

제 10회

INHA UNIVERSITY PROGRAMMING CONTEST

인하대학교 프로그래밍 경진대회

주최



인하대학교 컴퓨터공학과
INHA UNIVERSITY



인하대학교
SW 중심 대학 사업단

주관



Challenge the programming

후원

Inha Venture
Startup Academy



STARTLINK

제 10회

INHA UNIVERSITY PROGRAMMING CONTEST

인하대학교 프로그래밍 경진대회

대회 결과 및 해설

주최



인하대학교 컴퓨터공학과



인하대학교
S W 중심 대학 사업단

주관



Challenge the programming

후원

Inha Venture
Startup Academy 

STARTLINK 

문제	의도한 난이도	출제자
A 창고지기	EASY	김현민
B 카드 교환	EASY	안용모
C 문자열 접기	EASY	최준형
D 엉성한 도토리 분류기	NORMAL	황진익
E 편세권	NORMAL	송현성
F 벽록의 가면	HARD	김현민
G 삼각형 찾기	CHALLENGING	김동현
H 부등호 퍼즐	HARD	황진익
I 격자 순회하기	CHALLENGING	정치훈
J 인경호수공원	CHALLENGING	안용모
K 사탕공장	HARD	정치훈
L 간단한 순열 문제	NORMAL	김민겸


A. 참고지키


</ 문자열 >

출제 의도

- 문자열을 다룰 수 있는가?
- 조건문을 활용할 수 있는가?

출제자 김현민
의도한 난이도 EASY
처음 푼 사람 김성준 @ 3분

 제출
98회

 정답
59명 (정답률 60.20%)

A. 참고지기

- 로봇이 있는 칸, 박스가 있는 칸, 박스를 놓아야 하는 칸을 각각 문자열의 세 인덱스 x, y, z 라고 합시다.
- $x < y < z$ 또는 $x > y > z$ 가 참이라면 항상 가능합니다.
 - 로봇은 (x 와 z 사이의 거리) - 1만큼 이동하면 됩니다.
- 그렇지 않으면 항상 불가능합니다.


C. 문자열 접기


</ 완전 탐색 > </ 구현 >

출제 의도

- 완전 탐색 알고리즘을 구현할 수 있는가?

출제자 최준형
의도한 난이도 EASY
처음 푼 사람 최준석 @ 18분

 제출
63회

 정답
48명 (정답률 76.19%)

C. 문자열 접기

- 질문 하나를 푼다고 생각해봅시다.
 - S 의 부분 문자열 $S[l, r]$ 이 적힌 종이를 접어서 얻을 수 있는 최대 점수는?
- 접는 방법은 총 $r - l$ 가지
- 모든 방법으로 전부 한 번씩 접어보고, 그 중 최대 점수를 구하면 됩니다.
- 접는 방법이 $O(N)$ 가지이므로, $O(N)$ 에 점수를 구하면 한 번의 질문당 $O(N^2)$ 에 최대 점수를 구할 수 있습니다.
- 총 시간복잡도 $O(QN^2)$ 에 전체 문제를 풀 수 있습니다.

B. 카드 교환


</ 그리디 >

출제 의도

- 문제의 의도를 파악할 수 있는가?
- 수열에서 가장 큰 K개의 수를 찾을 수 있는가?

출제자 안용모
의도한 난이도 EASY
처음 푼 사람 김건우 @ 14분

 제출
172회

 정답
44명 (정답률 25.58%)

B. 카드 교환

- 플레이어가 한 턴에 할 수 있는 일은 2가지입니다.
- 1. 딜러의 정수 x 가 적힌 카드를 가져와서 A_x 점 얻기
 - 이후 딜러에게 정수 y 가 적힌 카드를 준다 (A_y 점은 영원히 얻을 수 없다)
- 2. 딜러의 더미 카드를 가져와서 점수를 얻지 않기
 - 이후 딜러에게 더미 카드를 돌려준다
 - 굳이 딜러에게 정수가 적힌 카드를 주어 그 카드를 쓰지 못하도록 할 이유는 없다
- 점수를 최대화 하기 위해서는 A_x 가 가장 큰 x 를 가져오고, A_y 가 가장 작은 y 를 주면 됩니다.

B. 카드 교환

- 2번 선택지는 어떨 때 선택하면 될까요?
- 남아있는 가장 큰 A_x 가 음수라면 굳이 점수를 잃지 말고 더미 카드를 이용해 0점을 획득하면 됩니다.
- 위 과정을 M번 반복해 시뮬레이션 합니다.
- $O(N)$ 에 한 번 시뮬레이션 할 수 있고, $O(NM)$ 에 충분히 해결할 수 있습니다.

D. 엉성한 도토리 분류기


</ 이분 탐색 >

출제 의도

- 이분 탐색을 사용하기 위한 조건을 알고 있는가?

출제자	황진익
의도한 난이도	NORMAL
처음 푼 사람	전병현 @ 12분

 제출
239회

 정답
32명 (정답률 13.38%)

D. 엉성한 도토리 분류기

- 뒤에 있는 구멍이 앞에 있는 구멍보다 작다면, 뒤에 있는 구멍에는 도토리가 절대 빠질 수 없습니다.
 - 이런 구멍들을 제외하고 시작합니다.
- 도토리의 크기가 작을 수록 번호가 작은 구멍에, 클 수록 번호가 큰 구멍에 빠질 것이라는 사실을 알 수 있습니다.
 - 즉, 도토리의 크기와 빠질 구멍의 번호의 관계가 단조 증가하므로 이분탐색을 사용할 수 있습니다.
- 한 도토리가 어떤 구멍에 빠질지 $O(\log N)$ 에 구할 수 있고, 총 $O(Q \log N)$ 에 문제를 해결할 수 있습니다.

