

El viaje más largo

¡Los organizadores del IOI 2023 tienen un gran problema! Se olvidaron de planificar el viaje a Ópusztaszer para el día siguiente. Pero quizás aún no sea demasiado tarde...

Hay N puntos de referencia en Ópusztaszer indexados de 0 a $N - 1$. Algunos pares de estos puntos de referencia están conectados por **carreteras bidireccionales**. Cada par de puntos de referencia está conectado como máximo por una carretera. Los organizadores *no saben* qué puntos de referencia están conectados por carreteras.

Decimos que la **densidad** de la red de carreteras en Ópusztaszer es **al menos** δ si cada 3 puntos de referencia distintos tienen al menos δ carreteras entre ellos. En otras palabras, para cada triplete de puntos de referencia (u, v, w) tal que $0 \leq u < v < w < N$, entre los pares de puntos de referencia (u, v) , (v, w) y (u, w) al menos δ pares están conectados por una carretera.

Los organizadores *conocen* un número entero positivo D tal que la densidad de la red de carreteras es al menos D . Tenga en cuenta que el valor de D no puede ser superior a 3.

Los organizadores pueden realizar **llamadas** al expedidor telefónico de Ópusztaszer para recoger información sobre las conexiones por carretera entre determinados puntos de referencia. En cada llamada deben especificarse dos matrices no vacías de puntos de referencia $[A[0], \dots, A[P - 1]]$ y $[B[0], \dots, B[R - 1]]$. Los puntos de referencia deben ser distintos por pares, es decir,

- $A[i] \neq A[j]$ para cada i y j tal que $0 \leq i < j < P$;
- $B[i] \neq B[j]$ para cada i y j tal que $0 \leq i < j < R$;
- $A[i] \neq B[j]$ para cada i y j tal que $0 \leq i < P$ y $0 \leq j < R$.

Para cada llamada, el expedidor informa de si hay una carretera que conecte un punto de referencia de A y un punto de referencia de B . Más concretamente, el expedidor itera sobre todos los pares i y j tales que $0 \leq i < P$ y $0 \leq j < R$. Si, para cualquiera de ellos, los puntos de referencia $A[i]$ y $B[j]$ están conectados por una carretera, el despachador devuelve `true`. En caso contrario, el despachador devuelve `false`.

Un **viaje** de longitud l es una secuencia de *puntos de referencia distintos* $t[0], t[1], \dots, t[l - 1]$, donde para cada i entre 0 y $l - 2$, ambos inclusive, el punto de referencia $t[i]$ y el punto de referencia $t[i + 1]$ están conectados por una carretera. Un viaje de longitud l se denomina **viaje más largo** si no existe ningún viaje de longitud igual o superior a $l + 1$.

Tu tarea es ayudar a los organizadores a encontrar un viaje más largo en Ópusztaszer haciendo llamadas al despachador.

Detalles de la implementación

Usted debe implementar el siguiente procedimiento

```
int[] longest_trip(int N, int D)
```

- N : el número de puntos de referencia en Ópusztaszer.
- D : la densidad mínima garantizada de la red de carreteras.
- Este procedimiento debe devolver una matriz $t = [t[0], t[1], \dots, t[l-1]]$, que representa un viaje más largo.
- Este procedimiento puede ser llamado **múltiples veces** en cada caso de prueba.

El procedimiento anterior puede hacer llamadas al procedimiento siguiente:

```
bool are_connected(int[] A, int[] B)
```

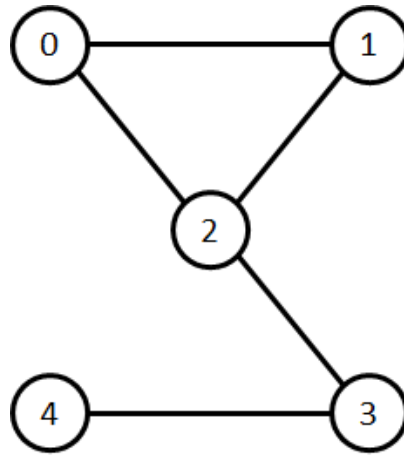
- A : una matriz no vacía de puntos de referencia distintos.
- B : una matriz no vacía de puntos de referencia distintos.
- A y B deben ser disjuntos.
- Este procedimiento devuelve true si hay un punto de referencia de A y un punto de referencia de B conectados por una carretera. En caso contrario, devuelve false.
- Este procedimiento puede invocarse un máximo de 32.640 veces en cada invocación de longest_trip, y un máximo de 150.000 veces en total.
- La longitud total de las matrices A y B pasadas a este procedimiento en todas sus invocaciones no puede exceder de 1 500 000.

El calificador **no es adaptativo**. Cada envío se califica con el mismo conjunto de casos de prueba. Es decir, los valores de N y D , así como los pares de puntos de referencia conectados por carreteras, son fijos para cada llamada de longest_trip dentro de cada caso de prueba.

Ejemplos

Ejemplo 1

Consideremos un escenario en el que $N = 5$, $D = 1$, y las conexiones de las carreteras son las que se muestran en la siguiente figura:



El procedimiento `longest_trip` es llamado de la siguiente manera:

```
longest_trip(5, 1)
```

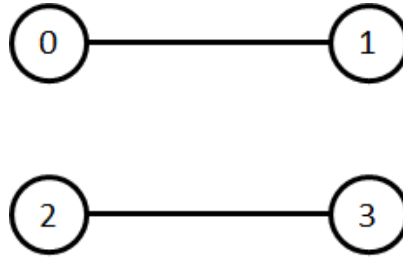
El procedimiento puede hacer llamadas a `are_connected` como sigue.

