



2019 숭고한 연합

Algorithm Contest

- 고급반 -

대회가 시작되기 전까지 절대 표지를 넘기지 마세요

< Hosted by >



< Sponsored by >



SAMSUNG
SOFTWARE
MEMBERSHIP

고려대학교 SW 중심대학사업단



대회는 총 7문제이며, 구성은 다음과 같습니다. 총 문제지가 표지를 제외하고 15쪽인지 확인하시길 바랍니다.

- A 이진수 변환
- B 백도어
- C 깃발춤
- D 일하는 세포
- E 트리의 외심
- F 시간 끌기
- G 가장 높고 넓은 성

- 대회 시간은 4시간입니다.
- 대회가 진행되는 동안 대회 사이트를 제외한 웹에 접속/열람/검색 등을 할 수 없습니다. 예외로, 공지를 통해 명시된 레퍼런스 사이트는 열람할 수 있습니다.
- 이전에 저장해두었던 자료/코드 등을 열람 및 사용할 수 없습니다. 예외로, 주최진 측에서 제공하는 ‘템플릿 코드 문서’는 열람할 수 있습니다.
- 그 외에 운영진이 판단하기에 악의적인 제출을 진행하거나 부정 행위를 저지를 경우 경고 없이 대회 참가 자격이 박탈될 수 있습니다.
- 컴파일 환경은 BOJ의 기본 환경을 따릅니다.
- 대회 중 ‘질문’ 탭을 통해 주최진에게 물을 수 있습니다. 예/아니오로 답이 나오는 질문을 해주시기 바랍니다.
- 문제지는 참가자의 편의를 위해 제공되었을 뿐, 웹사이트의 내용이 공식 문제입니다. 인쇄본과 사이트의 내용 중 상이한 부분이 있을 경우, 사이트에 기재된 설명을 참고하시길 바랍니다.

문제 A. 이진수 변환

시간 제한: 1 초

메모리 제한: 512 MB

당신에게 자연수 x_0 와 N 이 주어졌다. 지금부터 당신은 이 자연수 x_0 를 N 번의 ‘변환’을 통해 0으로 바꿀 것이다. 변환이란, 양의 정수를 이진법으로 표기했을 때, 1개 이상의 1을 0으로 바꾸는 작업이다. 예를 들어 9를 이진법으로 나타내면 $1001_{(2)}$ 인데, 9는 $0(0000_{(2)})$, $1(0001_{(2)})$, 또는 $8(1000_{(2)})$ 로 변환될 수 있다 (바뀐 자릿수는 밑줄로 표기되었다). 여러분의 목표는 x_i 를 변환하여 x_{i+1} 를 만드는 과정을 반복해, x_N 을 0으로 만드는 것이다.

위 조건을 만족하는 수열 $X = [x_0, x_1, x_2, \dots, x_N]$ 는 존재하지 않을 수도 있지만, 여러 개가 존재할 수도 있다. 만약 존재한다면, 각 수열별로 인접한 원소들의 차들의 집합 $D(X) = \{x_0 - x_1, x_1 - x_2, \dots, x_{N-1} - x_N\}$ 를 정의하자. 이 집합의 원소들의 최대값과 최소값의 차이를 최소화하도록, 수열 X 를 만들려고 한다. 즉, 가능한 모든 수열 X_i 중 ($D(X_i)$ 에 속한 원소의 최댓값 - $D(X_i)$ 에 속한 원소의 최솟값)이 최소가 되는 X_i 를 찾고자 한다.

이상해보일 수 있는 문제지만, 당신은 대답해야 한다. 과연 1초 안에 답할 수 있을까?

입력 형식

첫 번째 줄에 변환할 자연수와 변환 횟수를 의미하는 두 자연수 x_0 과 N 이 공백으로 구분되어 주어진다. ($1 \leq x_0 \leq 10^{16}, 1 \leq N \leq 50$)

출력 형식

만약 조건을 만족하는 수열이 존재하지 않으면 첫 번째 줄에 -1을 출력한다.

