2024

GIST ALGORITHM MASTERS

Official Problem Set

주관 알고마 준비위원회 (GIST X 전남대학교)

후원 GIST AI대학원 AI Graduate School

GIST 연구혁신센터

STARTLINK

aws

문제 목록

문제지에 있는 문제가 총 9문제가 맞는지 확인하시기 바랍니다.

A GIST 찍기

B 커플링 매치

C FOCUS

D 이진수 격자

E ChatGPT 만들기

F 유리병 속 무한히 터지는 기포

G 돈 복사

H 기초마법학

I DevNight 운영

모든 문제의 메모리 제한은 1GB로 동일합니다.

문제는 **만점을 받기 쉬운 문제부터 오름차순**으로 정렬되어 있습니다. 만점을 받기 어려운 문제 중에서도 부분점수를 받기는 쉬운 문제가 있을 수 있으므로, 모든 문제를 읽어볼 것을 권장합니다.

자세한 대회 규칙이나 FastIO (빠른 입출력) 사용법 등의 정보는 대회 플랫폼의 **유의사항** 란을 참조하세요.

문제 A. GIST 찍기

시간 제한	1 초
메모리 제한	1024 MB

GIST 학생 홍보대사 지온나래는 GIST를 홍보하기 위해 큰 현수막을 준비하고 있다. 현수막에는 다음과 같은 그림이 들어갈 것이다.

```
G...
.I.T
..S.
```

정수 K가 주어지면, 주어진 그림을 K배 확대한 그림을 예제 출력과 같이 출력하시오.

주어진 그림을 K배 확대한다는 것은 그림을 하나의 행렬로 보고 이를 확대한다는 것이다. 자세한 정의는 다음과 같다. 가로 길이가 M이고 세로 길이가 N인 그림은 $N \times M$ 행렬 $X = (x_{ij})$로 나타낼 수 있으며, 이 행렬 X를 K배 확대한 행렬 Y는 $KN \times KM$ 행렬로, 다음과 같이 정의된다.

$$Y = (y_{ij}), y_{ij} = x_{\lfloor \frac{i+K-1}{K} \rfloor \lfloor \frac{j+K-1}{K} \rfloor}$$

입력

첫째 줄에 정수 K가 주어진다. ($1 \leq K \leq 100$)

출력

주어진 그림을 K배 확대한 그림을 예제 출력과 같이 출력한다.

서브태스크 1 (점수: 25)

$K = 1$

서브태스크 2 (점수: 75)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
1	G... .I.T ..S.
5	GGGGG..... GGGGG..... GGGGG..... GGGGG..... GGGGG..... .....IIIII.....TTTTT .....IIIII.....TTTTT .....IIIII.....TTTTT .....IIIII.....TTTTT .....IIIII.....TTTTT .....SSSSS..... .....SSSSS..... .....SSSSS..... .....SSSSS..... .....SSSSS.....

문제 B. 커플링 매치

시간 제한	1 초
메모리 제한	1024 MB

종현이는 학교에 커플이 얼마나 많은지 확인하고자 한다.

커플은 왼손 약지에 남들과 구별되는 반지를 커플링으로 착용하고 있다는 사실을 알게 된 종현이는, 모든 재학생의 왼손 약지를 확인하기로 했다. 특징이 같은 반지를 착용한 사람이 정확히 두 명이라면, 종현이는 두 명을 커플 의심군으로 판단한다.

주어진 입력에 대해 종현이가 커플 의심군으로 판단할 사람들을 구하시오.

입력

첫째 줄에 종현이가 왼손 약지를 확인할 사람의 수 N이 주어진다. ($1 \le N \le 1\,000$)

둘째 줄부터 N개 줄에 걸쳐 사람의 이름 p_i와 반지의 특징 s_i가 공백으로 구분되어 주어진다. 만약 그 사람이 반지를 착용하고 있지 않다면 반지의 특징 대신 `-`가 주어진다. p_i와 s_i는 길이가 50 이하인 문자열이다.

모든 사람의 이름 p_i는 서로 다르고, 모든 사람의 이름과 반지의 특징은 영문자 알파벳과 숫자, 언더스코어(`_`)의 조합으로 이루어져 있다.

출력

첫째 줄에 커플 의심군 쌍의 수를 출력한다.

커플 의심군이 한 쌍 이상 존재한다면, 둘째 줄부터 순서와 상관없이 각 줄에 한 쌍씩, 공백으로 구분하여 각 쌍의 이름을 출력한다.

서브태스크 1 (점수: 25)

동일한 반지의 특징은 세 번 이상 등장하지 않는다.

