

**EXAMEN PARCIAL 2 - Clases y Colecciones**

Leé con cuidado el enunciado y por lo menos dos veces para resolver lo pedido. Pensá bien la estrategia de resolución antes de comenzar el desarrollo de lo que te solicitan. El objetivo de este examen es **evaluar la correcta aplicación de los conceptos y técnicas** vistos hasta el momento:

- Correcta implementación de constructores.
- Modularización reutilizable y mantenible con uso de métodos con correcta parametrización y correcto encapsulamiento, publicando *setters* y *getters* sólo cuando corresponda.
- Manejo de clases, enumerados y colecciones.

**Enunciado**

Un grupo de amigos suele juntarse a comer cada tanto y decide juntarse a comer un asado. Desde hace un tiempo ya definieron una mecánica que los ayuda a definir cuándo terminarán haciendo la Reunión:

- Primero se anotan los interesados en participar en esa reunión.
- A la Reunión se le agregan las distintas propuestas que se hacen. Para crear una Propuesta de Reunión hace falta el nombre de un amigo (que ya debe haberse anotado) y en qué momento se propone (SABADO MEDIODIA, DOMINGO NOCHE, etc.).
- Luego, cualquiera de los amigos puede adherir a alguna de las Propuestas existentes, “bajarse” de una propuesta o cambiar de opinión (“moverse” a otra de las Propuestas existentes). Siempre se indica el nombre del Amigo y la ubicación de la Propuesta (por ejemplo, “Juan”, SABADO y MEDIODIA).
- Finalmente se listan las propuestas “sobrevivientes” y quiénes están anotados para cada una de ellas.

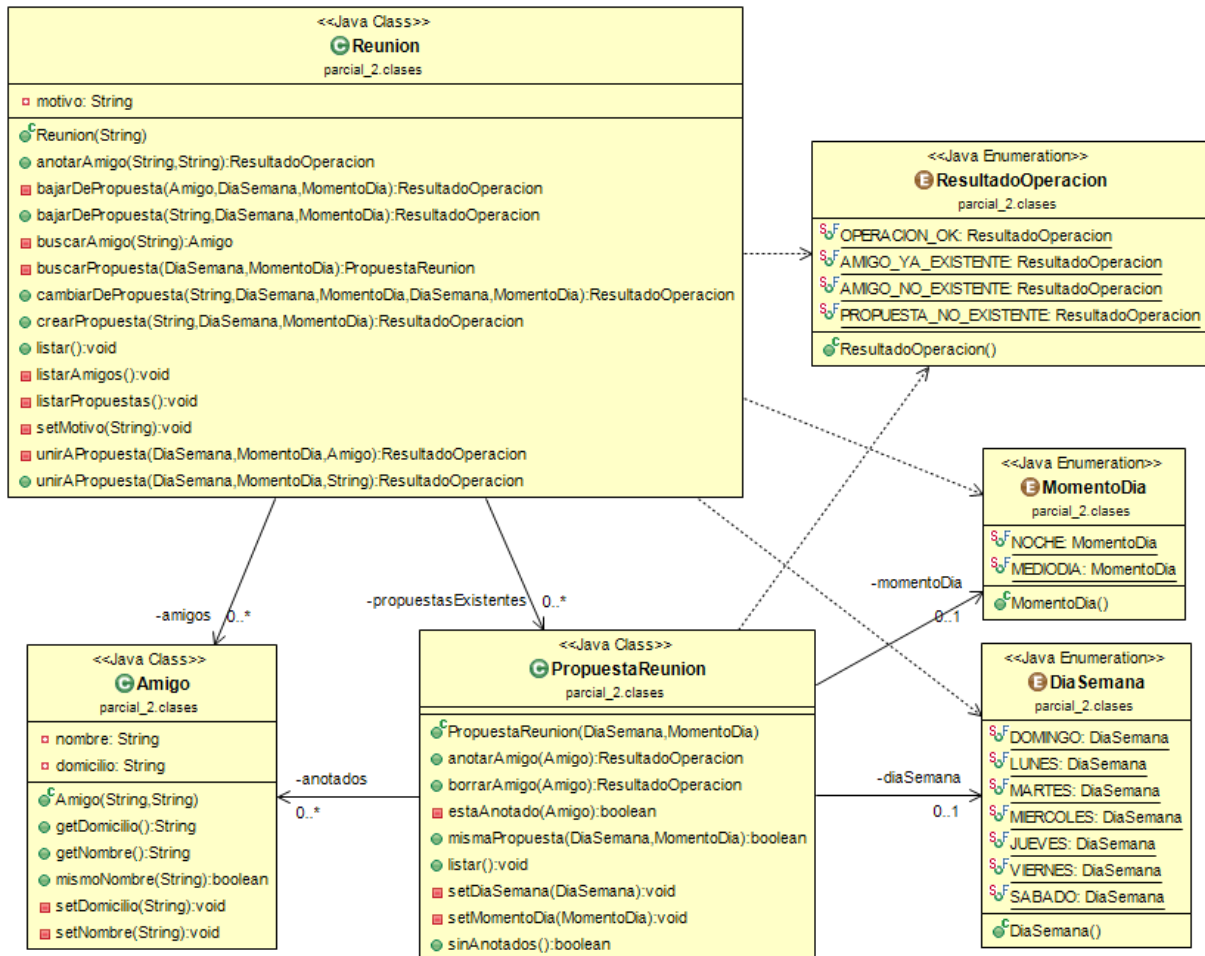
Uno de los amigos ya había escrito un programa que realiza casi completamente todo lo necesario para cumplir con esta mecánica pero el programa se perdió y nuestro amigo sólo encontró una versión incompleta, donde faltan algunas partes.

