

Gassen

Auf ihren Reisen durch Absurdistan besuchte Lea auch Ancientia, eine alte Stadt mit vielen engen Gassen. Manche davon sind einwegig und dürfen nur in eine Richtung passiert werden, während andere zweiwegig sind und in beide Richtungen passiert werden dürfen.

Auf der letzten „Tour d’Absurdistan“ fuhren viele Rennfahrer durch die Gassen der Stadt. Weil die Gassen so eng sind, blockierten sich die Fahrer auf den zweiwegigen Gassen häufig gegenseitig, und wegen des komplizierten Gassennetzes fuhren sie viel sinnlos im Kreis. Beides führte zu Staus und Verzögerungen. Um das Problem zu beheben, will der Stadtrat Ancientias nun das Gassennetz azyklisch machen, indem er alle zweiwegigen in einwegige Gassen umwandelt.

Derzeit ist sich der Stadtrat noch nicht sicher, ob das überhaupt möglich ist. Da Lea für ihre Problemlösekompetenz in ganz Absurdistan bekannt ist, wurde sie um Hilfe gebeten. Für ein gegebenes Gassennetz mit einwegigen (gerichteten) und zweiwegigen (ungerichteten) Gassen sucht der Stadtrat eine Zuteilung von Richtungen an die ungerichteten Gassen, sodass das resultierende Gassennetz (dessen Gassen alle gerichtet sind) azyklisch ist. Kannst Du für Lea eine solche Zuteilung finden, sofern es eine gibt?

Eingabe

Die erste Zeile der Eingabe enthält eine Ganzzahl t . Darauf folgen t Testfälle, jeweils getrennt durch eine Leerzeile.

Jeder Testfall besteht aus einer Zeile mit drei Ganzzahlen n , m und l . n ist die Zahl der Kreuzungen (die nummeriert sind von 1 bis n). m ist die Zahl der einwegigen (gerichteten) Gassen. l ist die Zahl der zweiwegigen (ungerichteten) Gassen. Es folgen m Zeilen. Die j -te Zeile enthält zwei Ganzzahlen a_j und b_j , die angeben, dass es eine einwegige (gerichtete) Gasse von der Kreuzung a_j zur Kreuzung b_j gibt. Es folgen l Zeilen. Die j -te Zeile enthält zwei Ganzzahlen a_j und b_j , die angeben, dass es eine zweiwegige (ungerichtete) Gasse zwischen der Kreuzung a_j und der Kreuzung b_j gibt. Zwischen zwei Kreuzungen gibt es höchstens eine Gasse.

Ausgabe

Für jeden Testfall soll die Ausgabe wie folgt sein:

Wenn man das Gassennetz nicht zu einem azyklischen Graphen umwandeln kann, indem man den zweiwegigen Gassen Richtungen gibt, gib „Case # i : no“ aus, wobei i bei 1 beginnen die Nummer des Testfalls ist. Ansonsten gib eine Zeile „Case # i : yes“, gefolgt von l weiteren Zeilen, die eine Zuteilung beschreiben, aus. Die j -te dieser Zeilen soll zwei Ganzzahlen a_j und b_j enthalten, wenn die bisher zweiwegige Gasse zwischen a_j und b_j (oder b_j und a_j) jetzt eine einwegige Gasse von a_j nach b_j ist. Wenn es mehrere mögliche Zuteilungen gibt, wird jede beliebige davon akzeptiert.

Beschränkungen

- $1 \leq t \leq 20$
- $1 \leq n \leq 1000$
- $0 \leq m + l \leq 10000$
- $1 \leq a_j, b_j \leq n$ für alle $1 \leq j \leq m + l$
- $a_j \neq b_j$ für alle $1 \leq j \leq m + l$
- Es gibt höchstens einen Eintrag aus $a_j b_j$ und $b_j a_j$ für beliebige $1 \leq a_j, b_j \leq n$.

Sample Input 1

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

4 4 1
1 2
4 3
3 1
2 4
2 3

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

Sample Output 1

```
Case #1: yes
1 3
4 2
2 3
Case #2: no
Case #3: yes
4 1
2 5
3 7
4 8
7 6
6 8
```

Sample Input 2

```
3
7 4 4
6 2
6 5
4 2
6 4
3 7
5 7
4 7
3 5

8 4 2
1 6
6 4
3 1
4 3
7 6
1 2

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

Sample Output 2

```
Case #1: yes
3 7
7 5
7 4
3 5
Case #2: no
Case #3: yes
6 4
1 5
1 2
1 6
```

City Roads

On her tour of Absurdistan, Lea visited Ancientia, an old city with many narrow roads. Some of its roads are one-way roads and can only be passed in one direction, while others are two-way roads and can be passed in two directions.

During the last “Tour d’Absurdistan”, many racers went through this city and its roads. However, due to the narrow roads, many racers often blocked each other in the two-way roads, and due to the complex road network, they happened to cycle around needlessly, both of which lead to traffic congestions and delays. To solve this problem, the city council of Ancientia wants to make the road network acyclic by turning all two-way roads into one-way roads.

Currently, the city council is still unsure if this is even possible to achieve. As Lea’s problem solving skills are known throughout Absurdistan, they ask her to help. Given a road network with one-way (directed) and two-way (undirected) roads, they ask for an assignment of directions to the undirected roads such that the resulting road network (consisting only of directed roads) is acyclic. Can you tell Lea how to find such an assignment, if there exists one?

Input

The first line of the input contains an integer t . t test cases follow, each of them separated by a blank line.

Each test case consists of a line containing three integers $n\ m\ l$. n is the number of intersections (indexed from 1 to n). m is the number of one-way (directed) roads. l is the number of two-way (undirected) roads. m lines follow. The j -th line contains two integers $a_j\ b_j$, meaning there is a one-way (directed) road from intersection a_j to intersection b_j . l lines follow. The j -th line contains two integers $a_j\ b_j$, meaning there is a two-way (undirected) road between intersections a_j and b_j . Two intersections are connected by at most one road.

Output

For each test case, if there is no way to turn the road network into an acyclic graph by assigning directions to the two-way roads, print a line “Case # i : *no*”. Otherwise, print a line “Case # i : *yes*”, followed by l more lines, describing an assignment. The j -th line should contain two integers $a_j\ b_j$ if the previous two-way road between a_j and b_j (or b_j and a_j) is now a one-way road from a_j to b_j . If there are multiple valid assignments, any one of them is accepted.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 1000$
- $0 \leq m + l \leq 10000$
- $1 \leq a_j, b_j \leq n$ for all $1 \leq j \leq m + l$
- $a_j \neq b_j$ for all $1 \leq j \leq m + l$
- There is at most one entry $a_j\ b_j$ or $b_j\ a_j$ for any $1 \leq a_j, b_j \leq n$.

Sample Input 1

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

4 4 1
1 2
4 3
3 1
2 4
2 3

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

Sample Output 1

```
Case #1: yes
1 3
4 2
2 3
Case #2: no
Case #3: yes
4 1
2 5
3 7
4 8
7 6
6 8
```

Sample Input 2

```
3
7 4 4
6 2
6 5
4 2
6 4
3 7
5 7
4 7
3 5

8 4 2
1 6
6 4
3 1
4 3
7 6
1 2

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

Sample Output 2

```
Case #1: yes
3 7
7 5
7 4
3 5
Case #2: no
Case #3: yes
6 4
1 5
1 2
1 6
```