서브태스크 2 (점수: 75)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
6 jonghyeon - minkyu twisted_silver_matte hyegeun thin_plated_gold somin text_engraved_light_blue daeun twisted_silver_matte jiyoung thin_plated_gold	2 minkyu daeun hyegeun jiyoung
6 eardrum_buster motdoen_goyangi ksap motdoen_goyangi beenzino chanel bongjunho chanel poison motdoen_goyangi bts chanel	0

이 페이지는 공백입니다

문제 C. FOCUS

시간 제한	1 초
메모리 제한	1024 MB

한 공장에서 인공지능 로봇이 오작동을 일으켜 폭주하고 있다. 로봇의 작동을 멈추기 위해서는 주어진 8개의 키 중 어느 것이 로봇을 정지시키는 키인지 찾아야 한다. 그러나 로봇이 그 키마저도 섞기 시작했다. 섞는 속도가 매우 빨라 인간의 시력으로는 키의 움직임을 추적할 수 없다. 다행히 당신은 로봇이 키를 섞는 데 사용하는 알고리즘과 로봇을 정지시키는 키의 처음 위치를 알고 있다.

각 키에는 0번부터 7번까지의 번호가 매겨져 있고, 이들은 각각 서로 다른 점 $P_0, P_1, \cdots, P_7$ 위에 놓여 있다. 로봇은 미리 주어진 길이 N의 수열을 하나씩 순서대로 읽으며 키를 섞는다. 이때 i번 키와 j번 키$(i < j)$의 위치를 바꾸는 명령은 정수 $2^i + 2^j$로 인코딩된다. i번 키와 j번 키의 위치를 바꾼다는 것은, i번 키가 점 P_k에 놓여 있고 j번 키가 점 P_l에 놓여 있을 때 i번 키를 점 P_l 위치로 옮기고 j번 키를 점 P_k 위치로 옮긴다는 것이다.

로봇이 읽는 수열 안에는 유효하지 않은 명령도 존재할 수 있다. 유효하지 않은 명령 x는 어떤 두 정수 $i, j (0 \le i < j \le 7)$에 대해서도 $x = 2^i + 2^j$를 만족시키지 않는 명령이다. 로봇이 유효하지 않은 명령을 인식하면 그 명령을 무시하고 다음 명령으로 넘어간다.

로봇을 정지시키는 키의 번호와 명령들의 수열이 주어지면, 로봇을 정지시키는 키는 $P_0, P_1, \cdots, P_7$ 중 몇 번 점에 위치하는지 구하시오.

입력

첫째 줄에 로봇이 읽는 명령들의 수열의 길이 N이 주어진다. $(1 \le N \le 200000)$

둘째 줄에 로봇이 읽는 명령들의 수열 $a_1, a_2, \cdots, a_N$이 공백으로 구분되어 주어진다. $(0 \le a_i < 256 = 2^8)$

셋째 줄에 로봇을 정지시키는 키의 번호 K가 주어진다. $(0 \le K \le 7)$

출력

로봇을 정지시키는 키가 점 P_t $(0 \le t \le 7)$ 위에 존재할 때, t의 값을 출력한다.

서브태스크 1 (점수: 20)

$0 \le a_i < 4 = 2^2$

서브태스크 2 (점수: 30)

모든 명령이 유효하다.

서브태스크 3 (점수: 50)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
7 130 72 130 17 96 66 6 5	3
7 20 7 68 9 144 156 40 6	4
7 111 49 235 228 172 77 151 2	2

노트

예제 1에서 로봇은 각 명령에 따라 다음과 같이 움직인다.

- $130 = 2^7 + 2^1$ 이므로, 1번 키와 7번 키의 위치가 바뀐다.
- $72 = 2^6 + 2^3$ 이므로, 3번 키와 6번 키의 위치가 바뀐다.
- $130 = 2^7 + 2^1$ 이므로, 1번 키와 7번 키의 위치가 바뀐다.
- $17 = 2^4 + 2^0$ 이므로, 0번 키와 4번 키의 위치가 바뀐다.
- $96 = 2^6 + 2^5$ 이므로, 5번 키와 6번 키의 위치가 바뀐다.
- $66 = 2^6 + 2^1$ 이므로, 1번 키와 6번 키의 위치가 바뀐다.
- $6 = 2^2 + 2^1$ 이므로, 1번 키와 2번 키의 위치가 바뀐다.

다섯 번째 명령에서 5번 키와 6번 키의 위치가 바뀌는데, 두 번째 명령 때문에 6번 키는 처음에 3번 키가 위치했던 곳에 가 있다. 따라서 정답은 3이다.

예제 2에서 명령 $a_2 = 7$은 유효하지 않은 명령이다.

문제 D. 이진수 격자

시간 제한	1 초
메모리 제한	1024 MB

각 칸에 0 또는 1이 적혀 있는 $N \times N$ 격자가 있다. 인선이는 1행 1열에서 출발해 N행 N열까지 이동하는데, 오른쪽(열 번호가 증가하는 방향) 또는 아래(행 번호가 증가하는 방향)로만 한 칸씩 이동할 수 있다. 인선이는 어떤 칸에 방문할 때마다 그 칸에 적힌 문자를 자신이 갖고 있는 문자열의 뒷부분에 덧붙인다. 예를 들어, 주어진 그림에서 인선이가 1행 1열에서 출발하여 순서대로 오른쪽, 아래, 오른쪽으로 한 칸씩 이동한다면 인선이가 얻게 될 문자열은 `1101`이다.

