

문제 1. 빌딩 장식 2

입력 파일: standard input
출력 파일: standard output
시간 제한: 1초
메모리 제한: 64MB

일본에는 N 개의 거리가 있고, 그 사이를 $N-1$ 개의 양방향으로 통행할 수 있는 도로가 연결하고 있다. 어떤 거리에서 다른 어떤 거리까지도 몇 개의 도로를 거쳐서 갈 수 있다. 각 도로에는 빌딩이 하나 있고, i 번째 도로 빌딩의 높이를 H_i 라고 하자.

국제정보올림피아드에 참가하기 위해 일본에 온 세계 선수단을 환영하기 위해 관광여행을 개최하기로 했다. 관광여행은 어느 거리에서 시작하고, 도로를 통해 다른 거리로 이동하는 것을 반복해 어떤 거리를 방문한 후 끝낸다. 이때, 같은 거리를 두 번 이상 방문하지는 않는다.

세계 선수들을 더욱 환영하기 위해 관광에서 방문하는 몇몇 거리에 있는 빌딩을 장식하기로 했다. 어떤 저명한 디자이너에게 디자인을 의뢰했는데, 장식하는 빌딩의 높이는 출발하는 도시에서 도착하는 도시로 갈수록 높이가 높아질 필요가 있다고 했다. 즉, 장식하는 빌딩이, 방문하는 순서대로 i_1, i_2, \dots, i_k 번째 거리의 빌딩이라고 할 때, $H_{i_1} < H_{i_2} < \dots < H_{i_k}$ 여야 한다(방문하는 모든 빌딩을 장식할 필요는 없음에 주의하여라).

가능한 한 화려하게 꾸미기 위해서 최대한 많은 빌딩을 장식하고 싶다. 관광여행을 시작하는 거리, 끝내는 거리, 그리고 빌딩을 장식하는 거리를 마음대로 선택할 수 있다면, 장식할 수 있는 빌딩 수의 최댓값을 구하는 프로그램을 작성하여라.

제한

$2 \leq N \leq 100\,000$ 거리의 수
 $1 \leq H_i \leq 1\,000\,000\,000$ i 번째 거리에 있는 빌딩의 높이

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N 이 주어지고, 거리의 수를 의미한다.
- 다음 N 개의 줄에는 각 거리에 있는 빌딩의 높이에 관련된 정보가 주어진다. i 번째 줄에는 정수 H_i 가 주어진다. 이는 i 번째 거리에 있는 빌딩의 높이가 H_i 라는 것을 의미한다.
- 다음 $N-1$ 개의 줄에는 도로에 관련된 정보가 주어진다. i 번째 줄에는 정수 A_i, B_i ($1 \leq A_i < B_i \leq N$) 이 공백으로 구분되어 주어진다. 이는 i 번째 도로가 A_i 번 거리와 B_i 번 거리를 잇는다는 것을 의미한다.

출력 형식

표준 출력으로, 장식 할 수 있는 빌딩 수의 최댓값을 나타내는 정수를 출력하여라.

배점

채점 데이터 중, 배점의 10%에 대해 $N \leq 100$ 을 만족한다.

채점 데이터 중, 배점의 30%에 대해 $N \leq 2000$ 을 만족한다.

예제

standard input	standard output
7 4 2 5 3 1 8 7 1 2 2 3 3 4 4 5 3 6 6 7	4

문제 2. 물고기

입력 파일: standard input
출력 파일: standard output
시간 제한: 1.5초
메모리 제한: 64MB

JOI 군은 문득 물고기를 기르고 싶어졌다.

JOI 군의 근처에 있는 애완동물 가게에 가 보니, N 마리의 물고기를 팔고 있었다. i 번째 물고기의 길이는 L_i 이며, 색은 빨강, 초록, 파랑 중 하나였다. JOI 군은 이 N 마리의 물고기 중 1마리 이상을 집으로 가져오려고 결심했다.

물고기를 기를 때 주의해야 할 것이 있는데, 큰 물고기와 작은 물고기를 동시에 기르면 큰 물고기가 작은 물고기를 먹어버린다. 구체적으로는 물고기 X 의 길이가 물고기 Y 의 길이의 2배 이상일 때 X 와 Y 를 동시에 기르면, X 가 Y 를 잡아먹어 버린다. 그렇기 때문에, 이렇게 두 마리의 물고기를 동시에 기를 수는 없다.

JOI 군은 기르려는 물고기의 색 조합이 몇 종류 있을 수 있는지 궁금해졌다. 두 가지 색 조합이 다르다는 것은 빨강, 초록, 파랑 중 적어도 하나의 색에서 그 색의 물고기의 수가 다르다는 것이다. 애완동물 가게에서 파는 물고기의 길이와 색이 주어졌을 때, JOI 군은 기르려는 물고기의 서로 다른 색 조합이 몇 가지 있는지 구하고 싶어졌다.

애완동물 가게에서 파는 물고기의 길이와 색이 주어졌을 때, JOI군이 기를 수 있는 물고기의 색 조합의 수를 구하는 프로그램을 작성하여라.

제한

$2 \leq N \leq 500\,000$ 물고기의 수
 $1 \leq L_i \leq 1\,000\,000\,000$ i 번째 물고기의 길이

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N 이 주어진다. N 은 애완동물 가게에서 팔고 있는 물고기의 수를 의미한다.
- $i + 1$ 번째 ($1 \leq i \leq N$) 줄에는, 정수 L_i 와 문자 C_i 가 공백으로 구분되어 주어진다. 문자 C_i 는 R, G, B 중 하나이다. 이것은 i 번째 물고기의 길이가 L_i cm 이며, C_i 가 R이면 i 번째 물고기의 색이 빨강, G이면 초록, B이면 파랑임을 의미한다.

출력 형식

표준 출력으로, JOI군이 기를 수 있는 물고기의 색 조합의 개수를 첫째 줄에 출력하여라.

배점

채점 데이터 중, 배점의 10%에 대해 $N \leq 100$ 을 만족한다.

채점 데이터 중, 배점의 30%에 대해 $N \leq 2\,000$ 을 만족한다.

예제

standard input	standard output
4 10 R 4 G 8 B 5 B	6

첫 번째 물고기는 두 번째 물고기를 먹어버리기 때문에, 이 둘을 동시에 기를 수 없다. 첫 번째와 네 번째, 두 번째와 세 번째 물고기에 대해서도 같다. 따라서, 가능한 색 조합은 빨간색 한 마리, 초록색 한 마리, 파란색 한 마리, 빨간색과 파란색 한 마리씩, 초록색과 파란색 한 마리씩, 파란색 두 마리로 총 6종류 이다.

standard input	standard output
10 26 B 10 B 16 G 20 R 6 R 5 G 13 G 40 R 8 R 33 R	13

문제 3. JOI 깃발

입력 파일: standard input
출력 파일: standard output
시간 제한: 3초
메모리 제한: 64MB

당신은 일본정보올림피아드의 새로운 깃발로, 레벨 K 의 **JOI Flag**를 만들게 되었다.

