서로 다른 수에 넣어 모든 수를 양수로 만듭니다.
- ✓ 이후 남은 A_i 를 각 수의 해당 자리가 9를 넘지 않도록 분배합니다.
- ✓ $A_i \leq 9N$ 이므로 모두 분배할 수 있습니다.
- ✓ 이렇게 만들면 각 자리의 합이 A_i 이므로 N 개 수의 합은 M , 모든 자리의 숫자 합은 K 가 됩니다.
- ✓ 전체 시간복잡도는 $\mathcal{O}(N + K)$ 입니다.



순열 LCA

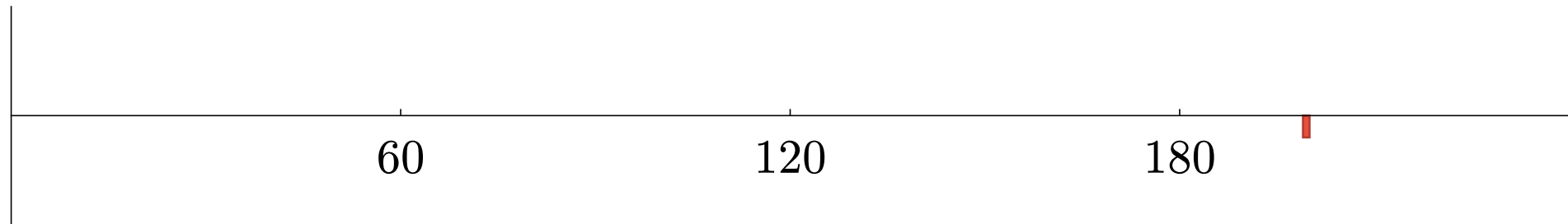
#constructive, #disjoint_set

출제진 의도 - Challenging

✓ 출제자: kangsm02, wapas

✓ 정답: 0 명 · 정답률 0.000%

✓ 처음 푼 참가자 -



초록: AC · 빨강: AC 이외 판정



순열 LCA

- ✓ $d(A, B)$ 를 순열 A 에서 B 로 만드는 데 필요한 최소 swap 연산 횟수라고 정의하고, $D = d(A, B)$ 라고 합시다.
- ✓ 순열의 각 위치를 하나의 정점으로 생각합시다. A 의 i 번째 원소가 B 의 j 번째 원소와 같다면, i 에서 j 로 가는 간선을 그립니다.
- ✓ 길이가 l 인 사이클을 정리하려면 최소 $l - 1$ 번의 swap이 필요하므로, 두 순열 사이의 최소 swap 횟수는 N 에서 사이클 수를 뺀 값입니다.
- ✓ 불가능한 경우부터 살펴보겠습니다.
- ✓ $d(C, A) = d(C, B) = K$ 이므로, $D \leq 2K$ 여야 합니다.
- ✓ 또한 swap은 inversion 개수의 홀짝성을 반전시킵니다. A 와 B 는 C 로부터 각각 K 번의 swap으로 얻어지므로 홀짝성이 같아야 하며, 따라서 D 는 짝수여야 합니다.
- ✓ $D > 2K$ 이거나 D 가 홀수라면 -1을 출력합니다.



순열 LCA

- ✓ 이제 순열을 구성해봅시다. 먼저 A 와 B 의 정확히 중간 지점에 있는 순열을 찾습니다.
- ✓ 순열의 각 위치를 하나의 정점으로 생각합시다. A 의 i 번째 원소가 B 의 j 번째 원소와 같다면, i 에서 j 로 가는 간선을 그립니다.
- ✓ 위 그래프의 각 사이클에서 한 위치를 기준으로 삼습니다. 간선을 따라 방문하는 순서대로 나머지 위치와 swap합니다. 이 연산들을 모두 수행하면 정확히 D 번의 swap으로 B 가 됩니다.
- ✓ 따라서 전체 swap 횟수가 $\frac{D}{2}$ 가 되었을 때 멈추면, 만들어진 순열 C 는 $d(A, C) = d(B, C) = \frac{D}{2}$ 를 만족합니다.
- ✓ 이제 C 의 i 번째 원소가 A 의 j 번째 원소와 같을 때 i 에서 j 로 가는 간선을 그립니다. 이렇게 얻은 그래프를 A' 이라고 하겠습니다.
- ✓ 마찬가지로 C 와 B 에 대해서 만든 그래프를 B' 라고 하겠습니다.



순열 LCA

- ✓ 서로 다른 사이클에 속하는 두 위치를 swap하면 두 사이클이 하나로 합쳐집니다. 따라서 A' 과 B' 각각에서 서로 다른 사이클에 속하는 두 위치를 선택하면, 두 최소 swap 거리가 정확히 1씩 증가합니다.
- ✓ 이러한 작업을 $K - \frac{D}{2}$ 번 수행하면 됩니다.



순열 LCA

- ✓ 필요한 위치 쌍을 효율적으로 찾기 위해 하나의 위치를 고정합니다. 각 기준에서 이 위치를 포함하는 사이클을 anchor라고 부르겠습니다.
- ✓ 먼저 모든 위치를 한 번 순회하면서, 양쪽 기준에서 모두 anchor 밖에 있는 위치를 찾습니다. 그런 위치가 있으면 고정한 위치와 swap합니다.
- ✓ 이 연산은 양쪽에서 각각 외부 사이클 하나를 anchor에 합칩니다. 따라서 두 거리가 동시에 1씩 증가합니다.
- ✓ 중간 순열에서 구한 각 사이클이 anchor에 합쳐졌는지를 표시해 두면, 각 위치가 anchor 안에 있는지 빠르게 확인할 수 있습니다.
- ✓ 모든 병합이 anchor로 이루어지므로, swap 이후에는 새로 합쳐진 사이클의 표시만 갱신하면 됩니다. 사이클을 매번 다시 구할 필요는 없습니다.
- ✓ 두 거리가 K 가 되면 종료합니다. 그렇지 않고 전체 순회를 마쳤다면, 양쪽 모두 anchor 밖에 있는 위치는 더 이상 없습니다.
- ✓ 그런 위치는 순회 중에 선택되었을 것이고, 한 번 anchor에 포함된 위치는 이후에도 계속 포함되기 때문입니다. 따라서 남은 외부 사이클의 모든 위치는 반대쪽 기준의 anchor 안에 있습니다.