1	**1**	**0**	**0**
1	**0**	**1**	**0**
0	**1**	**1**	**1**
1	**0**	**0**	**0**

N행 N열에 도착하면 인선이는 자신이 갖고 있는 문자열을 이진수로 해석한 값 M을 계산한다. 예를 들어, 인선이가 가진 문자열이 `1101`일 경우 $M = 13$이다. 인선이가 계산하게 될 M의 최댓값을 구하시오.

입력

첫째 줄에 격자의 크기를 의미하는 정수 N이 주어진다. $(2 \le N \le 30)$

둘째 줄부터 N개의 줄에 걸쳐 한 줄에 N개의 정수가 공백으로 구분되어 주어진다. 이 정수는 반드시 0 또는 1이다. $(i+1)$번째 줄에 주어진 j번째 수는 i행 j열에 적힌 격자의 칸에 적힌 수를 의미한다.

출력

M의 최댓값을 출력한다. **정답이 32비트 정수 범위를 넘을 수 있음에 주의하시오.**

서브태스크 1 (점수: 20)

$N \le 5$

서브태스크 2 (점수: 30)

행 번호와 열 번호가 서로 다른 칸에는 모두 0이 적혀 있다.

서브태스크 3 (점수: 50)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 1 0 1 0 0 0 0 1 0	20
2 1 0 0 1	5
5 1 0 0 0 1 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 1	289

노트

예제 1에서 인선이가 오른쪽, 오른쪽, 아래, 아래로 순서대로 움직이면 문자열 `10100`을 얻고, 이는 20의 이진수 표현법이다.

예제 3에서 인선이가 오른쪽, 오른쪽, 아래, 오른쪽, 오른쪽, 아래, 아래, 아래로 순서대로 움직이면 문자열 `100100001`을 얻고, 이는 289의 이진수 표현법이다.

문제 E. ChatGPT 만들기

시간 제한	1 초
메모리 제한	1024 MB

GIST AI대학원에 재학 중인 지원이는 원칙적으로는 집에서 가정용 ChatGPT 정도는 쉽게 만들 수 있어야 한다. 하지만 지원이는 귀찮기 때문에 딥 러닝을 사용한 언어 모델 대신에 기본 NLP 기술인 n-gram 모델을 약간 수정하여 사용할 것이다.

지원이의 언어 모델은 문자 하나가 주어지면 훈련 코퍼스를 참조하여 다음 문자를 예측해 낸다. 훈련 코퍼스는 문자열 여러 개로 이루어져 있다. 어떤 문자 c가 주어지면, 지원이의 언어 모델은 훈련 코퍼스에 있는 모든 문자 c를 검색한 다음, 훈련 코퍼스에서 c 바로 다음 위치에 가장 많이 나오는 문자 c'를 찾아서 c 다음 문자가 c'일 것이라고 예측한다. $c = c'$일 수도 있음에 주의하라. 만약 그런 문자가 여러 개 있다면, 아스키코드가 가장 작은 것을 선택한다. 만약 훈련 코퍼스에 문자 c가 존재하지 않는다면, 언어 모델이 다음 문자를 예측할 수 없어 오류가 발생한다.

훈련 코퍼스는 문장의 시작을 나타내는 문자 [와 문장의 끝을 나타내는 문자], 알파벳 소문자 a-z, 그리고 띄어쓰기를 대체하는 문자 -로만 구성되어 있다. 훈련 코퍼스 내의 모든 문장은 [로 시작하고]로 끝나며 길이가 3 이상이다. 문자 [는 반드시 문장의 시작에만 등장하고, 문자]는 반드시 문장의 끝에만 등장한다.

언어 모델을 사용하여 문장을 생성하는 방법은 다음과 같다. 먼저 언어 모델에 문장의 시작을 나타내는 문자 [을 입력한다. 그다음 언어 모델이 다음 문자 c_1을 예측하면, 생성하고 있는 문장의 끝에 예측한 문자를 추가한다. 그다음 문자 c_1을 다시 언어 모델에 입력해 다음 문자 c_2를 예측한다. 이를 반복한다. 만약 반복 과정에서 문장의 끝을 나타내는 문자]가 등장하면 반복을 끝내고 예측을 종료한다. 그렇지 않다면, 다음 문자를 예측하는 과정은 무한히 반복된다. 훈련 코퍼스가 공집합이 아닌 한, 이 과정에서 언어 모델이 다음 문자를 예측할 수 없는 오류가 발생하지 않음을 증명할 수 있다.

훈련 코퍼스가 주어지면, 지원이와 함께 이 모델이 생성하는 문자열의 부분 문자열을 추측해 보자.