조건을 만족하는 수열이 존재한다면, 수열의 원소를 의미하는 N 개의 정수 x_1, x_2, \dots, x_N 을 공백으로 구분하여 출력한다.

조건을 만족하는 수열이 여러 개 존재한다면, 아무 것이나 출력해도 좋다.

예제

표준 입력	표준 출력
23 2	16 0
48 5	-1

설명

첫 번째 예제에서 $x_0 = 23 = 10111_{(2)}$, $x_1 = 16 = 10000_{(2)}$, $x_2 = 0 = 00000_{(2)}$ 로 수열을 구성하면 인접한 원소의 차의 최대값과 최소값의 차이는 $16 - 7 = 9$ 가 되며, 이보다 작게 만들 수 있는 방법은 존재하지 않는다. 또 다른 답으로는 $x_0 = 23$, $x_1 = 7$, $x_2 = 0$ 이 있다.

두 번째 예제에서 주어진 조건을 만족하는 x_1, x_2, x_3, x_4, x_5 는 존재하지 않는다.

문제 B. 백도어

시간 제한: 2 초

메모리 제한: 512 MB

유섭이는 무척이나 게으르다. 오늘도 할 일을 모두 미뤄둔 채 열심히 물을 하던 유섭이는 오늘까지 문제를 내야 한다는 사실을 깨달았다. 그러나 게임은 시작되었고 지는 걸 무척이나 싫어하는 유섭이는 어쩔 수 없이 백도어를 해 게임을 최대한 빠르게 끝내기로 결심하였다.

최대한 빨리 게임을 끝내고 문제를 출제해야 하기 때문에 유섭이는 최대한 빨리 넥서스가 있는 곳으로 달려가려고 한다. 유섭이의 챔피언은 총 N 개의 분기점에 위치할 수 있다. 0번째 분기점은 현재 유섭이의 챔피언이 있는 곳을, $N - 1$ 번째 분기점은 상대편 넥서스를 의미하며 나머지 $1, 2, \dots, N - 2$ 번째 분기점은 중간 거점들이다. 그러나 유섭이의 챔피언이 모든 분기점을 지나칠 수 있는 것은 아니다. 백도어의 핵심은 안 들키고 살금살금 가는 것이기 때문에 적 챔피언 혹은 적 와드(시야를 밝혀주는 토탈), 미니언, 포탑 등 상대의 시야에 걸리는 곳은 지나칠 수 없다.

입력으로 각 분기점을 지나칠 수 있는지에 대한 여부와 각 분기점에서 다른 분기점으로 가는데 걸리는 시간이 주어졌을 때, 유섭이가 현재 위치에서 넥서스까지 갈 수 있는 최소 시간을 구하여라.

입력 형식

첫 번째 줄에 분기점의 수와 분기점들을 잇는 길의 수를 의미하는 두 자연수 N 과 M 이 공백으로 구분되어 주어진다. ($1 \leq N \leq 100,000$, $1 \leq M \leq 300,000$)

두 번째 줄에 각 분기점이 적의 시야에 보이는지를 의미하는 N 개의 정수 a_0, a_1, \dots, a_{N-1} 가 공백으로 구분되어 주어진다. a_i 가 0이면 i 번째 분기점이 상대의 시야에 보이지 않는다는 뜻이며, 1이면 보인다는 뜻이다. 추가적으로 $a_0 = 0$, $a_{N-1} = 1$ 이다. $N - 1$ 번째 분기점은 상대 넥서스이기 때문에 어쩔 수 없이 상대의 시야에 보이게 되며, 또 유일하게 상대 시야에 보이면서 갈 수 있는 곳이다.

다음 M 개의 줄에 걸쳐 세 정수 a, b, t 가 공백으로 구분되어 주어진다. ($0 \leq a, b < N$, $a \neq b$, $1 \leq t \leq 100,000$) 이는 a 번째 분기점과 b 번째 분기점 사이를 지나는데 t 만큼의 시간이 걸리는 것을 의미한다. 연결은 양방향이며, 한 분기점에서 다른 분기점으로 가는 간선은 최대 1개 존재한다.

