

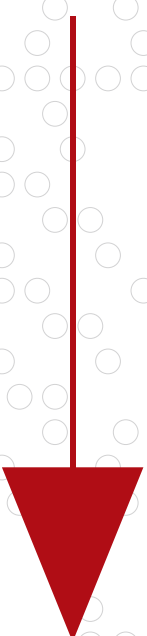
# Sogang Programming Contest

2021

2021 제17회  
서강대학교  
프로그래밍 대회

Division  
**Master**

문제지



Master  
Official Problemset

## 대회 규칙

- 대회 중 제출한 소스코드는 채점 서버에 의해 자동으로 채점되며, 실시간으로 결과를 알 수 있습니다. 문제를 제출하였을 때 ‘맞았습니다!!’ 를 받으면 문제를 푼 것으로, 이외의 결과를 받으면 틀린 것으로 생각합니다.
- 문제를 풀 때마다 패널티 점수가 누적됩니다. 패널티 점수는 모든 맞은 문제에 대해, 대회 시작 시간부터 그 문제를 풀기까지 걸린 시간을  $t$ 분, 처음으로 문제를 맞기 직전까지 제출한 횟수를  $w$ 번이라고 할 때  $(t + 20w)$  점입니다.
- 순위는 푼 문제가 많은 순서대로, 푼 문제가 같을 경우에는 패널티 점수의 합이 적은 순서대로 결정됩니다.
- 사용 가능 언어는 C, C++, Java, Python 3, Kotlin입니다. 모든 문제는 출제진이 C++로 정답을 작성했음이 보장되며, 문제를 해결하지 못하는 언어가 존재할 수도 있습니다. 각 언어의 컴파일 옵션과 컴파일러 버전은 아래와 같습니다.

### C11 gcc (GCC) 11.1.0

```
컴파일 gcc Main.c -o Main -O2 -Wall -lm -static -std=gnu11 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

### C++11 g++ (GCC) 11.1.0

```
컴파일 g++ Main.cc -o Main -O2 -Wall -lm -static -std=gnu++11 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

### C++14 g++ (GCC) 11.1.0

```
컴파일 g++ Main.cc -o Main -O2 -Wall -lm -static -std=gnu++14 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

### C++17 g++ (GCC) 11.1.0

```
컴파일 g++ Main.cc -o Main -O2 -Wall -lm -static -std=gnu++17 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

### C++20 g++ (GCC) 11.1.0

```
컴파일 g++ Main.cc -o Main -O2 -Wall -lm -static -std=gnu++20 -DONLINE_JUDGE -DBOJ
실행 ./Main
```

### Java 8 openjdk version "16.0.1" 2021-04-20

```
컴파일 javac -release 8 -J-Xms1024m -J-Xmx1920m -J-Xss512m -encoding UTF-8
Main.java
실행 java -Xms1024m -Xmx1920m -Xss512m -Dfile.encoding=UTF-8 -XX:+UseSerialGC
-DONLINE_JUDGE=1 -DBOJ=1 Main
```

### Python 3 Python 3.9.5

```
컴파일 python3 -c "import py_compile; py_compile.compile(r'Main.py')"
실행 python3 Main.py
```

### PyPy3 PyPy 7.3.4 with GCC 7.3.1 20180303 (Red Hat 7.3.1-5) (Python 3.7.10)

```
컴파일 pypy3 -c "import py_compile; py_compile.compile(r'Main.py')"
실행 pypy3 Main.py
```

**Kotlin/JVM** kotlinc-jvm 1.5.0 (JRE 1.8.0\_201-b09)

컴파일 `kotlinc-jvm -J-Xms1024m -J-Xmx1920m -J-Xss512m -include-runtime -d Main.jar Main.kt`

실행 `java -Xms1024m -Xmx1920m -Xss512m -Dfile.encoding=UTF-8 -XX:+UseSerialGC -DONLINE_JUDGE=1 -DBOJ=1 -jar Main.jar`

- 네트워크 사용은 금지됩니다. 단, 솔루션을 제출하거나 언어 레퍼런스를 확인하는 것은 가능합니다. 접속이 허용된 사이트의 예는 다음과 같습니다.

