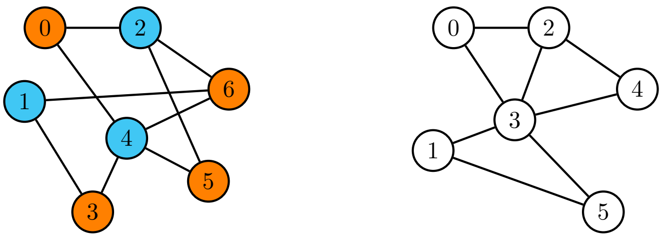
30

Grafo bipartito

Un grafo no dirigido es *bipartito* si sus vértices pueden repartirse en dos conjuntos disjuntos de tal forma que todas las aristas tengan un extremo en cada uno de esos conjuntos.

Dicho de otra forma, un grafo es bipartito si sus vértices pueden colorearse utilizando dos colores de tal forma que no exista ninguna arista que conecte dos vértices del mismo color. De los dos grafos no dirigidos siguientes, el de la izquierda es bipartito, pero el de la derecha no lo es.



Entrada

La entrada está compuesta por diversos casos de prueba. Para cada caso, la primera línea contiene el número de vértices del grafo, V (entre 1 y 100), y la segunda el número de aristas, A. A continuación aparecen A líneas, cada una con dos enteros que representan los extremos de cada una de las aristas (valores entre 0 y $V-1$). Los grafos no contienen aristas de un vértice a sí mismo ni más de una arista que conecte un mismo par de vértices.

Salida

Para cada caso de prueba se escribirá en una línea independiente la palabra `SI` si el grafo es bipartito y `NO` en caso contrario.

Entrada de ejemplo

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

Salida de ejemplo

```
SI
NO
```

Autor: Alberto Verdejo.

31

Colocar a los guardias

En el pequeño país de Abudajh hay algunas calles y cruces. Cada calle conecta dos cruces. El rey de Abudajh quiere colocar guardias en algunos cruces de forma que todas las calles queden custodiadas por ellos. Un guardia en un cruce puede vigilar todas las calles adyacentes a él.

Pero los guardias no son muy gentiles. Si una calle está custodiada por dos guardias, estos terminarán peleándose. Y el rey no quiere que los guardias se peleen.

Dada la información sobre las calles y cruces de Abudajh, ayuda al rey a encontrar la cantidad mínima de guardias necesarios para proteger todas las calles de su país sin que haya peligro de peleas.

Entrada

La entrada está compuesta por diversos casos de prueba. Para cada caso, primero aparece el número N de cruces en el país y a continuación el número de calles C entre cruces. En las siguientes C líneas se describen estas calles, dando los cruces que conectan (numerados entre 1 y N).

Salida

Para cada caso de prueba se escribirá el número mínimo de guardias necesarios para vigilar todas las calles. Si no fuera posible colocar los guardias para que todas las calles quedaran custodiadas y no hubiera peleas, entonces se escribirá `IMPOSIBLE`.

Entrada de ejemplo

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

Salida de ejemplo

```
4
1
IMPOSIBLE
```

Autor: Alberto Verdejo.

32
¡Las noticias vuelan!

En una red social hay una serie de usuarios que se comunican entre ellos dentro de una serie de grupos de amigos. Queremos analizar el proceso de distribución de noticias entre estos usuarios.

Inicialmente, algún usuario recibe una noticia de una fuente externa. Entonces envía la noticia a sus amigos en la red (dos usuarios son amigos si hay al menos un grupo al que pertenezcan ambos). Los amigos envían a su vez esa noticia a sus amigos y el proceso continua así hasta que no exista una pareja de amigos tal que uno de ellos conozca la noticia y el otro no.

Para cada usuario de la red, queremos saber cuántos usuarios terminarían conociendo la noticia si inicialmente solamente ese usuario la conocía.

Entrada

La entrada está formada por una serie de casos, cada uno de los cuales ocupa varías líneas. En la primera línea de cada caso aparecen dos números: el número N de usuarios de la red y el número M de grupos ($1 \leq N, M \leq 100.000$). A continuación aparecen M líneas describiendo esos grupos. Para cada grupo, la descripción comienza con el número de usuarios del grupo (entre 0 y N) seguido de los identificadores de esos usuarios (todos distintos), números entre 1 y N. La suma de los tamaños de todos los grupos no es mayor que 500.000.

