

2019 Summer Algorithm

3차 Test

A. 등록	2p	
B. 축구 게임	3p	
C. 선발 명단	4p	
D. 암호 코드	5p	
E. 거의 최단 경로	6p	
F. 문자열 뽑기	8p	
G. 화학 제품	9p	
H. 최종 순위	10p	
I. 점 분리	12p	
J. Vacation Plans	13p	
K. 맹독 방벽	16p	
L. 첼로	17p	
M. Networks with Unidirectional Links	19p	(Korean Translated Available)
N. 랜선 자르기	23p	
O. Closest Pair	24p	
P. Sticker	26p	(Korean Translated Available)
Q. Term Project	29p	(Korean Translated Available)
R. Meteor Shower	31p	
S. How Many to Be Happy	33p	
T. The Mountain of Gold	35p	
U. Proving Equivalences	39p	(Korean Translated Available)
V. 종이의 개수	41p	

2019.09.21. 15:00 ~ 18:30

Problem A

등록

제한 시간 : 1초, 메모리 제한 : 128MB

자신이 속한 팀명과 그 비밀번호를 그대로 출력하는 프로그램을 작성하시오.

Input

이 문제는 입력이 없다.

Output

표준출력(standard output)으로 출력해야 한다. 첫 줄에 자신의 팀명을 출력하고, 둘째 줄에 비밀번호를 출력한다.

다음은 팀명이 Smoothes, 비밀번호가 thisisit2!인 경우의 입출력 예제이다. 참고로 입력이 없는 것에 주의한다.

예제

Sample Input	Sample Output
(이 문제는 입력이 없다.)	Smoothes thisisit2!

Problem B

축구 게임

제한 시간 : 2초, 메모리 제한 : 512MB

문제

축구는 지구에서 가장 인기있는 스포츠 중의 하나입니다. n 팀으로 이루어진 축구 리그가 있습니다. 하나의 팀은 다른 모든 팀과 정확히 한 번씩 만 경기를 합니다. 그러므로, 각 팀은 $n - 1$ 번의 경기를 하게 됩니다. 무승부는 승부차기를 하기 때문에 없습니다. 한 경기 후에 이긴 한 팀은 1 점을 얻게 되고, 진 팀은 0 점을 얻게 됩니다.

베스트 팀 선정을 위해 경기 일정이 끝난 후에 각 팀은 리그 사무소에 획득한 점수를 보고하게 됩니다. 리그 사무소는 각 팀이 보고한 점수가 실수가 없는지 확실히 해두고 싶습니다. 즉, 보고한 점수가 유효한지 아닌지 알고 싶은 것이고, 이 말은 리그 룰에 따르는 경우 이 점수들을 각 팀에 할당하는 것이 가능해야 합니다.

주어진 n 개의 정수들은 각 팀에서 보고한 점수들로 이 점수들이 유효한지 아닌지 알아내는 프로그램을 작성해야 합니다.

입력

프로그램은 표준 입력에서 읽어야 합니다. 입력은 두 줄로 이루어져 있고, 첫째 줄은 하나의 정수 n ($2 \leq n \leq 10,000$) 이고, 팀의 개수를 의미합니다. 다음 줄은 각 팀에서 보고한 점수들입니다. 각 정수는 0 보다 같거나 크고 $n - 1$ 보다 같거나 작습니다.

출력

프로그램은 표준 출력에 써야 합니다. 보고한 점수들이 유효한 경우라면 1 을 출력하고, 그렇지 않으면 -1 을 출력합니다.

예제 입력 1 복사

```
4
0 2 1 3
```

예제 입력 2 복사

```
4
3 3 0 0
```

예제 출력 1 복사

```
1
```

예제 출력 2 복사

```
-1
```

Problem C

선발 명단

제한 시간 : 1초, 메모리 제한 : 128MB

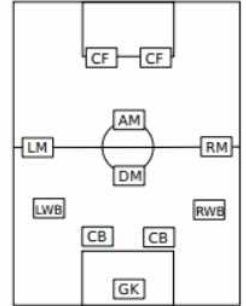
문제

챔피언스 리그 결승전을 앞두고 있는 맨체스터 유나이티드의 명장 퍼거슨 감독은 이번 경기에 4-4-2 다이아몬드 전술을 사용하려고 한다.

오늘 결승전에 뭘 선발 선수 11명은 미리 골라두었지만, 어떤 선수를 어느 포지션에 배치해야 할지 아직 결정하지 못했다.

수석코치 마이크 펠란은 11명의 선수가 각각의 포지션에서의 능력을 0부터 100가지의 정수로 수치화 했다. 0은 그 선수가 그 포지션에 적합하지 않다는 뜻이다.

이때, 모든 선수의 포지션을 정하는 프로그램을 작성하시오. 모든 포지션에 선수를 배치해야 하고, 각 선수는 능력치가 0인 포지션에 배치될 수 없다.



입력

입력은 여러 개의 테스트 케이스로 이루어져 있다. 첫째 줄에는 테스트 케이스의 개수 C 가 주어진다. 각각의 케이스는 11줄로 이루어져 있고, i 번 줄은 0과 100사이의 11개의 정수 s_{ij} 를 포함하고 있다. s_{ij} 는 i 번 선수가 j 번 포지션에서 뭘 때의 능력이다. 모든 선수에게 적합한 포지션의 수는 최대 5개이다. (능력치가 0보다 크다)

출력

각각의 테스트 케이스에 대해서, 모든 포지션의 선수를 채웠을 때, 능력치의 합이 최대값을 출력한다. 항상 하나 이상의 올바른 라인업을 만들 수 있다.

예제 입력 1 복사

```
1
100 0 0 0 0 0 0 0 0 0 0
0 80 70 70 60 0 0 0 0 0 0
0 40 90 90 40 0 0 0 0 0 0
0 40 85 85 33 0 0 0 0 0 0
0 70 60 60 85 0 0 0 0 0 0
0 0 0 0 0 95 70 60 60 0 0
0 45 0 0 0 80 90 50 70 0 0
0 0 0 0 0 40 90 90 40 70 0
0 0 0 0 0 0 50 70 85 50 0
0 0 0 0 0 0 66 60 0 80 80
0 0 0 0 0 0 50 50 0 90 88
```

예제 출력 1 복사

```
970
```

Problem D

암호코드

제한 시간 : 2초, 메모리 제한 : 128MB

문제

상근이와 선영이가 다른 사람들이 남매간의 대화를 듣는 것을 방지하기 위해서 대화를 서로 암호화하기로 했다. 그래서 다음과 같은 대화를 했다.

- 상근: 그냥 간단히 암호화 하자. A를 1이라고 하고, B는 2로, 그리고 Z는 26으로 하는거야.
- 선영: 그럼 안돼. 만약, "BEAN"을 암호화하면 25114가 나오는데, 이것 다시 글자로 바꾸는 방법은 여러 가지가 있어.
- 상근: 그럼네. 25114를 다시 영어로 바꾸면, "BEAAD", "YAAD", "YAN", "YKD", "BEKD", "BEAN" 총 6가지가 나오는데, BEAN이 맞는 단어라는건 쉽게 알수 있잖아?
- 선영: 예가 적절하지 않았네 $\pi\pi$ 만약 내가 500자리 글자를 암호화 했다고 해봐. 그 때는 나올 수 있는 해석이 정말 많은데, 그걸 언제 다해봐?
- 상근: 얼마나 많은데?
- 선영: 구해보자!

어떤 암호가 주어졌을 때, 그 암호의 해석이 몇 가지가 나올 수 있는지 구하는 프로그램을 작성하시오.

입력

첫째 줄에 5000자리 이하의 암호가 주어진다. 암호는 숫자로 이루어져 있다.

출력

나올 수 있는 해석의 가짓수를 구하시오. 정답이 매우 클 수 있으므로, 1000000으로 나눈 나머지를 출력한다.

암호가 잘못되어 암호를 해석할 수 없는 경우에는 0을 출력한다.

예제 입력 1 복사

예제 출력 1 복사

Problem E

거의 최단 경로

제한 시간 : 1초, 메모리 제한 : 256MB

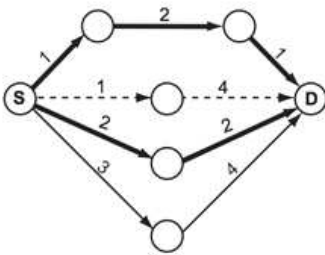
문제

요즘 많은 자동차에서는 GPS 네비게이션 장비가 설치되어 있다. 네비게이션은 사용자가 입력한 출발점과 도착점 사이의 최단 경로를 검색해 준다. 하지만, 교통 상황을 고려하지 않고 최단 경로를 검색하는 경우에는 극심한 교통 정체를 경험할 수 있다.

