



UCPC
greetings

22/23

Hello, BOJ 2023!

Official Problemset

주최 전국 대학생 프로그래밍 대회 동아리 연합

후원 STARTLINK



HYUNDAI
MOBIS

Presto

HYUNDAI
AutoEver

Proudly Supported By:

leejseo.com

한빛미디어
Hanbit Media, Inc.

문제 목록

문제지에 있는 문제가 총 8문제가 맞는지 확인하시기 바랍니다.

- A** 2023년이 기대되는 이유
- B** 청소
- C** 카드 플러리쉬
- D** 컵 쌓기
- E** 신도시 개발
- F** 줄넘기
- G** 편지 배달 2
- H** 정밀지도 제작

모든 문제의 메모리 제한은 1GB로 동일합니다.

문제지와 대회 페이지에 적혀 있는 내용이 다를 경우, 대회 페이지가 우선합니다.

문제 A. 2023년이 기대되는 이유

시간 제한 1 초
메모리 제한 1024 MB

Hello, BOJ 2023! 대회는 2023년의 시작을 기념하는 알고리즘 문제해결 대회이다.

다들 저마다의 이유로 올 한 해를 기대하고 있겠지만, Good Bye, BOJ 2022!와 Hello, BOJ 2023! 두 대회의 운영진들은 대회장에서 참가자 여러분을 만날 수 있게 되어 감사한 마음으로 2023년을 한껏 기대하고 있다.

하지만 위대한 수학자 준겸이는 2023년을 조금 다른 방식으로도 기대하고 있는 것 같다. 준겸이에 의하면, 2023이라는 수는 다음의 두 등식:

- $2^1 + 0^1 + 2^1 + 3^1 = 2 + 0 + 2 + 3$
- $2^3 + 0^3 + 2^3 + 3^3 = 20 + 23$

을 동시에 만족하기 때문에, 올해도 좋은 해가 될 것이라고 한다.

여러분은 이 얘기를 듣고, 다른 연도에 대해서도 준겸이가 말한 좋은 성질이 성립하는지가 궁금해졌다.

10진법으로 표기된 양의 정수 $n = \overline{d_0 d_1 \dots d_k}$ 가 주어진다. n 의 자릿수 사이사이에 0개 이상의 + 기호를 넣어서 계산했을 때, 그 값이 $d_0^m + d_1^m + \dots + d_k^m$ 과 같아지도록 하는 양의 정수 m 의 개수를 구하면 된다.

+ 기호를 삽입하는 방법의 수가 아닌, m 의 개수를 구해야 함에 유의하여라.

입력

첫째 줄에 테스트 케이스의 개수 T 가 주어진다.

이후 T 개의 줄에 걸쳐 10진법으로 표현된 양의 정수 n 이 주어진다.

출력

각 테스트 케이스에 대해, m 의 개수를 출력하여라.

만약 그러한 m 이 무수히 많다면, **Hello, BOJ 2023!**을 출력한다.

제한

- $1 \leq T \leq 1000$
- $1 \leq n \leq 10^9$
- 입력으로 주어지는 n 의 가장 큰 자리 수는 0이 아니다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5	Hello, BOJ 2023!
1	1
9	2
2022	2
2023	Hello, BOJ 2023!
1000000000	

설명

$n = 1$ 의 경우, 모든 양의 정수 m 에 대해서 $1^m = 1$ 이 성립하므로, 조건을 만족하는 m 은 무수히 많다. 따라서 Hello, BOJ 2023!을 출력한다.

$n = 2022$ 의 경우,

- $2^1 + 0^1 + 2^1 + 2^1 = 2 + 0 + 2 + 2$
- $2^3 + 0^3 + 2^3 + 2^3 = 2 + 0 + 22$

이므로, 가능한 m 의 개수는 2개이다. 따라서 2를 출력한다.

문제 B. 청소

시간 제한 2 초
메모리 제한 1024 MB

준석이는 청소 업체에 다니고 있다. 준석이가 청소할 장소는 1번부터 N 번까지 차례로 번호가 붙은 일렬의 N 개의 구역으로 나누어져 있다. i 번 구역은 $i-1$ 번과 $i+1$ 번 구역과 인접해 있어, 두 인접한 구역 사이를 이동하려면 1만큼 걸어야 한다.

각 구역에는 우선순위가 있다. i 번 구역의 우선순위 A_i 는 1 이상 N 이하의 정수로 이 값이 클수록 우선순위가 높다. 임의의 두 구역의 우선순위는 항상 다르다.