입력

첫째 줄에 세 정수 N, K, M이 공백으로 구분되어 주어진다. 이는 훈련 코퍼스가 총 N개의 문장으로 이루어져 있고, 프로그램은 이 훈련 코퍼스로 훈련한 언어 모델이 생성한 문자열의 K번째 문자부터 $K+M-1$번째 문자까지를 출력해야 한다는 것을 의미한다. ($1 \le N \le 100\,000$, $1 \le K \le 10^{18}$, $1 \le M \le 1\,000$)

둘째 줄부터 N개의 줄에 걸쳐 훈련 코퍼스에 들어 있는 문장들이 문자열의 형태로 한 줄에 하나씩 주어진다. 각 줄은 [로 시작하고]로 끝나며 길이가 3 이상이다. 문자 [는 반드시 각 줄의 시작에만 등장하고, 문자]는 반드시 각 줄의 끝에만 등장한다.

훈련 코퍼스에 존재하는 모든 문자의 길이의 합은 300 000을 넘지 않는다.

출력

입력된 훈련 코퍼스로 훈련한 언어 모델이 생성한 문자열의 K번째 문자부터 $K+M-1$번째 문자까지를 출력한다. i번째 문자를 출력할 때, 만약 언어 모델이 생성한 문자열의 길이가 짧아 i번째 문자가 존재하지 않는다면, 문자 .를 대신 출력한다.

노트

문자 -의 아스키코드는 45, [의 아스키코드는 91,]의 아스키코드는 93, a의 아스키코드는 97이다.

서브태스크 1 (점수: 25)

$K \le 100\,000$

서브태스크 2 (점수: 75)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 1 15 [i-love-you] [my-name-is-james] [gist-algorithm-masters]	[gis].....

노트

예제 1에서 언어 모델이 문장을 생성하는 방식은 다음과 같다.

언어 모델에 문장의 시작을 나타내는 문자 `[`를 입력한다. 훈련 코퍼스에서 문자 `[` 바로 다음 위치에 등장하는 문자는 `g` 1개, `i` 1개, `m` 1개이다. 따라서 `g`, `i`, `m`이 가장 많이 등장하므로 이들 중 아스키 코드가 가장 작은 `g`를 다음 문자로 선택한다.

언어 모델에 `g`를 입력한다. 훈련 코퍼스에서 `g` 바로 다음 위치에 등장하는 문자는 `i` 1개, `o` 1개이다. 따라서 `i`, `o`가 가장 많이 등장하므로 이들 중 아스키 코드가 가장 작은 `i`를 다음 문자로 선택한다.

언어 모델에 `i`를 입력한다. 훈련 코퍼스에서 `i` 바로 다음 위치에 등장하는 문자는 `-` 1개, `s` 2개, `t` 1개이다. 따라서 `s`가 가장 많이 등장하므로 `s`를 다음 문자로 선택한다.

언어 모델에 `s`를 입력한다. 훈련 코퍼스에서 `s` 바로 다음 위치에 등장하는 문자는 `-` 1개, `]` 2개, `t` 2개이다. 따라서 `]`, `t`가 가장 많이 등장하므로 이들 중 아스키 코드가 가장 작은 `]`를 다음 문자로 선택한다.

언어 모델이 문장의 끝을 나타내는 문자 `]`를 선택했으므로 더 이상 예측을 진행하지 않는다. 따라서 그 이후 자리를 모두 `.`로 채운다.

문제 F. 유리병 속 무한히 터지는 기포

시간 제한	3 초
메모리 제한	1024 MB

너에게서는 멸종된 과일 향기가 난다
투룸 신축 빌라 보증금 이천에 월세 구십, 어떻게 해야 너를 웃길 수 있을까 하는 생각, 두 시간 동안의 폭우, 일주일 동안의 아침, 유리병 속 무한히 터지는 기포

고선경, [샤워젤과 소다수]

세로로 길쭉한 유리병 하나가 있다. 이 속에서 기포가 무한히 터지고 있다.

과학자 정우가 이 현상에 호기심을 느끼고 계속해서 관찰한 결과 다음과 같은 사실을 알아냈다. 먼저, 유리병을 세로로 N등분하여 가장 아래에 위치한 1번째 구역부터 가장 위에 위치한 N번째 구역까지 총 N개의 구역으로 나눌 수 있다. 각 구역마다 최대 M개의 기포가 존재할 수 있다. 처음에는 어느 구역에도 기포가 존재하지 않고, 1초마다 유리병 내에서 다음 두 현상이 순서대로 번갈아 가며 일어난다.

- **기포 생성**: 유리병 내의 구역 하나에서 기포가 생성된다. i번째 기포 생성이 일어날 때, y_i번째 구역에서 k_i개의 기포가 생성된다. 만약 그 구역에서 원래부터 존재하던 기포 수와 새로 생성된 기포 수의 합이 M을 넘으면, M개만 남고 나머지 기포는 모두 사라진다.
- **기포 이동**: 모든 구역에서 기포들이 동시에 위 또는 아래로 이동한다. i번째 구역에 a_i개의 기포가 존재한다면, $\left\lfloor \frac{a_i}{5} \right\rfloor$개의 기포가 바로 위 구역으로 이동하고, $\left\lfloor \frac{a_i}{5} \right\rfloor$개의 기포가 바로 아래 구역으로 이동하며, $\left\lfloor \frac{a_i}{5} \right\rfloor$개의 기포는 제자리에 남아 있고, 나머지 기포는 사라진다. 만약 1번째 구역의 아래 또는 N번째 구역의 위로 움직이려 하는 기포가 있을 경우 그 기포는 사라진다. 기포 이동 후 각 구역마다 그 구역의 기포 수가 M을 넘으면, M개만 남고 나머지 기포는 모두 사라진다.