Salida

Para cada caso se escribirá una línea con N números. El número i-ésimo indicará el número de usuarios que terminarían conociendo la noticia si el usuario i fuera quién comenzara a distribuirla.

Entrada de ejemplo

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

Salida de ejemplo

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

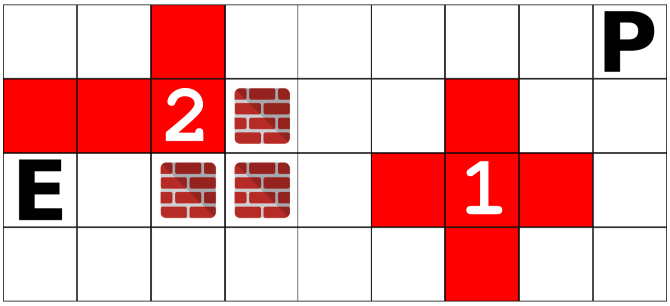
Autor: Alberto Verdejo.

33

La ronda de la noche

En Ciudad Decoro les preocupa que los jóvenes se reúnan con sus parejas en la noche, a espaldas de sus padres. Por ello en la última Ordenanza Municipal se ha obligado a que todas las casas tengan un jardín-laberinto para acceder a su puerta principal y que (opcionalmente) se instalen sensores de movimiento en los mismos. Conscientes de la posible dificultad de implantación de la Ordenanza, se ha decidido que todos los jardines-laberinto estén organizados en cuadrículas donde cada casilla está libre, tiene un muro o alberga un sensor de movimiento.

Como no podría ser de otra forma, la plataforma activista *#FreeLove* va a hacer todo lo posible para evitar que estas nuevas medidas supongan un obstáculo real para el amor. Durante las últimas semanas ha ido investigando y realizando mapas de todos los jardines-laberinto, anotando la posición de los sensores de movimiento. *#FreeLove* ha detectado que únicamente hay 10 tipos diferentes de sensores de movimiento, etiquetados como *CAT-k* donde k es un número natural entre 0 y 9. Los sensores captan movimiento en línea recta en las 4 direcciones (norte, sur, este y oeste) y la k de su categoría indica el número de casillas que cubren en cada dirección. De esta manera, un sensor *CAT-0* únicamente capta movimiento en la casilla en la que está instalado, mientras que un sensor *CAT-2* capta movimiento en su casilla y en 2 casillas en cada una de las 4 direcciones (en total cubre 9 casillas). Sin embargo, lo más interesante que ha descubierto *#FreeLove* es que los sensores no pueden traspasar los muros del laberinto, así que en ocasiones su alcance en alguna dirección se ve limitado.



#FreeLove necesita descubrir cuáles de los jardines-laberinto permiten que un amante vaya de la entrada del jardín a la puerta principal de la casa sin ser descubierto por ningún sensor. ¿Podrías ayudarles a diferenciar los jardines-laberinto impenetrables de aquellos favorables al amor, y en esos casos calcular el mínimo número de casillas que hay que atravesar para llegar de la entrada del jardín a la puerta principal?

Entrada

La entrada comienza con una línea conteniendo el número de jardínes a analizar. Cada jardín comienza con dos números $0 < ancho, alto \leq 1000$ con el ancho y alto del jardín-laberinto en una línea. Le sigue la descripción del jardín en *alto* líneas de *ancho* caracteres cada una. Estos caracteres son:

- '`#`': Una pared.
- '`.`': Una casilla libre.
- '`E`': La casilla donde está la entrada al jardín-laberinto.
- '`P`': La casilla donde está la puerta principal de la casa.
- k, con $0 \leq k \leq 9$: Casilla que alberga un sensor *CAT-k*.

La entrada al jardín y la puerta principal de la casa pueden estar en cualquier parte del jardín, incluidas casillas interiores. Además, nada impide que algún sensor vigile la casilla donde está la entrada al jardín o la puerta principal de la casa, por lo que en esos casos no será posible recorrer el jardín sin ser descubiertos.

Salida

Por cada jardín-laberinto la salida será una línea con el mínimo número de casillas del jardín que hay que atravesar para llegar de la entrada a la puerta principal de la casa sin ser descubierto, o la palabra `NO` en caso de que sea imposible.

Entrada de ejemplo

```
3
9 4
........P
..2#.....
E.##..1..
.........
6 4
....#P
..#...
.####.
E....1
10 5
E.........
..........
....2.....
..........
.........P
```

Salida de ejemplo

```
12
10
NO
```

Autor: Enrique Martín Martín.