**C/C++** <https://en.cppreference.com/w/>

**Java** <https://docs.oracle.com/javase/8/docs/api/>

**Python** <https://docs.python.org/3/>

**Kotlin** <https://kotlinlang.org/docs/>

- 대회 종료 전에 퇴실할 수 없습니다.
- 책이나 개인이 준비한 인쇄된 참고자료를 열람할 수 있습니다. 참고자료의 양에는 제한이 없습니다.
- 대회 중에 휴대폰 및 전자기기는 사용할 수 없습니다.

## 문제 목록

문제지에 있는 문제가 총 8문제가 맞는지 확인하시기 바랍니다.

- A Ресторан
- B 벼락치기
- C 야바위 게임
- D 버스 노선 개편하기
- E Player-based Team Distribution
- F 방탈출
- G 알고리즘 과외
- H 내가 몇 등이었지??

모든 문제의 메모리 제한은 1GB로 동일합니다.

## 문제 A. Ресторан

시간 제한 1 초  
메모리 제한 1024 MB

팀 레드시프트를 따라 모스크바 여행에 간 한별이는 거리에 늘어난 ‘Ресторан’가 적힌 간판들을 보고 어떤 가게일까 고민하기 시작했다.

러시아어 알파벳에는 영어 알파벳과 비슷하게 생긴 글자들이 있는데, A, B, E, K, M, H, O, P, C, T, Y, X의 12가지가 있다. A, K, M, O, T의 다섯 글자는 보이는 대로 읽으면 되지만, 나머지 7가지 글자는 영어 알파벳의 발음과 다르게 읽어야 한다!

- ‘B’는 ‘B’처럼 보이지만 ‘v’로 읽는다.
- ‘E’는 ‘E’처럼 보이지만 ‘ye’로 읽는다.
- ‘H’은 ‘H’처럼 보이지만 ‘n’으로 읽는다.
- ‘P’은 ‘P’처럼 보이지만 ‘r’으로 읽는다.
- ‘C’는 ‘C’처럼 보이지만 ‘s’로 읽는다.
- ‘Y’는 ‘Y’처럼 보이지만 ‘u’로 읽는다.
- ‘X’는 ‘X’처럼 보이지만 ‘h’로 읽는다.

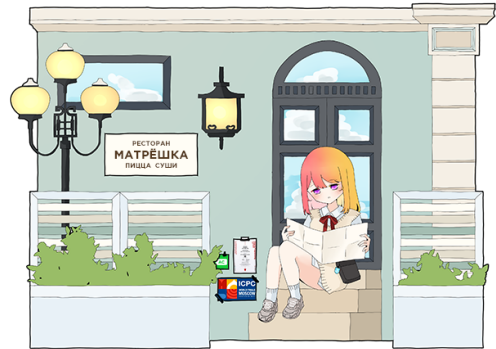


그림 A.1: Ресторан «Матрёшка»

고민에 빠진 한별이를 위해 Ресторан가 무슨 가게인지 알려 주자.

### 입력

최대 100글자의 단어가 주어진다. 모든 글자는 영어 대문자 A, B, E, K, M, H, O, P, C, T, Y, X 중 하나로 이루어져 있다. 입력이 러시아어 대문자로 주어지지 않음에 주의하자.

### 출력

입력으로 주어진 단어를 비슷한 러시아어 알파벳으로 읽었을 때, 어떤 발음이 되는지 영어 소문자로 나타내 출력한다.

### 입출력 예시

표준 입력(stdin)	표준 출력(stdout)
РЕСТОРАН	ryestoran
МОСКВА	moskva
НЕТ	nyet
ТРАКТОР	traktor

이 페이지는 공백입니다

## 문제 B. 벼락치기

시간 제한 1 초  
메모리 제한 1024 MB

유체이탈을 할 줄 아는 지환이는 수업에 출석은 했으나 수업을 듣지 않았다. 늘 그랬듯이 시험기간은 찾아오는 법, 지환이는 이제야 공부를 시작했다.

$N$ 개의 챕터를 순서대로 공부해야 하는데,  $i$ 번째 챕터를 공부하는 데  $T_i$ 분이 걸린다. 하지만 시간이 많지 않기 때문에 모두 볼 수는 없다. 어쩔 수 없이 벼락치기 공부법을 통해  $F$  학점만은 피하고자 한다.