순열 LCA

- ✓ 아직 거리를 더 늘려야 한다면, 두 종류의 대표 위치를 모읍니다. 먼저 외부 B' 사이클마다 위치를 하나씩 선택합니다. 이 위치들은 모두 A' 기준으로 anchor 안에 있습니다.
- ✓ 마찬가지로 외부 A' 사이클마다 위치를 하나씩 선택합니다. 이 위치들은 모두 B' 기준으로 anchor 안에 있습니다. 두 종류 모두 같은 외부 사이클에서 대표를 중복해서 선택하지 않습니다.
- ✓ 두 종류의 대표를 하나씩 짝지어 swap합니다. A' 기준에서는 anchor에 새로운 외부 A' 사이클이 합쳐지고, B' 기준에서는 anchor에 새로운 외부 B' 사이클이 합쳐집니다.
- ✓ 각 외부 사이클에서 대표를 하나만 선택했으므로, 앞선 swap 이후에도 다음 대표의 외부 사이클은 아직 합쳐지지 않은 상태입니다. 따라서 미리 선택한 쌍들을 차례로 처리할 수 있습니다.
- ✓ 이 단계를 시작할 때의 두 거리를 t 라고 합시다. 양쪽 사이클 수는 각각 $N - t$ 이므로, 두 종류의 대표를 각각 $N - t - 1$ 개씩 얻습니다. 필요한 추가 swap 횟수는 $K - t$ 이고,

$$K - t \leq N - t - 1$$

이므로 대표의 수는 항상 충분합니다.



순열 LCA

- ✓ 필요한 만큼 swap하면 $d(A, C) = d(B, C) = K$ 가 됩니다.
- ✓ 전체 시간복잡도와 공간복잡도는 모두 $O(N)$ 입니다.



. 발전소

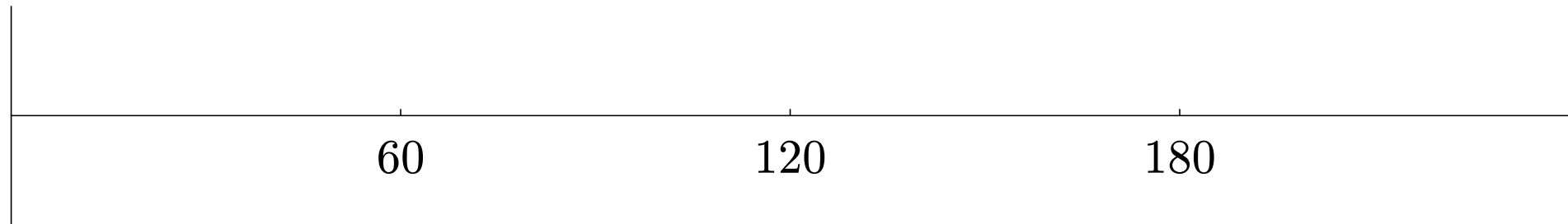
#constructive, #heuristic

출제진 의도 - [Extreme](#)

✓ 출제자: yyyy7089

✓ 정답: 0 명 · 정답률 -

✓ 처음 푼 참가자 -

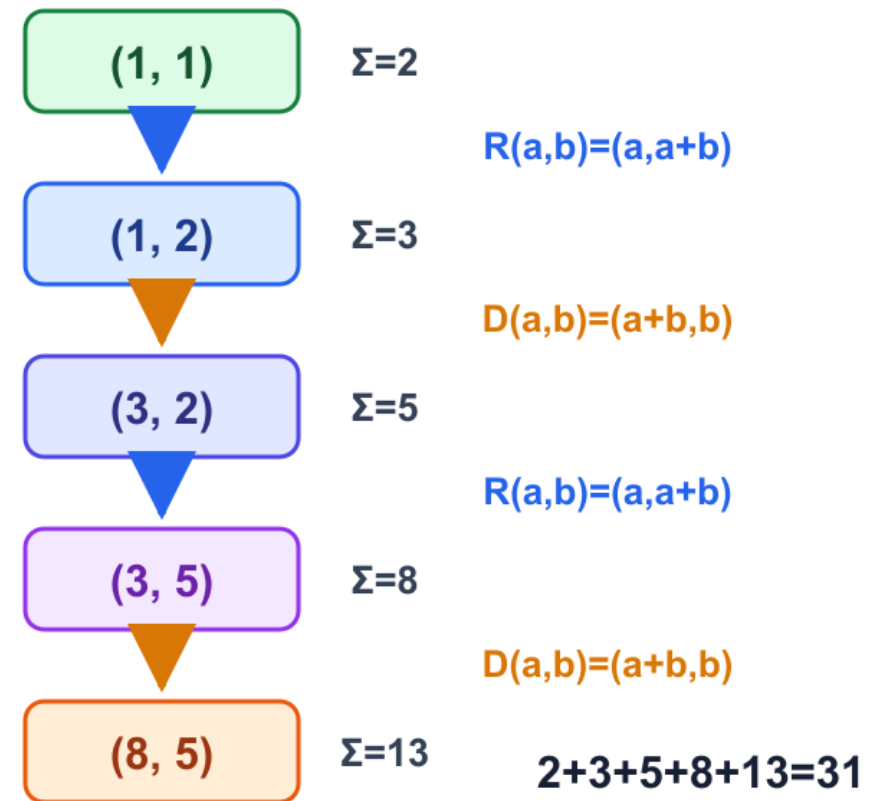
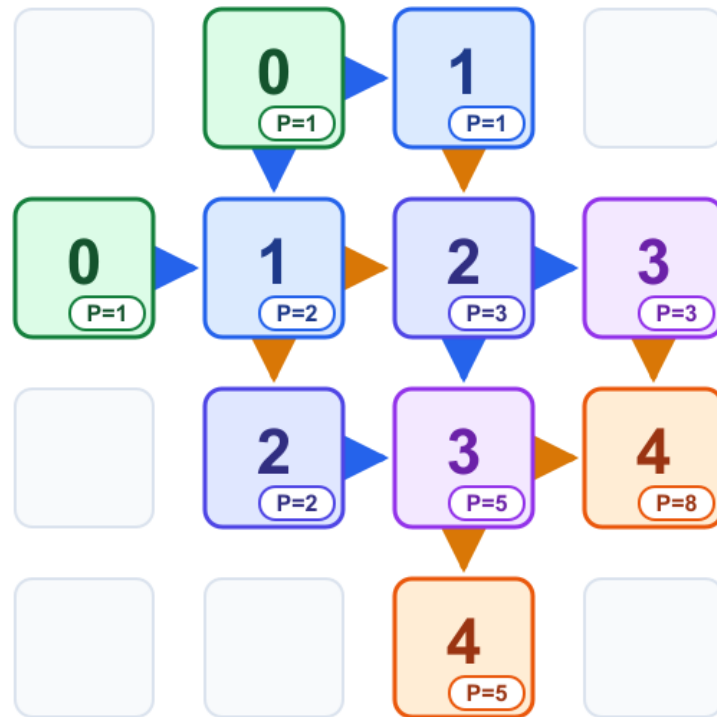