상근이는 오직 자기 자신만 사용 가능한 네비게이션을 만들고 있다. 이 네비게이션은 절대로 최단 경로를 찾아주지 않는다. 항상 거의 최단 경로를 찾아준다.

거의 최단 경로란 최단 경로에 포함되지 않는 도로로만 이루어진 경로 중 가장 짧은 것을 말한다.

예를 들어, 도로 지도가 아래와 같을 때를 생각해보자. 원은 장소를 의미하고, 선은 단방향 도로를 나타낸다. 시작점은 S, 도착점은 D로 표시되어 있다. 굵은 선은 최단 경로를 나타낸다. (아래 그림에 최단 경로는 두 개가 있다) 거의 최단 경로는 점선으로 표시된 경로이다. 이 경로는 최단 경로에 포함되지 않은 도로로 이루어진 경로 중 가장 짧은 경로이다. 거의 최단 경로는 여러 개 존재할 수도 있다. 예를 들어, 아래 그림의 길이가 3인 도로의 길이가 1이라면, 거의 최단 경로는 두 개가 된다. 또, 거의 최단 경로가 없는 경우도 있다.



입력

입력은 여러 개의 테스트 케이스로 이루어져 있다. 각 테스트 케이스의 첫째 줄에는 장소의 수 N ($2 \leq N \leq 500$)과 도로의 수 M ($1 \leq M \leq 10^4$)가 주어진다. 장소는 0부터 $N-1$ 번까지 번호가 매겨져 있다. 둘째 줄에는 시작점 S 와 도착점 D 가 주어진다. ($S \neq D$; $0 \leq S, D < N$) 다음 M 개 줄에는 도로의 정보 U, V, P 가 주어진다. ($U \neq V$; $0 \leq U, V < N$; $1 \leq P \leq 10^3$) 이 뜻은 U 에서 V 로 가는 도로의 길이가 P 라는 뜻이다. U 에서 V 로 가는 도로는 최대 한 개이다. 또, U 에서 V 로 가는 도로와 V 에서 U 로 가는 도로는 다른 도로이다.

입력의 마지막 줄에는 0이 두 개 주어진다.

출력

각 테스트 케이스에 대해서, 거의 최단 경로의 길이를 출력한다. 만약, 거의 최단 경로가 없는 경우에는 -1을 출력한다.

예제 입력 1 복사

```
7 9
0 6
0 1 1
0 2 1
0 3 2
0 4 3
1 5 2
2 6 4
3 6 2
4 6 4
5 6 1
4 6
0 2
0 1 1
1 2 1
1 3 1
3 2 1
2 0 3
3 0 2
6 8
0 1
0 1 1
0 2 2
0 3 3
2 5 3
3 4 2
4 1 1
5 1 1
3 0 1
0 0
```

예제 출력 1 복사

```
5
-1
6
```

Problem F

문자열 뽑기

제한 시간 : 2초, 메모리 제한 : 128MB

문제

a와 b로만 이루어진 문자열 s가 있다. 그룹은 같은 글자로 이루어진 가장 긴 연속 부분 문자열이다. 길이가 2 이상인 s의 모든 그룹 g는 제거할 수 (뽑을 수) 있고, 남은 왼쪽 부분과 오른쪽 부분을 연결해서 새 문자열을 만들 수 있다. 이러한 과정은 문자열이 빈 문자열이 되거나, 길이가 2 이상인 그룹이 없을 때 까지 계속한다.

예를 들어, s = babbbbaaabb일 때, s에는 그룹이 다섯 개 있다. s는 다음과 같은 단계를 거쳐서 빈 문자열로 바꿀 수 있다. (밑 줄이 그어져 있는 그룹이 뽑히는 그룹)

babbbbaaabb → baaaabb → bbb → 빈 문자열

하지만, 아래와 같은 단계를 거친다면, 빈 문자열로 바꿀 수 없다.

babbbbaaabb → babbbbaaa → baaaa → b

문자열이 주어졌을 때, 적절한 과정을 거쳐 빈 문자로 바꿀 수 있는지 없는지를 구하는 프로그램을 작성하시오.

입력

첫째 줄에 테스트 케이스의 개수 T가 주어진다. 각 테스트 케이스는 a와 b로 이루어진 문자열로 이루어져 있다. 문자열의 길이는 1보다 크거나 같고, 25보다 작거나 같다.

출력

각 테스트 케이스에 대해서, 입력으로 주어진 문자열을 빈 문자열로 바꿀 수 있으면 1을, 없으면 0을 출력한다.

예제 입력 1 복사

```
3
babbbbaaabb
aabbaabb
abab
```

예제 출력 1 복사

```
1
1
0
```


Problem G

화학 제품

제한 시간 : 1초, 메모리 제한 : 128MB

문제

상근이는 각기 다른 병에 담긴 세 화학 물질 A, B, C를 가지고 있다. 두 화학 물질을 같은 양만큼 혼합하면, 화학 제품을 얻을 수 있다.

A와 B를 혼합하면 AB가 되고, B와 C를 혼합하면 BC, C와 A를 혼합하면 CA가 된다. (A 하나와 B 하나를 혼합하면 AB 하나를 얻게 된다) AB, BC, CA의 가격은 모두 다르다. 따라서, 만드는 화학 제품에 따라서 얻는 이익은 달라진다. 항상 정수 단위 만큼 두 화학 물질을 혼합할 수 있다.

예를 들어, A를 5만큼, B를 3만큼, C를 7만큼 가지고 있고, 각 화학 제품의 가격은 다음과 같은 경우를 생각해보자.

화학 제품	단위 가격
AB	\$100
BC	\$30
CA	\$80

A 하나와 B하나를 혼합해 AB 하나를 만들면 \$100을 얻게 된다. 그 다음 B 2개와 C 2개를 혼합하면 \$60을 얻게 되고, C 4개와 A 4개를 혼합해 \$320을 얻게 된다. 총 이익은 \$480이 되고 이 이익이 가능한 이익 중 최댓값이다. 한 화학 물질은 모두 사용하지 않을 수도 있다. 이 예제에서는 C를 하나 사용하지 않았다.

다른 예로, A가 3개, B가 3개, C가 3개가 있고, AB, BC, CA의 가격이 모두 \$100인 경우를 생각해보자. A 2개와 B 2개를 혼합해 \$200을 얻고, B 1개와 C 1개를 혼합해 \$100을 얻을 수 있다. 마지막으로 C 1개와 A 1개를 섞으면 \$100을 얻는다. 총 이익은 \$400이 되고, 이 값도 가능한 이익 중 최댓값이다.

상근이가 가지고 있는 A, B, C의 양과 AB, BC, CA의 가격이 주어졌을 때, 상근이가 얻을 수 있는 최대 이익을 구하는 프로그램을 작성하시오.

입력

첫째 줄에 테스트 케이스의 개수 T가 주어진다. 각 테스트 케이스는 두 줄로 이루어져 있다. 첫째 줄에는 상근이가 가지고 있는 A, B, C의 양이 주어진다. 둘째 줄에는 AB, BC, CA의 가격이 주어진다. 입력으로 주어지는 모든 숫자는 정수이며, 1보다 크거나 같고, 1,000보다 작거나 같다.

출력

각 테스트 케이스 마다 최대 이익을 출력한다.

예제 입력 1 복사

```
2
5 3 7
100 30 80
3 3 3
100 100 100
```

예제 출력 1 복사

```
480
400
```

Problem H

최종 순위

제한 시간 : 1초, 메모리 제한 : 256MB

문제

올해 ACM-ICPC 대전 인터넷 예선에는 총 n 개의 팀이 참가했다. 팀은 1번부터 n 번까지 번호가 매겨져 있다. 놀랍게도 올해 참가하는 팀은 작년에 참가했던 팀과 동일하다.

올해는 인터넷 예선 본부에서는 최종 순위를 발표하지 않기로 했다. 그 대신에 작년에 비해서 상대적인 순위가 바뀐 팀의 목록만 발표하려고 한다. (작년에는 순위를 발표했다) 예를 들어, 작년에 팀 13이 팀 6보다 순위가 높았는데, 올해 팀 6이 팀 13보다 순위가 높다면, (6, 13)을 발표할 것이다.

창영이는 이 정보만을 가지고 올해 최종 순위를 만들어보려고 한다. 작년 순위와 상대적인 순위가 바뀐 모든 팀의 목록이 주어졌을 때, 올해 순위를 만드는 프로그램을 작성하시오. 하지만, 본부에서 발표한 정보를 가지고 확실한 올해 순위를 만들 수 없는 경우가 있을 수도 있고, 일관성이 없는 잘못된 정보일 수도 있다. 이 두 경우도 모두 찾아내야 한다.