출력 형식

첫 번째 줄에 유섭이의 챔피언이 상대 넥서스까지 안 들키고 가는데 걸리는 최소 시간을 출력하여라. 만약 상대 넥서스까지 갈 수 없으면 -1을 출력하여라.

예제

표준 입력	표준 출력
5 7 0 0 0 1 1 0 1 7 0 2 2 1 2 4 1 3 3 1 4 6 2 3 2 3 4 1	12
5 7 0 1 0 1 1 0 1 7 0 2 2 1 2 4 1 3 3 1 4 6 2 3 2 3 4 1	-1

설명

첫 번째 예제에서 위 그래프의 최단거리는 $0 \rightarrow 2 \rightarrow 3 \rightarrow 4$ 를 지나는 시간인 $5(2+2+1)$ 이지만, 3번 분기점이 상대의 시야에 있기 때문에 $0 \rightarrow 2 \rightarrow 1 \rightarrow 4$ 를 지나는 시간인 $12(2+4+6)$ 이 최소 시간이 된다.

문제 C. 깃발춤

시간 제한: 2 초

메모리 제한: 512 MB

매일 코딩만 하던 상현이는 두뇌가 너무 코드에만 범벅이 되고 있는 것 같아 두뇌를 다른 방면으로 몰입시킬 수 있는 깃발춤 공연을 보러 가기로 했다. 깃발춤 공연은 N 명의 공연자들이 일렬로 서서 깃발을 박력 있게 흔들며 진행된다. 깃발을 들고 있는 공연자들은 각각 카리스마의 정도 c_i 를 지니고 있어, 몇몇 공연자들은 보다 절도 있게 깃발을 흔든다.

상현이는 깃발춤을 보다가, 문득 깃발춤 공연 중 몇몇 연속된 공연자들이 깃발을 교대로 흔드는 것을 목격했다. 상현이는 이를 ‘교대 깃발춤’이라 명명하였다. 교대 깃발춤은 L 번째 공연자부터 R 번째 공연자까지 깃발을 각자 왼쪽 또는 오른쪽으로 흔드는 동작인데, L 번째 공연자를 포함하여 L 번째 공연자와의 거리가 짝수인 공연자는 깃발을 왼쪽으로 흔들고, 거리가 홀수인 공연자는 오른쪽으로 흔든 뒤, 다시 깃발을 몸 쪽으로 원위치시킨다. 여기서 x 번째 공연자와 y 번째 공연자 사이의 거리는 $|x - y|$ 로 표현된다.

문제 해결에서 벗어날 수 없었던 상현이는 교대 깃발춤에서 왼쪽으로 깃발을 흔든 공연자들의 카리스마의 합과 오른쪽으로 깃발을 흔든 공연자들의 카리스마의 합의 차이의 절댓값을 교대 깃발춤의 균일도라고 부르기로 하였다. 교대 깃발춤의 균일도가 큰 값을 가지면, 한쪽이 다른 쪽보다 압도적으로 카리스마가 느껴진다는 뜻이기에 비대칭적으로 보일 수 있다. 상현이는 교대 깃발춤의 균일도가 중요한 의미를 지닌다고 생각한다. 또 깃발을 흔드는 공연자들은 공연의 열기와 순간순간의 실수에 휩쓸리기 때문에, 공연자들의 카리스마가 증가하거나 감소할 수 있다. 이런 모든 상황을 고려하며 상현이는 매 교대 깃발춤의 균일도를 구하고 싶어졌다. 상현이를 도와주자!

입력 형식

첫 번째 줄에는 깃발춤을 진행하는 공연자의 명수인 자연수 N 과 상황 변화의 개수인 자연수 Q 가 공백으로 구분되어 주어진다. ($1 \leq N \leq 300,000, 1 \leq Q \leq 300,000$)

두 번째 줄에는 정수 c_1, c_2, \dots, c_N 이 공백으로 구분되어 주어지며, c_i 는 i 번째 공연자의 카리스마를 의미한다. ($-100,000 \leq c_i \leq 100,000$)

세 번째 줄부터 Q 개의 줄에 걸쳐 다음 형식 중 하나로 세 정수가 공백으로 구분되어 주어진다.