초록: AC · 빨강: AC 이외 판정



. 발전소

- ✓ 본 에디토리얼에서는 출제진이 생각하는 가장 좋은 풀이를 설명합니다
- ✓ 이 문제를 풀기 위해서는 다음과 같은 사고 흐름이 필요합니다.
- ✓ 각 칸에 놓인 기계의 기본 효과 1이 최종 생산 전력에 기여하는 총량을 그 칸의 전력 기여량이라고 하겠습니다.
- ✓ 전체 생산 전력은 모든 칸의 전력 기여량을 합한 값입니다.
- ✓ 먼저 같은 단계의 두 칸을 대각선으로 엇갈리게 놓고, 엇갈리는 방향을 계속 이어 붙이는 배치를 생각합니다.
- ✓ 낮은 단계부터 전력 기여량을 계산하면, 한 단계 높은 칸의 전력 기여량은 인접한 한 단계 낮은 칸들의 전력 기여량을 합한 값이 됩니다. 엇갈리는 방향을 번갈아 바꾸면 두 칸의 전력 기여량이 피보나치 수열 형태로 증가합니다. 앞으로 이 구조를 ‘피보나치 배치법’이라고 정의합니다.



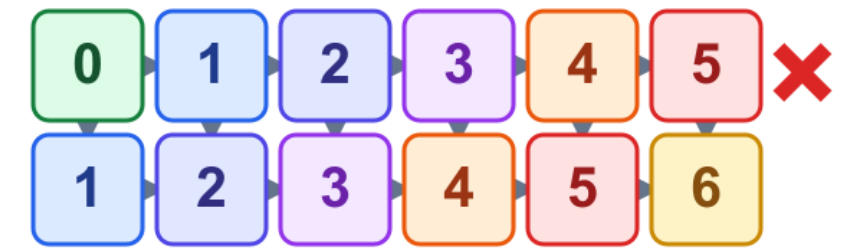


. 발전소

- ✓ 엇갈린 두 칸짜리 배치를 같은 가로나 세로 방향으로만 계속 이어 붙이면 두 줄이 서로 합쳐지지 않거나, 두 칸 중 한쪽의 전력 기여량이 그대로 유지되어 다른 쪽에 같은 값만 반복해서 더해집니다. 따라서 전력 기여량이 빠르게 증가하지 않습니다.
- ✓ 피보나치 배치법의 효율을 높이려면 같은 단계인 두 칸의 전력 기여량을 더한 값이 양쪽 칸에 번갈아 반영되어야 합니다.
- ✓ 같은 단계의 두 칸을 대각선으로 배치하고 진행 방향을 번갈아 꺾으면, 두 칸의 전력 기여량에 $(a, b) \rightarrow (a, a + b)$ 와 $(a, b) \rightarrow (a + b, b)$ 를 번갈아 적용할 수 있습니다. 따라서 전력 기여량이 피보나치 수열 형태로 증가합니다.