준석이는 오늘 이 구역 중 K 개의 구역을 먼저 청소하려고 한다. 단, 준석이가 청소하는 K 개의 구역은 반드시 연속해야 한다. 또한 청소할 때 선택한 구역들 내에서는 우선순위가 높은 구역부터 낮은 구역 순서대로 이동하며 청소해야 한다.

두 구역이 멀리 떨어져 있으면 이동하는 시간이 오래 걸리기 때문에, 준석이는 이동 거리의 합이 최소화되도록 연속한 K 개의 구역을 선택하려고 한다. 이동 거리가 최소가 되도록 구역을 선택했을 때 총 이동 거리를 출력한다.

입력

첫째 줄에 청소할 구역의 길이 N 과 오늘 청소할 구역의 개수 K 가 공백으로 구분되어 주어진다.

둘째 줄에 각 구역의 우선순위 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어진다.

출력

K 개의 구역을 선택했을 때 가능한 총 이동 거리 중 최솟값을 출력한다.

제한

- $1 \leq K \leq N \leq 500000$
- $1 \leq A_i \leq N$
- $i \neq j$ 이면 $A_i \neq A_j$ 이다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5 3 3 2 4 1 5	3
8 4 4 3 2 5 8 1 6 7	4
3 1 1 2 3	0

설명

첫 번째 예제의 설명이다.

- 1번 구역부터 3번 구역까지 청소한다면 3-1-2 순서로 이동하게 되고, 이동 거리는 3이다.
- 2번 구역부터 4번 구역까지 청소한다면 3-2-4 순서로 이동하게 되고, 이동 거리는 3이다.

- 3번 구역부터 5번 구역까지 청소한다면 $5-3-4$ 순서로 이동하게 되고, 이동 거리는 3이다.

모든 경우의 이동 거리가 3으로 동일하므로 정답은 3이다.

두 번째 예제의 설명이다.

- 1번 구역부터 4번 구역까지 청소한다면 $4-1-2-3$ 순서로 이동하게 되고, 이동 거리는 5이다.
- 2번 구역부터 5번 구역까지 청소한다면 $5-4-2-3$ 순서로 이동하게 되고, 이동 거리는 4이다.
- 3번 구역부터 6번 구역까지 청소한다면 $5-4-3-6$ 순서로 이동하게 되고, 이동 거리는 5이다.
- 4번 구역부터 7번 구역까지 청소한다면 $5-7-4-6$ 순서로 이동하게 되고, 이동 거리는 7이다.
- 5번 구역부터 8번 구역까지 청소한다면 $5-8-7-6$ 순서로 이동하게 되고, 이동 거리는 5이다.

이 중 2번 구역부터 5번 구역까지 청소하는 경우가 이동 거리가 가장 작으므로 정답은 4이다.

문제 C. 카드 플러리쉬

시간 제한 1 초
메모리 제한 1024 MB

주원이는 세계적인 마술사로부터 카드를 다루는 손기술을 배우고 있다. 주원이는 기술 연습을 위해 N 장의 카드로 구성된 카드 덱을 샀다. 카드에는 1번부터 N 번까지의 번호가 붙어 있으며, 덱의 위에서부터 오름차순으로 정렬되어 있다.

주원이는 오늘 한 손으로 카드를 섞는 기술 두 가지를 배웠다.

- 찰리어 컷: 카드를 연속한 두 묶음으로 나눈 뒤, 두 묶음의 순서를 뒤집는다.
- 트리플 컷: 카드를 연속한 세 묶음으로 나눈 뒤, 세 묶음의 순서를 뒤집는다.

각 묶음의 크기는 주원이가 원하는 대로 정할 수 있다. 주원이의 손은 누구보다도 정확하기 때문에, 카드를 정확히 원하는 크기의 묶음으로 나눌 수 있다. 단, 각 카드 묶음에는 반드시 하나 이상의 카드가 포함되어야 한다.

예시로 6장의 카드를 가지고 트리플 컷을 하는 과정을 살펴보자. 처음에는 덱의 맨 위에서부터 (1,2,3,4,5,6) 번 순서대로 카드가 정렬되어 있다. 카드를 크기 2,1,3의 묶음으로 나눈다면 각 묶음에 들어 있는 카드 번호는 (1,2), (3), (4,5,6)과 같다. 세 묶음의 순서를 뒤집으면 (4,5,6), (3), (1,2)가 된다. 이 순서로 묶음을 합치면 최종적인 카드 순서는 (4,5,6,3,1,2)번이 된다.

주원이는 두 가지 손기술을 최대 $N-1$ 번 사용해서 덱을 원하는 순서로 섞으려고 한다. 덱을 원하는 순서로 만들 수 있는지 판별하고, 가능하다면 그 방법을 출력하시오.