입력

첫째 줄에는 테스트 케이스의 개수가 주어진다. 테스트 케이스는 100개를 넘지 않는다. 각 테스트 케이스는 다음과 같이 이루어져 있다.

- 팀의 수 n 을 포함하고 있는 한 줄. ($2 \leq n \leq 500$)
- n 개의 정수 t_i 를 포함하고 있는 한 줄. ($1 \leq t_i \leq n$) t_i 는 작년에 i 등을 한 팀의 번호이다. 1등이 가장 성적이 높은 팀이다. 모든 t_i 는 서로 다르다.
- 상대적인 등수가 바뀐 쌍의 수 m ($0 \leq m \leq 25000$)
- 두 정수 a_i 와 b_i 를 포함하고 있는 m 줄. ($1 \leq a_i < b_i \leq n$) 상대적인 등수가 바뀐 두 팀이 주어진다. 같은 쌍이 여러 번 발표되는 경우는 없다.

출력

각 테스트 케이스에 대해서 다음을 출력한다.

- n 개의 정수를 한 줄에 출력한다. 출력하는 숫자는 올해 순위이며, 1등팀부터 순서대로 출력한다. 만약, 확실한 순위를 찾을 수 없다면 "?"를 출력한다. 데이터에 일관성이 없어서 순위를 정할 수 없는 경우에는 "IMPOSSIBLE"을 출력한다.

예제 입력 1 복사

```
3
5
5 4 3 2 1
2
2 4
3 4
3
2 3 1
0
4
1 2 3 4
3
1 2
3 4
2 3
```

예제 출력 1 복사

```
5 3 2 4 1
2 3 1
IMPOSSIBLE
```

Problem I

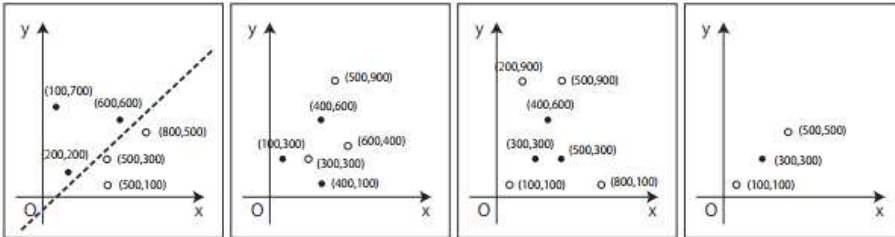
점 분리

제한 시간 : 1초, 메모리 제한 : 128MB

문제

평면 위에 여러 개의 검정 점과 흰 점이 있다. 이때, 길이가 무한대인 직선을 그려 흰 점과 검은 점을 분리하려고 한다. 직선은 어떤 점과도 만나면 안 된다. 직선으로 인해서 나누어지는 두 그룹 중 한 그룹에는 흰 점만 있어야 하고, 다른 그룹에는 검은 점만 있어야 한다.

아래 그림에서 제일 왼쪽 예제는 점선으로 표시된 직선으로 두 점을 나눌 수 있다. 하지만 나머지 예제는 직선으로 점을 분리할 수 없다.



흰 점과 검은 점의 좌표가 주어졌을 때, 직선으로 점을 분리할 수 있는지 없는지를 알아내는 프로그램을 작성하시오.

입력

첫째 줄에는 테스트 케이스의 개수 T 가 주어진다. 각 테스트 케이스의 첫째 줄에 검정 점의 개수 n 과 흰 점의 개수 m 이 공백으로 구분되어 주어진다. n 과 m 은 100보다 작거나 같다. 다음 줄부터 n 개의 줄에는 검정 점의 좌표가 공백으로 구분되어 주어진다. 그 다음 m 개의 줄에는 흰 점의 좌표가 주어진다.

모든 점의 x, y 좌표의 값은 0보다 크거나 같고, 10000보다 작거나 같다. 또한, 같은 위치에 점이 2개 이상 있는 경우는 없다.

출력

각각의 테스트 케이스에 대해서, 점을 문제의 설명대로 분리할 수 있으면 YES를, 아니면 NO를 출력한다.

Problem J

Vacation Plans

제한 시간 : 1초, 메모리 제한 : 512MB

문제

A group of people plan to have a vacation in a remote island near the equator over the winter holiday. All members of the group live in different countries and the destination island is only reachable via airplane. Therefore, each member has to go to their own country's airport to take a flight to the destination island. We assume that each country has only one airport. Now, for the sake of holiday spirit, all group members agree to start the journey on the same day from their home cities. Also, they plan to be at their country's airports on the same day, which is not necessarily the first day of their travel. However, the airports might not be in each member's home city, so some members may have to travel to another city over the course of a few days. On the first day of the winter holiday, all members are in their respective home cities. Then, every day, each member has to individually decide between traveling to an adjacent city (meaning that the two cities are connected by a road), or staying the day in the city they are currently in. Since the travelling cost between two adjacent cities and the cheapest hotel price in each city are already known to the world, one knows exactly how much it will cost either to move to an adjacent city or to stay in that city for each day. All members want to have as much money as possible for the vacation on the island, so they pool their money together and decide to calculate the travel plans as a group. Their goal is that all the members end up at their designated countries' airports on the same day, while spending the least amount of money.

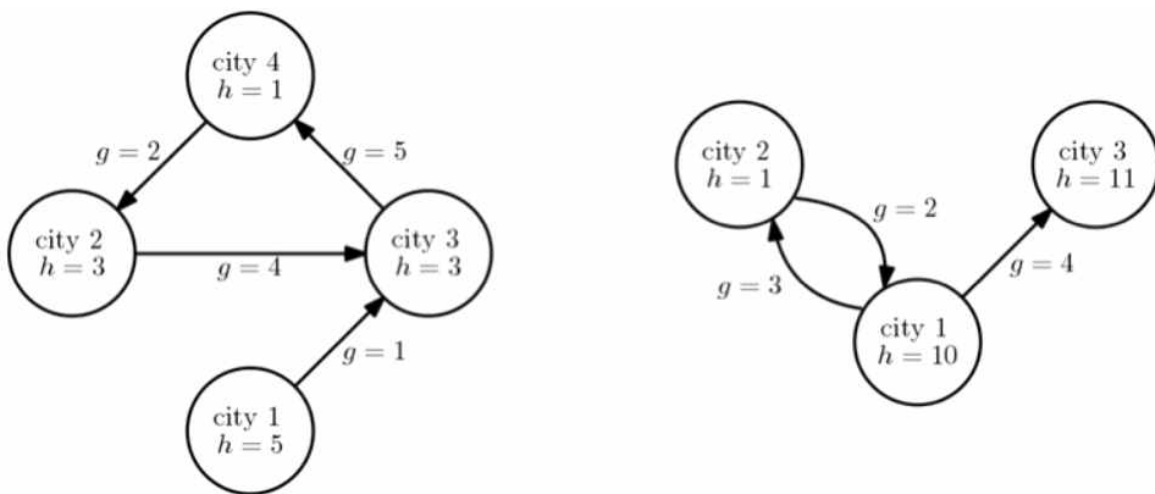


Figure L.1. Two members' country layouts, where the designated airports are in cities 4 and 3, respectively. The home city is 1 for both. g denotes the travelling cost between two cities and h denotes the cheapest hotel price.

Consider an example in Figure L.1 with two members and their designated airports being in cities 4 and 3, respectively. The cheapest travel plans with both members starting in their hometowns (always city 1) would be: (day 1) member-1 moves to city 3, member-2 moves to city 2; (day 2) member-1 moves to city 4, member-2 moves to city 1; (day 3) member-1 stays at city 4, member-2 moves to city 3. This has cost $(1 + 5 + 1) + (3 + 2 + 4) = 16$.

Note that the travelling cost between two cities is not necessary symmetric. Additionally, no city has a road connecting it to itself. You can always assume that, in each country, there is at least one path from home to the designated airport.

You should write a program that finds the minimum cost required to get all members from their home city to their country's designated airport such that everyone is at the airport on the same day. Note that every day, one has to either move to an adjacent city or stays at the current city hotel.

입력

Your program is to read from standard input. The first line contains a single integer $1 \leq p \leq 3$ denoting the number of people in the group that will be going on vacation (and therefore also the number of countries to be considered). Then the next input represents each country as follows: The next line consist of two integers $1 \leq n \leq 50$ and $n - 1 \leq m \leq 4 \times n$, corresponding to the number of cities and roads in the country. The next n lines contain exactly one integer, $0 \leq h \leq 1,000,000$, representing the cheapest hotel cost of each city (the costs are given in order from city 1 to n for each country). The next lines contain the road information. Each road is represented by three integers, separated by spaces: $1 \leq u, v \leq n$ and $0 \leq g \leq 1,000,000$, which are the cities at the start and the end of the road, respectively, and the cost to travel on the road from u to v . Finally, one more line is given containing a single integer $1 \leq a \leq n$, denoting the city containing that country's airport. In each country, city 1 is each one's home city.