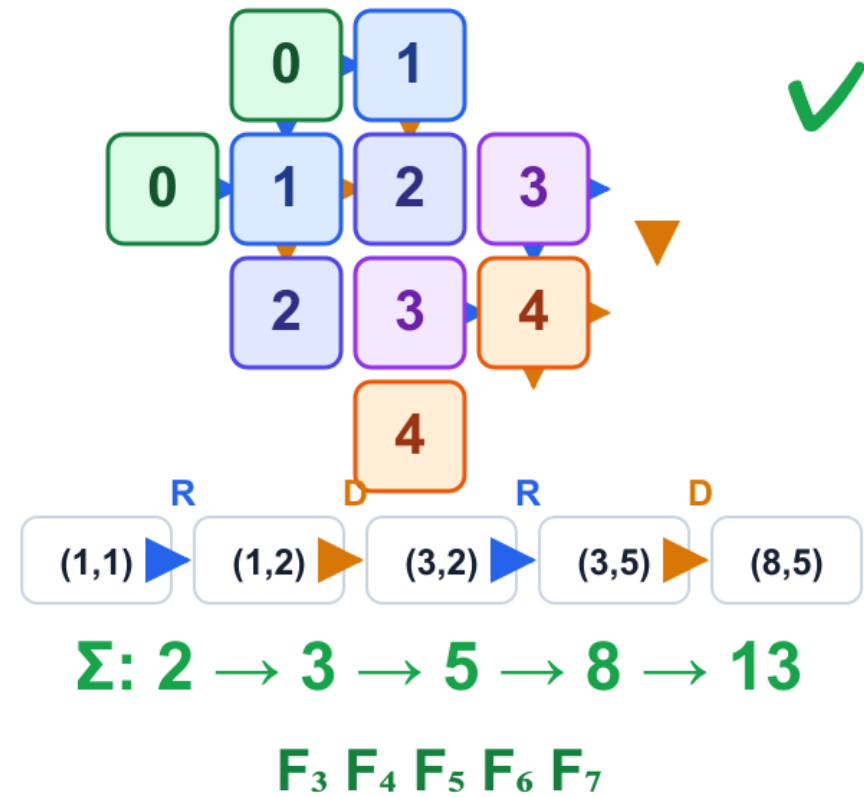
⚡ . 발전소



$\Sigma: 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$

$(1, m+1)$

+1 +1 +1 +1



$\Sigma: 2 \rightarrow 3 \rightarrow 5 \rightarrow 8 \rightarrow 13$

$F_3 F_4 F_5 F_6 F_7$



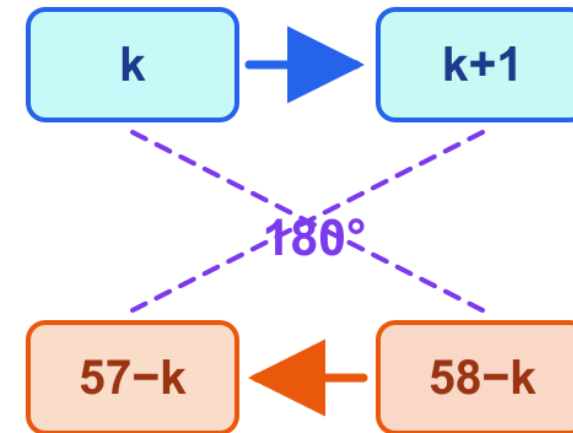
. 발전소

- ✓ 따라서 같은 단계의 두 칸이 대각선으로 맞물리는 피보나치 배치법을 10×10 격자 안에 여러 번 접어 넣습니다.
- ✓ 마지막으로 직접 정해야 하는 칸의 수를 절반으로 줄이기 위해, 각 칸의 값과 그 칸을 180° 회전한 위치의 값의 합이 58이 되도록 구성합니다. 행과 열을 0부터 세면 $A_{r,c} + A_{9-r,9-c} = 58$ 입니다.
- ✓ 그러면 서로 대응하는 인접한 두 칸의 단계 차이는 부호만 반대이고 절댓값은 같아집니다. 따라서 단계 차이가 1인 인접 관계의 배치는 보드의 중심을 기준으로 점대칭이 됩니다.
- ✓ 여기서 점대칭인 것은 각 칸의 값 자체가 아니라 인접한 칸 사이의 단계 차이 배치입니다.
- ✓ 이와 같이 피보나치 배치법과 점대칭을 함께 사용하면 문제의 조건을 만족하는 구성법에 도달할 수 있습니다.



10×10

38	39	40	41	42	43	58	57	56	55
37	38	39	42	43	44	45	58	55	54
36	35	34	33	44	45	46	47	54	53
35	34	33	32	31	46	47	48	51	52
34	33	32	31	30	29	48	49	50	51
7	8	9	10	29	28	27	26	25	24
6	7	10	11	12	27	26	25	24	23
5	4	11	12	13	14	25	24	23	22
4	3	0	13	14	15	16	19	20	21
3	2	1	0	15	16	17	18	19	20



$$A(Rp) = 58 - A(p)$$

$$k + (58 - k) = 58$$

$$(k+1) + (57-k) = 58$$

$$|k - (k+1)| = |(57-k) - (58-k)| = 1$$

$$A[r,c] + A[9-r,9-c] = 58$$



🌙. 달빛 등대의 섬

#constructive, #number_theory

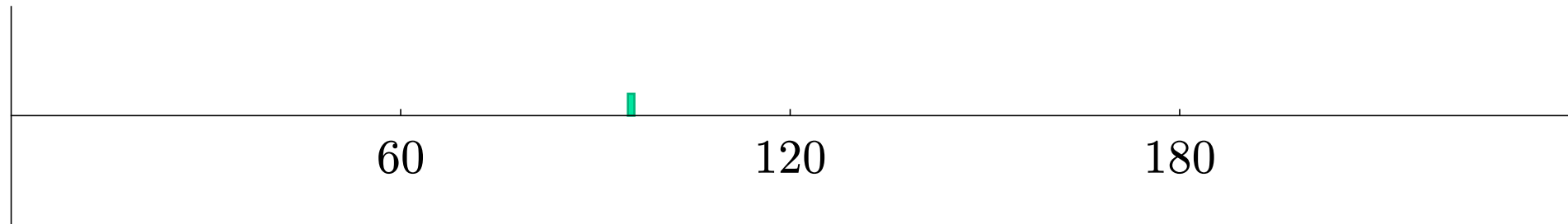
출제진 의도 - Challenging

✓ 출제자: saywoo, wapas

✓ 정답: 1 명 · 정답률 100.000%

✓ 처음 푼 참가자 cubious

95 분



초록: AC · 빨강: AC 이외 판정



🌙. 달빛 등대의 섬

- ✓ 전체 등대 수 K 는 각 행의 등대 수의 합이자, 각 열의 등대 수의 합입니다.
- ✓ 각 행과 열에 홀수 개의 등대가 있어야 하므로, $N \equiv M \equiv K \pmod{2}$ 이어야 합니다.
- ✓ 각 행과 열에 홀수 개의 등대가 들어가므로, N 개의 홀수의 합과 M 개의 홀수의 합이 모두 K 가 되어야 합니다.
- ✓ 또한 각 행과 열에 최소 1개의 등대가 필요하므로, $K \geq \max(N, M)$ 이어야 합니다.
- ✓ N, M 이 모두 짝수라면 각 행과 열의 빈칸 수도 홀수여야 합니다.
- ✓ 따라서 빈칸도 최소 $\max(N, M)$ 개 필요하므로, $K \leq NM - \max(N, M)$ 이어야 합니다.
- ✓ 반면 N, M 이 모두 홀수라면 각 행과 열의 빈칸 수는 짝수여야 합니다.
- ✓ 이때 빈칸이 정확히 2개인 경우는 불가능합니다.
- ✓ 위 조건을 모두 통과했다면 항상 배치를 구성할 수 있습니다.
- ✓ 이하에서는 $N \leq M$ 이라고 가정합니다.



달빛 등대의 섬

✓ N 이 짝수일 때

- ✓ N, M 이 짝수이므로 전체 격자를 뒤집어도 각 행과 열의 홀짝 조건은 유지됩니다. 따라서 $k > \frac{NM}{2}$ 라면 $NM - k$ 개 짜리 답을 만든 뒤 전체를 뒤집으면 됩니다. $k \leq \frac{NM}{2}$ 인 경우만 구성하면 충분합니다.
- ✓ 우선 $(1, 1)$ 부터 오른쪽 아래 방향 대각선으로 $\#$ 을 채우고 남은 열의 가장 아래 행에 $\#$ 를 채워주면 $\#$ M 개를 채울 수 있습니다.
- ✓ 2×2 블록의 네 칸을 모두 뒤집어도 각 행과 열에서는 정확히 2칸씩 변하므로 $\#$ 개수의 홀짝은 유지된다는 것을 관찰 할 수 있습니다.
- ✓ $\begin{pmatrix} \cdot & \# \\ \# & \# \end{pmatrix}$ 를 뒤집으면 $\begin{pmatrix} \# & \cdot \\ \cdot & \cdot \end{pmatrix}$ 가 되어 $\#$ 의 개수가 $3 \rightarrow 1$, 즉 2개 감소합니다.
- ✓ 마찬가지로 2×2 안의 격자를 채움으로써 전체 $\#$ 개수를 4개씩 추가할 수 있습니다. 따라서 처음 각 행과 열의 가장 앞 부분부터 하나 씩 2×2 $\#$ 격자를 배치한 뒤 필요 할 시에 위의 방법으로 2개 감소하여 원하는 k 를 만들 수 있습니다.