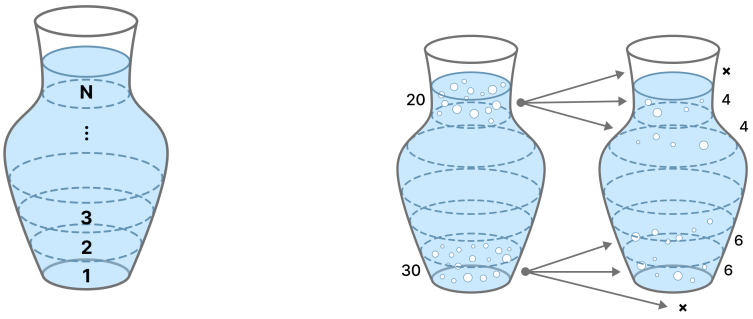


그림 1: (a) 유리병을 N개의 구역으로 나눈 모습. (b) 1번째 구역과 N번째 구역에서 각각 기포 이동이 일어나는 모습.

정우는 유리병 속 기포의 움직임을 모델링하는 프로그램을 만들려 한다. t초가 지난 후, 즉 기포 생성과 기포 이동이 각각 t번 일어난 후 유리병 속에 남은 모든 기포의 수를 $S(t)$라 하자. 관찰을 진행한 시간 T가 주어질 때, $S(1), S(2), \cdots, S(T)$의 값을 구하시오.

입력

첫째 줄에 유리병을 이루는 구역의 수 N, 각 구역마다 존재할 수 있는 최대 기포의 수 M, 관찰을 진행한 시간 T가 공백으로 구분되어 주어진다. ($1 \le N \le 100\,000$, $1 \le M \le 100\,000$, $1 \le T \le 50\,000$)

둘째 줄부터 T 개의 줄에 걸쳐, i번째 기포 생성에 대해 기포가 생성되는 구역의 번호 y_i와 그 구역에서 생성되는 기포의 수 k_i가 공백으로 구분되어 주어진다. ($1 \le y_i \le N$, $0 \le k_i \le M$)

출력

T 개의 줄에 걸쳐, i번째 줄에 $S(i)$의 값을 출력한다.

서브태스크 1 (점수: 20)

$N, T \le 1000$

서브태스크 2 (점수: 30)

모든 i에 대해 $y_i = 1$이다.

서브태스크 3 (점수: 50)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
6 100 6 3 20 2 40 4 60 3 80 1 100 5 100	12 24 44 66 73 92
6 100 2 1 100 1 100	40 52

예제 1에서 매초마다 각 구역에 존재하는 기포의 수는 다음과 같다.

1초가 지난 후 각 구역마다 존재하는 기포의 수는 $[0, 4, 4, 4, 0, 0]$이다.

2초가 지난 후 각 구역마다 존재하는 기포의 수는 $[8, 8, 8, 0, 0, 0]$이다.

3초가 지난 후 각 구역마다 존재하는 기포의 수는 $[2, 3, 14, 13, 12, 0]$이다.

4초가 지난 후 각 구역마다 존재하는 기포의 수는 $[0, 18, 20, 22, 4, 2]$이다.

5초가 지난 후 각 구역마다 존재하는 기포의 수는 $[23, 27, 11, 8, 4, 0]$이다.

6초가 지난 후 각 구역마다 존재하는 기포의 수는 $[9, 11, 8, 23, 21, 20]$이다.

노트

$\lfloor \cdot \rfloor$는 바닥 함수로, 주어진 실수 이하의 최대 정수를 나타내는 함수이다.

문제 G. 돈 복사

시간 제한	2 초
메모리 제한	1024 MB

GIST 하우스 중고장터에서는 돈을 특정 물건으로 바꿔주거나, 특정 물건을 다른 특정 물건으로 바꿔주는 이벤트를 진행하고 있다. 특정 물건을 특정 물건으로 바꿀 때는 추가적인 돈이 오갈 수 있다. GIST 하우스 중고장터는 무한히 많은 돈과 물건을 갖고 있기 때문에, 모든 거래는 재화가 존재하는 한 무한히 많이 할 수 있다. **단, 고객은 한 번에 물건을 최대 하나까지만 들고 있을 수 있다.**

GIST 하우스 중고장터 앱 개발팀은 물물교환을 통해 돈 복사가 가능한지 추적하고 있다. 예를 들어 다음과 같은 거래들이 존재한다고 가정하자.