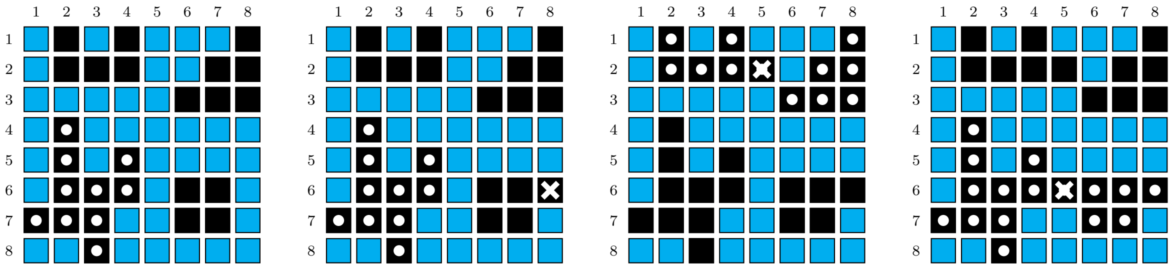
34

Petroleros hundidos

Desgraciadamente, las asociaciones ecologistas se ven obligadas a hacer frente, periódicamente, a las llamadas "*mareas negras*". En ellas, grandes cantidades de crudo son vertidas al mar, dejando enormes superficies de agua con petróleo flotando a la deriva.

Para estimar los daños medioambientales, se realizan fotografías de las zonas afectadas utilizando satélites geoestacionarios. La superficie del mar queda dividida en una rejilla de celdas, cada una marcada como zona contaminada o como zona limpia (al menos de momento). Las imágenes obtenidas son también usadas para organizar los trabajos de limpieza, para los que no es importante la superficie total contaminada, sino la superficie *contigua* más grande (*mancha* más grande). Dada una fotografía del satélite, dos celdas contaminadas de petróleo (negras) se consideran pertenecientes a la misma mancha si se puede llegar de una a otra atravesando solo celdas contaminadas realizando desplazamientos en cualquiera de las 8 direcciones (horizontal, vertical, y dos diagonales).

Cuando el petrolero se hunde, muchas veces sigue derramando crudo que, al emerger, aumenta la zona contaminada y puede hacer que cambie cual es la mayor mancha. Por ejemplo, en el siguiente esquema se muestra la primera imagen que se tomó, y la evolución horaria al ir subiendo más crudo desde las profundidades del mar. En cada una, se marca con puntos blancos las celdas de la mancha más grande (con su tamaño en el pie de la imagen), y con una cruz la última celda que ha pasado a estar contaminada.



10 10 12 16

Entrada

La entrada estará compuesta por diversos casos de prueba. Para cada caso, la primera línea contendrá el número F de filas y el número C de columnas de la rejilla (números entre 1 y 1.000). A continuación aparecerán F líneas, cada una con C caracteres. El espacio en blanco representa una celda azul (mar) y el carácter `#` representa una celda contaminada de negro petróleo. En la siguiente línea aparecerá un número no negativo N (no mayor de 100.000) indicando el número de imágenes adicionales tomadas (en cada una aparece una nueva celda de petróleo), seguido de N líneas cada una con dos enteros que indicarán la fila (entre 1 y F) y columna (entre 1 y C) donde aparecerá esa celda contaminada.

Salida

Para cada caso de prueba se escribirá una línea con el tamaño de la mancha de petróleo más grande inicialmente, seguido de los tamaños tras añadir cada una de las nuevas celdas contaminadas, separados por espacios. Las superficies se miden en número de celdas.

Entrada de ejemplo

```
8 8
 # #   #
 ###  ##
     ###
 #
 # #
 ### ##
###  ##
  #
3
6 8
2 5
6 5
```

Salida de ejemplo

```
10 10 12 16
```

Autores: Alberto Verdejo y Pedro Pablo Gómez Martín.

35
Camiones de reparto

Somos una empresa de transporte y hemos decidido renovar parte de nuestra flota de camiones de reparto. A nosotros nos conviene que los camiones sean anchos, porque así se puede repartir y colocar mejor la mercancia. Pero claro, hay ciudades con calles muy estrechas, por donde no todos los camiones pueden pasar.

Tenemos mapas actualizados de las ciudades donde trabajamos, donde hemos señalado para cada calle cuál es la anchura máxima que puede tener un camión para poder transitar por ella.

¿Nos ayudas a decidir si un camión de una anchura determinada puede circular por una ciudad para llegar desde un punto concreto a otro?