달빛 등대의 섬

✓ N 이 홀수일 때

- ✓ $M \leq K \leq N + M - 1$ 인 경우에는 각 행과 열에 기본적으로 하나씩 등대를 놓고, 첫 행과 첫 열을 이용해 필요한 만큼 늘려 직접 구성할 수 있습니다.
- ✓ 기본 배치는 첫 행과 첫 열을 모두 채운 형태입니다. 이때 등대의 수는 $N + M - 1$ 개이고 모든 행과 열의 등대 수는 홀수입니다.
- ✓ $K - (N + M - 1) \equiv 0 \pmod{4}$ 인 경우는 이후 격자의 오른쪽 아래부터 2×2 블록을 하나씩 추가하면 각 행과 열의 등대 수가 2씩 증가하므로 홀짝은 유지되고, 전체 등대 수는 4씩 증가합니다.
- ✓ $K - (N + M - 1) \equiv 2 \pmod{4}$ 인 경우에는 $(1, 1)$ 을 포함하는 2×2 칸을 뒤집어 $\begin{pmatrix} \cdot & \cdot \\ \cdot & \# \end{pmatrix}$ 의 배치로 사용합니다. 즉, 이후 위와 같은 방법으로 채우면 모든 경우를 채울 수 있습니다.



12. Max Mex Matrix

#constructive, #ad_hoc

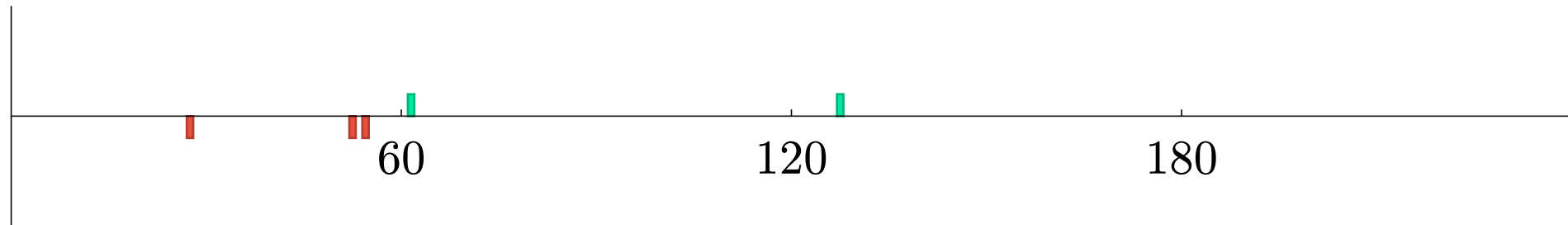
출제진 의도 - Hard

✓ 출제자: tapris

✓ 정답: 2 명 · 정답률 40.000%

✓ 처음 푼 참가자 azberjibiou

61 분



초록: AC · 빨강: AC 이외 판정



12 34. Max Mex Matrix

- ✓ $N = 1, M = 1$ 인 경우 0 하나로 이루어진 행렬이 정답입니다. 이 경우를 제외하고 $N = M$ 인 경우와 $N \neq M$ 로 나누어 봅시다.
- ✓ $N \neq M$ 인 경우 일반성을 잃지 않고 $N < M$ 으로 두겠습니다.
- ✓ $N < M$ 인 경우 한 열을 비우고 나머지 부분을 각 행에 0부터 $M - 2$ 까지 각 열에서 같은 숫자가 나오지 않도록 배치한 후 비워진 열에 $M - 1$ 부터 $N + M - 2$ 까지 배치하면 됩니다.
- ✓ 크기가 $N + M - 2$ 인 집합의 MEX값이 $N + M - 2$ 를 넘을 수 없으므로 이보다 최댓값이 큰 행렬은 존재하지 않습니다.
- ✓ $N = M$ 인 경우 한 행을 비우고 $N < M$ 일때의 방법으로 나머지 부분을 배치한 뒤 비워졌던 부분을 채우면 됩니다.
- ✓ 최댓값은 $N + M - 3$ 이 됩니다.
- ✓ 각 행과 열에 0이 정확히 하나가 필요합니다. 따라서 0이 아닌 원소는 행과 열에 0이 하나씩 존재합니다.
- ✓ 중복된 값이 항상 최소 하나 존재하므로 크기가 $N + M - 3$ 인 집합의 MEX값이 $N + M - 3$ 를 넘을 수 없고 이보다 최댓값이 큰 행렬은 존재하지 않습니다.