- 레벨 0의 JOI Flag는, 1×1 격자로 이루어져 있으며, J, 0, I 중 하나의 문자가 쓰여있다.
- 정수 $m > 0$ 에 대해, 레벨 m 의 JOI Flag는, $2^m \times 2^m$ 격자로 이루어져 있고, 각 격자 칸에 J, 0, I 중 하나의 문자가 쓰여 있으며, 다음 조건을 만족하는 것이다: 격자 전체를 $2^{m-1} \times 2^{m-1}$ 의 정사각형 네 개로 분할을 할 때, 레벨 $m-1$ 의 JOI Flag, 격자 전체에 J만 쓰인 부분, 격자 전체에 0만 쓰인 부분, 격자 전체에 I만 쓰인 부분의 네 부분으로 나뉘어 진다.

예를 들면,

```
OIJJ
JJJJ
00II
00II
```

는 레벨 2의 JOI Flag이다. 또한,

```
IIIIII00
IIIIII00
IIIIJ0JJ
IIII0IJJ
JJJJ0000
JJJJ0000
JJJJ0000
JJJJ0000
```

는 레벨 3의 JOI Flag이다.

당신의 손에는 몇 개의 격자 칸에 J, 0, I 중 하나의 문자가 쓰인 $2^K \times 2^K$ 의 깃발이 있다.

당신은 이 깃발에서 몇 개의 문자를 추가하거나, 이미 깃발에 있는 문자를 수정해서 레벨 K 의 JOI Flag를 완성하려고 한다. 문자가 쓰이지 않은 격자 칸에 새로운 문자를 써넣는 것은 0의 비용이 들며, 문자가 쓰인 격자 칸에 있는 문자를 바꾸는 데에는 1의 비용이 든다.

당신의 손에 있는 깃발에 정보가 주어질 때, 레벨 K 의 JOI Flag를 완성하는 데 필요한 비용 총합의 최솟값을 구하는 프로그램을 작성하여라.

제한

$1 \leq K \leq 30$	JOI Flag의 레벨
$1 \leq N \leq 1\,000$	JOI Flag에 쓰인 격자 칸의 개수
$1 \leq X_i \leq 2^K$	i 번째 문자가 쓰인 열 번호
$1 \leq Y_i \leq 2^K$	i 번째 문자가 쓰인 행 번호

C_i 는 J, 0, I 중 하나이다.
(X_i, Y_i)는 서로 다르다.

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 K, N 이 공백으로 구분되어 주어진다. K 는 JOI Flag의 레벨을, N 은 문자가 쓰인 격자 칸의 개수를 각각 의미한다. 쓰인 문자에는 1, 2, \dots , N 의 번호가 붙어있다.

- 다음 N 개의 줄에는 문자의 정보가 주어진다. $i + 1$ 번째 줄에는 X_i, Y_i, C_i 가 공백으로 구분되어 주어진다. 이는 문자 C_i 가 왼쪽으로부터 X_i 번째 열, 위쪽으로부터 Y_i 번째 행에 쓰여있다는 것을 의미한다.

출력 형식

표준 출력으로, 레벨 K 의 JOI Flag를 완성하는 데 필요한 비용의 최솟값을 의미하는 정수를 첫째 줄에 출력하여야.

배점

채점 데이터 중, 배점의 40%에 대해 $K \leq 10$ 을 만족한다.

예제

standard input	standard output
2 10 2 2 J 3 3 I 1 3 I 1 1 0 3 2 J 2 1 I 4 1 0 3 4 I 4 4 0 2 3 0	3

이 입력은,

0I-0
-JJ-
I0I-
--I0

이란 깃발을 의미한다. 단, 아무것도 쓰여있지 않은 칸은 -로 표시한다.

이 깃발을 비용 3으로

0IJJ
JJJJ
00II
00II

로 바꿀 수 있다.

standard input	standard output
4 30 16 14 J 2 8 0 10 9 J 10 13 I 6 6 0 11 14 I 1 2 I 3 2 0 3 10 0 1 12 I 4 11 I 9 5 J 15 1 0 12 4 I 16 5 J 10 7 J 3 8 J 4 10 I 4 7 I 2 11 I 2 12 0 15 5 J 15 7 J 6 9 J 5 7 0 14 5 J 12 11 J 15 10 0 13 16 I 13 11 I	9

문제 4. TV 방송

출력 파일: Output Only

이번에 JOI 왕국에서는 TV 방송을 시작하기로 했다. JOI 왕국에는 N 개의 집이 있다. 이 집 모두가 TV를 볼 수 있도록 기지국을 설치하고 싶다.

이번에 JOI 왕국의 왕은 K 개의 전파탑을 설치하기로 했다. i 번째 전파탑을 (X_i, Y_i) 에 설치하고 출력 레벨을 E_i 로 설정하면, (X_i, Y_i) 부터 거리가 $\sqrt{E_i}$ 이하인 집에 TV 방송을 볼 수 있게 된다(두 점 (a, b) 와 (c, d) 사이의 거리는 $\sqrt{(a-c)^2 + (b-d)^2}$ 로 주어진다). 단, 전파탑의 출력 레벨을 E_i 로 설정하면, E_i 의 에너지를 소비한다. 모든 집에서 TV 방송을 볼 수 있게 하면서, 가능한 한 K 개 전파탑에서 사용하는 에너지 소비의 총합(이하, 소비 에너지)을 최대한 줄이고 싶다. 전파탑은 집이 있는 곳에 세울 수도 있으며, 같은 위치에 두 개를 설치해도 된다.

JOI 왕국에 있는 집의 좌표가 주어졌을 때, 모든 집에서 TV 방송을 볼 수 있도록 전파탑을 세우는 장소와 출력 레벨을 정하고 싶다.

JOI 왕국에 있는 집의 위치와 K 가 주어질 때, 전파탑을 세우는 장소와 출력 레벨을 출력하여라. 소비 에너지가 적을수록 고득점이 된다.

제한

$1 \leq N \leq 500$	JOI 왕국에 있는 집의 수
$1 \leq K \leq 30$	설치할 전파탑의 수
$0 \leq A_i, B_i \leq 1\,000\,000$	집의 좌표

입력 형식

다음 정보가 입력 파일로 주어진다.

- 첫째 줄에는 정수 N, K 가 공백으로 구분되어 주어지며, JOI 왕국에 있는 집의 수가 N , 설치할 전파탑의 수가 K 라는 것을 의미한다.
- 다음 N 개의 줄의 i 번째 줄에는 정수 A_i, B_i 가 공백으로 구분되어 주어지며, i 번째 집의 좌표가 (A_i, B_i) 라는 것을 의미한다.

같은 좌표에 두 개 이상의 집이 있는 경우는 없다고 생각해도 된다.

출력 형식

출력의 i 번째 줄 ($1 \leq i \leq K$) 에는 세 개의 정수 X_i, Y_i, E_i ($0 \leq X_i \leq 1\,000\,000, 0 \leq Y_i \leq 1\,000\,000, 0 \leq E_i \leq 1\,000\,000\,000\,000(=10^{12})$)를 공백으로 구분하여 출력한다.