H. 부등호 퍼즐


</ 위상 정렬 >

출제 의도

- 답이 존재하기 위한 조건을 파악할 수 있는가?
- 위상 정렬 알고리즘을 알고 있는가?

출제자 황진익
의도한 난이도 HARD
처음 푼 사람 팽지원 @ 74분

 제출
27회

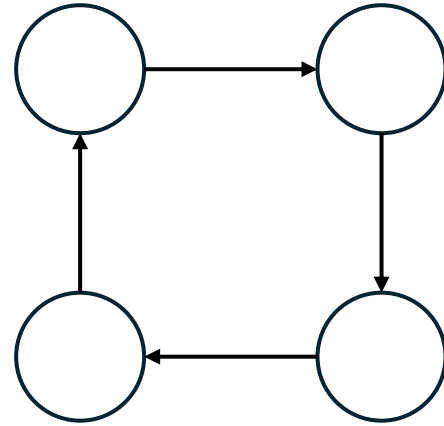
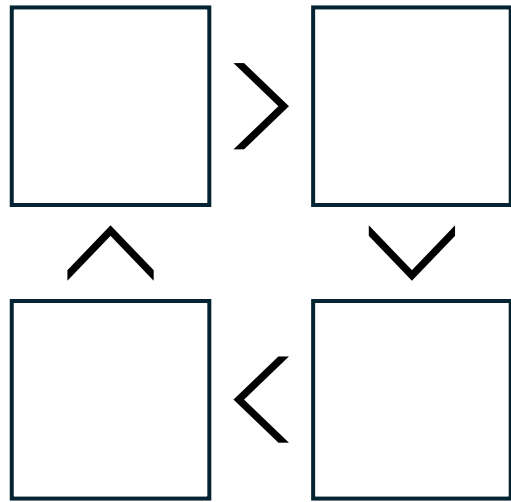
 정답
14명 (정답률 51.85%)

H. 부등호 퍼즐

- 각 칸이 정점, 이웃한 두 칸 사이가 간선인 그래프를 생각해봅시다.
- 부등호의 방향에 따라 간선의 방향이 정해진다고 생각할 수 있습니다.
- 문제에서 항상 풀이가 존재하는 입력이 주어진다고 했는데, 반대로 풀이가 존재하지 않는 경우는 무엇일까요?

H. 부등호 퍼즐

- 다음과 같이 사이클이 존재하는 경우, 문제의 답이 존재하지 않음을 알 수 있습니다.



- 따라서 문제에서 주어지는 그래프는 DAG(Directed Acyclic Graph)입니다.

H. 부등호 퍼즐

- DAG의 정점에 적절한 순서를 부여해 간선의 방향을 거스르지 않게 한다.
 - 위상 정렬의 정의와 같습니다.
- 가능한 위상 정렬 결과를 아무거나 하나 출력하면 됩니다.
- 시간복잡도는 $O(N^2)$ 입니다.


E. 편세권

</ BFS >

출제 의도

- 최단 경로 알고리즘을 활용할 수 있는가?

 제출
118회

 정답
13명 (정답률 11.01%)

출제자 송현성
의도한 난이도 NORMAL
처음 푼 사람 조성찬 @ 70분

E. 편세권

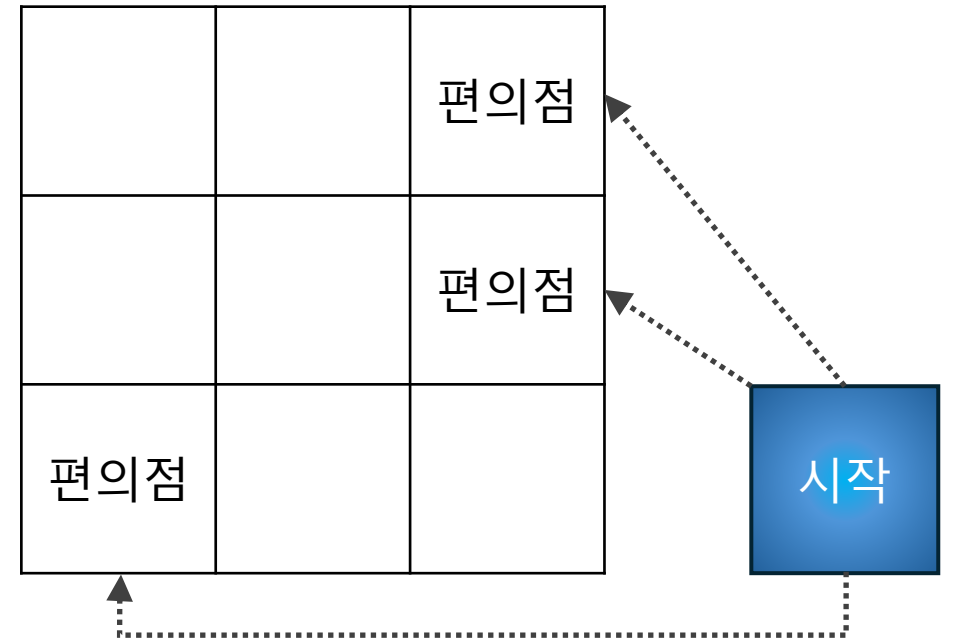
- 각 방에서 가장 가까운 편의점까지의 거리를 모든 방에 대해 계산해야 합니다.
- 방이 1개 뿐이라면 BFS를 사용해 $O(NM)$ 에 계산할 수 있습니다.
 - 가중치가 모두 1인 그래프이기 때문입니다.
- 하지만 이 문제는 방이 여러 개이므로, 모든 방에 대해 각각 BFS를 수행하면 총 시간복잡도 $O(N^2M^2)$ 로 너무 느립니다.