입력

첫 번째 줄에 카드의 수 N 이 주어진다.

두 번째 줄에 총 N 개의 카드 번호가 주어진다. 이는 주원이가 최종적으로 만들고 싶어하는 카드의 순서를 나타낸다.

출력

만약 주원이가 손기술을 최대 $N-1$ 번 사용하여 덱을 원하는 순서로 섞을 수 없으면, 첫 번째 줄에 -1 을 출력한다.

만약 주원이가 손기술을 최대 $N-1$ 번 사용하여 덱을 원하는 순서로 섞을 수 있으면, 첫 번째 줄에 주원이가 사용한 손기술의 횟수 K 를 출력하고 다음 K 개 줄에 걸쳐 아래 형식으로 그 과정을 순서대로 출력한다. ($0 \leq K \leq N-1$)

- **2 a:** 위에서부터 $[1, a]$ 번째 카드 묶음과 $[a+1, N]$ 번째 카드 묶음으로 나누는 찰리어 컷을 나타낸다. ($1 \leq a < N$)
- **3 a b:** 위에서부터 $[1, a]$ 번째 카드 묶음, $[a+1, b]$ 번째 카드 묶음, $[b+1, N]$ 번째 카드 묶음으로 나누는 트리플 컷을 나타낸다. ($1 \leq a < b < N$)

손기술을 사용한 횟수가 최소일 필요는 없으며, 답이 여러 가지인 경우 아무거나 하나 출력한다.

제한

- $2 \leq N \leq 3\,000$
- 주어지는 카드 번호는 1 이상 N 이하의 서로 다른 수이다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5 2 3 5 1 4	2 2 3 3 1 3

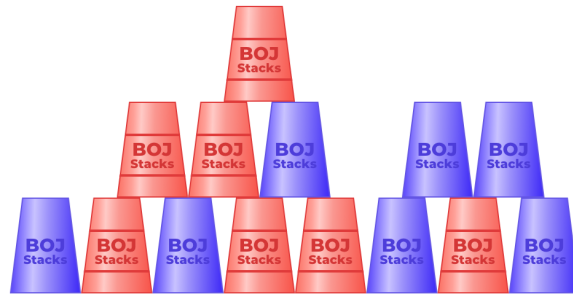
설명

다음과 같은 순서로 카드를 섞는 과정을 나타낸다.

- $(1, 2, 3, 4, 5) \rightarrow (1, 2, 3), (4, 5) \rightarrow (4, 5), (1, 2, 3) \rightarrow (4, 5, 1, 2, 3)$
- $(4, 5, 1, 2, 3) \rightarrow (4), (5, 1), (2, 3) \rightarrow (2, 3), (5, 1), (4) \rightarrow (2, 3, 5, 1, 4)$

문제 D. 컵 쌓기

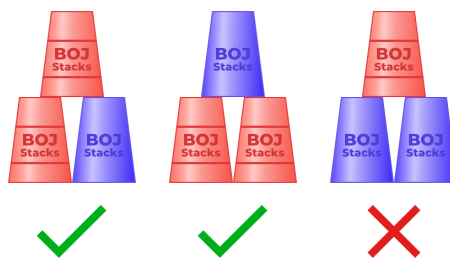
시간 제한 1 초
메모리 제한 1024 MB



스포츠스태킹은 N 개의 컵을 다양한 방법으로 빠르게 쌓는 스포츠 경기다. BOJ 스포츠스태킹 협회의 회장 한나는 이번에 새롭게 출시한 빨간 컵의 판매량을 늘리기 위해 새로운 스포츠스태킹 종목을 도입하고자 한다. (문제의 그림에서 빨간 컵은 가로줄이 그려져 있는 컵이다.) 그 규칙은 다음과 같다.

참가자는 빨간 컵 N 개와 파란 컵 N 개를 준비한다. 참가자는 이 중 N 개의 컵을 쌓아서 규칙에 맞는 배치를 아무거나 만들어야 한다. 컵을 쌓는 규칙은 다음과 같다.

- 맨 아랫층에는 컵이 일렬로 연속해 있어야 한다.
- 두 이웃한 컵의 중앙 위쪽에 다른 컵을 쌓을 수 있다. 이를 반복해서 여러 층을 쌓을 수도 있다.
- 두 이웃한 컵 위에 다른 컵을 쌓으려면, 두 컵 중 적어도 하나는 빨간 컵이어야 한다.



색깔이 같은 컵끼리는 구분할 수 없다고 할 때, 컵을 쌓을 수 있는 경우의 수를 주어지는 소수 P 로 나눈 나머지를 구하시오. 두 배치가 서로 대칭이어도, 서로 다른 배치로 센다.