배점

각 테스트케이스에 대해 득점은 다음과 같이 계산된다.

참가자가 제출한 출력 중 소비 에너지의 최솟값을 E_0 라고 하자. 제출한 출력이 문제의 조건을 만족하지 않을 경우, 점수는 0이다. 조건을 만족할 경우, 당신이 제출한 출력의 소비 에너지를 E 라고 할 때,

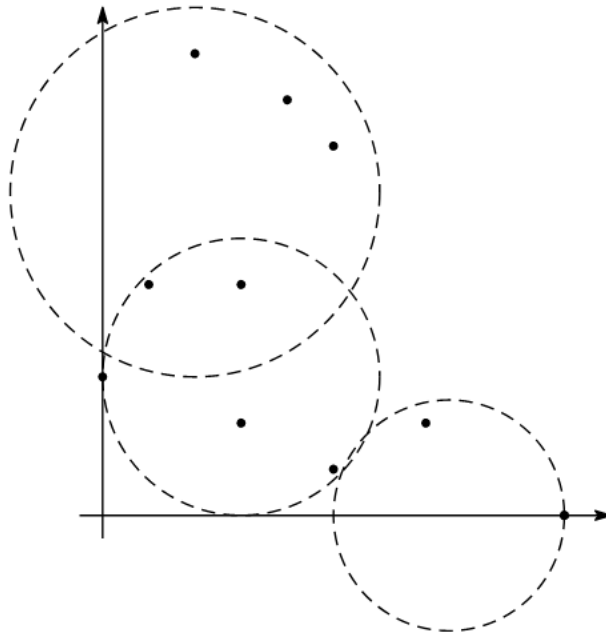
- $\frac{E}{E_0} > 1.5$ 인 경우, 0
- $\frac{E}{E_0} \leq 1.5$ 인 경우, $\left(4 \times \left(1.5 - \frac{E}{E_0}\right)^2\right) \times 20$ 을 소수점 첫째 자리에서 반올림한 값.

이 득점이 된다.

예제

Sample Input	Sample Output
10 3	200000 700000 1600000000000
0 300000	300000 300000 900000000000
500000 800000	750000 0 625000000000
700000 200000	
100000 500000	
400000 900000	
200000 1000000	
300000 500000	
300000 200000	
500000 100000	
1000000 0	

이 입출력은 다음 그림에 대응된다. 각 원은 좌표 (X_i, Y_i) 에서 거리 $\sqrt{E_i}$ 이하인 위치를 표시한다. 이 경우, 출력 에너지는 $160\,000\,000\,000 + 90\,000\,000\,000 + 62\,500\,000\,000 = 312\,500\,000\,000$ (3125억) 이다.



문제 5. 별자리

입력 파일: standard input
출력 파일: standard output
시간 제한: 1초
메모리 제한: 64MB

JOI 군은 별이 빛나는 밤하늘을 관찰하는 것을 사랑하는 소년이다. 매일 밤 밤하늘을 관찰해서, 각각 별이 어느 별자리에 속하는지 알아보는 것을 즐기고 있다.

어느 날 밤 JOI 군은 밤하늘을 관찰하면서 본 적 없는 별을 N 개 발견했다. 이 별이 어느 별자리에 위치하는지 신경 쓰인 JOI 군은 밤하늘의 사진을 찍어서 다음날 도서관에서 이 별들에 대해 조사했고, 이 별은 모두 별자리 A나 별자리 B에 속한다는 것과 이 별 중 몇 개가 어느 쪽의 별자리에 속하는지 알았다. 하지만, 그 이외의 별에 대해서는 어느 별자리에 속하는지 몰랐다. JOI 군은 별자리 A와 별자리 B를 구성하는 별의 집합으로 가능한 것이 몇 가지 있는지 궁금해졌다. 덧붙여서, 별의 크기는 매우 작으므로 점으로 봐도 좋다.

단, 별자리는 사진에 있는 한 개 이상의 별과 별 두 개를 잇는 선분 몇 개로 되어있고, 다음 조건을 만족한다. 별이 한 개만 있는 것도 올바른 별자리인 것에 주의하여라.

- 어떤 별자리에 속한 두 개의 별은, 사진에 있는 그 별자리에 속한 선분을 통해 서로 도달할 수 있다.
- 어떤 별자리에 속한 선분과 다른 별자리에 속한 선분은 서로 교차하지 않는다.

또한, JOI 군이 발견한 별들은 다음의 조건을 만족한다.

- 어떤 세 개의 별도, 사진에서 일직선 상에 존재하지 않는다.
- 모든 별은 별자리 A 혹은 별자리 B에 속하며, JOI 군이 발견한 별 이외에 별자리 A 또는 별자리 B에 속하는 별은 존재하지 않는다.

N 개의 별의 정보가 주어졌을 때, 별자리 A와 별자리 B를 구성하는 별의 집합으로 가능한 경우의 수를 $1\,000\,000\,007 (= 10^9 + 7)$ 로 나눈 나머지를 구하는 프로그램을 작성하여라.

제한

$2 \leq N \leq 100\,000$ JOI 군이 발견한 별의 수
 $0 \leq X_i \leq 10^9, 0 \leq Y_i \leq 10^9$ i 번째 별의 사진에서의 위치

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N 이 주어지며, JOI 군이 발견한 별의 수를 의미한다.
- 다음 N 개의 줄에는, 별의 정보가 주어진다. $i + 1$ ($1 \leq i \leq N$) 번째 줄에는, 세 개의 정수 X_i, Y_i, C_i 가 공백으로 구분되어 주어진다. X_i, Y_i 는 i 번째 별의 사진에서의 위치가 (X_i, Y_i) 라는 것을 의미한다. C_i 는 별 i 가 별자리 A와 별자리 B 중 하나에 속해있다는 것을 의미한다.
 - $C_i = 0$ 인 경우, i 번째 별이 어느 별자리에 속해 있는지 모른다는 것을 의미한다.
 - $C_i = 1$ 인 경우, i 번째 별이 별자리 A에 속해 있다는 것을 의미한다.
 - $C_i = 2$ 인 경우, i 번째 별이 별자리 B에 속해 있다는 것을 의미한다.

출력 형식

별자리 A와 별자리 B를 구성하는 별의 집합으로 가능한 경우의 수를 $1\,000\,000\,007 (= 10^9 + 7)$ 로 나눈 나머지를 첫째 줄에 출력하여라. 그런 별의 집합이 존재하지 않을 경우, 0을 출력하여라.

배점

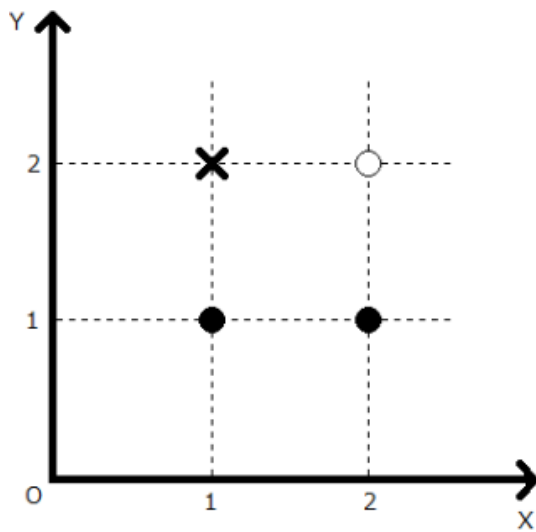
채점 데이터 중, 배점의 10%에 대해 $N \leq 10$ 을 만족한다.