벼락치기 공부법이란 30분간 공부한 후 휴식을 취하고 다시 30분 공부를 시작하는 방법이다. 한 챕터를 모두 공부하는 즉시 다음 챕터 공부를 시작한다. 이 공부법에 벼락치기라는 수식어가 붙은 이유는 한 가지 중요한 규칙이 더 있기 때문이다. 바로 **30분의 공부시간이 끝나면 공부하던 챕터는 가차 없이 덮고 넘어가 버리는 것이다!**

다행히 시험 전까지 벼락치기 공부법으로  $N$ 개의 챕터를 순서대로 공부할 수 있었다. 생각보다 쉽다고 생각한 지환이는 그대로 술을 마시러 갔다. 하지만 벼락치기 공부법의 한계일까? 안타깝게도 술을 마시고 나니 절반 이상 공부한 챕터를 제외하고 모두 머리에서 지워지고 말았다. 정신이 아득해진 지환이는 머리에 챕터가 몇 개 남았는지 세기 시작했다. 숙취에 힘들어하는 지환이를 도와 절반 이상 공부한 챕터의 개수를 구해주자.

### 입력

첫 번째 줄에  $N$ 이 주어진다. ( $1 \leq N \leq 100$ )

두 번째 줄부터  $N$ 개의 줄에 정수  $T_i$ 가 순서대로 주어진다. ( $1 \leq T_i \leq 100$ )

### 출력

지환이가 절반 이상 공부한 챕터의 개수를 출력한다.

### 입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5 10 20 30 40 50	5

처음 30분간 첫 번째 챕터를 10분, 두 번째 챕터를 20분 공부한다.

다음 30분간 세 번째 챕터를 30분 공부한다.

다음 30분간 네 번째 챕터를 30분 공부한다.

다음 30분간 다섯 번째 챕터를 30분 공부한다.

모든 챕터를 절반 이상 공부했다.

표준 입력(stdin)	표준 출력(stdout)
5 20 40 60 80 100	2

처음 30분간 첫 번째 챕터를 20분, 두 번째 챕터를 10분 공부한다.

다음 30분간 세 번째 챕터를 30분 공부한다.

다음 30분간 네 번째 챕터를 30분 공부한다.

다음 30분간 다섯 번째 챕터를 30분 공부한다.

첫 번째, 세 번째 챕터를 절반 이상 공부했다.



## 문제 C. 야바위 게임

시간 제한 1 초  
메모리 제한 1024 MB

준석이와 상원이는 게임을 한다. 게임판 위에는  $N$ 개의 정점과  $M$ 개의 간선이 있다. 각 정점 위에는 컵이 하나씩 있고 그중 하나에는 공이 들어 있다. 모든 간선은 서로 다른 두 정점을 양방향으로 잇는다.

준석이가 컵을 섞으면 상원이는 공이 어디에 있는지 맞혀야 한다. 준석이는 간선으로 연결된 두 컵을 잡고 위치를 바꿀 수 있다.

상원이는 준석이의 화려한 손기술에 속아서 컵이 어떻게 움직였는지 까먹었다. 하지만 처음에 공이 들어있던 컵과 그 컵을 움직인 횟수는 기억하고 있다.

처음에 공이 들어있는 컵이 있던 정점과 그 컵이 움직인 횟수가 주어지면, 지금 공이 들어있는 컵이 있을 수 있는 정점들의 후보를 모두 찾아보자!

### 입력

첫 번째 줄에 정점 수  $N$ , 간선 수  $M$ , 게임 시작 시 공이 놓여있는 정점 번호  $X$ , 공이 든 컵이 움직인 횟수  $Y$ 가 주어진다. ( $1 \leq N, Y \leq 10^3$ ,  $1 \leq M \leq 10^4$ ,  $1 \leq X \leq N$ )

다음 줄부터  $M$ 개의 줄 각각에 각 간선이 연결하는 두 정점의 번호가 주어진다. 정점의 번호는 1부터  $N$ 까지이다. 어떤 두 정점을 잇는 간선이 여러 개일 수 있다.

### 출력

공이 있을 수 있는 곳을 정점 번호가 작은 순서대로 한 줄에 출력한다.

가능한 후보가 없다면 -1을 출력한다.

