Trabajo práctico 2

Fecha límite de entrega: Viernes 7 de octubre, hasta las 17:00 hs.
Fecha estimada de devolución: Tres semanas después de la fecha en la que es entregado el TP.
Primer fecha de reentrega: Dos semanas después de devueltas las correcciones, hasta las 17:00 (viernes) o hasta las 22 (martes).
Segunda fecha de reentrega: Viernes 9 de diciembre, hasta las 17:00 hs.

Este trabajo práctico consta de 4 problemas. Para aprobar el mismo se requiere aprobar todos los problemas.
Los requisitos de aprobación del mismo así como los criterios de corrección están detallados en las pautas de aprobación que pueden encontrar en el repositorio de la materia.

En este trabajo práctico se evaluarán contenidos de algoritmos y problemas sobre grafos. Los problemas que serán evaluados en el mismo serán sobre:

- Flujo y matching bipartito.
- Teorema de Dilworth.
- Ejes puente, puntos de articulación y componentes biconexas.
- Orden topológico y componentes fuertemente conexas.

Problema A: Alumnos de secundario

Al fin tenemos la suerte
Que casi toda esta parte
De armar el TP fue un arte
Sino puede ser la muerte

Nos pide el departamento
Hoy difundir la carrera
A cambio de una remera
Si ya la tienen lo siento

Queremos tener contacto
Con chicos adolescentes
De entre toda la gente
Para generar impacto

Dificil tener presencia
Por temas combinatorios
En todos los territorios
Para difundir la ciencia

Por eso hoy necesitamos
Sabiendo que las personas
Se mueven en ciertas zonas
Que vos nos des una mano

Tenemos un objetivo
El plan de cubrir las rutas
Y de una manera astuta
Seamos abarcativos

A chicos de secundaria
Que van hoy para la escuela
Veamos si los desvela
La vida universitaria

Pongamos en su camino
A alguien que sea vidriera
Contando de la carrera
Que vean que esto no es chino

Para eso es que averiguamos
Los puntos donde hay colegios
Usemos el privilegio
De estar muy bien informados

También sabemos los puntos
Donde estos alumnos viven
Prohibamos que nos esquiven
Tratemos de hacerlo juntos

Un mapa les proveemos
Por eso les preguntamos
A cuantos necesitamos
Para que a todos lleguemos

Para todos esos pibes
Que en una esquina haya uno
Que espere muy oportuno
Por donde sea que camine

El problema consiste en dadas N esquinas, M calles bidireccionales que conectan pares de esas esquinas, sabiendo que en algunas de esas esquinas hay escuelas, en otras de esas esquinas hay alumnos que van a alguna de esas escuelas, y que todo par de esquinas están conectadas por un camino que usa algunas de esas calles, contar la mínima cantidad de esquinas en las que podemos podemos poner a un estudiante del departamento a contar sobre la carrera de Ciencias de la Computación, de modo tal que no importe qué camino utilice cada chico para llegar de su casa al colegio, siempre tenga que pasar por una esquina donde lo podamos interceptar para contarle de la carrera.

El algoritmo debe tener una complejidad temporal $\mathbf{O}(NM^2)$.

Entrada

La primera línea consta de un valor entero positivo N, que indica la cantidad de esquinas de la ciudad, y un entero positivo M que indica la cantidad de calles que conectan pares de esas esquinas.

A continuación, N líneas, una por cada esquina, indicando si en esa esquina hay una escuela ('E'), un alumno ('A') o ninguna de las dos ('X'). Si hay un alumno, además se indicará el número de esquina donde se encuentra la escuela donde estudia dicho alumno (que será una de las esquinas que sea indicada como una esquina con una escuela).

Por último, la entrada contará con M líneas indicando los pares de esquinas que conecta cada calle.