채점 데이터 중, 배점의 50%에 대해 $N \leq 300$ 을 만족한다.

예제

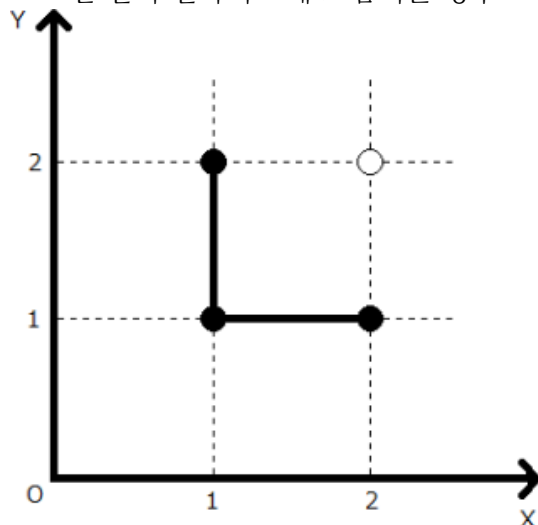
standard input	standard output
4 1 1 1 2 1 1 1 2 0 2 2 2	2

이 입력 예제는 다음의 그림에 대응된다. 검은색 점이 별자리 A를 구성하는 별들이고, 하얀색 점이 별자리 B를 구성하는 별들이며, ×표시는 어떤 별자리에 속하는지 모르는 별이다.

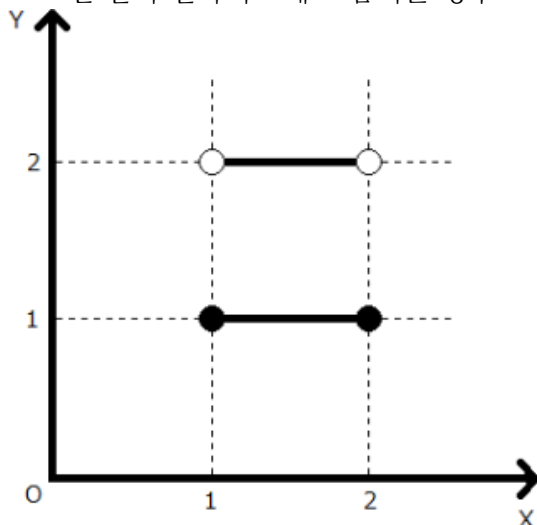


이 입력 예제에 대응하는 답은 3번 별이 별자리 A에 포함되는 경우와 별자리 B에 포함되는 경우 두 가지가 있으며, 각각의 경우 별자리 그림은 다음과 같다. 3번 별이 A에 포함되는 경우, 별자리 A의 선분을 그리는 방법이 여러 종류가 있지만 모두 한 종류로 셈에 주의하여라.

3번 별이 별자리 A에 포함되는 경우



3번 별이 별자리 B에 포함되는 경우



문제 6. 회전

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 64MB

정사각형 모양의 격자 칸이 세로로 N 행 가로로 N 열 있는 총 $N \times N$ 모양의 격자가 있다. 위에서 i 번째, 왼쪽에서 j 번째 격자 칸을 (i, j) 번 격자 칸이라고 부른다. (i, j) 번 격자 칸에는 문자 A_{ij} 가 쓰여 있다.

어떤 정사각형 영역을 반시계방향으로 90도 회전하는 것을 Q 번 반복한다. k 번째의 회전은 (I_k, J_k) 번 격자 칸이 가장 왼쪽 위인 $S_k \times S_k$ 개의 격자를 포함하는 정사각형을 회전한다.

최종 상태를 구해서 출력하여야.

제한

$2 \leq N \leq 1\,000$ 격자의 크기
 $1 \leq Q \leq 2\,000$ 회전의 회수

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N, Q 가 공백으로 구분되어 주어진다. N 은 격자의 크기이며, Q 는 회전의 회수를 의미한다.
- 다음 N 개의 줄에는 격자에 처음에 쓰인 문자가 주어진다. 이 중 i 번째 줄에는 길이 N 의 문자열이 주어진다. 이 문자열의 j 번째 문자는 A_{ij} 를 의미한다. 문자열은 알파벳 소문자만으로 이루어져 있다.
- 다음 Q 개의 줄에는 회전을 의미하는 정보가 주어진다. 이 중 k 번째 줄에는 정수 I_k, J_k, S_k ($1 \leq I_k \leq N - S_k + 1, 1 \leq J_k \leq N - S_k + 1, 2 \leq S_k \leq N$) 이 공백으로 구분되어 주어진다. 이는, k 번째 회전이 (I_k, J_k) 번 격자 칸이 가장 왼쪽 위인 $S_k \times S_k$ 개의 격자를 포함하는 정사각형을 회전한다는 것을 의미한다.

출력 형식

표준 출력으로, 최종 격자의 상태를 N 개의 줄로 출력하여야. 즉, i 번째 줄에는 N 개의 문자로 이루어진 문자열이고, j 번째 문자가 최종적으로 격자의 (i, j) 번 격자 칸에 쓰인 문자가 되도록 출력하여야.

배점

채점 데이터 중, 배점의 10%에 대해 $N \leq 100, Q \leq 100$ 을 만족한다.

예제

standard input	standard output
4 1 abcd efgh ijkl mnop 2 2 2	abcd egkh ifjl mnop

이 입력은, 다음과 같은 격자판을 의미한다.

```
abcd
efgh
ijkl
mnop
```

회전의 정보를 의미하는 첫째 줄은, “(2, 2)번 격자 격자 칸이 가장 왼쪽 위인 2×2 개의 격자를 포함하는 정사각형을 회전한다”는 것을 의미한다. 즉,

fg
jk

를 반시계방향으로 90도 회전해서,

abcd
egkh
ifjl
mnop

가 된다.

문제 7. 점치기

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 64MB

K 이사장은 점을 매우 좋아해서, 언제나 이런저런 점을 치고 있다. 오늘은 카드를 사용해서 올해 IOI에서 일본선수단의 미래를 점치기로 했다.

점을 치는 방법은 다음과 같다.

- 처음 카드를 세로로 M 행, 가로 N 열의 직사각형 모양으로 모두 앞면이 보이도록 배치한다.
- $i = 1, \dots, K$ 에 대해, “위에서 A_i 번째 행부터 B_i 번째 행까지, 왼쪽에서 C_i 번째 행부터 D_i 번째 열까지에 있는 모든 카드의 앞뒷면을 뒤집는다.”라는 조작을 한다. 즉, 위에서부터 a 번째 행, 왼쪽에서부터 b 번째 열에 있는 카드를 (a, b) 번 카드라고 하면, 각 i 에 대해 $A_i \leq a \leq B_i$ 이면서 $C_i \leq b \leq D_i$ 를 만족하는 (a, b) 번 카드를 모두 뒤집는다.
- 조작이 끝난 경우, 뒷면과 앞면이 되어있는 카드가 몇 장인지에 따라 점의 결과가 결정된다.