출력

Your program is to write to standard output. You should output exactly one line containing a single integer equal to the minimum cost required to get all members from their home city to their country's designated airport such that everyone is at the airport on the same day.

예제 입력 1 복사

```
2
4 4
5
3
3
1
1 3 1
2 3 4
3 4 5
4 2 2
4
3 3
10
1
11
1 2 3
1 3 4
2 1 2
3
```

예제 출력 1 복사

16

예제 입력 2 복사

```
2
4 4
2
8
15
1
1 2 5
2 3 7
3 4 10
4 1 3
3
5 4
1
1
1
1
1
1 2 3
2 3 5
3 4 7
4 5 1
5
```

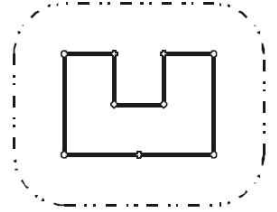
예제 출력 2 복사

```
32
```

Problem K

맹독 방벽

제한 시간 : 1초, 메모리 제한 : 128MB



문제

화학 제국의 왕 성준이는 계속되는 이웃나라의 침범으로부터 자유로워지기 위해 자국의 자랑 **화학 방벽**을 건설하기로 마음먹었다. 이 방벽은 근처에 다가오는 생명체에게 해로운 독성을 내뿜어서 더이상 다른 나라들이 열썬도 못하게 만들 것이다!

그러나 이 방벽은 만들기 까다롭기에 가능한 한 적게 지어야 하며, 자국민들에게도 악영향을 끼칠 수 있으므로 자국의 모든 건물들로부터 L 이상의 거리를 유지해야만 한다.

자국의 건물들의 좌표가 주어졌을 때, 모든 건물들로부터 L 이상의 거리를 두면서 모든 건물을 한번에 두르는 방법의 최소 길이를 구하시오.

입력

첫 번째 줄에 건물의 수 N 과 거리 L 이 주어진다. ($3 \leq N \leq 1000, 1 \leq L \leq 1000$)

다음 N개의 줄에 걸쳐 건물의 좌표 X_i 와 Y_i 가 정수로 주어진다. ($-10000 \leq X_i, Y_i \leq 10000$) 모든 건물의 좌표는 다르며, 건물은 충분히 작아서 점과 같다고 생각해도 좋다. 방벽은 자신들끼리 교차해서는 안 되며 끊어져서도 안 된다.

출력

첫째 줄에 답을 정수 단위로 반올림하여 출력한다.

예제 입력 1 복사

9	100
200	400
300	400
300	300
400	300
400	400
500	400
500	200
350	200
200	200

예제 출력 1 복사

1628

Problem L

철로

제한 시간 : 1초, 메모리 제한 : 512MB

문제

집과 사무실을 통근하는 n 명의 사람들이 있다. 각 사람의 집과 사무실은 수평선 상에 있는 서로 다른 점에 위치하고 있다. 임의의 두 사람 A, B에 대하여, A의 집 혹은 사무실의 위치가 B의 집 혹은 사무실의 위치와 같을 수 있다. 통근을 하는 사람들의 편의를 위하여 일직선 상의 어떤 두 점을 잇는 철로를 건설하여, 기차를 운행하려고 한다. 제한된 예산 때문에, 철로의 길이는 d 로 정해져 있다. 집과 사무실의 위치 모두 철로 선분에 포함되는 사람들의 수가 최대가 되도록, 철로 선분을 정하고자 한다.

양의 정수 d 와 n 개의 정수쌍, (h_i, o_i) , $1 \leq i \leq n$,이 주어져 있다. 여기서 h_i 와 o_i 는 사람 i 의 집과 사무실의 위치이다. 길이 d 의 모든 선분 L 에 대하여, 집과 사무실의 위치가 모두 L 에 포함되는 사람들의 최대 수를 구하는 프로그램을 작성하시오.

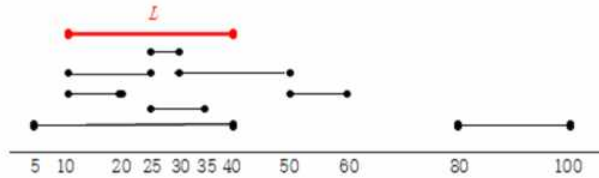


그림 1. 8 명의 집과 사무실의 위치

그림 1 에 있는 예를 고려해보자. 여기서 $n = 8$, $(h_1, o_1) = (5, 40)$, $(h_2, o_2) = (35, 25)$, $(h_3, o_3) = (10, 20)$, $(h_4, o_4) = (10, 25)$, $(h_5, o_5) = (30, 50)$, $(h_6, o_6) = (50, 60)$, $(h_7, o_7) = (30, 25)$, $(h_8, o_8) = (80, 100)$ 이고, $d = 30$ 이다. 이 예에서, 위치 10 과 40 사이의 빨간색 선분 L 이, 가장 많은 사람들에게 대하여 집과 사무실 위치 모두 포함되는 선분 중 하나이다. 따라서 답은 4 이다.

입력

입력은 표준입력을 사용한다. 첫 번째 줄에 사람 수를 나타내는 양의 정수 n ($1 \leq n \leq 100,000$)이 주어진다. 다음 n 개의 각 줄에 정수 쌍 (h_i, o_i) 가 주어진다. 여기서 h_i 와 o_i 는 $-100,000,000$ 이상, $100,000,000$ 이하의 서로 다른 정수이다. 마지막 줄에, 철로의 길이를 나타내는 정수 d ($1 \leq d \leq 200,000,000$)가 주어진다.

출력

출력은 표준출력을 사용한다. 길이 d 의 임의의 선분에 대하여, 집과 사무실 위치가 모두 그 선분에 포함되는 사람들의 최대 수를 한 줄에 출력한다.

예제 입력 1 복사

```
8
5 40
35 25
10 20
10 25
30 50
50 60
30 25
80 100
30
```

예제 출력 1 복사

```
4
```

예제 입력 2 복사

```
4
20 80
70 30
35 65
40 60
10
```

예제 출력 2 복사

```
0
```

예제 입력 3 복사

```
5
-5 5
30 40
-5 5
50 40
5 -5
10
```

예제 출력 3 복사

```
3
```

Problem M

Networks with Unidirectional Links

제한 시간 : 1초, 메모리 제한 : 128MB

문제

There is a multicomputer system that consists of a number of nodes, each with its own memory. The nodes in the system are interconnected via unidirectional communication links. The interconnection network of the system, representing the way in which the nodes are connected together, is modeled as a directed graph, where vertices and directed edges respectively represent nodes and unidirectional links of the network. Figure 1 shows an example of interconnection networks with fourteen nodes interconnected via nineteen unidirectional links.

Linear arrays and rings are two of the important computational structures in interconnection networks. In a linear array, each node except for the extreme ends has two neighboring nodes, one to its left and one to its right, where the two nodes at the ends are connected to their single neighbor. If the nodes at either end are connected, we refer to it as a ring. More specifically, the nodes of a linear array with k nodes can be numbered from 0 to $k-1$ so that there exists a unidirectional link from node i to node $i+1$ for every $0 \leq i < k-1$. The linear array becomes a ring if we add a unidirectional link from the node $k-1$ to the node 0.

To support parallel applications that require one of the two aforementioned computational structures, we need to decompose the system into several subsystems each of whose interconnection networks is a ring or a linear array. The subsystems should be pairwise node-disjoint, i.e., no two subsystems share a common node. According to a recent report, a ring composed of k nodes is worth k dollars while a linear array of k nodes is worth $k-1$ dollars. This motivates the study on how to decompose the system into subsystems in order to maximize profits. You are to write a program for decomposing the interconnection network of our multicomputer system into rings and/or linear arrays whose total value is the maximum possible. Note that a linear array may have only one node and in that case, its value is zero dollar.

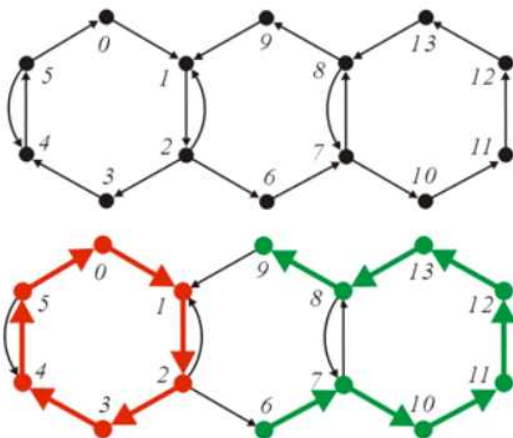


Figure 1. An interconnection network and its decomposition into a ring of six nodes (red) and a linear array of eight nodes (green). The total value of this decomposition is thirteen dollars, which is the maximum possible.