입력

첫 번째 줄에 쌓을 컵의 개수를 나타내는 정수 N 과 나누어 출력할 소수 P 가 공백을 사이에 두고 주어진다.

출력

규칙을 만족하도록 N 개의 컵을 쌓는 경우의 수를 P 로 나눈 나머지를 출력한다.

제한

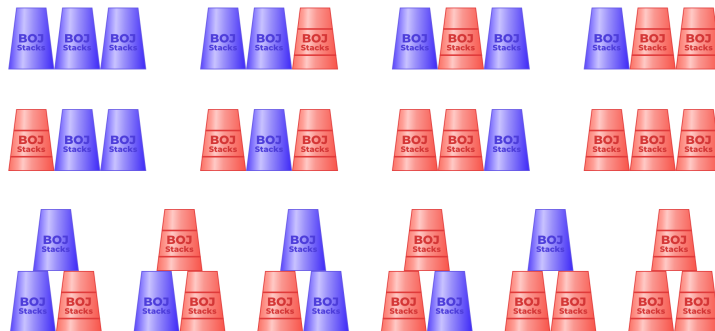
- $1 \leq N \leq 150$
- $2 \leq P \leq 10^9$
- P는 소수이다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 998244353	14
5 998244353	124
7 998244353	1248

설명

3개의 컵을 쌓는 방법은 다음 그림과 같이 14가지가 있다.



문제 E. 신도시 개발

시간 제한 1 초
메모리 제한 1024 MB

BOJ 도시에는 1번부터 N 번까지 번호가 붙은 N 개의 토지가 일렬로 있고, 이 중 이미 분양된 토지가 M 개 있다.

BOJ 도시의 시장인 청한이는 BOJ 도시를 개발하기 위해 아직 분양되지 않은 토지 중 K 개를 분양하려고 한다. 단 도시와 멀면 토지의 수요가 줄어들기 때문에, 중심으로부터 멀어질수록 할인하여 분양하려고 한다. 구체적으로, 어떤 토지를 분양하는 시점에 이 토지의 왼쪽과 오른쪽에 있는 분양된 토지의 개수 차이를 D 라고 할 때, 이 토지는 원래 가격에서 D 만큼 할인하여 분양한다.

청한이는 BOJ 도시의 토지 중 K 개를 골라 적절한 순서로 분양해서 최대한의 이익을 내고 싶다. 즉, 분양한 토지들이 할인된 양의 총합을 최소화해야 한다. 토지는 청한이가 정한 순서대로 하나씩 분양하며, 동시에 여러 개의 토지를 분양할 수 없다. 최적의 방법으로 토지를 분양했을 때, 할인된 양의 총합은 얼마인지 구하시오.

입력

첫 번째 줄에 BOJ 도시의 토지의 개수 N , 이미 분양된 토지의 개수 M , 앞으로 분양할 토지의 개수 K 가 주어진다.

두 번째 줄부터 M 개의 줄에 이미 분양된 토지의 번호 x_i 가 주어진다.

출력

최적의 방법으로 토지를 분양했을 때, 분양한 토지들의 할인된 양의 총합을 출력한다.

제한

- $1 \leq N \leq 300000$
- $1 \leq M, K \leq N$
- $M + K \leq N$
- $1 \leq x_i \leq N$
- 주어지는 모든 x_i 는 서로 다르다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
6 2 2 1 3	3

설명

1번 토지와 3번 토지가 이미 분양되어 있다. 6번 토지를 새롭게 분양하면, 왼쪽에 분양된 토지가 2개이고 오른쪽에 분양된 토지가 0개이므로 2만큼 할인해서 분양하게 된다. 그 이후 4번 토지를 새롭게 분양하면, 왼쪽에 분양된 토지가 2개이고 오른쪽에 분양된 토지가 1개이므로 1만큼 할인해서 분양하게 된다. 따라서 할인의 총합은 3이 된다. 이보다 더 적게 할인하면서 2개의 토지를 분양할 수 있는 방법은 없다.



Hello, BOJ 2023!

이 페이지는 공백입니다

문제 F. 줄넘기

시간 제한 3 초
메모리 제한 1024 MB

우경이는 N 개의 밧줄을 가지고 줄넘기를 한다. 각각의 밧줄은 평면 위의 직선으로 나타낼 수 있으며, 일정한 속도로 평행이동하고 있다. 어떤 두 밧줄도 서로 평행하거나 일치하지 않는다. 우경이는 평면상의 밧줄이 지나지 않는 위치 S 에서 줄넘기를 시작하며, 줄넘기를 하는 동안 원하는 경로와 속도로 자유롭게 이동할 수 있다. 이동 도중에 밧줄을 만날 때마다 밧줄을 한 번 넘은 것으로 센다. 한 번에 두 개 이상의 밧줄을 넘을 수는 없다.