K 이사장은 카드를 뒤집는 횟수가 생각보다 많다는 것을 깨달았기 때문에, 카드를 실제로 사용해서 점을 두는 대신 조작이 끝났을 때 앞면인 카드가 몇 장인지만 구하기로 했다.

행의 수 M , 열의 수 N , 조작의 횟수 K 와 K 번의 조작이 주어졌을 때, 조작 후에 앞면인 카드의 수를 구하는 프로그램을 작성하여라.

제한

$1 \leq M \leq 1\,000\,000\,000 (= 10^9)$ 행의 수
 $1 \leq N \leq 1\,000\,000\,000 (= 10^9)$ 열의 수
 $1 \leq K \leq 100\,000$ 조작의 회수

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 M, N, K 가 공백으로 구분되어 주어지며, 카드가 M 행 N 열로 나열되어 있고, 조작을 K 번 한다는 것을 의미한다.
- $1 + i$ 번째 행 ($1 \leq i \leq K$) 에는 네 개의 정수 A_i, B_i, C_i, D_i ($1 \leq A_i \leq B_i \leq M, 1 \leq C_i \leq D_i \leq N$)가 주어지며, i 번째 조작이 위에서 A_i 번째 행부터 B_i 번째 행까지, 왼쪽에서 C_i 번째 행부터 D_i 번째 열까지에 있는 모든 카드의 앞뒷면을 뒤집는다는 것을 의미한다.

출력 형식

표준 출력으로, K 번의 조작 이후에 앞면이 되어 있는 카드의 장수를 첫째 줄에 출력하여라.

배점

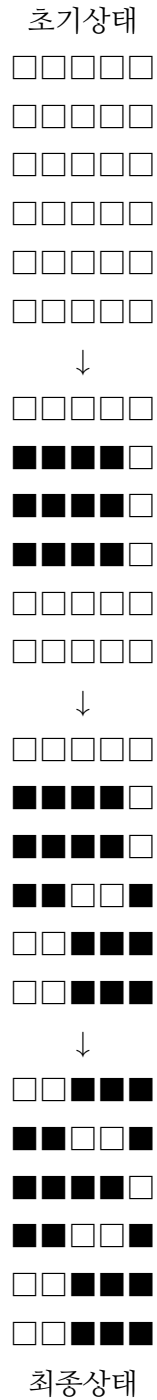
채점 데이터 중, 배점의 30%에 대해 $K \leq 3\,000$ 을 만족한다.

예제

standard input	standard output
6 5 3 2 4 1 4 4 6 3 5 1 2 3 5	11

이 예에서, $K = 3$ 번의 조작은 다음과 같이 진행된다.

앞면인 카드를 □, 뒷면인 카드를 ■라고 나타내면



최종상태에서 앞면인 카드의 수인 11을 출력한다.

문제 8. 켑거루

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 64MB

K 이사장은 켑거루에 흥미를 느껴서, 켑거루의 행동을 관찰하기로 했다. K 이사장은 N 마리의 켑거루를 관찰하고 있다. 켑거루에는 주머니가 하나씩 달려있다. 켑거루에는 번호 $1, 2, \dots, N$ 이 붙어있다. i 번 켑거루의 몸 크기는 A_i 이며, i 번 켑거루의 주머니 크기는 B_i 이다. 주머니 크기는 몸 크기보다 작다. ($A_i > B_i$) 처음에는 어떤 켑거루의 주머니 안에도 켑거루가 들어있지 않다. 켑거루는 다음의 조작을 **조작을 더 할 수 없을 때까지** 반복한다.

$A_i < B_j$ 를 만족하는 i 번 켑거루와 j 번 켑거루가 있어서, i 번 켑거루가 다른 켑거루의 주머니에 들어있지 않고, j 번 켑거루의 주머니 안에 어떤 켑거루도 들어있지 않는 두 켑거루가 존재할 때, i 번 켑거루는 j 번 켑거루 안에 들어간다. 이때, i 번 켑거루에 주머니 안에 있는 다른 켑거루가 있어도, j 번 켑거루가 다른 켑거루의 주머니 안에 있어도 상관없다. 이런 (i, j) 쌍이 여러 개 있으면, 어떤 쌍이 선택될지는 알 수 없다. i 번 켑거루 안에 다른 켑거루가 들어있을 경우, 안에 들어 있는 켑거루는 i 번 켑거루와 같이 이동한다.

켑거루의 몸 크기와 주머니 크기가 주어졌을 때, 최종 상태로 가능한 경우가 몇 가지인지를 $1\,000\,000\,007(=10^9+7)$ 로 나눈 나머지를 구하여라.

제한

$1 \leq N \leq 300$ 켑거루의 수
 $1 \leq B_i < A_i \leq 1\,000\,000\,000$ i 번째 켑거루의 주머니 크기와 몸 크기

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N 이 주어진다. N 은 켑거루의 수를 의미한다.
- 다음 N 개의 줄에는 켑거루의 정보가 주어진다. $i+1$ 번째 ($1 \leq i \leq N$) 줄에는 두 개의 정수 A_i, B_i 가 공백으로 구분되어 주어진다. A_i 는 i 번 켑거루의 몸 크기를, B_i 는 i 번 켑거루의 주머니 크기를 각각 의미한다.

출력 형식

표준 출력으로, 최종 상태로 가능한 경우가 몇 가지인지를 $1\,000\,000\,007(=10^9+7)$ 로 나눈 나머지를 첫째 줄에 출력하여라.

배점

채점 데이터 중, 배점의 50%에 대해 $N \leq 30$ 을 만족한다.

채점 데이터 중, 배점의 70%에 대해 $N \leq 70$ 을 만족한다.

예제

standard input	standard output
5 4 3 3 1 6 5 2 1 4 2	4

1, 2, 5번 캥거루는 3번 캥거루의 주머니 안에 들어갈 수 있다. 또한, 4번 캥거루는 1번 캥거루 혹은 3번 캥거루의 들어갈 수가 있고, 3번 캥거루는 다른 어떤 캥거루의 주머니에도 들어갈 수 없다. 그러므로, 최종 상태로 가능한 것은 다음의 네 종류이다.