E. 편세권

- 반대로 편의점에서 방까지의 최단거리를 구합시다.
- 모든 편의점에서 시작해, 편의점이 없는 모든 칸까지의 최단거리를 구합니다.
- 시작점이 여러 개인 그래프에서의 최단거리는 어떻게 구할 수 있을까요?

E. 편세권

- 가상의 시작점을 만들고, 모든 편의점까지 가중치가 0인 간선을 만듭시다.
- 가상의 시작점에서 시작하는 BFS로 다중 시작점에서의 최단거리를 구할 수 있습니다.
- 따라서 BFS에 $O(NM)$, 편세권 점수 계산에 $O(R)$ 로, 총 $O(NM)$ 에 문제를 해결할 수 있습니다.





F. 벽록의 가면

</ 기하학 >

출제 의도

- 2차원 기하 문제의 풀이를 코드로 옮길 수 있는가?
- 또는 기하 관련 알고리즘을 알고 있는가?

 제출
31회

 정답
6명 (정답률 19.35%)

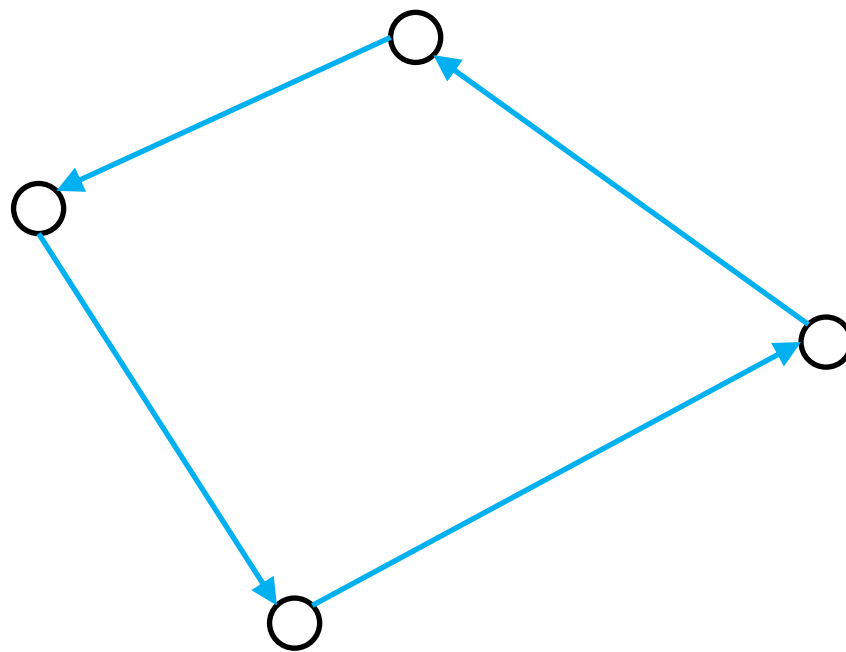
출제자 김현민
의도한 난이도 HARD
처음 푼 사람 임선종 @ 65분

F. 볼록의 가면

- 평면 위의 점 N 개가 주어졌을 때, 볼록한 사각형의 개수를 세어봅시다.
- N 이 충분히 작아 $O(N^4)$ 에 문제를 해결할 수 있으므로, 4개의 점을 고르는 모든 조합에 대해 볼록한 사각형을 만들 수 있는지 확인하면 됩니다.

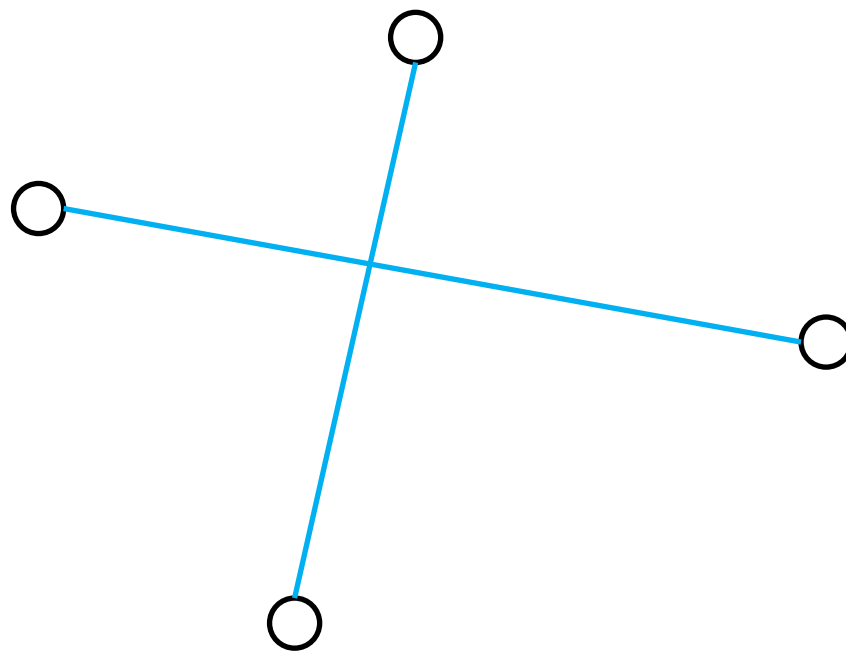
F. 벽록의 가면

- 풀이 1: 네 점을 적절히 배열해 4개의 벡터를 만들고, 이웃하는 두 벡터의 CCW가 모두 같은지 확인한다
- CCW 알고리즘을 사용할 수 있습니다.