입력

Your program is to read from standard input. The input consists of T test cases, where the positive integer T is given in the first line of the input, followed by the description of each test case. The first line of a test case contains two positive integers n and m , respectively indicating the numbers of nodes and unidirectional links in an interconnection network, in which we assume $n \leq 1,000$ and $m \leq 50,000$. The nodes are indexed 0 to $n-1$. In the following m lines, each line contains two integers u and v , which represent a unidirectional link from node u to node v . The two integers given in a single line are always separated by a space.

출력

Your program is to write to standard output. Print exactly one line for each test case. The line should contain an integer, representing the maximum total value in dollars that can be achieved by decomposing the interconnection network into rings and/or linear arrays.

예제 입력 1 복사

```
3
4 3
3 2
1 0
2 3
6 6
0 1
1 2
2 3
3 1
3 4
4 5
14 19
0 1
1 2
2 3
3 4
4 5
5 0
5 4
2 1
2 6
6 7
7 8
8 9
9 1
8 7
7 10
10 11
11 12
12 13
13 8
```

예제 출력 1 복사

```
3
5
13
```

Problem M

단방향 링크 네트워크

제한 시간 : 1초, 메모리 제한 : 128MB

문제

멀티 컴퓨터 시스템은 여러 개의 노드로 이루어져 있고, 각 노드는 자체 메모리를 가지고 있다. 시스템 상의 각 노드는 단방향 커뮤니케이션 링크로 서로 연결되어 있다. 시스템의 상호 접속 네트워크는 방향 그래프로 나타낼 수 있다. 정점은 노드를 나타내며, 간선은 네트워크의 단방향 링크를 나타낸다. 그림 1은 14개의 노드가 19개의 단방향 링크로 서로 연결된 상호 접속 네트워크를 나타낸다.

선형 배열과 링은 상호 접속 네트워크에서 가장 중요한 두 구조이다. 선형 배열에서 각 양 끝 노드를 제외한 모든 노드는 인접한 노드를 두 개 가진다. 한 노드는 왼쪽, 다른 노드는 오른쪽에 있으며, 두 노드는 모두 공통된 이웃을 통해 연결되어 있다. 이 구조에서 양 끝 점이 연결되었을 때, 그 구조를 링이라고 한다. 즉, 0번부터 $k-1$ 번까지, k 개 노드로 이루어진 선형 배열은 모든 $0 \leq i < k-1$ 에 대해서 노드 i 에서 노드 $i+1$ 로 단방향 링크가 존재해야 한다. 여기서 노드 $k-1$ 과 노드 0 을 연결하는 단방향 링크를 추가하면 링이 된다.

병렬 어플리케이션을 지원하려면, 앞에서 설명한 두 구조 중 하나가 필요하다. 따라서, 시스템을 여러 개의 서브 시스템으로 분해해야 한다. 각 서브 시스템은 링 또는 선형 배열이어야 한다. 서브 시스템은 공통된 노드를 공유할 수 없다.

최근에 발표된 한 보고서에 의하면, k 개 노드로 이루어진 링의 가치는 k 달러이고, k 개의 노드로 이루어진 선형 배열의 가치는 $k-1$ 달러이다. 따라서, 이익을 최대로 하면서, 시스템을 서브 시스템으로 분해하는 연구가 진행 중이다.

한 멀티 컴퓨터 시스템의 상호 접속 네트워크가 주어졌을 때, 이 시스템을 링 과/또는 선형 배열로 분해해서 가치를 최대로 하는 프로그램을 작성하시오. 노드를 1개만 가지는 선형 배열도 존재할 수 있다. 이때, 가치는 0 달러이다.

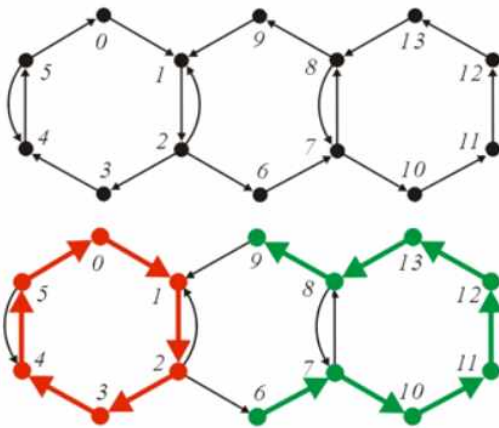


그림 1. 상호 연결 네트워크를 노드 6개로 이루어진 링 (빨간색)과 노드 8개로 이루어진 선형 배열 (초록색)으로 분해한 그림이다. 이 분해의 총 가치는 13달러이고, 이 값은 가능한 방법중 최댓값이다.

입력

첫째 줄에 테스트 케이스의 개수 T 가 주어진다. 각 테스트 케이스의 첫째 줄에는 상호 네트워크를 이루는 노드의 수 n 과 단방향 링크의 수 m 이 주어진다. ($n \leq 1,000$, $m \leq 50,000$) 노드는 0번부터 $n-1$ 번까지 번호가 매겨져 있다. 다음 m 개 줄에는 u 에서 v 로 향하는 단방향 링크를 나타내는 두 정수 u 와 v 가 주어진다. 두 정수는 한 줄에 주어지며, 항상 공백으로 구분되어져 있다.

출력

각 테스트 케이스 마다, 한 줄을 출력한다. 이 줄에는 정수를 하나 포함해야 하며, 정수는 입력으로 주어진 상호 접속 네트워크를 링 과/또는 선형 배열로 분해했을 때 얻을 수 있는 가치의 최댓값이다.

예제 입력 1 복사

```
3
4 3
3 2
1 0
2 3
6 6
0 1
1 2
2 3
3 1
3 4
4 5
14 19
0 1
1 2
2 3
3 4
4 5
5 0
5 4
2 1
2 6
6 7
7 8
8 9
9 1
8 7
7 10
10 11
11 12
12 13
13 8
```

예제 출력 1 복사

```
3
5
13
```

Problem N

랜선 자르기

제한 시간 : 2초, 메모리 제한 : 128MB

문제

집에서 시간을 보내던 오영식은 박성원의 부름을 받고 급히 달려왔다. 박성원이 캠프 때 쓸 N개의 랜선을 만들어야 하는데 너무 바빠서 영식에게 도움을 청했다.

이미 오영식은 자체적으로 K개의 랜선을 가지고 있다. 그러나 K개의 랜선은 길이가 제각각이다. 박성원은 랜선을 모두 N개의 같은 길이의 랜선으로 만들고 싶었기 때문에 K개의 랜선을 잘라서 만들어야 한다. 예를 들어 300cm 짜리 랜선에서 140cm 짜리 랜선을 두 개 잘라내면 20cm은 버려야 한다. (이미 자른 랜선은 붙일 수 없다.)

편의를 위해 랜선을 자르거나 만들 때 손실되는 길이는 없다고 가정하며, 기존의 K개의 랜선으로 N개의 랜선을 만들 수 없는 경우는 없다고 가정하자. 그리고 자를 때는 항상 센티미터 단위로 정수길이만큼 자른다고 가정하자. N개보다 많이 만드는 것도 N개를 만드는 것에 포함된다. 이때 만들 수 있는 최대 랜선의 길이를 구하는 프로그램을 작성하시오.

입력

첫째 줄에는 오영식이 이미 가지고 있는 랜선의 개수 K, 그리고 필요한 랜선의 개수 N이 입력된다. K는 1이상 10,000이하의 정수이고, N은 1이상 1,000,000이하의 정수이다. 그리고 항상 $K \leq N$ 이다. 그 후 K줄에 걸쳐 이미 가지고 있는 각 랜선의 길이가 센티미터 단위의 정수로 입력된다. 랜선의 길이는 $2^{31}-1$ 보다 작거나 같은 자연수이다.

출력

첫째 줄에 N개를 만들 수 있는 랜선의 최대 길이를 센티미터 단위의 정수로 출력한다.

예제 입력 1 복사

예제 출력 1 복사

```
4 11
802
743
457
539
```

```
200
```

힌트

802cm 랜선에서 4개, 743cm 랜선에서 3개, 457cm 랜선에서 2개, 539cm 랜선에서 2개를 잘라내 모두 11개를 만들 수 있다.

Problem O

Closest Pair

제한 시간 : 1.5초, 메모리 제한 : 512MB

문제

Given two sets P and Q of finitely many points in the plane, a closest pair of P and Q is a pair (p, q) of points $p \in P$ and $q \in Q$ such that the distance between p and q is the minimum among all pairs (p', q') with $p' \in P$ and $q' \in Q$.