- 4번 캥거루가 3번 캥거루의 주머니 안에 들어있다.
- 4번 캥거루는 1번 캥거루의 주머니 안에, 1번 캥거루는 3번 캥거루의 주머니 안에 들어있다.
- 4번 캥거루는 1번 캥거루의 주머니 안에, 2번 캥거루는 3번 캥거루의 주머니 안에 들어있다.
- 4번 캥거루는 1번 캥거루의 주머니 안에, 5번 캥거루는 3번 캥거루의 주머니 안에 들어있다.

standard input	standard output
20 7 6 7 3 10 1 7 2 10 7 10 7 8 6 3 2 5 4 7 2 3 2 10 9 9 4 7 2 8 6 5 4 8 6 7 4 10 5 9 3	21060

문제 9. 소코반

입력 파일: standard input
출력 파일: standard output
시간 제한: 2초
메모리 제한: 512MB

“소코반”은 오랫동안 사람들에게 사랑받아온 퍼즐이다. “소코반”은 세로 M 행, 가로 N 열의 격자에서 진행되는 게임이다. 격자 위에 있는 상자를 플레이어를 조작해서 목적지까지 옮기는 것이 목적이다. 이 문제에서는 상자가 한 개 있는 경우만 생각한다. 격자의 일부 격자 칸에는 벽이 있어서 플레이어, 상자, 목적지는 각각 벽이 아닌 한 격자 칸에 있다(플레이어나 상자를 격자 칸 밖으로 움직일 수는 없다). 플레이어는 다음 중 하나의 조작을 할 수 있다.

- 플레이어가 있는 격자 칸에 인접한 격자 칸 중 벽이 아닌 상자가 없는 격자를 한 칸 골라서, 플레이어가 그 칸으로 이동한다.
- 플레이어가 있는 격자 칸과 상자가 있는 격자 칸이 인접하면서, 플레이어의 반대쪽으로 상자와 인접한 벽이 아닌 격자 칸이 있으면, 상자는 해당 격자 칸으로 이동하며, 플레이어는 상자가 있는 격자 칸으로 이동한다.

여기서 격자와 격자가 인접한다는 것은 두 격자가 한 변을 공유한다는 것을 의미한다.

아래는 “소코반” 문제의 예이다. #은 벽, @은 플레이어, 0은 상자, X는 목표지점, .은 그 외의 격자 칸을 의미한다.

```
..#@.  
.X.0.  
##...#
```

이 상태에서, 다음의 조작을 통해 상자를 목적지에 놓을 수 있다.

1. 플레이어를 오른쪽으로 움직인다.
2. 플레이어를 아래쪽으로 움직인다.
3. 상자와 플레이어를 왼쪽으로 움직인다.
4. 상자와 플레이어를 왼쪽으로 움직인다.

한편, 다음 상태라면 상자를 목적지점까지 움직일 수 없다.

```
..#..  
.X.0.  
##.@#
```

당신은 격자의 벽의 위치와 목적지 지점이 정해져 있을 때, 플레이어와 상자를 배치해서 풀 수 있는 “소코반” 문제가 몇 종류 있는지 알고 싶다. 여기서 풀 수 있는 “소코반” 문제는, 조작을 몇 번 반복해서 목적지점으로 상자를 이동할 수 있는 초기상태를 의미한다. 단, 플레이어와 상자는 각각 벽이 아니면서 목적지점이 아닌 격자에 배치되어야 하며, 플레이어와 상자를 서로 다른 격자에 배치해야 한다.

격자의 크기와 벽의 위치와 목적지의 위치가 주어졌을 때, 풀 수 있는 “소코반” 문제가 몇 종류 있는지 구하는 프로그램을 작성하여라.

제한

$1 \leq M \leq 1\,000$ 격자의 세로 길이
 $1 \leq N \leq 1\,000$ 격자의 가로 길이

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 M, N 이 공백으로 구분되어 주어지며, 각각은 격자의 세로와 가로 길이이다.
- 다음 M 개의 줄에는 격자의 정보가 주어진다. 각 줄은 N 개의 문자로 되어있다. 각 문자는 $\#$ X $.$ 중 하나이며, $\#$ 은 벽, X 는 목적지, $.$ 은 그 외의 격자 칸(플레이어나 상자의 초기위치로 가능하다.)을 의미한다. 문자 X 는 정확히 한 번 주어진다.

출력 형식

표준 출력으로, 풀 수 있는 “소코반” 문제가 몇 종류 있는지 의미하는 정수를 첫째 줄에 출력하여야.

배점

채점 데이터 중, 배점의 20%에 대해 $M \leq 50, N \leq 50$ 을 만족한다.

예제

standard input	standard output
3 5 ..#.. .X.. ##..#	9

풀 수 있는 소코반 문제는 다음과 같이 9종류이다.

..#@. ..#.@ ..#.. ..#.. ..#.. ..#.. ..#@. ..#.@ ..#..
.X0.. .X0.. .X0@. .X0.@ .X0.. .X0.. .X.0. .X.0. .X.0@
##..# ##..# ##..# ##..# ##@.# ##.@# ##..# ##..# ##..#

standard input	standard output
2 3 .X. ...	0

이 예에서는, 풀 수 있는 “소코반” 문제를 만들 수 없다.

standard input	standard output
4 7 .### ##.##X.. ##.###	24

문제 10. 중화요리

입력 파일: standard input
출력 파일: standard output
시간 제한: 1.5초
메모리 제한: 64MB

K 이사장을 포함한 정보올림피아드일본위원회의 N 명 모두가 중화요리점에 갔다.

중화요리점의 식탁은 원탁이며, N 명의 자리가 같은 간격으로 떨어져 있다. 또한, 중앙에 요리가 위치한 회전대가 놓여있다. N 명의 위원은 N 개의 자리에 앉아 있으며, K 이사장을 1번으로 해서 반시계방향으로 $2, 3, \dots, N$ 의 위원 번호가 붙어있다. 위원회는 N 종류의 요리를 하나씩 주문해 각 요리를 회전대 위에 올렸다. 각 요리는 각 위원의 앞에 있으며, i 번 위원의 앞에 위치한 요리는 i 번 요리이다. K이사장 이외의 위원은 먹고 싶은 요리가 하나씩 정해져 있고, i 번 ($2 \leq i \leq N$) 위원이 먹고 싶은 요리는 A_i 번 요리이다.

회전대는 $(360/N)^\circ$ 단위로 시계방향 혹은 반시계방향 어느 방향으로든 회전할 수 있다. 예를 들어, 회전대를 반시계방향으로 한 단위만큼 회전시키면, K 이사장에 앞에는 N 번 요리가, i 번 ($2 \leq i \leq N$) 위원 앞에는 $i - 1$ 번 요리가 온다.

어떤 위원이 어떤 요리를 먹기 위해서는 그 요리가 그 위원의 앞에 오도록 회전대를 돌려야 한다.

정보올림피아드일본위원회에서 K 이사장은 존경받고 있기 때문에, 처음에 K 이사장이 회전대를 회전해서 k 번 ($1 \leq k \leq N$) 요리가 앞에 오도록 해서 요리를 먹는다.

K 이사장이 요리를 먹은 이후 K 이사장 이외의 위원들은 각각 자신이 먹고 싶은 요리가 앞에 오도록 회전대를 돌려서 그 요리를 먹는다. 단, K 이사장 이외의 위원이 회전대를 돌리는 순서는 어떤 순서여도 상관없다.

또한, 각 요리는 충분히 많기 때문에 요리가 바닥나는 경우는 없다고 한다.