### 입출력 예시

표준 입력(stdin)	표준 출력(stdout)
6 6 1 2 1 2 2 3 3 4 4 5 5 6 6 1	1 3 5
6 4 1 2 2 3 3 4 4 5 5 6	-1

이 페이지는 공백입니다

## 문제 D. 버스 노선 개편하기

시간 제한 2 초  
메모리 제한 1024 MB

서강 나라에서는 일직선 도로를 따라  $N$ 개의 버스 노선을 운영 중이다. 필요할 때마다 노선을 새로 만든 탓에 겹치거나 중복되는 노선이 많다. 복잡한 버스 노선에 지친 시민들을 위해 버스 노선을 개편하기로 했다.

각 버스 노선은 세 정수  $S, E, C$ 로 나타낼 수 있으며, 구간  $[S, E]$ 를 요금  $C$ 로 운행한다는 뜻이다. 어떤 두 버스 노선의 구간이 한 점 이상에서 겹친다면, 두 구간을 합친 새 노선으로 대체한다. 이때 요금은 더 낮은 금액의 요금을 따르기로 했다. 버스 노선 개편은 구간이 겹치는 버스 노선이 없을 때까지 진행한다.

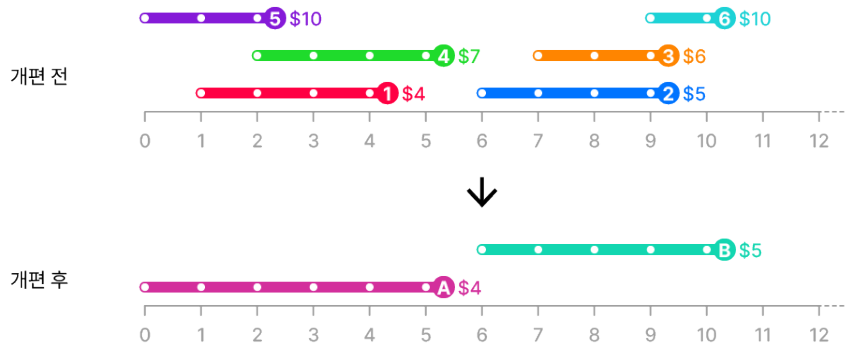


그림 D.1: 개편 전과 개편 후의 버스 노선도

버스 노선들의 정보가 주어지면, 개편이 끝난 후 버스 노선의 정보를 출력하는 프로그램을 작성하자.

### 입력

첫 번째 줄에 버스 노선의 수  $N$ 이 주어진다. ( $1 \leq N \leq 200\,000$ )

두 번째 줄부터  $N$ 개의 줄에 각 버스 노선을 나타내는 세 정수  $S, E, C$ 가 주어진다. ( $0 \leq S < E \leq 10^9, 1 \leq C \leq 10^9$ )

### 출력

첫 번째 줄에 개편이 끝난 후의 버스 노선의 수  $K$ 를 출력한다.

두 번째 줄부터  $K$ 개의 줄에 개편 후 각 버스 노선의  $S, E, C$ 를  $S$ 가 작은 순서대로 출력한다.

### 입출력 예시

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

표준 입력(stdin)	표준 출력(stdout)
5	5
1 2 4	1 2 4
3 7 3	3 7 3
8 9 5	8 9 5
10 14 10	10 14 10
17 18 3	17 18 3

## 문제 E. Player-based Team Distribution

시간 제한 1 초  
메모리 제한 1024 MB

플레이어  $N$ 명이 1개 이상의 팀으로 나누어 게임을 진행하려 한다. 플레이어는 각각 정확히 한 팀에 속해야 한다.  $i$  번째 플레이어는 같은 팀에 속한 인원 수와  $a_i$ 를 곱한 것만큼의 점수를 얻는다.

팀을 적절히 나누었을 때, 모든 플레이어의 점수의 합의 최댓값을 구해보자.

### 입력

첫째 줄에  $N$  ( $1 \leq N \leq 10^5$ )이 주어진다.

둘째 줄에  $N$ 개의 정수가 주어진다.  $i$ 번째 수는  $a_i$ 이다. ( $-10^5 \leq a_i \leq 10^5$ )

### 출력

첫째 줄에 팀을 적절히 나누었을 때 모든 플레이어들의 점수의 합의 최댓값을 출력한다.