La entrada contará con el siguiente formato:

```
N M
D1
D2
...
DN
A1 B1
A2 B2
...
AM BM
```

Indicando Ai y Bi los números de esquinas que están conectadas por la i-ésima calle bidireccional, y Di la descripción de la i-ésima esquina, que podrá ser alguna de las siguientes:

```
E
A i
X
```

Indicando que hay una escuela, la casa de un alumno que va a la escuela que se encuentra en la i-ésima esquina, o ninguna de las dos. Si Di es A j entonces Dj es necesariamente E.

Salida

La salida debe constar de una línea que indique la mínima cantidad de personas que necesita el departamento para cubrir al menos una esquina en cada ruta que conecte la casa de un alumno con su escuela, con el siguiente formato:

```
P
```

siendo P esta cantidad de personas.

Entrada de ejemplo 1	**Salida para la entrada de ejemplo 1**
4 3 E X X A 1 1 2 2 3 3 4	1

Entrada de ejemplo 2	**Salida para la entrada de ejemplo 2**
4 4 E X X A 1 1 2 1 3 2 4 3 4	1

Entrada de ejemplo 3	**Salida para la entrada de ejemplo 3**

`8 12`
`A 6`
`A 8`
`A 7`
`X`
`X`
`E`
`E`
`E`
`1 4`
`1 5`
`2 4`
`2 5`
`3 4`
`3 5`
`6 4`
`6 5`
`7 4`
`7 5`
`8 4`
`8 5`
	`2`

Entrada de ejemplo 4	**Salida para la entrada de ejemplo 4**

`3 2`
`A 2`
`E`
`A 2`
`1 2`
`2 3`
	`1`

Explicación del ejemplo 1:
Como hay un sólo alumno y un sólo camino de su casa a su escuela podemos poner una persona en cualquier lugar de su camino.

Explicación del ejemplo 2:
El único alumno tiene dos caminos distintos para llegar a la escuela, pero podemos interceptarlo tanto en la puerta de su casa como en la puerta de la escuela.

Explicación del ejemplo 3:
Hay casos en los que, si bien una persona puede cubrir todos los caminos de un mismo alumno, conviene poner varias personas que le cubran a un alumno sus caminos para llegar a una solución óptima. Además, las calles son bidireccionales por lo que un alumno puede ir por ejemplo de la esquina 4 a la esquina 6.

Explicación del ejemplo 4:
Puede haber dos alumnos que vayan a la misma escuela.

Pesos mínimo y máximo: 8 y 11.

Cotas recomendadas para testear: Se recomienda testear el problema con valores de $N \leq 100$, $M \leq 500$.

Problema B: Buenos gráficos

A todos hoy les pedimos
Que lean este enunciado
Para ayudar al mercado
Y ver como lo medimos

Los precios de las acciones
Este año han sido fluctuantes
Ya no son como eran antes
Sufren muchas variaciones

Por eso es que les pasamos
La evolución de los precios
Les pido que no sean necios
Y vean como graficamos

Impriman un resultado
Queremos que minimicen
La cantidad y que avisen
De gráficos que han usado

Un gráfico es un conjunto
De una o más mediciones
De precios de las acciones
Que no cruzan ni en un punto

Veamos como empezamos
Al grafico permitido
De acciones que hemos medido
Formalmente definamos

Dados dos días seguidos
Del rango que analizamos
Segmento de recta trazamos
Para que queden unidos

Los puntos que conectamos
Indican los precios diarios
De acciones en las que varios
Valores tenemos dados

Los gráficos con acciones
Tendrán segmentos de recta
Y una respuesta perfecta
No habrá con intersecciones

El problema consiste en, dados los valores de A acciones durante D días seguidos, calcular la mínima cantidad de gráficos que es necesaria, para poder graficar la evolución de los precios de las acciones (siendo un eje el tiempo y el otro eje el precio, uniendo las mediciones por segmentos de recta) sin que haya un gráfico en el que dos o más acciones se intersequen.

El algoritmo debe tener una complejidad temporal $\mathbf{O}(A^2D)$

Entrada

La primera línea consta de un valor entero positivo A, que indica la cantidad de acciones que medimos, y un entero positivo D que indica la cantidad de días durante los cuales medimos las acciones.