Specifically, in this problem, by the *distance* between two points a and b in the plane, we mean:

$$d(a, b) = |x_a - x_b| + |y_a - y_b|$$

where x_a and y_a denote the x - and y -coordinates of point a , and x_b and y_b denote the x - and y -coordinates of point b . Then, a pair (p, q) with $p \in P$ and $q \in Q$ is a closest pair of P and Q if and only if the following holds:

$$d(p, q) = \min\{d(p', q') \mid p' \in P \text{ and } q' \in Q\}$$

Given two sets P and Q , write a program that computes the distance between a closest pair of P and Q and the number of distinct closest pairs of P and Q .

Note that you can assume the following on the input points in P and Q :

1. All the points in P lie on the horizontal line $y = c_1$ while all the points in Q lie on the horizontal line $y = c_2$ for some integers c_1 and c_2 .
2. No two input points in P have the same coordinates; no two input points in Q have the same coordinates.

입력

Your program is to read from standard input. The input consists of four lines. The first line contains two integers, n ($1 \leq n \leq 500,000$) and m ($1 \leq m \leq 500,000$), where n is the number of points in set P and m is the number of points in set Q . In the second line, two integers c_1 and c_2 ($-10^8 \leq c_1, c_2 \leq 10^8$) are given in order, separated by a single space. In the third line, n distinct integers between -10^8 and 10^8 , inclusively, are given, separated by a single space, that are the x -coordinates of the points in set P , while their y -coordinates are all the same as c_1 . In the fourth line, m distinct integers between -10^8 and 10^8 , inclusively, are given, separated by a single space, that are the x -coordinates of the points in set Q , while their y -coordinates are all the same as c_2 .

출력

Your program is to write to standard output. Print exactly one line for the input. The line should contain two integers, separated by a single space, that represent the distance between a closest pair of P and Q and the number of closest pairs of P and Q in this order.

예제 입력 1 복사

```
3 4
1 -3
3 0 6
-2 5 4 2
```

예제 출력 1 복사

```
5 3
```

예제 입력 2 복사

```
5 5
1 2
-4 -10 -2 0 -1
3 18 0 1 5
```

예제 출력 2 복사

```
1 1
```

Problem P

Stickers

제한 시간 : 1초, 메모리 제한 : 256MB

문제

Nancy, your little sister, has a sheet of $2n$ stickers of rectangular shape that are arranged in 2 rows and n columns. See Figure 1(a). Nancy wants to decorate her desk with the stickers. But the quality of the stickers is poor, and tearing off one sticker from the sheet spoils the stickers sharing an edge with it. So, Nancy must lose the stickers above, below, to the left of, and to the right of the sticker she tears off.



Figure 1. A sheet of 10 stickers in 2 rows

Nancy had no idea about what to do. You looked at her and suggested that she should score each sticker and try to choose a possible set of stickers that maximizes the total score. Nancy marked scores to all the $2n$ stickers as in Figure 1(b). And Nancy had no idea, again. You again took a look at her and sighed. You cannot help doing something for her, and at last decided to help her with a fast computer program. Your program is to select a set of stickers of maximum total score from the $2n$ stickers such that no two of them share an edge.

In the example shown in Figure 1, the maximum total score is 260 when you select the four stickers of scores 50, 50, 100, 60. Unfortunately, in this case, it is not allowed to simultaneously select both of the two highest scored stickers (of score 100 and 70) because the two stickers share an edge between them.

입력

Your program is to read from standard input. The input consists of T test cases. The number of test cases T is given in the first line of the input. Each test case starts with a line that contains an integer ($1 \leq n \leq 100,000$), where $2n$ is the number of stickers in the sheet. In the next two lines, each line contains n integers, each of which represents Nancy's score for the sticker at that position in the sticker sheet. Every two consecutive integers in a line are separated by a blank. Note that the $2n$ stickers are of rectangular shape and are arranged in 2 rows and n columns in the sheet. Nancy's scores range from 0 to 100.

출력

Your program is to write to standard output. Print exactly one line for each test case. The line should contain the maximum possible total score for a subset of the $2n$ stickers such that no two stickers share an edge.

예제 입력 1 복사

```
2
5
50 10 100 20 40
30 50 70 10 60
7
10 30 10 50 100 20 40
20 40 30 50 60 20 80
```

예제 출력 1 복사

```
260
290
```

Problem P

스티커

제한 시간 : 1초, 메모리 제한 : 256MB

문제

상근이의 여동생 상냥이는 문방구에서 스티커 2n개를 구매했다. 스티커는 그림 (a)와 같이 2행 n열로 배치되어 있다. 상냥이는 스티커를 이용해 책상을 꾸미려고 한다.

상냥이가 구매한 스티커의 품질은 매우 좋지 않다. 스티커 한 장을 떼면, 그 스티커와 변을 공유하는 스티커는 모두 찢어져서 사용할 수 없게 된다. 즉, 떼는 스티커의 왼쪽, 오른쪽, 위, 아래에 있는 스티커는 사용할 수 없게 된다.



모든 스티커를 붙일 수 없게된 상냥이는 각 스티커에 점수를 매기고, 점수의 합이 최대가 되게 스티커를 떼어내려고 한다. 먼저, 그림 (b)와 같이 각 스티커에 점수를 매겼다. 상냥이가 떼 수 있는 스티커의 점수의 최댓값을 구하는 프로그램을 작성하시오. 즉, 2n개의 스티커 중에서 점수의 합이 최대가 되면서 서로 변을 공유 하지 않는 스티커 집합을 구해야 한다.

위의 그림의 경우에 점수가 50, 50, 100, 60인 스티커를 고르면, 점수는 260이 되고 이 것이 최대 점수이다. 가장 높은 점수를 가지는 두 스티커 (100과 70)은 변을 공유하기 때문에, 동시에 떼 수 없다.

입력

첫째 줄에 테스트 케이스의 개수 T가 주어진다. 각 테스트 케이스의 첫째 줄에는 n ($1 \leq n \leq 100,000$)이 주어진다. 다음 두 줄에는 n개의 정수가 주어지며, 각 정수는 그 위치에 해당하는 스티커의 점수이다. 연속하는 두 정수 사이에는 빈 칸이 하나 있다. 점수는 0보다 크거나 같고, 100보다 작거나 같은 정수이다.

출력

각 테스트 케이스 마다, 2n개의 스티커 중에서 두 변을 공유하지 않는 스티커 점수의 최댓값을 출력한다.

예제 입력 1 복사

```
2
5
50 10 100 20 40
30 50 70 10 60
7
10 30 10 50 100 20 40
20 40 30 50 60 20 80
```

예제 출력 1 복사

```
260
290
```

Problem Q

Term Project

제한 시간 : 3초, 메모리 제한 : 256MB

문제

Students who enrolled in the 'Problem Solving' course in this fall semester have to carry out a term project. There is no limit to the number of project team members. Even one team is allowed such that all students are members of the same team. In order to form project teams, every student should select a friend with whom he or she wants to work. A student who wants to work alone can select himself or herself. A student list (s_1, s_2, \dots, s_r) can be a team if either $r=1$ and s_1 selects s_1 or s_1 selects s_2 , s_2 selects s_3 , ..., s_{r-1} selects s_r , and s_r selects s_1 .

For example, let's assume that there are 7 students in the class. The students are numbered from 1 to 7. The following is the result of the selection.

1	2	3	4	5	6	7
3	1	3	7	3	4	6

From the above result, we can see that two teams (3) and (4, 7, 6) are formed. Students 1, 2, and 5 don't belong to any team.

Given the result of the selection, write a program to compute the number of students who don't belong to a project team.

입력

Your program is to read from standard input. The input consists of T test cases. The number of test cases T is given in the first line of the input. Each test case starts with a line containing an integer n ($2 \leq n \leq 100\,000$), where n is the number of students in the class. All students are numbered from 1 to n . The next line of each test case contains n integers s_1, s_2, \dots, s_n , where s_i is the student who was a student s_i selected by.

출력

Your program is to write to standard output. Print exactly one line for each test case. The line should contain the number of students who don't belong to a project team.

예제 입력 1 복사

```
2
7
3 1 3 7 3 4 6
8
1 2 3 4 5 6 7 8
```

예제 출력 1 복사

```
3
0
```

Problem Q

팀 프로젝트

제한 시간 : 3초, 메모리 제한 : 256MB

문제

이번 가을학기에 '문제 해결' 강의를 신청한 학생들은 팀 프로젝트를 수행해야 한다. 프로젝트 팀원 수에는 제한이 없다. 심지어 모든 학생들이 동일한 팀의 팀원인 경우와 같이 한 팀만 있을 수도 있다. 프로젝트 팀을 구성하기 위해, 모든 학생들은 프로젝트를 함께하고 싶은 학생을 선택해야 한다. (단, 단 한 명만 선택할 수 있다.) 혼자 하고 싶어하는 학생은 자기 자신을 선택하는 것도 가능하다.