### 입출력 예시

표준 입력(stdin)	표준 출력(stdout)
4 2 3 -4 1	14

이 페이지는 공백입니다

## 문제 F. 방탈출

시간 제한        2 초  
 메모리 제한     1024 MB

원빈이는 친구들과 함께 방탈출 카페에 갔다. 방탈출 카페에는 1번부터  $N$ 번까지 총  $N$ 개의 방이 있고, 각 방에는 친구들이 한 명씩 들어가 있다. 모든 방은 외부로부터 완전히 독립되어 있다.

방에서 탈출하지 못하는 친구들이 답답했던 원빈이는 모든 친구들이 출구로 탈출할 수 있도록 워프와 비상탈출구를 설치하려고 한다. 워프는 최대  $M$ 개까지 설치할 수 있는데,  $i$ 번째 워프를 설치하는 데  $c_i$ 의 시간이 걸리고, 워프를 설치하면  $a_i$ 번 방과  $b_i$ 번 방 사이를 이동할 수 있다. 또한 각 방에는 출구로 바로 연결되는 비상탈출구를 설치할 수 있는데,  $i$ 번 방에 비상탈출구를 설치하는 데 걸리는 시간은  $t_i$ 이다.

안타깝게도 원빈이는 머리가 나빠 워프나 비상탈출구의 설치 작업을 동시에 여럿 진행할 수 없다. 즉 한 작업이 끝나고 나서야 다음 작업을 이어서 시작할 수 있다.

원빈이를 도와 모든 친구들이 출구로 탈출할 수 있도록 워프와 비상탈출구를 설치하는 데 걸리는 최소 시간을 구해보자.

### 입력

첫 번째 줄에는 방의 개수  $N$ 과 설치할 수 있는 워프의 개수  $M$ 이 주어진다. ( $1 \leq N \leq 200\,000$ ,  $1 \leq M \leq 100\,000$ )

다음  $M$ 개의 줄에는 워프의 정보를 나타내는 세 정수  $a_i, b_i, c_i$ 가 공백으로 구분되어 주어지는데, 이는  $a_i$ 번 방과  $b_i$ 번 방 사이를 잇는 워프를 설치하는 데 걸리는 시간이  $c_i$ 라는 의미이다. 같은 두 개의 방을 잇는 워프가 여러 개 존재할 수 있다. ( $1 \leq a_i, b_i \leq N$ ,  $1 \leq c_i \leq 10^4$ ,  $a_i \neq b_i$ )

마지막 줄에는  $N$ 개의 정수  $t_1, \dots, t_n$ 이 주어지는데,  $t_i$ 는  $i$ 번째 방에 비상탈출구를 설치하는 데 드는 시간을 의미한다. ( $1 \leq t_i \leq 10^4$ )

### 출력

모든 친구들이 출구로 탈출할 수 있도록 워프와 비상탈출구를 설치하는 데 걸리는 최소 시간을 출력한다.

### 입출력 예시

표준 입력 (stdin)	표준 출력 (stdout)
3 3 1 2 2 2 3 2 3 1 2 3 3 3	7
3 1 1 2 2 3 3 3	8

이 페이지는 공백입니다



## 문제 G. 알고리즘 과외

시간 제한 2 초  
메모리 제한 1024 MB

지환이(라라라라)가 운영하는 알고리즘 학원에는  $N$ 명의 학생이 있고, 각 학생은 1부터  $N$ 까지의 번호를 가지고 있다,

알고리즘 학원에서는 학생의 수준을 나타내기 위해 레이팅 시스템을 사용하는데, 모든 학생은 자신만의 레이팅을 갖고 있고,  $i$ 번 학생의 레이팅은  $a_i$ 로 나타낼 수 있다.

지환이는 학원 수강생의 레이팅 상승을 위해 2명의 학생을 골라 몰래 과외를 해주려고 한다. 그런데 수강생은 자기와 번호가 너무 많이 차이 나거나 너무 적게 차이 나는 학생을 싫어한다. 따라서  $i$ 번 학생은 자기와의 번호 차이가  $l_i$  이상  $r_i$  이하인 학생들과만 과외를 하려고 할 것이다.

위의 조건을 만족하면서 레이팅의 차이가 최대가 되도록 2명의 학생을 고를 때, 그때의 레이팅의 차를 구하여라.