K 이사장이 어떤 요리를 먹어도 상관없도록, 모두가 요리를 먹을 수 있게 회전대를 돌리는 횟수의 합이 최소가 되도록 K 이사장 이외의 위원이 회전대를 돌리는 순서를 각 k 에 대해서 정해두고 싶다.

i 번 위원이 먹고 싶어 하는 요리의 번호 A_i 가 각각 주어졌을 때, K 이사장이 먹는 k 번 요리 ($1 \leq k \leq N$) 각각에 대해 회전대를 돌리는 횟수의 최솟값을 구하여라. 회전대를 $(360/N)^\circ$ 돌리는 것을 한 번 돌린다고 한다.

제한

$2 \leq N \leq 100\,000$ 정보올림피아드일본위원회의 사람 수
 $1 \leq A_i \leq N$ i 번 위원이 먹고 싶은 요리

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 N 이 주어지며, 정보올림피아드일본위원회의 사람 수를 의미한다.
- 다음 $N - 1$ 개의 줄에는 각 위원이 먹고 싶은 요리의 정보가 주어진다. i 번째 ($2 \leq i \leq N$) 줄에는 정수 A_i 가 주어진다. 이는 i 번 위원이 먹고싶은 요리가 A_i 번 요리라는 것을 의미한다.

출력 형식

출력은 N 개의 줄로 되어있다. k 번째 ($1 \leq k \leq N$) 줄에는 K이사장이 k 번 요리를 먹었을 때, 회전대를 돌리는 횟수의 최솟값을 의미하는 정수를 출력하여라. 단, 회전대의 회전량은 $(360/N)^\circ$ 돌리는 것을 한 번으로 생각한다.

배점

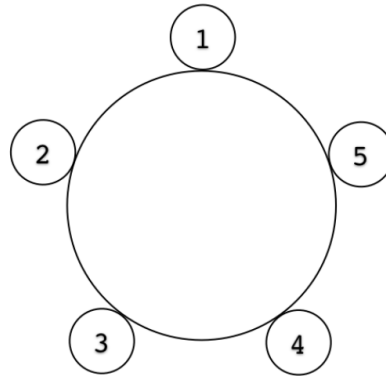
채점 데이터 중, 배점의 10%에 대해 $N \leq 10$ 을 만족한다.

채점 데이터 중, 배점의 40%에 대해 $N \leq 1\,000$ 을 만족한다.

예제

standard input	standard output
5	4
3	4
5	5
3	6
2	4

이 예에서, 원탁에 5명의 위원이 그림과 같이 앉아있다.



예를 들어, $k = 3$ 인 경우(K 이사장이 3번 요리를 먹는 경우)를 생각해 보면, 회전대를 돌리는 횟수가 최소가 되는 것은 다음과 같은 방법이다.

- K 이사장 (1번 위원)이 회전대를 시계방향으로 두 번 돌려서, 3번 요리를 먹는다.
- 3번 위원이 회전대를 돌리지 않고 5번 요리를 먹는다.
- 5번 위원이 회전대를 돌리지 않고 5번 요리를 먹는다.
- 2번 위원이 회전대를 반시계방향으로 한 번 돌려서, 3번 요리를 먹는다.
- 4번 위원이 회전대를 반시계방향으로 두 번 돌려서, 3번 요리를 먹는다.

이때 회전대를 돌리는 횟수는 총 $2 + 1 + 2 = 5$ 번이므로, 출력의 세 번째 줄에는 5를 출력한다.

문제 11. 복사 붙여넣기

입력 파일: standard input
출력 파일: standard output
시간 제한: 17초
메모리 제한: 512MB

텍스트 에디터의 제일 중요한 기능 중 하나로 복사 붙여넣기가 있다. JOI 회사는 복사 붙여넣기를 매우 고속으로 처리할 수 있는 텍스트 에디터의 개발을 시작했다. JOI 회사에 소속된 우수한 프로그래머인 당신은, 중요한 복사 붙여넣기 기능의 실제 구현을 담당하게 되었다. JOI 회사의 운명이 걸린 일이므로 정확하고 빠른 프로그램을 작성하고 싶다.

구체적인 사양은 다음과 같다. 우선, 파일의 내용을 문자열 S 라 하자. 이제 복사 붙여넣기가 N 번 일어난다. i 번째의 조작은 위치 A_i 부터 위치 B_i 까지의 문자열을 복사해서, 복사한 문자열을 원래 문자열의 위치 C_i 에 삽입한다. 여기서, 위치 x 는 앞에 부터 세어 x 개의 문자 직후를 의미한다(위치 0은 문자열의 가장 앞부분이다). 단, 조작 후의 문자열 길이가 M 을 넘을 경우, 길이가 M 이 되도록 오른쪽부터 차례로 문자를 삭제한다.

문자열 길이 상한 M , 처음 문자열 S , 조작 횟수 N 과 N 번의 복사 붙여넣기 조작이 주어졌을 때, 조작 후의 문자열을 출력하는 프로그램을 작성하여라.

제한

$1 \leq M \leq 1\,000\,000$ 문자열의 길이 상한
 $1 \leq N \leq 1\,000\,000$ 조작 횟수

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 M 이 주어지며, 문자열 길이 상한을 의미한다.
- 둘째 줄에는 문자열 S 가 주어지며, 처음 문자열을 의미한다. S 는 알파벳 소문자로 되어있고, 길이는 1 이상 M 이하이다.
- 셋째 줄에는 정수 N 이 주어지며, 조작 횟수를 의미한다.
- $3+i$ 번째 ($1 \leq i \leq N$) 줄에는 정수 A_i, B_i, C_i 가 공백으로 구분되어 주어지며, i 번째 조작은 위치 A_i 부터 위치 B_i 까지의 문자열을 복사해서, 복사한 문자열을 원래 문자열의 위치 C_i 에 삽입한다는 것을 의미한다. i 번째 문자열의 조작 직전의 문자 길이를 L_i 라고 하면, $0 \leq A_i < B_i \leq L_i$ 와 $0 \leq C_i \leq L_i$ 를 만족한다.

출력 형식

표준 출력으로, N 번의 조작 이후 문자열을 첫째 줄에 출력한다.

배점

채점 데이터 중, 배점의 10%에 대해 $M \leq 100\,000, N \leq 100\,000$ 을 만족한다.