Entrada

La entrada está formada por una serie de casos de prueba. En cada caso, primero se describe una ciudad. La primera línea contiene el número V de intersecciones de la ciudad (numeradas de 1 a V) y la segunda el número E de calles entre intersecciones. A continuación aparecen E líneas, cada una con tres números: las intersecciones que une esa calle, y la anchura máxima que puede tener un camión que transite por ella. Todas las calles son de doble sentido.

Tras la descripción de la ciudad, aparece un número K de consultas, seguido de K líneas, cada una con tres números: dos intersecciones distintas, el origen y el destino, y la anchura de un camión, del que estamos interesados en saber si podría viajar desde el origen hasta el destino.

Todos los casos cumplen que $2 \leq V \leq 10.000$, $0 \leq E \leq 100.000$ y $1 \leq K \leq 10$. Todas las anchuras son números entre 1 y 1.000.000.

Salida

Para cada caso de prueba se escribirán K líneas, una por consulta. La respuesta a una consulta será `SI` si un camión de la anchura correspondiente podría recorrer un camino que le llevara del origen al destino, y `NO` en caso contrario.

Entrada de ejemplo

```
5
5
1 2 10
1 3 30
2 4 20
3 4 15
4 5 12
3
1 5 8
1 4 12
2 5 15
```

Salida de ejemplo

```
SI
SI
NO
```

Autor: Alberto Verdejo.

36

Recorriendo el archipiélago en bicicleta

Al archipiélago *Milislotes* han llegado las bicicletas. Sus habitantes están estusiasmados y no se bajan de la bici en todo el día, yendo de un lado para otro sin descanso. Pero las islas son tan pequeñas que terminan todos mareados de dar vueltas sin parar, por lo que se están planteando crear una red de puentes rectos que les permita ir en bici de cualquier isla a cualquier otra.

Han pedido presupuesto al arquitecto, que ha confeccionado una lista con todos los puentes que podrían construirse entre islas del archipiélago y cuánto costaría construir cada uno de esos puentes. ¿Podrías ayudarles a decidir qué puentes construir de tal forma que se pueda ir en bici desde cualquier isla a cualquier otra y el coste total de la obra sea lo mínimo posible?

Entrada

La entrada está formada por varios casos de prueba. Cada uno ocupa varias líneas: en la primera aparece el número I de islas en el archipiélago (entre 1 y 1.000); en la siguiente aparece el número P de puentes presupuestados (entre 0 y 10.000); y a continuación aparece una línea por cada uno de estos puentes con tres enteros, las islas que une (numeradas entre 1 e I) y el coste de construir ese puente (un valor entre 1 y 100.000). Los puentes siempre van de una isla a otra distinta, son transitables en bici en ambos sentidos, y no se ha presupuestado más de un puente entre un mismo par de islas.

Salida

Para cada caso de prueba se escribirá, en una línea, el coste mínimo de construir los puentes necesarios para unir todas las islas por bici. Si no hubiera suficientes puentes presupuestados para lograrlo, se escribirá `No hay puentes suficientes`.

Entrada de ejemplo

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

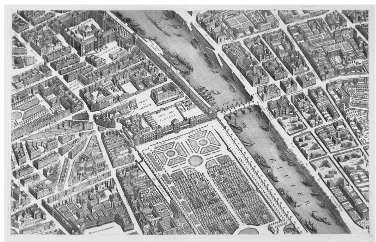
Salida de ejemplo

```
14
No hay puentes suficientes
```

Autor: Alberto Verdejo.

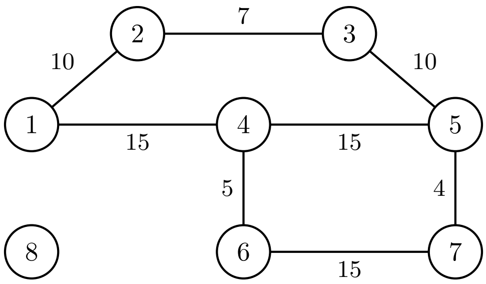
37

¿Cuál es el mejor camino?



Hace poco me he mudado a una nueva ciudad. Aprovechando que me gusta caminar, siempre que puedo voy andando a los sitios y así voy conociendo la ciudad. Cuando voy lejos y tengo prisa suelo coger las grandes avenidas porque las conozco más y sé que no voy a perderme, pero soy consciente de que, muchas veces, callejeando por calles cortas recorrería menos distancia, aunque tuviese que atravesar muchas de ellas. Me pregunto cuántas veces el camino más corto en distancia también es el que pasa por menos calles.