### 입력

첫 번째 줄에는 알고리즘 학원의 학생 수  $N$ 이 주어진다. ( $2 \leq N \leq 200\,000$ )

두 번째 줄부터  $N+1$ 번째 줄까지는 학생의 정보가 주어진다.  $i+1$ 번째 줄에는 세 정수  $a_i, l_i, r_i$ 가 공백으로 구분되어 주어지는데, 이는  $i$ 번 학생의 레이팅이  $a_i$ 이고, 자기와의 번호 차이가  $l_i$  이상  $r_i$  이하인 학생들과만 과외를 하려고 한다는 의미이다. ( $1 \leq a_i \leq 10^9, 1 \leq l_i \leq r_i \leq N$ )

### 출력

문제의 조건을 모두 만족하면서 레이팅의 차이가 최대가 되도록 2명의 학생을 고를 때, 그때의 레이팅의 차를 출력한다.

만약 조건을 만족하도록 2명의 학생을 고를 수 없다면  $-1$ 을 출력한다.

### 입출력 예시

표준 입력(stdin)	표준 출력(stdout)
4 1 1 4 2 1 1 3 1 2 7 1 2	4
5 10 1 1 3 1 1 2 1 3 3 1 1 9 2 4	7

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

## 문제 H. 내가 몇 등이었지??

시간 제한 1 초  
메모리 제한 1024 MB

길고 길었던 2학기가 끝나가고 있다. 종강에 앞서 성적 확인 기간만이 남아있다.

“알고리즘 설계와 분석” 과목의 수강생들은 자신의 과제 성적  $p$ , 중간고사 성적  $q$ , 기말고사 성적  $r$ 을 알고 있다. 이때 각 수강생의 최종 성적은 성적 반영 비율을 나타내는 세 개의 양의 정수  $A, B, C$ 에 따라 다음과 같이 계산된다.

$$Ap + Bq + Cr$$

이번 학기에는 모든 시험과 과제에 최선을 다하라는 뜻으로 성적 반영 비율  $A, B, C$ 의 값을 공개하지 않기로 했다. 따라서 성적 확인 기간에 알려주는 최종 성적을 알아두는 것이 중요하다.

다가올 종강이 신난 나머지 수강생들은 성적 확인을 제대로 하지 않았고 본인의 최종 성적이 얼마였는지 잊었다. 뒤늦게 최종 성적이 궁금해진 수강생들은 자신들의 성적을 다시 확인하려 했지만 어림도 없었다.

다행히도 몇몇 수강생들은 자신보다 최종 성적이 높은 수강생을 기억한다. 이 정보들을 모아 자신의 최종 성적을 가늠할 수 있을 것이다. 두 수강생에 대해 누가 더 최종 성적이 높은지 묻는 질문에 답해주는 프로그램을 작성하자.

### 입력

첫 번째 줄에 수강생의 수  $N$ , 수강생이 기억하는 정보의 수  $M$ , 질문의 수  $Q$ 가 주어진다. ( $2 \leq N \leq 2000$ ,  $1 \leq M, Q \leq 2000$ )

두 번째 줄부터  $N$ 개의 줄에 걸쳐 각 수강생의  $p, q, r$ 이 정수로 주어진다. ( $0 \leq p, q, r \leq 10000$ )

이어지는 줄부터  $M$ 개의 줄에  $i, j$ 가 정수로 주어진다.  $i$ 번째 학생이  $j$ 번째 학생보다 최종 성적이 높았다는 뜻이다. ( $1 \leq i, j \leq N, i \neq j$ )

이어지는 줄부터  $Q$ 개의 줄에  $i, j$ 가 정수로 주어진다.  $i$ 번째 학생과  $j$ 번째 학생 중 누가 더 최종 성적이 높은지 묻는 질문이다. ( $1 \leq i, j \leq N, i \neq j$ )

항상 올바른 입력만 주어진다.

### 출력

$Q$ 개의 줄에  $i$ 번째 학생이  $j$ 번째 학생보다 최종 성적이 더 높으면  $>$ , 더 낮으면  $<$ , 같으면  $=$ , 결정할 수 없으면  $?$ 를 출력한다.

## 입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5 3 4	>
2 1 4	<
1 2 4	?
1 1 5	=
2 2 0	
1 1 5	
1 3	
2 5	
5 4	
1 4	
3 2	
1 2	
3 5	