- t_1: 사용자가 1번 물건을 10원에 산다.
- t_2: 사용자가 1번 물건에 10원을 추가해 2번 물건과 교환한다.
- t_3: 사용자가 2번 물건을 21원을 받고 판다.
- t_4: 사용자가 2번 물건을 1번 물건과 교환하고 12원을 추가로 받는다.
- t_5: 사용자가 2번 물건을 7원에 산다.

이들 중 개발팀이 리스트 $L_1 = \{t_1, t_2, t_3\}$에 든 거래만 허용한다고 가정하자. 그렇다면, t_1, t_2, t_3를 이 순서대로 선택하면 돈 20원을 돈 21원으로 늘릴 수 있다. 그리고 이를 무한히 반복하면 돈을 무한정 늘릴 수 있다. 이렇게 돈이 복사되면 대한민국 경제에 악영향을 끼칠 수 있기 때문에, 개발팀은 그런 상황을 막고 싶다. 따라서 개발팀은 거래들의 리스트가 주어지면 그 리스트가 안전한지 판별하고 싶다.

리스트의 **안전도**는 다음과 같이 정의된다.

- 어떤 사람이 C원의 돈을 초기 자금으로 갖고 거래를 반복하여 돈을 무한정 늘릴 수 있다면, 리스트의 안전도는 C의 최솟값이다. 그러한 C가 존재하지 않는다면 리스트의 안전도는 ∞이다. 이때, 돈을 늘리기 위해 모든 물건을 판매할 필요는 없다.

예를 들어, 리스트 $L_1 = \{t_1, t_2, t_3\}$의 안전도는 20이다. 20원의 돈을 갖고 가장 먼저 t_1, t_2, t_3를 선택하기를 반복하면 되기 때문이다. 그러나 19원의 돈을 갖고 있는 사람은 어떻게 하더라도 돈을 무한정 늘릴 수 없다. 또 리스트 $L_2 = \{t_3, t_4, t_5\}$의 안전도는 7이다.

한편 리스트 $L_3 = \{t_2, t_3, t_4\}$의 안전도는 ∞이다. t_2와 t_4를 선택하기를 반복하면 돈을 무한정 늘릴 수 있을 것 같아 보이지만, 1번 물건 또는 2번 물건을 구할 수 있는 방법이 없어 그러한 거래를 시도할 수 없기 때문이다.

물물교환 거래 리스트가 주어지면 그 리스트의 안전도를 측정하는 프로그램을 작성하시오.

입력

첫째 줄에 물건의 개수 N과 물물교환 거래 리스트의 길이 M이 주어진다. $(1 \le N \le 1\,000, 1 \le M \le \min\left(N^2 + N, 3\,000\right))$

둘째 줄부터 M개의 줄에 걸쳐, 물물교환 거래 리스트 L의 정보가 다음 다섯 가지 형식 중 하나로 주어진다.

- 1 c i: c원을 들여 i번 물건을 살 수 있다.
- 2 i c: i번 물건을 팔아 돈 c원을 벌 수 있다.
- 3 i_1 i_2: i_1번 물건을 i_2번 물건과 교환할 수 있다.
- 4 i_1 c i_2: i_1번 물건과 돈 c원을 i_2번 물건으로 교환할 수 있다.
- 5 i_1 i_2 c: i_1번 물건을 i_2번 물건과 돈 c원으로 교환할 수 있다.

i, i_1, i_2는 1 이상 N 이하의 정수이다. c는 1 이상 1000 이하의 정수이다. 각각의 수는 공백으로 구분되어 주어진다.

거래 리스트 L 내의 3, 4, 5번 형식 정보는 각각 $i_1 \neq i_2$를 만족시킨다.

1번 형식의 임의의 두 정보가 같은 i를 공유하지 않는다.

2번 형식의 임의의 두 정보가 같은 i를 공유하지 않는다.

3번, 4번 또는 5번 형식 중 하나로 들어온 임의의 두 정보가 같은 i_1과 i_2를 동시에 공유하지 않는다. 예를 들어 `3 1 2`와 `4 1 1 2`는 동시에 입력으로 주어지지 않는다. 그러나 `3 1 2`와 `3 1 3`은 동시에 입력으로 주어질 수 있는데, 이 둘은 i_1을 공유하더라도 i_2는 공유하지 않기 때문이다.

모든 거래는 역을 보장하지 않는다. 예를 들어 입력 `3 1 2`는 2번 물건으로 1번 물건을 교환할 수 있음을 의미하지 않는다.

출력

리스트의 안전도를 출력한다. 리스트의 안전도가 ∞라면 문자열 `INF`를 출력한다.

서브태스크 1 (점수: 25)

$N \leq 5$

서브태스크 2 (점수: 75)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5 8 1 10 5 2 1 10 4 1 15 4 4 2 10 3 4 3 10 4 4 5 10 2 5 2 1 10 5 4 5 20	25

문제 H. 기초마법학

시간 제한	1 초
메모리 제한	1024 MB

GIST는 리버럴 아츠 칼리지 모델을 차용하여 학부생들에게 수준 높은 인문학/사회학 교양과목을 제공하고 있다.