- 1 $L R$: L 번째 공연자부터 R 번째 공연자까지 구성된 교대 깃발춤이 시연된다. ($1 \leq L \leq R \leq N$)
- 2 $L x$: L 번째 공연자의 카리스마가 정수 x 만큼 증가한다. ($1 \leq L \leq N, -100,000 \leq x \leq 100,000$)

첫 번째 종류(‘1 $L R$ ’ 꼴)의 쿼리는 한 번 이상 주어짐이 보장된다.

출력 형식

첫 번째 종류의 쿼리가 입력될 때마다 매 줄에 해당하는 교대 깃발춤의 균일도를 출력한다.

예제

표준 입력	표준 출력
6 3	2
3 1 4 1 5 9	9
1 2 4	
2 3 10	
1 3 6	

설명

첫 번째 쿼리에서 교대 깃발춤을 추는 공연자들의 카리스마는 순서대로 1, 4, 1이다. 균일도는 $|(1 + 1) - (4)| = 2$ 이다.

두 번째 쿼리로 인해 3번째 공연자의 카리스마가 10 증가하여, 전체 카리스마가 3, 1, 14, 1, 5, 9가 된다.

세 번째 쿼리에서 교대 깃발춤을 추는 공연자들의 카리스마는 순서대로 14, 1, 5, 9이다. 균일도는 $|(14 + 5) - (1 + 9)| = 9$ 이다.

문제 D. 일하는 세포

시간 제한: 1 초

메모리 제한: 512 MB

사람의 세포 수, 약 37조 개. 세포들은 몸 속에서 오늘도 열심히 일하고 있다. 그중에서도 우리의 적혈구는 혈관을 타고 돌아다니며 산소와 영양소를 운반해주는 중요한 역할을 맡고있다.

적혈구는 심장이나 폐 같은 거점들을 돌아다니면서 산소와 영양분을 운반한다. 몸 속에는 총 N 개의 거점이 있고, 몇몇 거점은 통로를 통해 서로 이어져 있다. 거점 사이의 통로를 통과하는데는 1초가 걸린다. 하지만 혈관의 곳곳에는 판막이나 공사중인 부분들이 있기 때문에 매 초 거점 사이의 연결관계가 바뀐다. 그럼에도 불구하고 몸의 곳곳이 산소와 영양분을 필요로 하므로 적혈구는 가만히 있을 수 없으며, 매 초 통로를 무조건 하나 타야 한다. 일부 통로는 출발 거점과 도착 거점이 같을 수도 있다. 일부 거점의 특정 순간에는 나가는 통로가 없을 수도 있는데, 이 때는 도착한지 1초 후에 파괴되어 몸과 다시 하나가 된다. 잔혹하지만 우리의 몸은 이렇게 돌아간다.

우리의 적혈구는 매 순간 변하는 몸속 혈관 지도에 길을 헤매지만 그래도 최선을 다해서 하루하루 열심히 일을 하고 있다. 옆에 있던 백혈구가 길을 헤매는 적혈구를 보고 도와주고 싶다는 생각을 했다.

수십 시간의 유주 경험을 통해 백혈구는 몸속 혈관 지도가 T 초를 주기로 반복된다는 것을 알게 되었다. 이 사실을 정리해서 적혈구가 거점 A 에서 출발하여 정확히 D 초 후 거점 B 에 도달하게 되는 경우의 수를 모든 거점의 순서쌍에 대해 구해주고자 하지만 너무나도 단세포이기 때문에 머리가 나빠서 계산을 하지 못했다. 한 경로는, D 초 동안 통과한 통로의 순열로 정의된다. 백혈구를 도와서 적혈구가 D 초 동안 한 거점에서 다른 거점까지 움직일 수 있는 경우의 수를 구해주자!