. 센트컵

#constructive, #dp

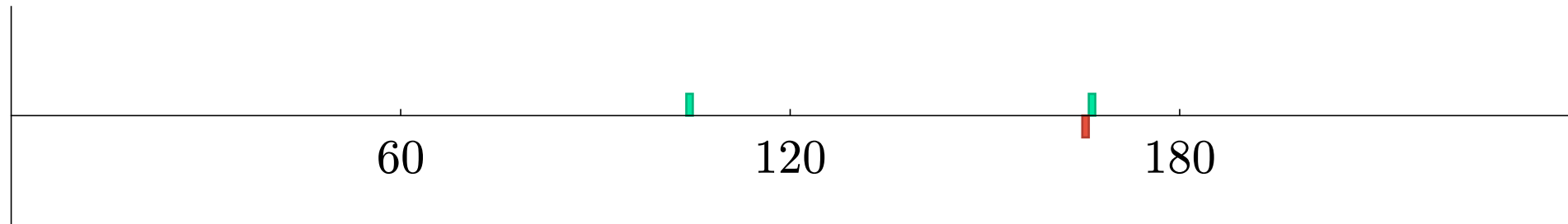
출제진 의도 - Hard

✓ 출제자: bluerini

✓ 정답: 2 명 · 정답률 66.667%

✓ 처음 푼 참가자 cubious

104 분



초록: AC · 빨강: AC 이외 판정



. 센트컵

- ✓ 위치 x 의 문자를 선택할 때 c 는 $-x$, u 는 0 , p 는 $+x$ 의 점수를 부여합니다. 그러면 컵 하나의 점수는 크기와 같은 $k - i$ 가 됩니다.
- ✓ $D_{i,j}$ 를 문자열의 앞 i 개 문자에서 목표 문자열의 앞 j 개 문자를 선택했을 때의 최대 점수로 정의합니다. 초깃값은 $D_{0,0} = 0$ 이며, 나머지는 $-\infty$ 입니다.
- ✓ 현재 문자를 선택하지 않는 경우, $D_{i,j} = D_{i-1,j}$ 입니다.
- ✓ 현재 문자가 목표 문자열의 j 번째 문자와 같고 이전 상태가 가능하다면, 해당 문자를 선택할 수 있습니다. 이 경우에는

$$D_{i,j} = \max(D_{i,j}, D_{i-1,j-1} + w_i)$$

입니다. 여기서 w_i 는 앞서 정한 문자의 점수입니다.



. 센트컵

- ✓ $D_{N,300}$ 이 정답입니다. 값이 $-\infty$ 라면 100개의 컵을 만들 수 없으므로 -1을 출력합니다.
- ✓ 구성을 복원하기 위해 각 상태에 마지막으로 선택한 위치를 함께 저장합니다. $(N, 300)$ 부터 역추적하면서, 저장된 위치가 i 라면 i 를 정답에 추가하고 $(i-1, j-1)$ 로, 아니라면 $(i-1, j)$ 로 이동합니다.
- ✓ 복원한 위치들을 뒤집은 뒤 3개씩 묶어 출력합니다. 시간 복잡도와 공간 복잡도는 모두 $O(300N)$ 입니다.



❤️. 회의실 예약 시스템 저격

#constructive, #greedy

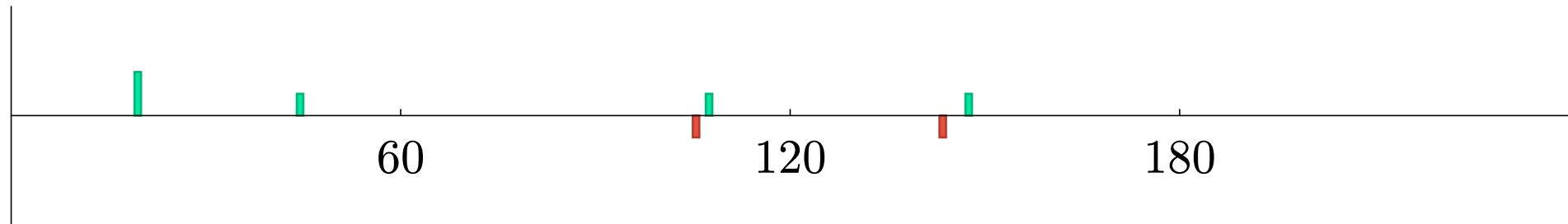
출제진 의도 - Hard

✓ 출제자: chan120714, wapas

✓ 정답: 5 명 · 정답률 71.429%

✓ 처음 푼 참가자 dadasO8

19 분



초록: AC · 빨강: AC 이외 판정



회의실 예약 시스템 저장

- ✓ 하한부터 생각해봅시다.
- ✓ 유검이가 아무리 배치를 잘해도 최적해보다 많이 배정할 수 없음은 자명합니다.
- ✓ 따라서 $A \leq B$ 입니다.



회의실 예약 시스템 저장

- ✓ 이제 상한을 증명해봅시다.
- ✓ 유검이가 선택된 회의들을 G , 최적해에서 선택된 회의들을 I 라고 해봅시다.
- ✓ I 에 대해서
 - ✓ I 가 유검이가 선택한 회의라면 자기 자신에 대응
 - ✓ I 가 유검이가 선택한 회의가 아니라면 유검이가 선택한 시간이 겹치는 회의 G_i 에 대응 해줍니다.
- ✓ 이때, I 가 G_i 에 의해 탈락했다면 $|G_i| \leq |I|$ 입니다.
- ✓ 이제, 하나의 $G_i = [l, r]$ 에 대응된 최적해 회의를 봅시다.
- ✓ 대응된 최적해 회의를 $I = [x, y]$ 라 하면, I 는 2가지 조건을 만족합니다.
 - ✓ I 와 G_i 가 겹친다.
 - ✓ I 의 길이는 G_i 이상이다.
- ✓ 이런 회의는 l 또는 r 과 겹칩니다.
- ✓ I 가 l 또는 r 과 겹치지 않는다고 가정해봅시다.



회의실 예약 시스템 저장

- ✓ 그러면 I 가 l 과 r 사이에 들어가야하므로 $l < x \leq y < r$ 이고, 따라서 $y - x < r - l$ 이 됩니다.
- ✓ 근데 $|G_i| \leq |I|$ 이므로 모순입니다.
- ✓ 따라서 최적해 회의는 l 을 포함하거나 r 을 포함해야 합니다.
- ✓ l 이라는 동일한 시점을 포함하는 회의가 2개 이상일수는 없으므로, 하나의 G_i 에 대응 되는 회의는 최대 2개입니다.
- ✓ 따라서 상한은 $B \leq 2A$ 입니다.
- ✓ 따라서 $A \leq B \leq 2A$ 일때 구성가능합니다.
- ✓ 해 구성은 $[10x, 10x + 3], [10x + 3, 10x + 5], [10x + 5, 10x + 8]$ 을 $B - A$ 개,
- ✓ $[10x, 10x + 1]$ $2A - B$ 개를 출력하면 됩니다.