F. 벽록의 가면

- 풀이 2 : 네 점을 적절히 배열해 두 선분을 만들고 교차하는지 확인한다
- 고등학교 수준의 수학 지식을 활용하거나, 선분 교차 알고리즘을 사용할 수 있습니다.





L. 간단한 순열 문제

</ 스택 > </ 자료구조 >

출제 의도

- 문제를 단순화 할 수 있는가?
- 문제 해결을 위해 적절한 자료구조를 선택할 수 있는가?

 제출
12회

 정답
5명 (정답률 41.66%)

출제자 김민겸
의도한 난이도 NORMAL
처음 푼 사람 조성찬 @ 197분

L. 간단한 순열 문제

- 식에서 $\max(0, P_{i+1}, P_{i+2}, \dots, P_{j-1})$ 값이 x 라고 가정해봅시다.
- $P_{i+1}, P_{i+2}, \dots, P_{j-1}$ 중 하나는 x 여야 합니다. x 의 위치를 pos 라고 합시다.
 - $P_{pos} = x$
- $\min(P_i, P_j)$ 는 x 보다 커야 합니다.
- 이를 만족하는 i, j 는 최대 1개밖에 존재하지 않습니다.
 - $i < pos \ \&\& \ P[i] > x$ 인 i 중 가장 큰 값, $pos < j \ \&\& \ P[j] > x$ 인 j 중 가장 작은 값

L. 간단한 순열 문제

- 즉, x 보다 큰 값이 x 의 왼쪽과 오른쪽에 1개 이상 존재하는지 확인하면 됩니다.
- 존재한다면 $\max(0, P_{i+1}, P_{i+2}, \dots, P_{j-1}) = x$ 를 만족하는 i, j 가 반드시 1개 있습니다.
- 이를 모든 x 에 대해 구해주면 됩니다.

L. 간단한 순열 문제

- 스택이나 BBST 등의 적절한 자료구조를 이용하면 각 x 에 대해 $O(\log N)$ 이하의 시간에 구할 수 있습니다.
- 마지막으로, $i + 1 = j$ 인 경우 항상 식을 만족하므로 $n - 1$ 을 더합니다.


K. 사탕 공장

</ 연결 리스트 >

출제 의도

- 문제에서 설명한 조작을 모두 충분히 빠르게 처리할 수 있는 자료구조를 알고 있는가?

 제출
9회

 정답
1명 (정답률 11.11%)

출제자 정치훈
의도한 난이도 HARD
처음 푼 사람 이하늘 @ 162분

K. 사탕 공장

- 사탕들을 어떤 자료구조에 저장해야 합니다.
- 이 자료구조에 필요한 연산들은 다음과 같습니다.
 - 자료구조의 맨 앞과 맨 뒤에서의 push, pop
 - 첫 번째 원소부터 R 번째 원소까지의 자료를 splice
 - R을 1씩 증가, 감소
- 이 3가지 연산을 모두 충분히 빠르게 할 수 있는 자료구조는?
 - Front에서의 push / pop이 느린 대부분의 자료구조는 사용할 수 없습니다. (배열, vector 등)
 - 배열 기반의 자료구조는 사용할 수 없습니다. (deque)

K. 사탕 공장

- "링크드 리스트"
- Doubly linked list는 다음과 같은 연산을 모두 상수 시간에 할 수 있습니다.
 - 맨 앞과 맨 뒤에서의 push, pop
 - 맨 앞부터 (이미 저장해 둔) 특정한 pointer까지 자르고 붙이기
 - 위 특정한 pointer를 앞뒤로 1칸씩 움직이기
- 따라서 링크드 리스트를 구현해 모든 조작을 처리하면 됩니다.
 - 참고 : C++ STL list는 splice 연산의 시간복잡도가 $O(N)$ 이므로 사용할 수 없습니다.


I. 격자 순회하기

</ 구현 > </ 애드 혹 >

출제 의도

- 문제를 일반화하여 규칙을 찾을 수 있는가?
- 효율적으로 코드를 작성할 수 있는가?

 제출
3회

 정답
0명 (정답률 N/A)

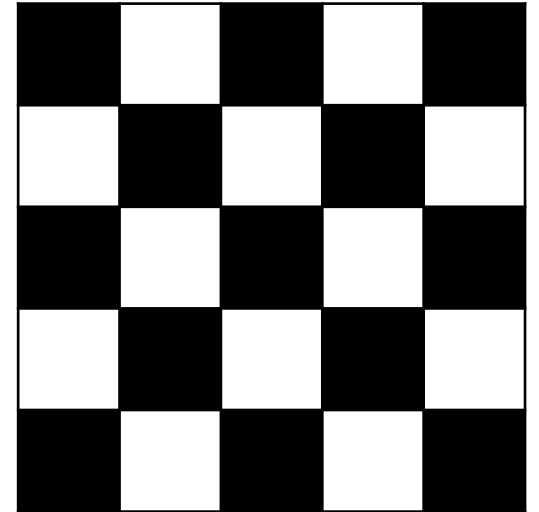
출제자	정치훈
의도한 난이도	CHALLENGING
처음 푼 사람	없음

I. 격자 순회하기

- 알고리즘이 필요하지 않은 문제입니다. 규칙을 찾으려고 노력해봅시다.
- 먼저 불가능한 경우에 대해 생각해봅시다.