우경이는 일정한 시간이 지나면 줄넘기를 멈출 것이다. 우경이는 줄넘기를 멈추기 전까지 밧줄을 넘는 횟수가 최소가 되도록 이동하고자 한다. 줄넘기를 멈췄을 때 우경이의 위치는 밧줄이 지나지 않는 곳이어야 하며, 시작한 위치와 달라도 된다. 우경이가 밧줄을 넘어야 하는 횟수는 최소 몇 번인지 구하라.

입력

첫 번째 줄에 밧줄의 수 N 이 주어진다.

두 번째 줄에 두 정수 X, Y 가 주어진다. 우경이의 시작 위치 좌표가 $S = (X, Y)$ 임을 나타낸다.

세 번째 줄부터 N 개의 줄에 밧줄의 정보를 나타내는 여섯 개의 정수 x_1, y_1, x_2, y_2, a, b 가 주어진다. 우경이가 줄넘기를 시작했을 때 밧줄이 (x_1, y_1) 과 (x_2, y_2) 를 잇는 직선 형태이고, 우경이가 줄넘기를 멈출 때까지 밧줄이 (a, b) 만큼 평행이동하게 됨을 의미한다.

출력

우경이가 줄넘기를 멈추기 전까지 밧줄을 넘어야 하는 횟수의 최솟값을 출력한다.

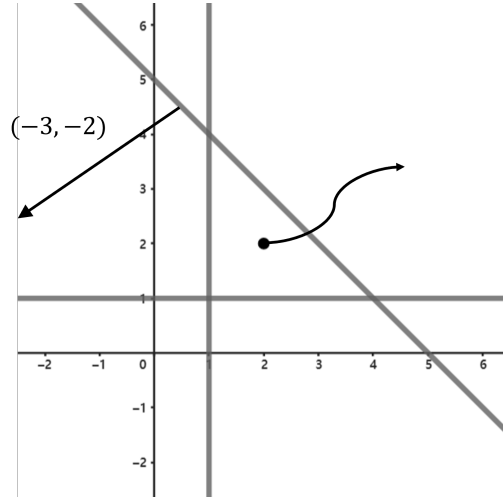
제한

- $1 \leq N \leq 2000$
- $|X|, |Y|, |x_1|, |y_1|, |x_2|, |y_2|, |a|, |b| \leq 1000$
- $(x_1, y_1) \neq (x_2, y_2)$
- 우경이가 줄넘기를 시작했을 때, S 는 평면상에서 밧줄 위에 있지 않다.
- 어떤 두 밧줄도 서로 평행하거나 일치하지 않는다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 2 2 0 1 1 1 0 0 1 0 1 1 0 0 5 0 0 5 -3 -2	1