2024년, GIST는 교양으로 마법학 과목을 신설했다. GIST 부지 내에는 N개의 마법원이 있고, 각각의 마법원은 좌표평면 위 하나의 점으로 표현된다. 마법원들은 각각 마나의 세기와 색깔을 가진다. 마나의 세기는 양수이고, 색깔은 1과 K 사이의 정수로 표현할 수 있다. 기초마법학 수업의 일환으로 마법진 그리기를 배우고 있는 해원이는 다음 조건을 만족시키며 마법진을 그려야 한다.

- 마법진은 직사각형이고, 각 변은 x축 또는 y축 둘 중 하나에 평행하다.
- 직사각형의 네 꼭짓점 중 x좌표와 y좌표가 동시에 최소인 점은 $(0,0)$이다.
- 마법진은 모든 색깔의 마법원을 포함해서는 안 된다. 마법진이 마법원을 포함한다는 것은 마법원이 마법진의 둘레 또는 내부에 있다는 것이다.

마법진의 힘은 직사각형의 내부 또는 네 변의 위에 있는 모든 마법원의 마나의 세기의 총합이다. 만약 마법진 내에 하나의 마법원도 포함되어 있지 않다면, 마법진의 힘은 0이다. 마법진의 힘을 최대화하시오.

입력

첫째 줄에 마법원의 개수 N과 마법원 색깔의 종류의 수 K가 공백으로 구분되어 주어진다. ($2 \le N \le 150\,000$, $2 \le K \le \min\{N, 10\,000\}$)

둘째 줄부터 N개의 줄에 걸쳐 네 정수 x_i, y_i, p_i, c_i가 공백으로 구분되어 주어진다. ($1 \le x_i, y_i \le 10^8$, $1 \le p_i \le 1\,000$, $1 \le c_i \le K$) 이는 i번째 마법원이 좌표 (x_i, y_i)에 위치하고, 마나의 세기 p_i를 가지며, c_i번째 색깔을 가진다는 뜻이다.

1 이상 K 이하의 모든 정수 j에 대하여 j번째 색깔을 가지는 마법원이 적어도 하나 존재한다.

출력

주어진 조건을 만족시키며 그릴 수 있는 마법진 힘의 최댓값을 출력한다.

서브태스크 1 (점수: 30)

$N \le 300$

서브태스크 2 (점수: 15)

모든 마법원의 y좌표가 1이다.

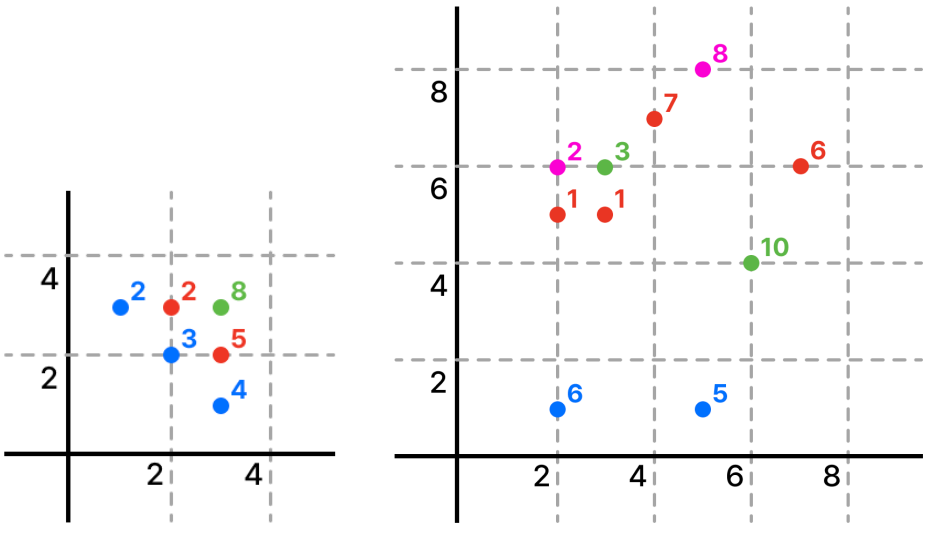
서브태스크 3 (점수: 55)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
6 3 1 3 2 1 2 2 3 1 3 1 4 1 2 3 2 3 3 2 5 3 3 3 8 2	12
10 4 5 1 5 1 2 5 1 2 2 6 2 4 3 6 9 3 3 5 1 2 5 8 8 4 6 4 10 3 7 6 6 2 4 7 7 2 2 1 6 1	23
2 2 1 1 1 1 1 1 1 2	0

노트



(a) **예제 1**에 해당하는 그림 (b) **예제 2**에 해당하는 그림

예제 3과 같이 x좌표와 y좌표가 모두 같은 마법원이 두 개 이상 존재할 수 있음에 주의하시오.