. 슬라이딩 퍼즐

#constructive, #math

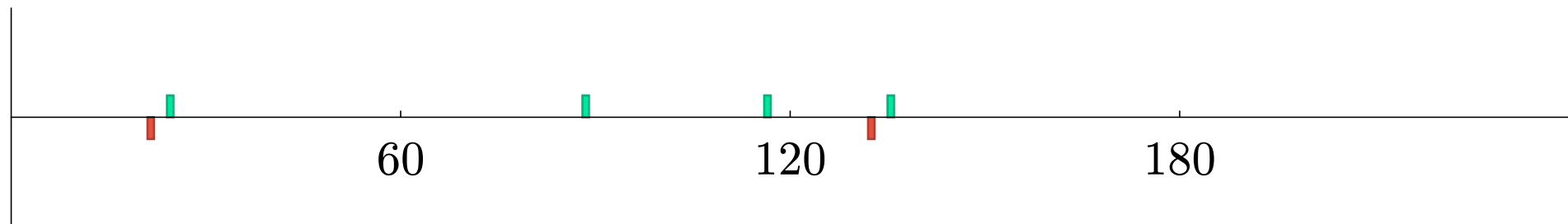
출제진 의도 - Medium

✓ 출제자: wapas

✓ 정답: 4 명 · 정답률 66.667%

✓ 처음 푼 참가자 azberjibiou

24 분



초록: AC · 빨강: AC 이외 판정



. 슬라이딩 퍼즐

- ✓ 먼저 필요한 가로 이동 횟수의 최솟값을 구해봅시다.
- ✓ 수 i 가 적힌 두 숫자의 시작 열은 i 와 $N + i$ 이고, 목표 열은 $2i - 1$ 과 $2i$ 입니다.
- ✓ 왼쪽 숫자를 왼쪽 목표에 대응하면 두 숫자가 필요한 가로 이동은 적어도 $(i - 1) + (N - i) = N - 1$ 번입니다. 두 숫자의 목표를 반대로 설정하면 $N + 1$ 번이 필요하므로, 왼쪽 숫자를 왼쪽 목표로 놓는게 이득입니다.
- ✓ 따라서 모든 i 를 합치면 가로 이동은 적어도 $N \times (N - 1)$ 번 필요합니다.
- ✓ 이제 필요한 세로 이동 횟수를 구해봅시다. 아랫줄에 한 번도 내려가지 않은 숫자끼리는 윗줄에서 서로의 순서를 바꿀 수 없습니다.
- ✓ 따라서 이 숫자들은 처음 수열 $A = (1, 2, \dots, N, 1, 2, \dots, N)$ 과 목표 수열 $B = (1, 1, 2, 2, \dots, N, N)$ 의 공통 부분수열을 이룹니다.
- ✓ A 의 첫 번째 블록(앞 N 개 원소)에서 고른 부분 수열을 P , 두 번째 블록(뒤 N 개 원소)에서 고른 부분 수열을 Q 라고 합시다. 두 부분 수열은 $a = \max(P)$, $b = \min(Q)$ 에 대해 $a \leq b$ 를 만족해야 합니다.



슬라이딩 퍼즐

- ✓ P 에서는 원소를 최대 a 개 고를 수 있고, Q 에서는 최대 $N - b + 1$ 개 고를 수 있습니다. 따라서 공통 부분수열의 길이는 $a + N - b + 1 \leq N + 1$ 입니다. 한 블록에서만 고른 경우에도 길이는 N 이하입니다.
- ✓ 한편 $(1, 2, \dots, N, N)$ 은 실제로 길이가 $N + 1$ 인 최장 공통 부분수열입니다. 따라서 두 수열의 LCS 길이는 정확히 $N + 1$ 입니다.
- ✓ 전체 $2N$ 개 중 적어도 $N - 1$ 개는 아랫줄을 방문해야 합니다. 각 숫자는 윗줄에서 시작해 윗줄에서 끝나므로 내려가는 이동과 올라오는 이동이 한 번씩 필요합니다.
- ✓ 따라서 세로 이동은 적어도 $2(N - 1)$ 번 필요하고, 전체 연산 횟수는 적어도 $N(N - 1) + 2(N - 1) = (N - 1)(N + 2)$ 번입니다.
- ✓ 이제 이 하한을 달성하는 이동을 만들어봅시다. $x = 1, 2, \dots, N - 1$ 을 차례로 처리하고, $L = 2x$, $R = N + x$ 라고 정의합니다.
- ✓ 단계 x 가 시작될 때 윗줄의 앞에는 1부터 $x - 1$ 까지가 두 개씩 확정되어 있고, 그 뒤에는 $(x, x + 1, \dots, N, x, x + 1, \dots, N)$ 이 놓여 있습니다. 아랫줄은 모두 비어 있습니다.



. 슬라이딩 퍼즐

- ✓ 따라서 두 번째 x 는 윗줄 R 열에 있습니다. 먼저 $1 \ R \ D$ 를 출력해 이 숫자를 아랫줄로 내립니다.
- ✓ $c = R - 1, R - 2, \dots, L$ 의 순서로 $1 \ c \ R$ 을 출력하면 윗줄 L 열부터 $R - 1$ 열까지의 숫자가 오른쪽으로 한 칸씩 이동합니다.
- ✓ 이어서 $c = R, R - 1, \dots, L + 1$ 의 순서로 $2 \ c \ L$ 을 출력해 아랫줄의 x 를 L 열까지 옮깁니다.
- ✓ 윗줄에서는 방금 생긴 빈칸을 오른쪽에서 왼쪽으로 옮기고, 아랫줄에서는 비어 있는 칸을 따라 x 를 옮겼으므로 지금까지의 모든 이동이 가능합니다.
- ✓ 마지막으로 $2 \ L \ U$ 를 출력하면 x 가 목표 위치로 올라옵니다. 윗줄의 앞부분은 $1, 1, 2, 2, \dots, x, x$ 로 확정되고, 아랫줄은 다시 모두 비게 됩니다.
- ✓ 단계 x 에서 출력하는 이동은 $1 + (R - L) + (R - L) + 1 = 2(N - x) + 2$ 개입니다.
- ✓ 따라서 전체 이동 수는 $\sum_{x=1}^{N-1} (2(N - x) + 2) = (N - 1)(N + 2)$ 입니다.
- ✓ 이는 하한값과 같으므로 최소입니다.



. 슬라이딩 퍼즐

- ✓ 첫 줄에 이 이동 수를 출력한 뒤 위 과정을 그대로 출력하면 됩니다. $N = 1$ 이면 이동 수는 0이고 아무 이동도 출력하지 않습니다.
- ✓ 시간 복잡도는 출력량과 같은 $\mathcal{O}(N^2)$ 입니다.
- ✓ 증명은 복잡한데, 다음 페이지 사진처럼 머릿속에 떠올린 그 전략으로 옮기면 최소가 맞습니다.