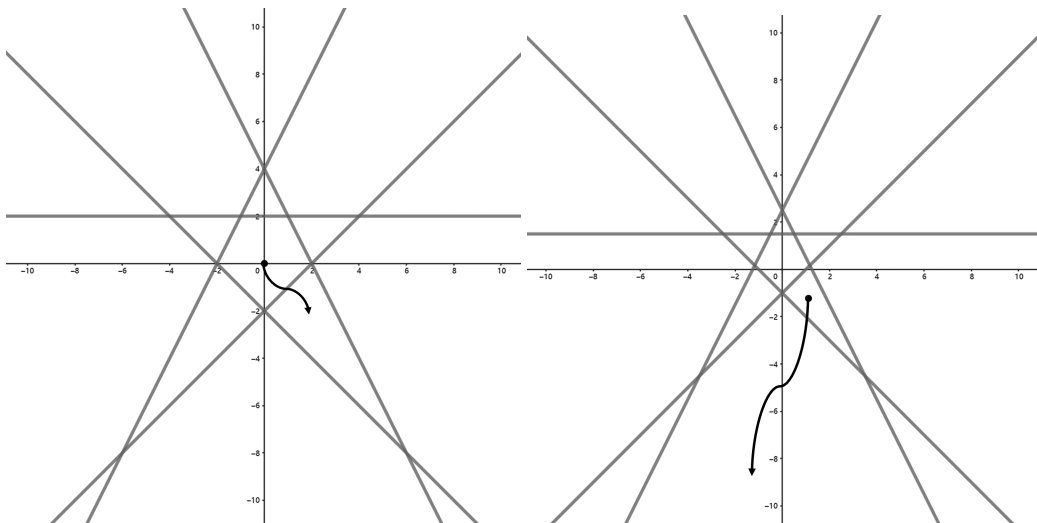
설명

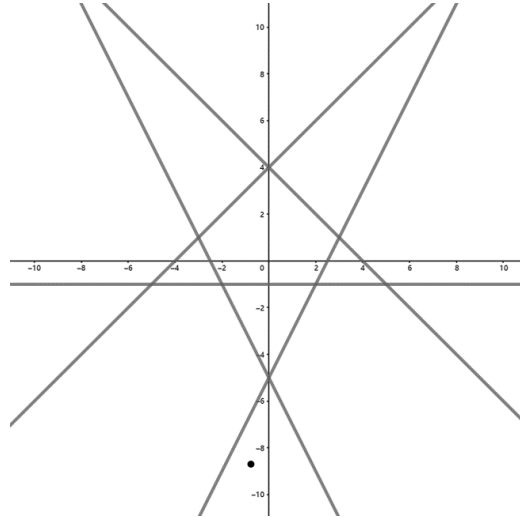


처음에 우경이는 (2,2)에 서 있다. 밧줄은 $y=1$, $x=1$, $x+y=5$ 총 3개가 있다. 이 중 $y=1$ 과 $x=1$ 은 움직이지 않고, $x+y=5$ 는 줄넘기를 하는 동안 $(-3, -2)$ 만큼 등속으로 평행이동한다. 우경이는 줄넘기를 하는 동안 자유롭게 이동하여 줄을 넘을 수 있다. 위 그림은 줄넘기를 시작할 때 $x+y=5$ 에 있었던 밧줄이 움직이는 도중, 이 밧줄을 넘으며 우상단으로 이동하는 예시이다. 이 경우에 우경이가 넘는 밧줄은 한 개이고, 이보다 더 적게 밧줄을 넘을 수 있는 방법은 없다.

표준 입력(stdin)	표준 출력(stdout)
5 0 0 -1 2 1 2 0 -3 1 2 2 0 -3 -3 2 0 0 -2 -3 3 0 -2 -2 0 3 3 -2 0 -1 2 3 -3	2

설명

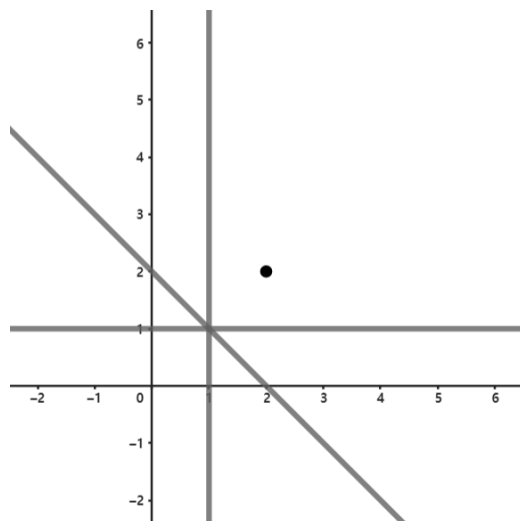




처음에 우경이는 (0,0)에 서 있다. 위 그림은 우경이가 두 번 밋줄을 넘는 예시이다. 첫 번째 그림에서 보이듯 우경이는 줄넘기 시작 후 줄이 이동하는 동안 우하단으로 이동하여 줄을 하나 넘을 수 있다. 그 이후 두 번째 그림처럼 좌하단으로 다시 이동하여 줄을 총 둘 넘을 수 있다. 이후 우경이는 줄넘기가 종료될 때까지 더 이상 줄을 넘지 않는 경로로만 이동할 수 있다. 이보다 더 적게 밋줄을 넘을 수 있는 방법은 없다.

표준 입력(stdin)	표준 출력(stdout)
<pre>3 2 2 0 1 1 1 0 0 1 0 1 1 0 0 5 0 0 5 -2 -1</pre>	<pre>1</pre>

설명



$x + y = 5$ 직선이 $(-2, -1)$ 만큼 평행이동하는 것을 제외하면 예제 1과 동일하다. 그림은 줄넘기가 끝나고 난 상태를 나타내고 있다.



Hello, BOJ 2023!

이 페이지는 공백입니다

문제 G. 편지 배달 2

시간 제한 3 초
메모리 제한 1024 MB

Hello 고등학교에서는 다른 반에 있는 친구와 편지를 주고받는 것이 유행이다. Hello 고등학교에는 1반부터 N 반까지 총 N 개의 반이 있고, 각 반의 교실은 반 번호 순서대로 직선 형태의 복도를 따라 나열되어 있다.