A esta línea le siguen A líneas con D enteros cada una, indicando los números $P_{i,j}(1 \leq i \leq A, 1 \leq j \leq D$ que indican el precio de la acción i en el día j. La entrada contará con el siguiente formato:

```
A D
P11 P12 ... P1D
P21 P22 ... P2D
...
PA1 PA2 ... PAD
```

Salida

La salida debe constar de una línea que indique la mínima cantidad de gráficos que se necesitan para poder graficar la evolución de los precios de las A acciones a lo largo de los D días, con el siguiente formato:

```
G
```

siendo G esta cantidad mínima de gráficos.

Entrada de ejemplo 1	Salida para la entrada de ejemplo 1
3 3 5 5 5 4 4 6 4 5 4	3

Entrada de ejemplo 2	Salida para la entrada de ejemplo 2
5 2 1 1 2 2 5 4 4 4 4 1	2

Entrada de ejemplo 3	Salida para la entrada de ejemplo 3
4 4 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4	1

Entrada de ejemplo 4	Salida para la entrada de ejemplo 4
5 2 1 5 2 4 3 3 4 2 5 1	5

Explicación del ejemplo 1:
Dos mediciones que no se crucen pero se toquen en un punto no pueden ir juntas en un mismo gráfico.

Pesos mínimo y máximo: 7 y 10.

Cotas recomendadas para testear: Se recomienda testear el problema con valores de $A \leq 1000$, $D \leq 100$.

Problema C: Cortes programados

Nos pide hoy el intendente
Nuestros servicios brindemos
Para que el caos evitemos
Siendo útiles a la gente

Las obras darán comienzo
Del nuevo pavimentado
Y ya nos han contratado
Para este trabajo intenso

Hay calles del municipio
Que están muy mal asfaltadas
Para eso serán cortadas
Un sólo día en principio

Nos piden saquemos cuentas
Para ayudar a la gente
Que ya les dijo de frente
Cuidá cuando pavimentas

Si es de lunes a viernes
Va a ser medio complicado
Ver algo desconectado
Y eso a mi me concierne

Por eso hoy a vos te pido
Respondas unas preguntas
Podés responderlas juntas
Y así serás bienvenido

La duda que hoy no distingo
Es ver si en día laborable
Un corte es algo viable
O queda para el domingo

Para eso te hago consultas
Con tus respuestas decido
No quiero hacer mucho ruido
Que afecte a la gente adulta

Te paso el mapa de calles
Decime lo que pregunto
Yo tus respuestas las junto
Y veo yo los detalles

En este problema, son dadas N esquinas (numeradas de 1 a N) y M calles bidireccionales (numeradas de 1 a M) de una ciudad donde existe para cada par de esquinas un camino utilizando estas calles que las conecta. Además, recibimos Q queries (consultas) que pueden ser de varios tipos:

- Tipo A: dadas dos esquinas e_1 y e_2, imprimir la cantidad de calles que, en caso de ser cortadas (solo cortando esa calle), impiden llegar de e_1 a e_2.
- Tipo B: dada una calle, imprimir un 1 si al cortar la calle existen al menos dos esquinas entre las que deja de haber un camino, y 0 en caso contrario.
- Tipo C: dada una esquina e_1, imprimir la cantidad de esquinas e_2 tales que de cortar una sóla calle, sea cual sea, seguirá habiendo camino de e_1 a e_2.

El algoritmo debe tener una complejidad temporal $\mathbf{O}(M + MQ_A + Q_B + Q_C)$ siendo Q_A la cantidad de queries de tipo A, Q_B la cantidad de queries de tipo B y Q_C la cantidad de queries de tipo C.

Entrada

La primera línea consta de un valor entero positivo N y un entero positivo M, que indican la cantidad de esquinas y calles.