I. 격자 순회하기

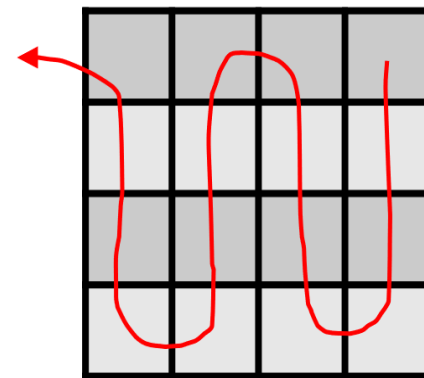
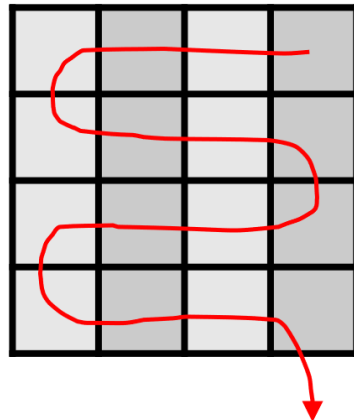
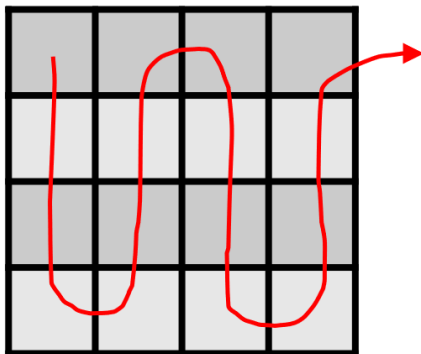
- 총 격자 칸의 개수가 홀수인 경우, 검은색 칸과 흰색 칸의 개수가 다릅니다.
 - 검은색 칸이 하나 더 많습니다.



- 반드시 검은색 칸에서 시작해 검은색 칸에서 끝나야 합니다.
- 따라서 흰색 칸에서 격자 밖으로 나가야 하는 입력이 들어온다면, 조건을 만족하는 경로를 만들 수 없습니다.

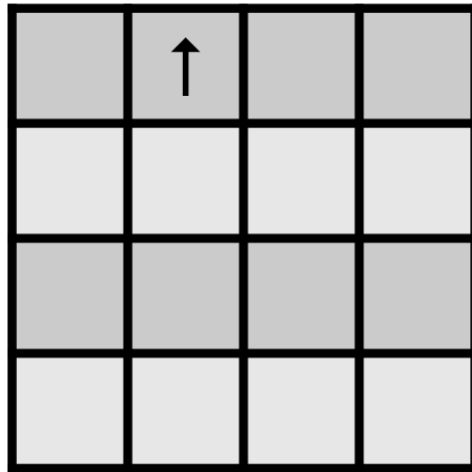
I. 격자 순회하기

- 이동방향이 정해져 있는 칸이 경로의 마지막 칸이 아닌 경우에 대해 생각해봅시다.
- 아래 그림과 같은 S자 경로를 하나 만듭니다.
- 이 경로를 회전하거나 뒤집어 만들 수 있는 8가지 경우 중 하나는 무조건 답입니다.
 - 자세한 증명은 생략합니다.



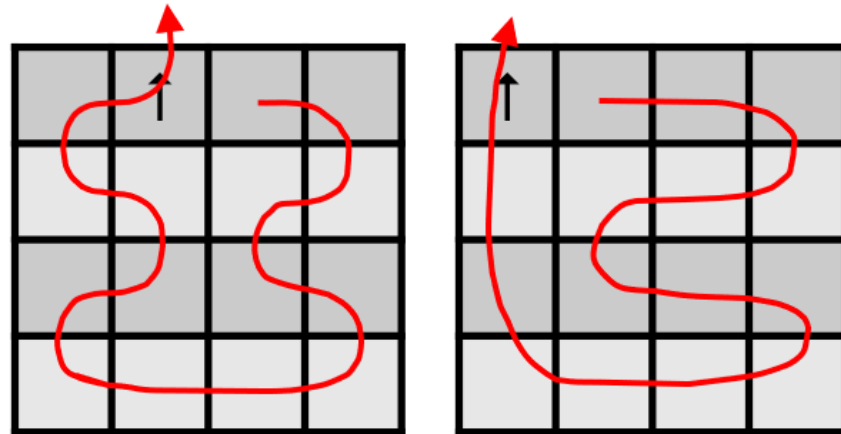
I. 격자 순회하기

- 이동 방향이 정해져 있는 칸이 마지막 칸인 경우에 대해 생각해봅시다.
- 일반적으로, 그 칸이 1번 행에서 위쪽으로 가는 경우라고 가정합시다.
 - 그렇지 않은 경우 격자를 회전해서 그렇게 되도록 만들 수 있습니다.