쉬는 시간마다 복도를 산책하는 것을 좋아하는 동규는, 산책하는 김에 학생들의 편지도 배달해 주기로 했다. 동규가 산책하는 방법은 다음과 같다. 먼저 동규는 $L+1$ 개의 반 번호 $c_0, c_1, \dots, c_{L-1}, c_L$ 를 정한다. 이때 $c_0 = c_L$ 은 동규의 반 번호로, 동규는 이곳에서 출발해서 이곳으로 도착해야 한다. 산책은 총 L 번의 이동으로 이루어진다. i 번째 이동에서는 현재 위치에서 c_i 반을 향해 일직선으로 이동한다. 교실이 직선 형태의 복도를 따라 나열되어 있기 때문에, 동규가 c_{i-1} 반에서 c_i 반을 향해 이동하는 동안 두 반 사이에 있는 모든 반을 지나게 된다.

학생들은 두 명씩 쌍을 이뤄 편지를 주고받는다. 편지를 주고받을 두 학생은 서로 다른 반에 있어야 한다. a 반 학생 A와 b 반 학생 B가 편지를 주고받는 방법은 다음과 같다.

- 먼저 A가 B에게 보낼 편지를 준비한다. 동규는 산책 도중 처음으로 a 반을 지날 때 A의 편지를 건네받는다.
- 동규는 A의 편지를 받은 이후 처음으로 b 반을 지날 때 B에게 편지를 전달하고, B는 편지를 받은 즉시 답장 편지를 작성해서 동규에게 건넨다.
- 동규는 B의 편지를 받은 이후 처음으로 a 반을 지날 때 A에게 편지를 전달하고, A는 편지를 받은 즉시 답장 편지를 작성해서 동규에게 건넨다.
- 동규가 산책을 끝낼 때까지 두 학생은 계속 번갈아서 답장 편지를 보낸다.

총 M 쌍의 학생들이 편지를 주고받고자 한다. 동규는 편지를 동시에 몇 개든지 지닐 수 있으며, 산책을 시작할 때와 끝낼 때에도 자기 반에서 편지를 건네받거나 전달할 수 있다. 산책이 끝났을 때 동규에게 남은 편지는 배달하지 못한 것으로 간주한다. $i = 1, \dots, L$ 에 대해, 동규가 i 번째 이동을 마쳤을 때 지금까지 전달한 편지의 개수를 구하시오.

입력

첫 번째 줄에 정수 N, L, M 이 공백으로 구분되어 주어진다.

두 번째 줄에 동규의 산책 경로를 나타내는 L 개의 정수 c_0, \dots, c_{L-1} 이 공백으로 구분되어 주어진다. c_L 은 c_0 과 같으므로 주어지지 않음에 유의하라.

세 번째 줄부터 M 개의 줄에, 편지를 주고받을 학생 쌍에 대한 정보 a, b 가 공백으로 구분되어 주어진다. a 반과 b 반에 있는 학생이 서로 편지를 주고받고자 함을 나타내며, a 반의 학생이 먼저 편지를 보낸다.

출력

총 L 개의 줄을 출력한다. i 번째 줄에는 동규가 i 번째 이동을 마쳤을 때 지금까지 배달한 편지의 개수를 출력한다.

제한

- $2 \leq N \leq 200\,000$
- $1 \leq L, M \leq 200\,000$
- $1 \leq c_i \leq N$
- 모든 학생 쌍에 대해서, $1 \leq a, b \leq N$ 이고 $a \neq b$ 를 만족한다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
10 4 5	2
4 8 2 6	6
5 8	8
6 3	9
4 9	
7 2	
5 6	

문제 H. 정밀지도 제작

시간 제한 2.5 초
메모리 제한 1024 MB

현대오토에버는 미래차 성장 동력의 한 축인 인포테인먼트 산업 전반을 혁신적으로 리드합니다. GIS 분야의 축적된 기술력과 노하우를 바탕으로 디지털 맵, 내비게이션 SW, 자율주행 정밀 지도를 정교하게 제작하고 글로벌 위치 기반 콘텐츠와 통합 OTA 서비스를 제공하여 사용자가 전혀 다른 모빌리티 서비스, 혁신적인 카 라이프를 경험할 수 있도록 합니다.