문제 I. DevNight 운영

시간 제한	2 초
메모리 제한	1024 MB

GIST에서는 2022년부터 매년 가을 'GIST DevNight' (개발자의 밤, Developers' Night) 행사가 열린다. DevNight 행사를 통해 개발자들끼리 정보를 공유하고 친목을 도모할 수 있다. 행사는 컨퍼런스 룸에서 열리는 컨퍼런스와 커뮤니케이션 룸에서 열리는 친목 활동으로 나누어져 있다.

2024년도 GIST DevNight는 정점 N개와 길이가 있는 양방향 간선 M개로 구성된 건물에서 열린다. 각각의 정점에는 컨퍼런스 룸 또는 커뮤니케이션 룸 둘 중 하나가 자리잡고 있다. 커뮤니케이션 룸의 숫자는 K개이며, 1번 정점부터 K번 정점까지는 커뮤니케이션 룸이고 $K+1$번 정점부터 N번 정점까지는 컨퍼런스 룸이다.

컨퍼런스 하나가 끝나면 개발자들은 커뮤니케이션 룸으로 이동한다. 이때 자신의 위치에 따라 각 커뮤니케이션 룸에 대한 선호도가 달라진다. 모든 사람은 컨퍼런스 룸의 위치로부터 최단 경로의 길이가 짧은 커뮤니케이션 룸을 더 선호하며, 만약 그런 커뮤니케이션 룸이 2개 이상 있으면 정점의 번호가 작은 커뮤니케이션 룸을 더 선호한다.

2023년도 DevNight 때는 개발자들이 가장 선호도가 높은 커뮤니케이션 룸을 이용했다. 그러나 그러다 보니 방 하나에만 지나치게 사람이 많이 모이는 일이 발생하여, 2024년도 DevNight에서는 모든 사람이 **두 번째로** 선호도가 높은 커뮤니케이션 룸을 이용하도록 정책이 바뀌었다. DevNight 운영자인 보성이는 각각의 컨퍼런스 룸에 대하여 그 방에서 컨퍼런스가 끝나면 개발자들이 두 번째로 선호도가 높은 커뮤니케이션 룸으로 가기 위해 얼마나 걸어가야 하는지 구하려고 한다. 보성이를 도와주는 프로그램을 만들어 보자.

입력

첫째 줄에 정점의 개수 N, 간선의 개수 M, 커뮤니케이션 룸의 개수 K가 공백으로 구분되어 주어진다. $(3 \le N \le 100\,000, 2 \le M \le 300\,000, 2 \le K < N)$

둘째 줄부터 M줄에 걸쳐 간선의 정보를 나타내는 세 정수 s, e, d가 공백으로 구분되어 주어진다. $(1 \le s, e \le N, s \ne e, 1 \le d \le 10^6)$ 이는 두 정점 s와 e를 잇는 길이 d의 간선이 존재함을 의미한다.

임의의 두 정점 사이를 잇는 간선은 최대 1개이다. 또 주어진 그래프는 반드시 연결 그래프이다.

출력

$K+1$ 이상 N 이하의 모든 자연수 x에 대하여 x번 정점에 있는 사람이 두 번째로 선호도가 높은 커뮤니케이션 룸까지 가기 위한 최단 경로의 길이를 순서대로 공백으로 구분하여 출력한다. **정답이 32비트 정수 범위를 넘을 수 있음에 주의하시오.**

서브태스크 1 (점수: 20)

$K \le 5$

서브태스크 2 (점수: 25)

주어지는 그래프가 일직선 모양이다. 즉, 어떤 수열 $a_1, a_2, \cdots, a_n$이 존재하여 이 수열에는 1부터 n까지의 수가 모두 한 번씩만 나타나고, a_1번 정점과 a_2번 정점을 잇는 간선, a_2번 정점과 a_3번 정점을 잇는 간선, $\cdots$, a_{n-1}번 정점과 a_n번 정점을 잇는 간선이 존재하며 다른 간선은 존재하지 않는다.

서브태스크 3 (점수: 55)

추가 제약 조건이 없다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
10 15 3 1 4 10 1 5 9 1 9 3 2 4 1 2 6 6 2 7 5 3 6 2 3 8 11 3 9 11 4 8 4 5 6 8 5 10 7 7 9 11 7 10 11 8 10 11	9 10 6 13 11 11 16

노트

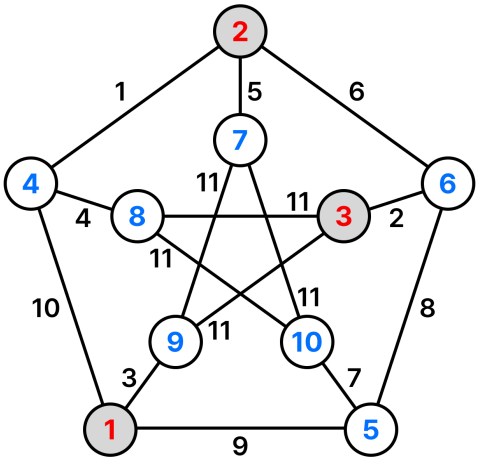


그림 3: **예제 1**에 해당하는 그래프