I. 격자 순회하기

- 1. 높이가 짝수인 경우
- 화살표가 있는 칸 기준으로 좌우를 나누어 가로 방향으로 채우면 됩니다.



I. 격자 순회하기

- 설명한 내용을 최대한 효율적으로 구현하면 됩니다.





J. 인경호수공원

</ 자료구조 >

출제 의도

- 구간 최댓값을 빠르게 구할 수 있는 자료구조를 알고 있는가?

 제출
4회

 정답
0명 (정답률 N/A)

출제자	안용모
의도한 난이도	CHALLENGING
처음 폰 사람	없음

- 원형으로 된 것은 일자로 펴면 간단 해집니다.
- 0번과 N-1번 갈림길 사이의 길을 끊고, 0번 출입구에서 시작해 항상 시계방향으로만 이동할 수 있다고 가정해봅시다.
- 답은 수열 $[a_0 + b_1, a_0 + a_1 + b_2, \dots, a_0 + a_1 + \dots + a_{n-2} + b_{n-1}]$ 중 최댓값입니다.
- 위 수열을 x라고 합시다.

J. 인경호수공원

- 1번 출입구에서 시작했을 때의 답도 같은 방법으로 계산해봅시다.
 - 0번과 1번 갈림길 사이의 길을 끊고, 1번 출입구에서 시작해 시계방향으로만 이동할 수 있다
- 답은 수열 $[a_1 + b_2, a_1 + a_2 + b_3, \dots, a_1 + a_2 + \dots + a_{n-1} + b_0]$ 중 최댓값입니다.
- 위 수열을 y 라고 합시다.

J. 인경호수공원

- 한 출입구에서 시작했을 때의 최댓값을 구하기 위해 N개의 항을 모두 계산하는 건 너무 느립니다.
- 이전에 계산한 값을 재활용할 방법을 생각해봅시다.
- x 를 이용해 y 를 빠르게 구하려면 어떻게 해야 할까요?

J. 인경호수공원

- $x = [a_0 + b_1, a_0 + a_1 + b_2, \dots, a_0 + a_1 + \dots + a_{n-2} + b_{n-1}]$
- $y = [a_1 + b_2, a_1 + a_2 + b_3, \dots, a_1 + a_2 + \dots + a_{n-1} + b_0]$
- x 에서 초항을 제거하고, (y 의 마지막 항) + a_0 을 추가해봅시다.
 - $x' = [a_0 + a_1 + b_2, \dots, a_0 + a_1 + \dots + a_{n-2} + b_{n-1} + a_0]$
- y 의 최댓값은 (x' 의 최댓값) - a_0 와 같습니다.

J. 인경호수공원


- x, y 와 같은 수열을 적절한 자료구조에 저장하고 있으면 됩니다.
- 해당 자료구조는 임의 원소의 추가/삭제 및 자료구조 내의 원소 중 최댓값 구하기를 빠르게 할 수 있어야 합니다.
- BBST 또는 Segment tree등을 사용할 수 있습니다.


G. 삼각형 찾기

</ 누적 합 > </ 세그먼트 트리 >

출제 의도

- 문제를 단순화 할 수 있는가?
- 구간 안의 길이가 2인 증가 수열의 개수를 빠르게 셀 수 있는가?

 제출
15회

 정답
0명 (정답률 N/A)

출제자	김동현
의도한 난이도	CHALLENGING
처음 푼 사람	없음

G. 삼각형 찾기

- 가장 직관적인 방법으로 M개의 선분들 중 3개를 고르는 모든 경우에 대해 세 선분이 교차하는지 확인하는 방법이 있습니다.
- 하지만 이 문제는 $O(N^3)$ 에 해결할 수 없습니다.

G. 삼각형 찾기

- $O(M^2 \log M)$ 풀이
- 모든 선분이 $u < v$ 를 만족하도록 하고, u 가 커지는 순서로 정렬합니다.
- 정렬된 선분들을 순서대로 1번, 2번, ..., M 번 선분이라고 합시다.
- 이제 k 번 선분을 반드시 사용해 삼각형을 만드는 경우의 수를 세어봅시다.
- 나머지 두 개의 선분을 $k+1$ 번 ~ M 번 중에서 고르면 중복 없이 셀 수 있습니다.
 - 이 두 선분을 i, j ($i < j$) 번 이라고 합시다. 따라서 $k < i < j$ 를 만족하는 선분의 수를 셉니다.

G. 삼각형 찾기

- k번 선분과 i번, j번 선분이 교차할 조건 : $v_k < v_i, v_k < v_j$
- i번 선분과 j번 선분이 교차할 조건 : $v_i < v_j$
- 따라서 k+1번 ~ M번 선분에 대해 v가 v_k 보다 큰 선분을 골라내고, 그 중 $v_i < v_j$ 를 만족하는 (i, j) 쌍이 몇 개인지 세면 됩니다.
- 즉, 수열이 주어졌을 때 길이가 2인 증가하는 부분 수열의 개수를 세면 됩니다.

G. 삼각형 찾기

- 수열 A 가 주어졌을 때 길이가 2인 증가하는 부분수열의 개수
 - 세그먼트 트리로 해결할 수 있는 잘 알려진 문제입니다.
- 수열 A 를 차례로 순회하면서,
 - $[1, A_{i-1}]$ 구간에 해당하는 원소의 개수를 구해서 더한다
 - A_i 를 세그먼트 트리에 추가한다
- 시간복잡도 $O(M \log M)$ 에 해결할 수 있습니다.

