

## Trabajo Práctico: Grafos

La resolución de este trabajo práctico debe ser enviada a través de GitHub Classroom.

El código provisto como parte de la solución a los ejercicios deberá estar documentado apropiadamente (por ejemplo, con comentarios en el código). Aquellas soluciones que no requieran programación, como así también la documentación adicional de código que se desee proveer, debe entregarse en archivos de **texto convencionales**, en el mismo repositorio de entrega de las soluciones, con nombres que permitan identificar fácilmente su contenido.

Los grupos para la resolución del trabajo deben ser conformados por dos o tres integrantes.

La fecha límite para la entrega de la solución del TP es en **13** días desde la fecha de entrega del enunciado.

### Enunciado

Debido a su popularidad, las redes sociales han sido objeto de diferentes análisis. Un campo de investigación en esta área es la identificación de varios tipos de usuarios y grupos.

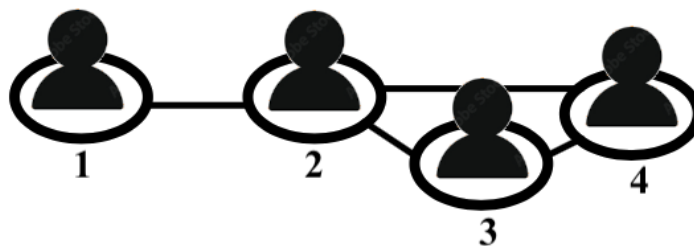
En ciertas redes las personas se relacionan mediante el concepto de amistad mutua. Por ejemplo, si Juan es amigo de Margarita, entonces Margarita es amiga de Juan.

Llamaremos comunidad a un conjunto de personas relacionadas mediante una amistad directa, o bien mediante una amistad indirecta. Una amistad es indirecta cuando existe una cadena de amigos entre una persona y la otra. En una red social compuesta de comunidades, dos personas que son amigas, están en la misma comunidad.

Hay amistades que son “esenciales” para mantener la unidad de la comunidad, en el sentido de que si la amistad directa entre dos personas desaparece, puede ocurrir que dos personas que antes pertenecían a la misma comunidad ahora ya no, quedando en comunidades diferentes.

Se pide escribir un algoritmo que, dada la información de una comunidad específica de una red social, es decir personas y amistades que las relacionan, calcule el número de amistades esenciales.

### Ejemplo



En la imagen, se describe una comunidad, con 4 usuarios, 4 amistades directas y 2 amistades indirectas. En el ejemplo, la única amistad esencial, es entre el usuario 1 y 2, por lo tanto el número de amistades esenciales es 1.

Se describe a continuación los archivos que se provee para la solución de este trabajo y que podrá encontrar en el repositorio GitHub correspondiente a su grupo:

- Una Interface *Grafo*, que contiene la especificación de las operaciones públicas necesarias para manipular el tipo de datos *Grafo*.
- Una clase *GrafoNoDirigido*, que deben implementar.
- Una Clase *Vertice*, con métodos para manipular la unidad fundamental de un grafo.
- Una Clase *Arista*, que contiene dos enteros que representan vértices.
- Un archivo de entrada *amigos.in* que contiene el ejemplo en la imagen.
- Una Clase *Amistades*, con un método principal que se encarga de cargar el archivo de entrada. Y además un método *amistadEsencial*. Este método tiene el siguiente perfil:

```
public static int amistadEsencial(Grafo grafo)
```

Donde:

- El parámetro es un grafo que representa una comunidad de amigos específica.

### Consideraciones para la solución

Para la solución del trabajo práctico debe tener en cuenta que el grafo que representa una comunidad de una red social, se construye a partir de los datos tomados de un archivo de entrada con el siguiente formato:

- La primera línea del archivo contiene dos enteros, el primero representa el número de usuarios de la red que pertenecen a la comunidad ( $U$ ),  $1 < U \leq 100000$  y el segundo valor representa el número de amistades ( $A$ ),  $1 \leq A \leq 150000$ . Los usuarios son identificados con enteros de 1 a  $U$ .
- Cada una de las siguientes  $A$  líneas contiene información acerca de las amistades, enteros  $a, b$ , donde  $a$  y  $b$  son los números que representan usuarios con una relación de amistad directa. El orden en que aparecen  $a$  y  $b$  es irrelevante.

El archivo de entrada, correspondiente a la imagen del ejemplo sería:

4	4
1	2
2	3
3	4
2	4

amigos.in