입력 형식

첫 번째 줄에는 백혈구가 알아낸 혈관 지도들의 주기인 자연수 T 와 거점의 개수인 자연수 N , 적혈구가 움직이는 시간인 정수 D 가 공백으로 구분되어 주어진다. ($1 \leq T \leq 100$, $2 \leq N \leq 20$, $0 \leq D \leq 10^9$)

그 뒤 거점 사이의 연결 관계를 나타내는 혈관 지도 T 개가 순서대로 1번부터 T 번까지 주어지는데, 혈관 지도가 주어지는 형식은 다음과 같다.

- 첫 번째 줄에는 거점 사이를 잇는 혈관의 개수인 자연수 M_i 가 주어진다. ($0 \leq M_i \leq N^2$)
- 그 뒤 M_i 개의 줄에 걸쳐 세 자연수 a, b, c 가 공백으로 구분되어 주어진다. 이는 거점 a 에서 거점 b 로 가는 서로 다른 단방향 통로가 c 개 있음을 의미한다. ($1 \leq a, b \leq N$, $1 \leq c \leq 1000$)

매 혈관 지도에 중복된 연결 관계는 주어지지 않으며, i 초에서 $(i+1)$ 초 동안 이동할 때는 $(i \% T + 1)$ 번 혈관 지도가 적용된다. $i \% T$ 는 i 를 T 로 나눈 나머지를 의미한다.

출력 형식

출력은 N 개의 줄로 구성되며, i 번째 줄에는 N 개의 정수 $x_{i1}, x_{i2}, \dots, x_{iN}$ 를 공백으로 구분하여 출력해야 한다. x_{ij} 는 0초 때 거점 i 에서 출발하여 정확히 D 초 때 거점 j 에 위치하게 되는 경로의 수를 1,000,000,007로 나눈 나머지도다.

예제

표준 입력	표준 출력
1 2 4 2 1 1 2 2 2 3	16 0 0 81
3 4 5 4 1 2 1 2 3 1 3 4 1 4 1 1 4 1 1 2 2 3 1 3 4 1 4 2 1 5 1 2 1 2 4 1 4 3 1 3 1 1 1 4 1	0 0 1 0 0 1 0 0 2 0 0 0 4 0 0 2
1 8 3 12 1 2 1 1 8 1 2 3 1 2 8 1 3 4 1 3 7 1 3 8 1 4 5 1 4 7 1 5 6 1 6 7 1 7 8 1	0 0 0 1 0 0 1 1 0 0 0 0 1 0 1 1 0 0 0 0 0 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0

표준 입력	표준 출력
1 8 100000000	261245548 769313318 167840464
24	862450688 445583828 270525651
1 2 1	828293276 976953408
1 8 1	769313318 542054741 65223362
2 1 1	957807341 63263610 545566670
2 3 1	134857214 863984679
2 8 1	167840464 65223362 959076197
3 2 1	916983285 988077461 199284746
3 4 1	461375786 371787307
3 7 1	862450688 957807341 916983285
3 8 1	119640157 267995930 978327505
4 3 1	847171719 483910227
4 5 1	445583828 63263610 988077461
4 7 1	267995930 394594439 718634258
5 4 1	715295769 69712722
5 6 1	270525651 545566670 199284746
6 5 1	978327505 718634258 131562703
6 7 1	197248645 728310434
7 3 1	828293276 134857214 461375786
7 4 1	847171719 715295769 197248645
7 6 1	322189612 142912983
7 8 1	976953408 863984679 371787307
8 1 1	483910227 69712722 728310434
8 2 1	142912983 162707920
8 3 1	
8 7 1	

설명

네 번째 예제의 경우 적혈구가 3.17년 동안 몸 속을 돌아다니는 경우이다. 표의 너비가 좁아 8개의 숫자들이 한 줄에 있어야 하나 3개 / 3개 / 2개로 나뉘어 표시되었다.