G. 삼각형 찾기

- 지금까지의 풀이는 특정 k 번 선분이 반드시 삼각형에 포함된다고 가정한 것입니다.
- 이를 모든 선분 M 개에 대해 반복하면 됩니다.
- 총 시간복잡도는 $O(M^2 \log M)$ 입니다.
- 단, M 이 크기 때문에 이 풀이로 통과하기 위해서는 상수 최적화가 필요합니다.
- Ex) 세그먼트 트리 대신 Fenwick Tree 사용하기

G. 삼각형 찾기

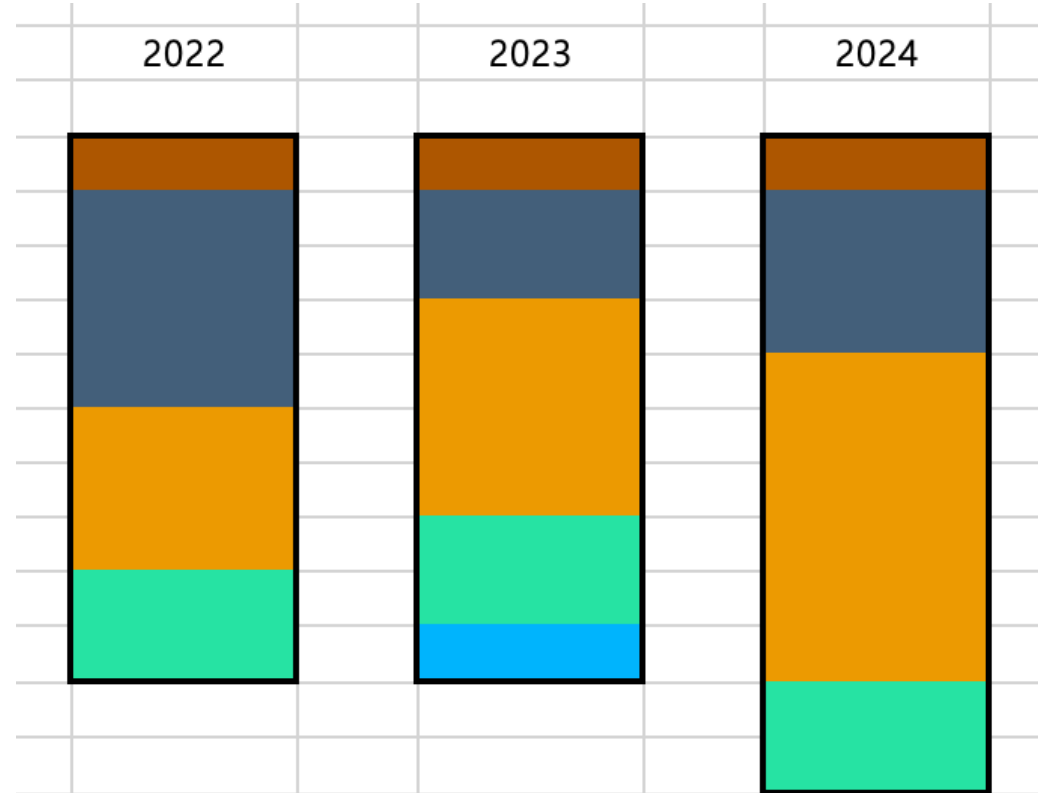
- $O(M^2)$ 풀이
- 앞의 풀이에 누적합을 이용하면 로그를 하나 뺄 수 있습니다.
- 똑같이 k 번 선분을 무조건 사용하고, $k < i < j$ 를 만족하는 세 선분이 삼각형을 이루는 경우를 세어 봅시다.
- k 를 고정 시켜두고 j 를 1씩 증가 시켜주면서 답을 구해 봅시다.
- 1~ i 번 선분 중 j 번과 교차하는 선분의 개수를 누적 합을 이용해 저장해 둡시다.

G. 삼각형 찾기

- 그러면 우선 i 번 선분과 j 번 선분이 교차하는 경우의 수를 구할 수 있습니다.
- 여기에서 i, j 번 선분이 k 번 선분과 교차하지 않는 경우만 빼 주면 됩니다.
- j 를 1씩 증가시켜주는 과정에서, j 번 선분이 k 번 선분과 교차하는지 여부를 바로 알 수 있습니다. 교차하지 않는 선분의 개수를 누적해서 구해 두면 됩니다.
- 시간 복잡도는 이중 반복문 만을 사용하므로, $O(M^2)$ 입니다.

Appendix 1 </ 2024 IUPC >

작년 / 재작년과 비교



풀린 문제 수	8	9	9
참가자 수	85	102	77

Appendix 2

</ Open Contest 챌린지 >

문자열 접기 (Hard)

- $N \leq 5,000, Q \leq 200,000$

벽록의 가면 (Hard)

- $N \leq 500$

편세권 (Hard)

- $N, M, R + C \leq 500,000$



오픈 컨테스트에서 풀어보세요

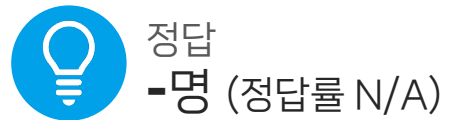
C2. 문자열 접기 (Hard)

</ DP >

출제 의도

- 오픈 컨테스트 전용

	출제자	최준형
의도한 난이도	-	-
처음 푼 사람	-	-



C2. 문자열 접기 (Hard)

- $DP[i][j] := S[l..r]$ 에서 얻을 수 있는 최대 점수
- $H[i][j] := S[l..r]$ 을 정확하게 절반으로 접었을 때 얻을 수 있는 점수
- 다음과 같은 점화식을 얻을 수 있습니다.
- $DP[i][j] = \max(DP[i-1][j], DP[i][j-1], H[i][j])$
- $H[i][j] = H[i-1][j+1] + (S[i] == S[j] ? 1 : 0)$
- 총 $O(N^2)$ DP로 해결할 수 있습니다.