Llamada	Pares unidos por una carretera	Valor de retorno
<code>are_connected([0], [1, 2, 4, 3])</code>	(0,1) y (0,2)	true
<code>are_connected([2], [0])</code>	(2,0)	true
<code>are_connected([2], [3])</code>	(2,3)	true
<code>are_connected([1, 0], [4, 3])</code>	ninguno	false

Después de la cuarta llamada, resulta que *ninguno* de los pares (1,4), (0,4), (1,3) y (0,3) está conectado por una carretera. Como la densidad de la red es al menos $D = 1$, vemos que del triplete (0,3,4), el par (3,4) debe estar conectado por una carretera. Del mismo modo, los puntos de referencia 0 y 1 deben estar conectados.

Llegados a este punto, se puede concluir que $t = [1, 0, 2, 3, 4]$ es un viaje de longitud 5, y que no existe un viaje de longitud mayor que 5. Por lo tanto, el procedimiento `longest_trip` puede devolver `[1, 0, 2, 3, 4]`.

Consideremos otro escenario en el que $N = 4$, $D = 1$, y las carreteras entre los puntos de referencia son como se muestra en la siguiente figura:



El procedimiento `longest_trip` es llamado de la siguiente manera

```
longest_trip(4, 1)
```

En este escenario, la longitud del viaje más largo es de 2. Por lo tanto, después de unas cuantas llamadas al procedimiento `are_connected`, el procedimiento `longest_trip` puede devolver uno de $[0, 1]$, $[1, 0]$, $[2, 3]$ o $[3, 2]$.

Ejemplo 2

La subtarea 0 contiene un caso de prueba de ejemplo adicional con $N = 256$ puntos de referencia. Este caso de prueba se incluye en el paquete adjunto que puede descargar del sistema del concurso.

Restricciones

- $3 \leq N \leq 256$
- La suma de N en todas las llamadas a `longest_trip` no exceda de 1024 en cada caso de prueba.
- $1 \leq D \leq 3$

Subtareas

1. (5 puntos) $D = 3$
2. (10 puntos) $D = 2$.
3. (25 puntos) $D = 1$. Sea l^* la longitud del viaje más largo. El procedimiento `longest_trip` no tiene por qué devolver un viaje de longitud l^* . En su lugar, debe devolver un viaje de longitud al menos $\left\lceil \frac{l^*}{2} \right\rceil$.
4. (60 puntos) $D = 1$

En la subtarea 4 tu puntuación se determina en base al número de llamadas al procedimiento `are_connected` sobre una única invocación de `longest_trip`. Sea q el número máximo de llamadas entre todas las invocaciones de `longest_trip` en cada caso de prueba de la subtarea. Tu puntuación para esta subtarea se calcula de acuerdo con la siguiente tabla:

Condición	Puntos
$2\,750 < q \leq 32\,640$	20
$550 < q \leq 2\,750$	30
$400 < q \leq 550$	45
$q \leq 400$	60

Si, en cualquiera de los casos de prueba, las llamadas al procedimiento `are_connected` no se ajustan a las restricciones descritas en Detalles de la implementación, o el array devuelto por `longest_trip` es incorrecto, la puntuación de su solución para esa subtask será de 0.

Calificador de ejemplo Grader

Sea C que denota el número de escenarios, es decir, el número de llamadas a `longest_trip`. El calificador de ejemplo lee la entrada en el siguiente formato:

- línea 1: C

A continuación se describen los siguientes C escenarios.

El calificador de ejemplo lee la descripción de cada escenario en el siguiente formato:

- línea 1: $N\ D$
- línea $1 + i$ ($1 \leq i < N$): $U_i[0]\ U_i[1]\ \dots\ U_i[i - 1]$

Aquí, cada U_i ($1 \leq i < N$) es una matriz de tamaño i , que describe qué pares de puntos de referencia están conectados por una carretera. Para cada i y j tales que $1 \leq i < N$ y $0 \leq j < i$:

- si los puntos de referencia j y i están conectados por una carretera, entonces el valor de $U_i[j]$ debe ser 1;
- si no hay carretera que conecte los puntos de referencia j y i , entonces el valor de $U_i[j]$ debe ser 0.

En cada escenario, antes de llamar a `longest_trip`, el calificador de ejemplo comprueba si la densidad de la red de carreteras es al menos D . Si no se cumple esta condición, imprime el mensaje `Insufficient Density` y termina.

Si el calificador de ejemplo detecta una violación del protocolo, la salida del calificador de ejemplo es `Protocol Violation: <MSG>`, donde `<MSG>` es uno de los siguientes mensajes de error:

- `invalid array`: en una llamada a `are_connected`, al menos una de las matrices A y B
 - está vacía, o
 - contiene un elemento que no es un entero entre 0 y $N - 1$, ambos inclusive, o
 - contiene el mismo elemento al menos dos veces.

- `non-disjoint arrays`: en una llamada a `are_connected`, las matrices A y B no son disjuntas.
- `too many calls`: el número de llamadas realizadas a `are_connected` supera los 32 640 en la invocación actual de `longest_trip`, o supera los 150 000 en total.
- Demasiados elementos: el número total de puntos de referencia pasados a `are_connected` en todas las llamadas supera los 1 500 000.

En caso contrario, que los elementos del array devuelto por `longest_trip` en un escenario sean $t[0], t[1], \dots, t[l-1]$ para algún l no negativo. El calificador de ejemplo imprime tres líneas para este escenario en el siguiente formato:

- línea 1: l
- línea 2: $t[0] \ t[1] \ \dots \ t[l-1]$
- línea 3: el número de llamadas a `are_connected` en este escenario

Finalmente, el calificador de ejemplo imprime:

- línea $1 + 3 \cdot C$: el número máximo de llamadas a `are_connected` sobre todas las llamadas a `longest_trip`