예제

standard input	standard output
18 copypaste 4 3 6 8 1 5 2 4 12 1 17 18 0	acyppypastoopypypyp

- 처음 문자열은 `copypaste`이다.
- 첫 번째 조작에서, 위치 3부터 위치 6까지의 문자열 `ypa`를 복사해서, 위치 8에 삽입해서, 문자열은 `copypastypae`가 된다.
- 두 번째 조작에서, 위치 1부터 위치 5까지의 문자열 `opyp`를 복사해서, 위치 2에 삽입해서, 문자열은 `coopypypastypae`가 된다.
- 세 번째 조작에서, 위치 4부터 위치 12까지의 문자열 `ypypast`를 복사해서, 위치 1에 삽입해서, 문자열은 `cyppypastoopypypastypae`가 되지만, 길이가 $M = 18$ 을 넘으므로, 오른쪽에서 부터 문자열을 삭제해서, 문자열 `cyppypastoopypypa`가 된다.
- 네 번째 조작에서, 위치 17부터 위치 18까지의 문자열 `a`를 복사해서, 위치 1에 삽입해서, 문자열은 `acyppypastoopypypa`가 되지만, 길이가 $M = 18$ 을 넘으므로, 오른쪽에서 부터 문자열을 삭제해서, 문자열 `acyppypastoopypyp`가 된다.

standard input	standard output
100 joi 3 0 1 0 3 4 3 2 3 3	jjooii

문제 12. 초대

입력 파일: standard input
출력 파일: standard output
시간 제한: 3초
메모리 제한: 128MB

20XX년, 드디어 IOI가 JOI 나라의 JOI 마을에서 열리게 되어, 이를 기념하기 위한 파티가 열린다. JOI 마을에는 A 마리의 개($1, 2, \dots, A$ 의 번호가 붙어있다.)와, B 마리의 고양이($1, 2, \dots, B$ 의 번호가 붙어있다.)가 있다. 당신은 이 $A + B$ 마리 모두를 파티에 초대하고 싶다.

개와 고양이 사이에는 N 개의 **사이좋은 그룹**이 있다. i 번째의 사이좋은 그룹은 번호가 P_i 이상 Q_i 이하인 개 $Q_i - P_i + 1$ 마리와 번호가 R_i 이상 S_i 이하인 고양이 $S_i - R_i + 1$ 마리로 구성되어있다. 또한, 각 사이좋은 그룹에는 **사이좋은 정도**라는 정수가 정해져 있다. i 번째 사이좋은 그룹의 사이좋은 정도는 T_i 이다. 한 마리의 개나 한 마리의 고양이가 여러 사이좋은 그룹에 들어 있을 수도 있고, 어떤 사이좋은 그룹에도 들어 있지 않은 개나 고양이가 있을 수도 있다.

당신은 번호가 C 인 개와 매우 사이가 좋아, 그 개를 초대하는 데에 성공했다. 당신은 다음과 같은 행동을 반복해서 남은 개나 고양이를 초대하려 한다.

- $A + B$ 마리 모두를 이미 초대한 경우에는 종료한다.
- 아직 초대하지 않은 개나 고양이 각각에 대해, 초대할 때 **행복도**를 구한다. 행복도는 그 개나 고양이가 들어 있는 사이좋은 그룹 중에서, 이미 초대에 성공한 개나 고양이가 한 마리 이상 들어 있는 사이좋은 그룹의 사이좋은 정도의 최댓값이다. 그런 사이좋은 그룹이 존재하지 않은 경우에 행복도는 0이다.
- 행복도가 최대인 개나 고양이를 고른다. 그런 개나 고양이가 여럿 있으면 개를 우선시하고, 그래도 여럿 있으면 번호가 작은 동물을 고른다.
- 선택한 개나 고양이의 행복도가 0인 경우 초대는 실패한다. 그렇지 않은 경우, 선택된 개나 고양이를 초대하는 데에 성공한다.

당신은, 이 초대 방법이 어떤 결과가 될 지 미리 계산하고 싶어졌다.

개의 수 A , 고양이의 수 B , 당신과 매우 사이가 좋은 개의 번호 C 와 N 개의 사이좋은 그룹의 정보가 주어졌을 때, $A + B$ 마리 모두를 초대하는데 성공하는지 판단하고, 성공한 경우에는 각 단계에서 초대한 개나 고양이의 행복도 총합이 얼마인지 구하는 프로그램을 작성하여라.

제한

$1 \leq A \leq 1\,000\,000\,000$	개의 수
$1 \leq B \leq 1\,000\,000\,000$	고양이의 수
$1 \leq N \leq 100\,000$	사이좋은 그룹의 수
$1 \leq T_i \leq 1\,000\,000\,000$	사이좋은 정도

입력 형식

다음 정보가 표준 입력으로 주어진다.

- 첫째 줄에는 정수 A, B, C ($1 \leq C \leq A$) 가 공백으로 구분되어 주어지고, 각각 개의 수, 고양이의 수, 당신과 매우 사이가 좋은 개의 번호를 의미한다.
- 둘째 줄에는 정수 N 이 주어지고 사이좋은 그룹의 수를 의미한다.
- $2 + i$ 번째 ($1 \leq i \leq N$) 줄에는 정수 P_i, Q_i, R_i, S_i, T_i ($1 \leq P_i \leq Q_i \leq A, 1 \leq R_i \leq S_i \leq B$) 가 공백으로 구분되어 주어지고, i 번째 사이좋은 그룹은 번호가 P_i 이상 Q_i 이하인 개와 번호가 R_i 이상 S_i 이하인 고양이로 구성되어있다는 것을 의미한다.

출력 형식

표준 출력으로, 다음 정수 중 하나를 첫째 줄에 출력하여야라.

- $A + B$ 마리 모두를 초대하는 데에 성공한 경우, 각 단계에서 선택된 개나 고양이의 행복도 총합을 의미하는 정수.
- 초대가 도중에 실패했을 경우, 정수 -1 .

배점

채점 데이터 중, 배점의 30%에 대해 $A \leq 1\,000$, $B \leq 1\,000$, $N \leq 2\,000$ 을 만족한다.

채점 데이터 중, 배점의 50%에 대해 $N \leq 2\,000$ 을 만족한다.

예제

standard input	standard output
5 6 3 4 2 4 1 3 20 1 2 2 4 40 4 5 2 3 30 4 4 4 6 10	280

이 예에서, 개나 고양이는 다음과 같이 초대된다.

- 당신은 3번 개를 초대했다.
- 행복도를 계산해 보면, 2번 개: 20, 4번 개: 20, 1번 고양이: 20, 2번 고양이: 20, 3번 고양이: 20, 다른 초대를 받지 않은 개나 고양이는: 0이다. 당신은 2번 개를 골라서, 초대에 성공한다.
- 행복도를 계산해 보면, 1번 개: 40, 4번 개: 20, 1번 고양이: 20, 2번 고양이: 40, 3번 고양이: 40, 4번 고양이: 40, 다른 초대를 받지 않은 개나 고양이는: 0이다. 당신은 1번 개를 골라서, 초대에 성공한다.
- 위와 같이 초대가 계속되어, 다음의 표에 나타난 차례대로 모든 개나 고양이를 초대한다.

동물	번호	행복도
개	3	—
개	2	20
개	1	40
고양이	2	40
고양이	3	40
고양이	4	40
개	4	30
개	5	30
고양이	1	20
고양이	5	10
고양이	6	10

표의 “행복도” 열의 값은, 해당 개나 고양이를 초대 할 때 행복도를 의미한다. 총합인 280을 출력한다.

standard input	standard output
10 10 1 2 1 5 1 5 3 6 10 6 10 4	-1

이 예에서는 1번 개, 2번 개, 3번 개, 4번 개, 5번 개, 1번 고양이, 2번 고양이, 3번 고양이, 4번 고양이, 5번 고양이 10마리를 초대할 후 선택된 6번 개의 행복도가 0이므로, 초대가 중간에 실패한다.