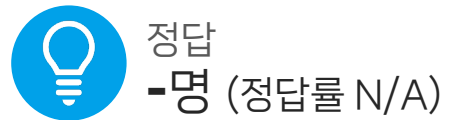
F2. 벽록의 가면 (Hard)

</ 기하학 >

출제 의도

- 오픈 컨테스트 전용

출제자	김현민
의도한 난이도	-
처음 푼 사람	-

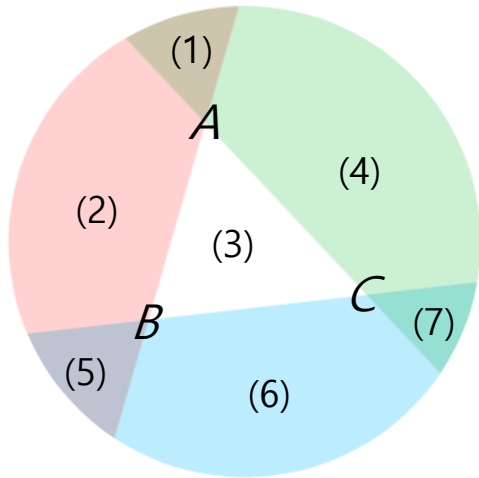


F2. 벽록의 가면 (Hard)

- 네 점을 골랐을 때, 세 점을 꼭짓점으로 하는 삼각형 안에 한 점이 있는 형태라면 볼록한 사각형을 만들 수 없습니다.
- 만들 수 있는 모든 삼각형에 대해 삼각형 안에 있는 점의 개수를 세어 모두 더합니다.
 - 네 점을 골랐을 때 볼록한 사각형을 만들 수 없는 모든 경우의 수와 같습니다.
- 각 점에 대해, 다른 모든 점을 각도 순으로 정렬해 순서를 매깁시다.
- 그러면 점 A, B, C 가 정해져 있을 때, 벡터 AB 와 벡터 AC 사이에 있는 점의 개수를 상수 시간에 구할 수 있습니다.

F2. 벽록의 가면 (Hard)

- 다음과 같이 삼각형 안에 있는 점의 개수를 상수 시간에 구할 수 있습니다.



AB, AC 사이 = (3) + (6), 반대 (1)
BA, BC 사이 = (3) + (4), 반대 (5)
CA, CB 사이 = (3) + (2), 반대 (7)

모두 더하고 $n-3$ 을 빼면 (3)의 2배

- 시간 복잡도는 만들 수 있는 삼각형의 개수와 동일한 $O(N^3)$ 입니다.
- 별개로 $O(N^4 / 64)$ 또는 $O(N^2 \log N)$ 에 해결할 수도 있습니다.

E2. 편세권 (Hard)

</ 세그먼트 트리 >

출제 의도

- 오픈 컨테스트 전용



제출
-회



정답
-명 (정답률 N/A%)

출제자	송현성
의도한 난이도	-
처음 푼 사람	-

E2. 편세권 (Hard)

- Taxicab geometry에서 Nearest neighbor 문제
- 어떤 방의 위치가 (x_i, y_i) , 편의점의 위치가 (x_j, y_j) 일 때, $x_i \geq y_j$ & $y_i \geq y_j$ 가 참이라고 가정합니다.
- 방과 편의점 사이의 거리는 $(x_i - y_j) + (y_i - y_j) = (x_i + y_i) - (x_j + y_j)$ 입니다.
- 즉, 모든 편의점에 대해 $x_i + y_i$ 에서 $x_j + y_j$ 의 최댓값을 빼면 가장 가까운 편의점까지의 거리를 구할 수 있습니다.

E2. 편세권 (Hard)

- 편의점의 x 좌표가 더 작거나 같아야 한다는 조건은 x 좌표에 대해 스위핑하여 만족할 수 있습니다.
- 세그먼트 트리의 y_j 번째 인덱스에 $x_j + y_j$ 의 최댓값을 저장해 놓읍시다.
- 편의점의 y 좌표가 더 작거나 같아야 한다는 조건은 세그먼트 트리에서 $[1..x_i]$ 범위에서의 최댓값을 구해 만족할 수 있습니다.

E2. 편세권 (Hard)

- 우리는 편의점이 $x_i \geq x_j \ \&\& \ y_i \geq y_j$ 의 한 방향에 존재하는 경우에 대해서만 문제를 해결했습니다.
- 이를 네 방향 모두에 대해 같은 방식으로 풀면 됩니다.
- 다시 말해, 지도 전체를 90도씩 회전하며 같은 문제를 4번 반복해 풀면 됩니다.

제 10회

INHA UNIVERSITY PROGRAMMING CONTEST

인하대학교 프로그래밍 경진대회

종료

감사합니다!

주최



인하대학교 컴퓨터공학과



인하대학교
SW 중심 대학 사업단

주관



Challenge the programming

후원

Inha Venture
Startup Academy 

STARTLINK 