문제 E. 트리의 외심

시간 제한: 2 초

메모리 제한: 512 MB

알고리즘에 푼 빠진 동판이가 트리에 심취한 나머지 트리에서 외심을 정의하려 한다. 트리란, 모든 정점이 연결되어 있으면서 사이클이 존재하지 않는 그래프이다. 하지만 동판이는 트리에서 외심을 정의하기 위해서는 “트리에서 두 정점 사이의 거리”도 정의해야 한다는 사실을 깨달았다!

트리에서 두 정점 사이의 거리는 한 정점에서 다른 정점으로 가기 위해 거쳐야 하는 최소한의 간선의 개수로 정의된다. 이 때 트리의 세 정점에 대해, 트리의 외심은 세 정점으로부터 거리가 같으면서, 그 거리를 최소로 하는 정점이 존재한다면 해당 정점으로 정의된다. 수학적으로 트리의 세 정점에 대해 외심이 존재한다면, 유일하다는 것을 보일 수 있다.

자명하게도, 외심을 정의하는 3개의 정점이 달라지면 같은 트리라 해도 외심이 달라진다. 동판이는 다양한 외심들을 찾아보고 싶지만 코딩에 귀찮음을 겪고 있다.....동판이를 위해 여러분들이 대신 코드를 짜주도록 하자.

입력 형식

첫 번째 줄에 정점의 개수 N 이 주어진다. ($1 \leq N \leq 100,000$) 이 트리는 1번 정점, 2번 정점, ..., N 번 정점으로 구성된다.

두 번째 줄부터 N 번째 줄까지, 트리의 간선 정보를 의미하는 두 자연수 X, Y 가 공백으로 구분되어 주어진다. 이는 X 번 정점과 Y 번 정점이 연결되어있음을 의미한다. ($1 \leq X, Y \leq N, X \neq Y$)

주어지는 연결관계는 트리를 구성한다.

$N + 1$ 번째 줄에는 쿼리의 개수 Q 가 주어진다. ($1 \leq Q \leq 100,000$)

다음 Q 개의 줄에 걸쳐, 외심을 정의하기 위한 세 개의 정점 번호를 뜻하는 세 자연수 A, B, C 가 공백으로 구분되어 주어진다. ($1 \leq A, B, C \leq N$)

출력 형식

Q 개의 줄에 걸쳐 각 쿼리마다 입력으로 주어진 세 정점에 대해 트리의 외심이 존재하면 외심의 정점 번호를, 존재하지 않으면 -1을 출력한다.

예제

표준 입력	표준 출력
4 1 2 1 3 1 4 2 2 3 4 1 2 3	1 -1
2 1 2 2 1 1 2 2 2 2	-1 2
6 1 2 2 3 2 4 3 5 5 6 1 1 4 6	3

문제 F. 시간 끌기

시간 제한: 1 초

메모리 제한: 512 MB

영희와 철수는 공부하기에 앞서 게임을 하고 있다. 게임은 $N \times M$ 크기의 게임판에서 진행되는데 몇몇 칸은 X 표시가 있다.

게임은 두 명이 번갈아 가며 진행되고 다음의 규칙을 따른다.

1. 아직 선택되지 않은 행 또는 열을 하나 선택하되, 기존에 선택되었던 행과 열과의 교차점에 X가 있어서는 안 된다.
2. 선택한 후, 상대의 차례가 된다.
3. 만약 조건을 만족하는 행 또는 열이 없다면 (또는 이미 모든 행 또는 열을 선택했다면) 그 사람은 지고 게임은 끝난다.

예를 들어 다음의 게임판을 보자.

X		X	
			X
	X		X
		X	

지금까지 두 번째 행과 네 번째 행, 두 번째 열을 선택한 상태라고 하자. 그렇다면 첫 번째 행을 선택하거나 네 번째 열을 선택하는 것 외에는 선택된 행, 열들의 교차점에 X가 생기게 된다. 따라서 첫 번째 행이나 네 번째 열을 선택해야만 하고 그 뒤에는 어떤 행이나 열을 골라도 교차점에 X가 생기므로 총 선택 횟수는 4번이며, 이보다 더 많이 선택할 수 있는 방법은 없다.