학생들이 (s_1, s_2, \dots, s_r) 이라 할 때, $r=1$ 이고 s_1 이 s_1 을 선택하는 경우나, s_1 이 s_2 를 선택하고, s_2 가 s_3 를 선택하고, ..., s_{r-1} 이 s_r 을 선택하고, s_r 이 s_1 을 선택하는 경우에만 한 팀이 될 수 있다.

예를 들어, 한 반에 7명의 학생이 있다고 하자. 학생들을 1번부터 7번으로 표현할 때, 선택의 결과는 다음과 같다.

1	2	3	4	5	6	7
3	1	3	7	3	4	6

위의 결과를 통해 (3)과 (4, 7, 6)이 팀을 이룰 수 있다. 1, 2, 5는 어느 팀에도 속하지 않는다.

주어진 선택의 결과를 보고 어느 프로젝트 팀에도 속하지 않는 학생들의 수를 계산하는 프로그램을 작성하라.

입력

첫째 줄에 테스트 케이스의 개수 T 가 주어진다. 각 테스트 케이스의 첫 줄에는 학생의 수가 정수 n ($2 \leq n \leq 100,000$)으로 주어진다. 각 테스트 케이스의 둘째 줄에는 선택된 학생들의 번호가 주어진다. (모든 학생들은 1부터 n 까지 번호가 부여된다.)

출력

각 테스트 케이스마다 한 줄에 출력하고, 각 줄에는 프로젝트 팀에 속하지 못한 학생들의 수를 나타내면 된다.

예제 입력 1 복사

```
2
7
3 1 3 7 3 4 6
8
1 2 3 4 5 6 7 8
```

예제 출력 1 복사

```
3
0
```


Problem R

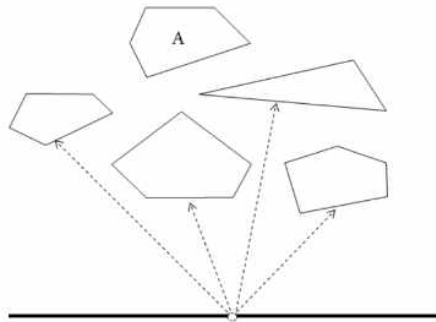
Meteor Shower

제한 시간 : 1초, 메모리 제한 : 512MB

문제

NSC(Naro Space Center) has just discovered that n large meteorites are falling to Korea. NSC is planning to blow up the meteorites using laser-guided missile system. In order not to miss a single meteorite, NSC needs to identify the meteorites which are completely blocked by others at a specific moment. We call them invisible meteorites.

Each meteorite is represented by a convex polygon. All meteorites are separated from each other, i.e., any two convex polygons do not intersect each other. The figure below shows an example of a situation with 5 meteorites and a laser-guided missile launcher. In the figure, a meteorite labeled with A is invisible because any point on it can't be touched by a laser beam from the launcher.



Given a list of convex polygons representing meteorites at some moment, write a program to find the number of the meteorites which are invisible from the laser-guided missile launcher.

입력

Your program is to read from standard input. The input starts with a line containing an integer n ($1 \leq n \leq 100,000$), where n is the number of convex polygons representing meteorites at a specific moment. In the following n lines, each line contains $2m+1$ integers $m, x_1, y_1, x_2, y_2, \dots, x_m, y_m$ ($3 \leq m \leq 10^5, -10^8 \leq x_i \leq 10^8, 1 \leq y_i \leq 10^8$), where m is the number of vertices of a convex polygon Q and (x_i, y_i) 's are coordinates of m vertices of Q in the counter-clockwise order. The laser-guided missile launcher is located at $(0, 0)$, i.e., the origin of the coordinate system. The total number of vertices of all convex polygons is less than or equal to 10^6 . Notice that any two convex polygons do not intersect each other. Also, you may assume that the line connecting any two vertices of all convex polygons does not pass through the origin, i.e., the location of the laser-guided missile launcher.

출력

Your program is to write to standard output. Print exactly one line which contains an integer representing the number of the meteorites which are invisible from the laser-guided missile launcher.

예제 입력 1 복사

```
5
3 -2 13 9 12 7 15
5 1 16 -1 18 -5 18 -6 16 -5 14
5 -12 13 -13 11 -11 10 -7 12 -8 13
5 -7 9 -5 7 0 7 1 9 -3 12
5 9 9 6 10 3 9 4 6 9 7
```

예제 입력 2 복사

```
4
4 -800 500 -800 300 700 300 700 500
4 -700 1100 -900 1100 -900 700 -700 700
4 100 700 100 900 -500 900 -500 700
4 300 700 600 700 600 1200 300 1200
```

예제 출력 1 복사

```
1
```

예제 출력 2 복사

```
3
```


Problem S

How Many to Be Happy

제한 시간 : 0.5초, 메모리 제한 : 512MB

문제

Let G be a connected simple undirected graph where each edge has an associated weight. Let's consider the popular MST (Minimum Spanning Tree) problem. Today, we will see, for each edge e , how much modification on G is needed to make e part of an MST for G . For an edge e in G , there may already exist an MST for G that includes e . In that case, we say that e is *happy* in G and we define $H(e)$ to be 0. However, it may happen that there is no MST for G that includes e . In such a case, we say that e is *unhappy* in G . We may remove a few of the edges in G to make a *connected* graph G' in which e is happy. We define $H(e)$ to be the minimum number of edges to remove from G such that e is happy in the resulting graph G' .

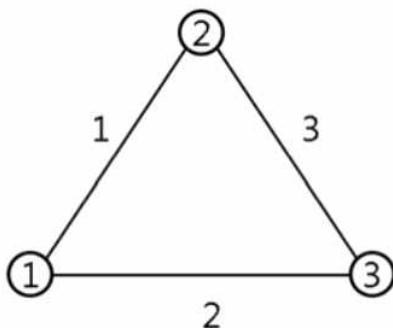


Figure E.1. A complete graph with 3 nodes.

Consider the graph in Figure E.1. There are 3 nodes and 3 edges connecting the nodes. One can easily see that the MST for this graph includes the 2 edges with weights 1 and 2, so the 2 edges are happy in the graph. How to make the edge with weight 3 happy? It is obvious that one can remove any one of the two happy edges to achieve that.

Given a connected simple undirected graph G , your task is to compute $H(e)$ for each edge e in G and print the total sum.

입력

Your program is to read from standard input. The first line contains two positive integers n and m , respectively, representing the numbers of vertices and edges of the input graph, where $n \leq 100$ and $m \leq 500$. It is assumed that the graph G has n vertices that are indexed from 1 to n . It is followed by m lines, each contains 3 positive integers u , v , and w that represent an edge of the input graph between vertex u and vertex v with weight w . The weights are given as integers between 1 and 500, inclusive.

출력

Your program is to write to standard output. The only line should contain an integer S , which is the sum of $H(e)$ where e ranges over all edges in G .

예제 입력 1 복사

```
3 3
1 2 1
3 1 2
3 2 3
```

예제 출력 1 복사

```
1
```

예제 입력 2 복사

```
7 9
1 2 8
1 3 3
2 3 6
4 2 7
4 5 1
5 6 9
6 7 3
7 4 2
4 6 2
```

예제 출력 2 복사

```
3
```

Problem T

The Mountain of Gold?

제한 시간 : 1초, 메모리 제한 : 256MB

문제

Ancient history and myth points to the Gunung Ledang (a.k.a. Ledang Mountain, as “gunung” is mountain in Malaysian) being the site of rich gold deposits, luring traders from as far as Greece and China. In the 14th Century, the Chinese seafarers plying the Straits of Melaka called it ‘Kim Sua’ meaning the ‘Golden Mountain’. The mountain was named ‘Gunung Ledang’, which means ‘mount from afar’, during the period of the Majapahit empire. Legend has it, that before the death of Princess Gunung Ledang, she hid a huge amount of gold far back in time during the creation of earth in Ledang Mountain. It is (apparently, becoming “was”) a mystery as to how the princess was able to go so far back in time. However, the princess was known to possess mystical powers that could enchant any pool she bath in into a portal that could manipulate time and space. A Malaysian historian of this time have discovered many of these pools located near many mountains around the world and named them “Ledang Pools”. A Ledang Pool is a portal through space and time connecting two pools. Ledang Pools have a few peculiar properties:

- A Ledang Pool is a one way portal with two end points, i.e. it connects exactly two mountains.
- The time it takes to travel through a Ledang Pool is negligible.
- Each mountain might have multiple Ledang Pools’ end point in its area.
- For some unknown reasons, starting from Ledang Mountain in Malaysia, it is always possible to end up in any mountain (of course, which also has Ledang Pool end point) on earth by hopping a sequence of Ledang Pools.
- There are no Ledang Pools with both end points on the same mountain area.
- Each Ledang Pool has a fixed time difference (distortion) between their end points. For example, traveling through a certain Ledang Pool may cause that person to end up 42 years in the past at the other end point. It’s a space and time traveling!