Por ejemplo, si el siguiente esquema representa la ciudad, con 8 intersecciones y 8 calles, donde junto a cada calle aparece su longitud medida en metros, para ir del punto 1 al punto 5 el camino más corto es el $1 \to 2 \to 3 \to 5$, que recorre 27 metros y atraviesa tres calles, mientras que el camino $1 \to 4 \to 5$, aunque es más largo (30 metros) pasa solamente por dos calles. En cambio, el camino más corto que une los puntos 6 y 5 sí utiliza el menor número de calles (no hay ningún otro camino con menos calles).



Entrada

La entrada consta de varios casos de prueba, ocupando cada uno de ellos varias líneas.

En la primera aparece el número N (entre 1 y 10.000) de intersecciones en la ciudad, y en la segunda el número C (entre 0 y 100.000) de calles (entre intersecciones). A continuación, aparece una línea por cada calle con tres enteros, que indican los números de las intersecciones que une la calle (números entre 1 y N) y su longitud (un valor entre 1 y 5.000) medida en metros. Todas las calles pueden recorrerse en ambos sentidos.

A continuación aparece el número K de consultas (no más de 10) seguido de esas consultas: dos números que representan las intersecciones origen y destino. Se garantiza que para cada consulta, el camino más corto entre origen y destino es único.

Salida

Para cada caso de prueba se escribirá una línea por cada consulta que contendrá la distancia, medida en metros, del camino más corto que conecta el origen con el destino, seguida de la palabra `SI` si ese camino además atraviesa el menor número de calles o `NO` en caso contrario. Si para una consulta no existiera camino que conecte el origen con el destino, entonces se escribiría `SIN CAMINO`.

Después de la salida de cada caso se escribirá una línea con `---`.

Entrada de ejemplo

```
8
8
1 2 10
2 3 7
3 5 10
1 4 15
4 5 15
4 6 5
5 7 4
6 7 15
3
1 5
3 8
6 5
```

Salida de ejemplo

```
27 NO
SIN CAMINO
19 SI
---
```

Autor: Alberto Verdejo.

38

Ratones en un laberinto

Un grupo de ratones de laboratorio está siendo entrenado para escapar de un laberinto. El laberinto está compuesto por una serie de celdas, donde cada celda está conectada a otras a través de pasadizos donde hay obstáculos, por lo que los ratones tardan un tiempo conocido en superar cada uno de estos tramos. Además, algunos pasajes permiten a los ratones ir en un solo sentido, pero no al revés.

Todos los ratones han sido muy bien entrenados y, cuando son colocados en una celda arbitraria del laberinto, siguen un camino que los lleva a la celda de salida en un tiempo mínimo.

Vamos a realizar el siguiente experimento: se coloca un ratón en cada celda del laberinto (excepto en la celda de salida) y se inicia un temporizador de cuenta atrás. Cuando el cronómetro se detiene, contamos la cantidad de ratones que han salido del laberinto.

¿Puedes escribir un programa que, dada la descripción del laberinto y el límite de tiempo, prediga la cantidad de ratones que saldrán del laberinto? Puedes suponer que no hay cuellos de botella en el laberinto, es decir, que todas las celdas tienen espacio para un número arbitrario de ratones.

Entrada

La entrada está compuesta por diversos casos de prueba. La primera línea de cada caso contiene 4 números: el número N de celdas del laberinto (numeradas de 1 a N), el número S de la celda donde se encuentra la salida, el número T de segundos con el que se inicia el cronómetro para la cuenta atrás, y el número P de pasadizos. Las siguientes P líneas describen cada una de ellas un pasadizo, dando 3 números: dos números de celda A y B y los segundos que tarda un ratón en llegar de A a B.

Obsérvese que cada conexión es unidireccional, es decir, los ratones no pueden viajar de B a A a menos que haya otra línea que especifique ese pasadizo. Además, el tiempo requerido para viajar en cada dirección puede ser diferente.

Salida

Para cada caso de prueba el programa debe escribir una línea con el número de ratones que alcanzarán la celda de salida S en como mucho T segundos.

Entrada de ejemplo

```
5 5 20 5
1 2 5
1 4 10
2 4 7
3 4 15
4 5 10
3 1 10 2
2 3 5
3 2 6
```

Salida de ejemplo

```
3
0
```

Autor: Alberto Verdejo.