철수와 영희는 공부가 매우 하기 싫었기에 누가 이기는지와는 별개로 가장 오래가는 선택을 한다고 가정하자. 이때 철수와 영희는 도합 최대 몇 번의 선택을 할 수 있는지를 출력하시오.

입력 형식

첫 번째 줄에 게임판의 사이즈인 N 과 M , 그리고 X 표시된 칸의 개수 K 가 공백으로 구분되어 주어진다. ($1 \leq N, M \leq 200, 1 \leq K \leq NM$)

이후 K 개의 줄에 걸쳐 두 자연수 x, y 가 공백으로 구분되어 주어진다. ($1 \leq x \leq N, 1 \leq y \leq M$) 이는 x 행 y 열에 있는 칸에 X 표시가 있다는 것을 의미한다.

X 표시가 있는 칸이 중복되어서 입력되는 경우는 없다.

출력 형식

첫 번째 줄에 철수와 영희가 합쳐서 총 몇 번의 선택을 하게 되는지 출력한다.

예제

표준 입력	표준 출력
4 4 6 1 1 1 3 2 4 3 2 3 4 4 3	4

문제 G. 가장 높고 넓은 성

시간 제한: 1 초

메모리 제한: 512 MB

당신은 어마어마한 크기의 토지를 소유한 부자다. 평지에는 n 개의 표지판이 있다. 당신은 지금부터 이 곳에 민성이를 위한 성을 지을 것이다. 성은 여러 개의 층으로 구성된 구조물이다.

- 1층은 평지 위에 지어지고, 2층은 1층 위에, 3층은 2층 위에, \dots , n 층은 $n-1$ 층 위에 지어진다.
- 각 층의 경계는 다각형이며, 각 층의 경계의 꼭짓점에는 표지판이 위치하여야 한다. 다시 말하면, 표지판 위에만 각 층의 경계의 꼭짓점을 세울 수 있다.
- 한 표지판은 여러 층의 경계의 꼭짓점에 동시에 사용될 수 없다. 그러나 한 표지판이 다른 층의 꼭짓점이 아닌 경계에 위치할 수는 있다.
- 넓이가 0인 층은 허용되지 않는다.
- 쓸모가 없는 표지판들은 버려도 된다.

민성이는 숙소를 고를 때 전망을 최우선으로 고려하므로, 당신이 성을 지을 때 고려해야 할 우선순위는 다음과 같다.

1. 성의 높이를 가능한 한 제일 높게 지어야 한다.
2. 성의 모든 층의 넓이의 합이 가능한 한 제일 크게 지어야 한다.
3. 가능한 적은 수의 표지판을 사용하여야 한다.

당신은 성을 몇 층까지 지을 수 있으며, 그 때 각 층에 사용될 표지판들이 무슨 표지판인지 알아내야 한다.

입력 형식

첫 번째 줄에 표지판의 개수 n 이 주어진다. ($1 \leq n \leq 10^3$)

두 번째 줄부터 n 개의 줄에 걸쳐 각 표지판들의 위치를 의미하는 정수 x, y 가 공백으로 구분되어 주어진다. ($-10^4 \leq x, y \leq 10^4$) 이는 (x, y) 에 표지판이 위치함을 의미한다.

모든 표지판은 서로 다른 위치에 세워져 있다.

출력 형식

첫 번째 줄에 n 개의 정수 x_1, x_2, \dots, x_n 를 공백으로 구분하여 출력한다. x_i 는 i 번째 표지판이 사용되었을 경우 사용된 층수이며, 사용되지 않았으면 0이다.

예제

표준 입력	표준 출력
9 0 0 -1 3 -1 -2 -5 -5 2 -2 2 2 3 1 3 -5 1 -1	2 1 2 1 2 1 1 1 0
12 0 0 1 0 2 0 3 0 4 0 3 1 2 2 1 3 0 4 0 3 0 2 0 1	1 2 3 2 1 2 3 2 1 2 3 2

설명

첫 번째 예제를 시각화하면 다음과 같다. 마지막 점인 $(1, -1)$ 은 사용되지 않았다.

