

A - Where's My Internet??

Se está construyendo una nueva ciudad lejos en el país, y actualmente hay N casas. La gente ya ha comenzado a mudarse. Sin embargo, algunas de las casas aún no están conectadas a Internet y, naturalmente, los residentes están indignados.

Las casas están numeradas 1 a N . Casa número 1 ya se ha conectado a Internet a través de un cable de red largo a una ciudad vecina. El plan es proporcionar Internet a otras casas conectando pares de casas con cables de red separados. Una casa está conectada a Internet si tiene un cable de red a otra casa que ya está conectada a Internet.

Dada una lista de qué pares de casas ya están conectadas mediante un cable de red, determine qué casas aún no están conectadas a Internet.

Input

La primera línea de entrada contiene dos números enteros $1 \leq N, M \leq 200000$, donde N es el número de casas y M es el número de cables de red ya desplegados. Luego siga M líneas, cada una de las cuales contiene un par de números de casa distintos $1 \leq a, b \leq N$, lo que significa que el número de casa a y el número de casa b ya están conectados por un cable de red. Cada par de casas se enumera como máximo una vez en la entrada.

Output

Si todas las casas ya están conectadas a Internet, muestre por pantalla una línea que contenga la cadena "Connected". De lo contrario, genere una lista de números de casas en orden creciente, uno por línea, que represente las casas que aún no están conectadas a Internet.

Ejemplo

Input

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

Output

```
1 5
2 6
```

Input

```
1 2 1
2 2 1
```

Output

```
1 Connected
```

Input

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

Output

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

B - Vertex

Escriba un programa que busque en un grafo dirigido los nodos que son inaccesibles desde un nodo inicial dado.

- Un grafo dirigido está representado por n nodos donde $1 \leq n \leq 100$, numerados consecutivamente $1 \dots n$, y una serie de aristas $p \rightarrow q$ que conectan el par de nodos p y q en una sola dirección.
- Un nodo r es alcanzable desde un nodo p si hay una arista $p \rightarrow q$, o si existe algún nodo q para el cual q es alcanzable desde p y r es alcanzable desde q .
- Un nodo r es inaccesible desde un nodo p si r no es alcanzable desde p .

Input

Los datos de entrada para este programa constan de varios grafos dirigidos y nodos iniciales.

Para cada grafo, hay una primera línea que contiene un solo entero n . Este es el número de nodos en el grafo.

A continuación, habrá un grupo de líneas, cada una de las cuales contiene un conjunto de números enteros. El grupo termina con una línea que contiene solo el número entero "0". Cada conjunto representa una colección de aristas. El primer número entero del conjunto, i , es el nodo inicial, mientras que el siguiente grupo de números enteros, $j \dots k$, define la serie de aristas $i \rightarrow j \dots i \rightarrow k$, y el último número entero de la línea es siempre "0". Cada posible nodo inicial i , $1 \leq i \leq n$ aparecerá una vez o no aparecerá en absoluto. Después de cada definición de un grafo, habrá una línea que contiene una lista de enteros. El primer entero de la línea especificará cuántos enteros siguen. Cada uno de los siguientes números enteros representa un nodo inicial que su programa debe investigar. Luego sigue el siguiente grafo. Si no hay más grafos, la siguiente línea del archivo contendrá solo un número entero "0".

Output

Para que se investigue cada nodo, su programa debe identificar todos los nodos que son inaccesibles desde el nodo dado. Cada lista debe aparecer en una línea, comenzando con el recuento de nodos inaccesibles y seguida por los números de nodos inaccesibles.

Ejemplo

Input

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

Output

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

C - Reachable Roads

Después del gran terremoto y los incendios en su ciudad, los planificadores están evaluando los daños a las carreteras de la ciudad. Algunas carreteras simplemente se han ido y otras están intransitables. El resultado es que es imposible ir de unos lugares a otros. Los planificadores quieren que la ciudad funcione rápidamente. Por lo tanto, quieren agregar la menor cantidad de carreteras posible para obtener una ciudad donde todos puedan llegar a todos los demás.

Haremos las suposiciones simplificadoras de que construir cualquier camino tiene el mismo costo que construir cualquier otro camino, y que todos los caminos son bidireccionales.

Los planificadores le han pedido que encuentre la cantidad mínima de carreteras que pueden agregar a la cuadrícula actual para que todos puedan llegar a todas partes.

Input

La entrada comienza con una línea que contiene un número entero $1 \leq n \leq 100$ que indica el número de ciudades que siguen. La descripción de cada ciudad comienza con una línea que describe el número de extremos de carreteras $1 \leq m \leq 1000$. Cada punto final está en el rango de 0 a $m - 1$. A esto le sigue un número $0 \leq r \leq \min(\frac{m \cdot (m-1)}{2}, 10 \cdot m)$, luego una lista de r pares de extremos que están conectados directamente por una carretera utilizable. Cada par aparece en su propia línea, todos los pares son distintos y todas las carreteras se conectan a diferentes puntos finales.

Output

Para cada descripción de ciudad, muestre por pantalla el número mínimo de carreteras que se deben agregar.

Ejemplo

Input

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

Output

```
1 1
2 0
```