현대자동차그룹이 레벨3 자율주행차 상용화 목표에 발맞춰 충력을 다하고 있는 가운데, 국내 최고 수준의 지도 구축 기술력을 보유한 현대오토에버는 자율주행에 필요한 정밀지도를 제작해 배포하고, 기술 고도화를 위한 연구에 매진하고 있습니다. 카메라와 센서만으로는 탐재 장치에 이상이 생기거나 눈, 질은 안개 등 기상 상황이 악화되었을 때 주변 상황을 제대로 인지할 수 없기 때문입니다. 정밀지도에는 도로의 경계와 차선은 물론 각종 시설물의 정보가 cm 단위로 매우 상세하게 표시됩니다. 또한 중앙선과 경계선 등 차선 단위 정보와 신호등, 횡단보도, 표지판, 노면 마크 등을 3D로 확보해 악천후나 차량 센서 고장처럼 긴급 상황에서도 자율주행차가 안정적으로 주행할 수 있도록 도와줍니다.

현대오토에버에서 근무하는 종서는 도로 데이터를 기반으로 자동으로 정밀지도를 구축하는 지도 자동 구축(Map Auto Creation) 기술을 개발해 지도 제작 시간을 단축하고 정밀도를 향상시키는 연구를 진행 중이다. 종서는 MAC 기술을 이번에 새로 지어지는 신도시에 적용하기로 결정했다.

종서가 분석하고 있는 신도시에 1번부터 N 번까지 번호가 붙여진 N 개의 교차로가 있다. 앞으로 신도시에 서로 다른 두 교차로를 잇는 도로가 M 개 지어질 것이다. 신도시는 교차점을 정점으로, 도로를 간선으로 하는 단순 연결 그래프 형태이다.

도로는 각각 1번부터 M 번까지 번호가 붙여져 있다. i 번 도로는 a_i 번 교차로와 b_i 번 교차로를 이으며, k_i 시점에 건설이 완료될 예정이다. 종서는 적당한 음이 아닌 정수 t 를 정한 뒤에, 각 도로마다 건설된 직후부터 $t + 0.5$ 만큼의 시간 동안 도로를 분석하여 정밀지도를 제작할 예정이다. 따라서 i 번 도로는 정확히 $[k_i, k_i + t + 0.5)$ 구간에서 분석될 예정이다.

교차로 중 P 개의 교차로 s_1, s_2, \dots, s_P 에는 건물이 지어져 있다. 이번 정밀지도 구축의 핵심 목적 중 하나는, 이 건물들 사이에서 자율주행차량의 원활한 이동이 가능한지를 확인하는 것이다.

종서는 서로 다른 Q 개의 시각 T_1, T_2, \dots, T_Q 를 잘 정해서, 서로 다른 Q 개의 지도 데이터를 확보하고자 한다. 만약 시각 T_i 에 분석 중인 도로들만으로 모든 임의의 두 건물 사이를 이동할 수 있다면 종서는 지도 데이터 i 를 얻는다. 두 지도 데이터 i 와 j 가 다르다는 것은, 시각 T_i 에 분석 중인 도로들의 집합이 시각 T_j 에 분석 중인 도로들의 집합과 다를음을 의미한다.

종서는 분석 비용을 절감하기 위해 t 를 최소화하고 싶다. 서로 다른 Q 개의 지도 데이터를 확보 가능한 최소 t 는 얼마인가?

입력

첫째 줄에 N, M, P, Q 가 공백을 구분으로 주어진다.

다음 줄에 건물이 있는 교차로의 번호 s_i 가 공백을 구분으로 주어진다.

이후 M 개의 줄에는 i 번째 줄에 i 번 도로에 대한 정보인 a_i, b_i, k_i 가 주어진다.

출력

조건을 만족하는 최소 t 를 출력한다. 만약 조건을 만족할 수 없다면 **-1**을 출력한다.

제한

- $2 \leq N \leq 100\,000$

- $1 \leq M \leq 300000$
- $2 \leq P \leq N$
- $1 \leq a_i, b_i, s_i \leq N$
- $1 \leq k_i \leq 10^9$
- $1 \leq Q \leq 10^6$
- k_i 는 모두 다르다.
- s_i 는 모두 다르다.
- 신도시를 그래프로 나타내면, 단순 연결 그래프이다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
4 4 2 3 2 4 1 2 1 2 3 3 3 4 6 4 1 10	7

설명

이 도시에서 건물은 2번 교차로와 4번 교차로에 지어져 있다. 만약 $t = 7$ 이라면, 종서는 시각 6, 9, 10을 정할 수 있다. 이때 분석중인 도로의 집합은 각각 (1,2,3), (2,3), (2,3,4)이고, 모든 경우에 대해 2번 교차로의 건물과 4번 교차로의 건물을 지날 수 있다. 따라서 $t = 7$ 일 때 서로 다른 Q 개의 지도 데이터를 확보할 수 있다. 이는 가능한 최소인 음이 아닌 정수이다.