A esta línea le siguen M líneas con dos enteros U_i, V_i ($1 \leq U_i, V_i \leq N, U_i \neq V_i$) indicando los números de calles que conecta la i-ésima arista. No habrá dos calles que conecten las mismas esquinas ($U_i = V_i \rightarrow U_j \neq V_j$ y $U_i = V_j \rightarrow U_j \neq V_i$)

Luego sigue una línea con un entero positivo Q indicando la cantidad de queries, y Q líneas, una con cada query, cuya descripción será en alguno de los siguientes formatos:

- A e_1 e_2 (siendo A el caracter 'A', e_1 y e_2 números válidos de esquinas ($1 \leq e_1, e_2 \leq N, e_1 \neq e_2$)).
- B c (siendo B el caracter 'B', c un número válido de calle ($1 \leq c \leq M$))
- C e (siendo C el caracter 'C', e un número válido de esquina ($1 \leq e \leq N$)).

La entrada contará con el siguiente formato:

```
N M
U1 V1
U2 V2
...
```

```
UM VM
Q
D1
D2
...
DQ
```

Siendo Di la descripción de las queries en el formato provisto anteriormente.

Salida

La salida deberá constar de Q líneas, una con la respuesta para cada query:

```
R1
R2
...
RQ
```

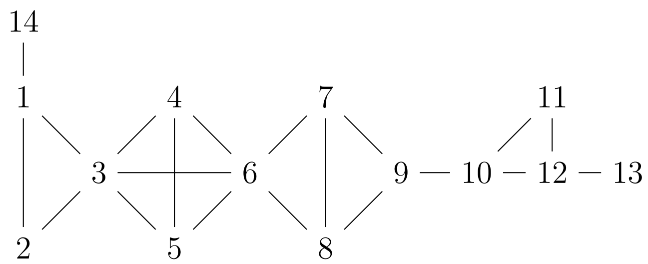
siendo R_i la respuesta de la i-ésima query.

Entrada de ejemplo 1	**Salida para la entrada de ejemplo 1**
14 20 1 2 2 3 3 1 3 4 3 5 3 6 4 5 4 6 5 6 6 7 6 8 7 8 7 9 8 9 9 10 10 11 10 12 11 12 12 13 1 14 Q A 13 14 A 2 8 B 1 B 20 C 10 C 14	3 0 0 1 2 0

Explicación del ejemplo 1:
Veamos una por una las queries:

- A 13 14: Cortar las calles 15 (9-10), 19 (12-13) o 20 (1-14) desconecta a las esquinas 13 y 14 entre sí.
- A 2 8: No importa qué calle sea cortada, las esquinas 2 y 8 seguirán conectadas

- B 1: La calle 1 es la calle (1,2) que en caso de ser cortada no deja de haber camino entre ningún par de esquinas.
- B 20: La calle 2 es la calle (1,14) que es la única calle que conecta ambas esquinas, luego las esquinas 1 y 14 quedan desconectadas al desconectar esta calle.
- C 10: Las únicas esquinas que siguen conectadas a la esquina 10 sin importar qué calle es cortada son las esquinas 11 y 12. Notar que si no desconectamos la calle 19 (12-13) la esquina 13 sigue siendo accesible, pero si cortamos esta calle las demás esquinas son accesibles desde la esquina 10, pero no hay ninguna esquina que no sea la 11 o la 12 y que sea accesible desde la 10 cortemos la calle que cortemos.
- C 14: Si cortamos por ejemplo, la calle 20 (1,14) ninguna esquina será accesible desde la esquina 14.



Pesos mínimo y máximo: 8 y 9.

Cotas recomendadas para testear: Se recomienda testear el problema con valores de $N \leq 10^4$, $M \leq 10^5$, $Q \leq 10^5$, $Q_A \leq 10^3$.

Pista 1: Si entre todo par de esquinas hay un camino $N \in \mathbf{O}(M)$.

Pista 2: Q_A puede ser igual a 0, por lo que $MQ_A \notin \mathbf{O}(M + MQ_A)$

Problema D: Después de la próxima clase

El problema 4 será agregado al enunciado después de la clase (de martes) de orden topológico y componentes fuertemente conexas. (Pista: sale con temas vistos en esa clase!).