The Malaysian historian suspects that a large amount of gold is hidden in Ledang Mountain in the past because there’s no gold found in this mountain at this time (but then, where does the myth come from?). Starting from Ledang Mountain, the historian wants to reach the same mountain (Ledang Mountain) in the past. In order to this, he has to hop into two or more Ledang Pools, doing the space-and-time-travel things, and end up at Ledang Mountain in the past. Note that he does not care in which time in the past he will end up at Ledang Mountain as long as it is in the past. Your task is to help this historian to determine whether it is possible to reach Ledang Mountain in the past from Ledang Mountain in the present time by hopping a sequence of Ledang Pools.

입력

The input file starts with a line containing the number of cases T ($1 \leq T \leq 20$). Each case starts with two numbers N and M in a line. These indicate the number of mountains which have Ledang Pool end point ($1 \leq N \leq 1,000$) and the number of Ledang Pools ($0 \leq M \leq 2,000$). The pools are numbered from 0 (Ledang Mountain in Malaysia) up to $N-1$. The next M lines each contains three integers A, B, C ($0 \leq A, B < N$; $A \neq B$; $-1,000 \leq C \leq 1,000$) denoting the properties of one Ledang Pool, i.e. someone can do a space-time-travel from mountain A to mountain B and he will end up C year in the future/past. If C is positive, then it is in the future, otherwise it is in the past.

출력

For each case, output in a line "Case #X: Y" (without quotes) where X is the case number starting from 1, and Y is "possible" (without quotes) if it is indeed possible to reach Ledang Mountain in the past, otherwise output Y as "not possible" (without quotes).

예제 입력 1 복사

```
2
2 2
0 1 15
1 0 -20
4 4
0 1 10
1 2 20
2 3 30
3 0 -60
```

예제 출력 1 복사

```
Case #1: possible
Case #2: not possible
```

Problem U

Proving Equivalences

제한 시간 : 2초, 메모리 제한 : 128MB

문제

Consider the following exercise, found in a generic linear algebra textbook.

Let A be an $n \times n$ matrix. Prove that the following statements are equivalent:

- (a) A is invertible.
- (b) $Ax = b$ has exactly one solution for every $n \times 1$ matrix b .
- (c) $Ax = b$ is consistent for every $n \times 1$ matrix b .
- (d) $Ax = 0$ has only the trivial solution $x = 0$.

The typical way to solve such an exercise is to show a series of implications. For instance, one can proceed by showing that (a) implies (b), that (b) implies (c), that (c) implies (d), and finally that (d) implies (a). These four implications show that the four statements are equivalent.

Another way would be to show that (a) is equivalent to (b) (by proving that (a) implies (b) and that (b) implies (a)), that (b) is equivalent to (c), and that (c) is equivalent to (d). However, this way requires proving six implications, which is clearly a lot more work than just proving four implications!

I have been given some similar tasks, and have already started proving some implications. Now I wonder, how many more implications do I have to prove? Can you help me determine this?

입력

On the first line one positive number: the number of testcases, at most 100. After that per testcase:

- One line containing two integers n ($1 \leq n \leq 20\,000$) and m ($0 \leq m \leq 50\,000$): the number of statements and the number of implications that have already been proved.
- m lines with two integers s_1 and s_2 ($1 \leq s_1, s_2 \leq n$ and $s_1 \neq s_2$) each, indicating that it has been proved that statement s_1 implies statement s_2 .

출력

Per testcase:

- One line with the minimum number of additional implications that need to be proved in order to prove that all statements are equivalent.

예제 입력 1 복사

```
2
4 0
3 2
1 2
1 3
```

예제 출력 1 복사

```
4
2
```


Problem U

동치 증명

제한 시간 : 2초, 메모리 제한 : 128MB

문제

위대한 수학자 김선영은 선형대수학 교과서를 집필하는 과정에서 다음과 같은 문제를 만들었다.

$N \times N$ 행렬 A 에 대해 다음 명제들이 동치임을 증명하라:

- A 의 역행렬이 존재한다.
- 임의의 $N \times 1$ 행렬 b 에 대해 $Ax = b$ 는 단 하나의 해만을 갖는다.
- 임의의 $N \times 1$ 행렬 b 에 대해 $Ax = b$ 는 해를 갖는다.
- $Ax = 0$ 의 해는 오직 $x = 0$ 하나밖에 존재하지 않는다.

이런 문제를 해결하는 일반적인 방법은 일련의 함축(implication)을 이용하는 것이다.

예를 들어, 선영이의 예시 답안은

(a)이면 (b)이고, (b)이면 (c)이며 (c)이면 (d)이고, 마지막으로 (d)이면 (a)이다. 이 네 함축은 네 가지 명제가 동치임을 보여준다.

라고 되어있다.

다른 방법으로는 (a)이면 (b)이고, (b)이면 (a)이므로 (a)와 (b)가 동치라고 증명한 다음,

같은 방식으로 (b)와 (c)가 동치임을 증명하고, 마지막으로 (c)와 (d)가 동치임을 증명하는 방법이 있다.

하지만 이건 무려 여섯 번의 함축을 필요로 한다.

선영이는 선형대수학 책을 집필하는 과정에서 수없이 많은 명제가 동치임을 증명해야 하기 때문에 이러한 비효율성은 치명적이다.

선영이가 다음 학기 시작 전에 교재를 모두 집필할 수 있도록 되도록이면 적은 수의 함축을 이용해서 명제가 동치임을 증명할 수 있도록 도와주자.

입력

첫 번째 줄에는 테스트 케이스의 개수 $T(1 \leq T \leq 100)$ 가 주어지고,

각 테스트 케이스에 대해서는 다음과 같은 입력이 주어진다:

- 명제의 수 $n(1 \leq n \leq 20,000)$ 와 이미 증명된 함축의 수 $m(0 \leq m \leq 50,000)$ 이 첫 번째 줄에 주어진다.
- m 개의 줄에 " s_1 이면 s_2 이다."를 나타내는 s_1 과 $s_2(1 \leq s_1, s_2 \leq n$ 이고 $s_1 \neq s_2)$ 가 주어진다.

출력

각 테스트 케이스마다 한 줄을 출력한다:

- 위대한 수학자 김선영이 주어진 명제들이 동치임을 증명하기 위해 사용해야 하는 함축의 수의 최솟값.

예제 입력 1 복사

```
2
4 0
3 2
1 2
1 3
```

예제 출력 1 복사

```
4
2
```


Problem V

종이의 개수

제한 시간 : 2초, 메모리 제한 : 256MB

문제

$N \times N$ 크기의 행렬로 표현되는 종이가 있다. 종이의 각 칸에는 -1, 0, 1의 세 값 중 하나가 저장되어 있다. 우리는 이 행렬을 적절한 크기로 자르려고 하는데, 이때 다음의 규칙에 따라 자르려고 한다.

- 만약 종이가 모두 같은 수로 되어 있다면 이 종이를 그대로 사용한다.
- (1)이 아닌 경우에는 종이를 같은 크기의 9개의 종으로 자르고, 각각의 잘린 종이에 대해서 (1)의 과정을 반복한다.

이와 같이 종이를 잘랐을 때, -1로만 채워진 종이의 개수, 0으로만 채워진 종이의 개수, 1로만 채워진 종이의 개수를 구해내는 프로그램을 작성하시오.

입력

첫째 줄에 $N(1 \leq N \leq 3^7, N \text{은 } 3^k \text{ 꼴})$ 이 주어진다. 다음 N 개의 줄에는 N 개의 정수로 행렬이 주어진다.

출력

첫째 줄에 -1로만 채워진 종이의 개수를, 둘째 줄에 0으로만 채워진 종이의 개수를, 셋째 줄에 1로만 채워진 종이의 개수를 출력한다.

예제 입력 1 복사

```
9
0 0 0 1 1 1 -1 -1 -1
0 0 0 1 1 1 -1 -1 -1
0 0 0 1 1 1 -1 -1 -1
1 1 1 0 0 0 0 0 0
1 1 1 0 0 0 0 0 0
1 1 1 0 0 0 0 0 0
0 1 -1 0 1 -1 0 1 -1
0 -1 1 0 1 -1 0 1 -1
0 1 -1 1 0 -1 0 1 -1
```

예제 출력 1 복사

```
10
12
11
```