. 슬라이딩 퍼즐

1

1	2	3	1	2	3

2

1	2	3		2	3
			1		

3

1		2	3	2	3
			1		

4

1		2	3	2	3
	1				

5

1	1	2	3	2	3

6

1	1	2		3	3
				2	

7

1	1	2		3	3
			2		

8

1	1	2	2	3	3



Tesselation

#constructive, #implementation

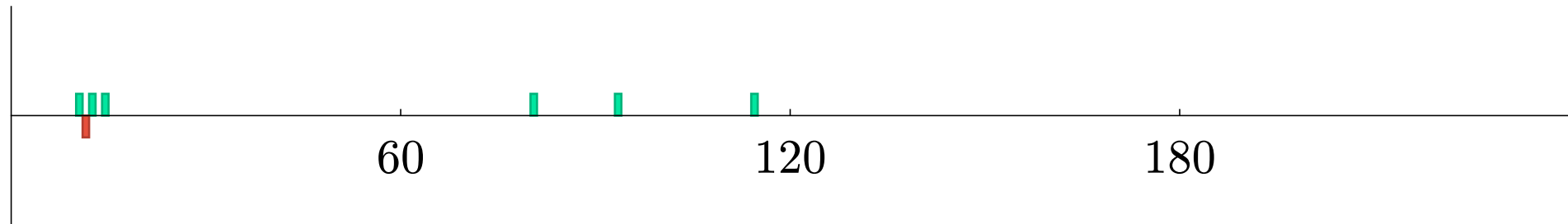
출제진 의도 - Medium

✓ 출제자: wapas

✓ 정답: 6 명 · 정답률 85.714%

✓ 처음 푼 참가자 azberjibiou

10 분



초록: AC · 빨강: AC 이외 판정



- ✓ 모든 $K \times K$ 부분격자의 합을 정확히 S 로 만들기 위해, 행과 열 방향으로 주기가 K 인 격자를 구성해봅시다.
- ✓ 행과 열을 0번부터 세고, 크기가 $K \times K$ 인 기본 패턴 B 를 하나 만듭니다.
- ✓ B 의 칸을 행 우선 순서로 나열했을 때 앞의 S 칸만 1로 두고, 나머지 칸은 0으로 둡니다. 즉, $B_{r,c}$ 는 $rK + c < S$ 이면 1, 아니면 0입니다.
- ✓ $0 \leq S \leq K^2$ 이므로 기본 패턴에는 항상 1을 정확히 S 개 놓을 수 있습니다. $K = 1$ 인 경우에도 같은 정의가 그대로 성립합니다.
- ✓ 이제 기본 패턴을 행과 열 방향으로 반복합니다. 출력할 격자의 각 칸을 $A_{i,j} = B_{i \bmod K, j \bmod K}$ 로 둡니다.
- ✓ 임의의 연속한 K 개 행을 생각해봅시다. 각 행 번호를 K 로 나눈 나머지는 $0, 1, \dots, K - 1$ 이 정확히 한 번씩 등장합니다.
- ✓ 임의의 연속한 K 개 열에서도 같은 성질이 성립합니다.



Tesselation

- ✓ 따라서 임의의 $K \times K$ 부분격자에서 $(i \bmod K, j \bmod K)$ 의 쌍은 기본 패턴의 모든 위치와 정확히 한 번씩 대응합니다.
- ✓ 그러므로 그 부분격자에는 기본 패턴의 모든 값이 정확히 한 번씩 들어가고, 합은 항상 S 가 됩니다.
- ✓ 기본 패턴을 실제로 저장할 필요는 없습니다. 각 칸에서 $(i \% K) * K + j \% K < S$ 인지 확인하여 바로 출력하면 됩니다.
- ✓ 각 칸을 상수 시간에 계산하므로 시간 복잡도는 $\mathcal{O}(NM)$ 입니다.
- ✓ 증명은 복잡한데, 다음 페이지 사진처럼 머릿속에 떠올린 그 배치로 옮기면 정답이 맞습니다.



1	1	1
1	1	0
0	0	0

1	1	1	1	1	1	1	1
1	1	0	1	1	0	1	1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
1	1	0	1	1	0	1	1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1



■. 제곱수^제곱수^제곱수

#constructive, #math

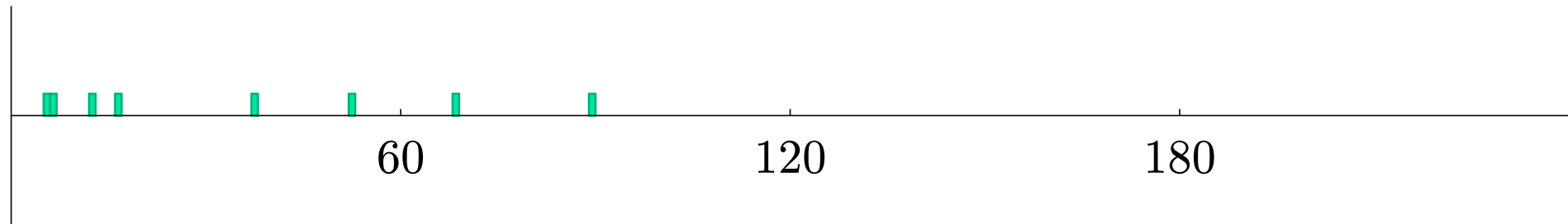
출제진 의도 - Easy

✓ 출제자: hjkim9911, wapas

✓ 정답: 8 명 · 정답률 100.000%

✓ 처음 푼 참가자 dadasO8

5 분



초록: AC · 빨강: AC 이외 판정



■. 제곱수^제곱수^제곱수

✓ 복잡한 사고 없이 아래의 두 가지 방법으로 N 을 제곱수 개의 제곱수의 합으로 무조건 나타낼 수 있습니다.

✓ 1과 N 이 모두 제곱수이므로,

$$N = \overbrace{1 + 1 + \cdots + 1 + 1}^{N\text{개}}$$

로 나타낼 수 있습니다.

✓ 같은 이유로 $N = N$ 으로도 나타낼 수 있습니다.



KPSC × A.N.S.i × CTP × ZeroPage