**NOTA IMPORTANTE:** Se deben tener en cuenta las siguientes consideraciones.

- No pueden repetirse ni personas ni días propuestos. Si alguien propone juntarse el mismo día/momento que otro ya propuso simplemente se lo agrega a la propuesta coincidente anterior.
- Si al “bajar” a alguien de una propuesta ésta quedase vacía, la misma debe eliminarse.
- Cada operación (el agregado de una fecha, de un amigo, agregar o sacar a alguien de una fecha propuesta, etc.) da por resultado una de los siguientes valores:
  - **OPERACION OK,**
  - **AMIGO YA EXISTENTE,**
  - **AMIGO NO EXISTENTE,**
  - **PROPUESTA NO EXISTENTE**

Por suerte, también encontró un diagrama de clases donde se refleja el diseño completo del programa que existía. El diagrama es el siguiente:

**EXAMEN PARCIAL 2 - Clases y Colecciones**



Lo que quedó sin completar (y que es necesario para que el programa funcione) es:

- Completar los constructores de las clases **Reunion** y **PropuestaReunion**.
- Completar la búsqueda de una propuesta.
- Completar el agregado de un amigo a una propuesta existente.
- Completar la cancelación de una propuesta (borrando la misma si queda "sin amigos").
- Por último, quieren agregar algo que faltaba en la versión que existía del programa: un "Listado de amigos faltantes" (que debería mostrar a los amigos que no se unieron a ninguna propuesta existente ni propusieron nada).

El programa funcionando debería mostrar algo semejante a esto:

```

Agregando a Amigo 1 que vive en Domicilio 1
Agregando a Amigo 2 que vive en Domicilio 2
Agregando a Amigo 3 que vive en Domicilio 3
Agregando a Amigo 4 que vive en Domicilio 4
Agregando a Amigo 5 que vive en Domicilio 5
Agregando propuesta SABADO (NOCHE). Propuesta por Amigo 1
Agregando propuesta DOMINGO (MEDIODIA). Propuesta por Amigo 2
Agregando propuesta SABADO (NOCHE). Propuesta por Amigo 3
Amigo 3 se quiere agregar al DOMINGO (MEDIODIA)
Colado se quiere agregar al SABADO (NOCHE)
  
```

```
AMIGO_NO_EXISTENTE
Amigo 3 se borra del SABADO (MEDIODIA)
PROPUESTA_NO_EXISTENTE
Amigo 3 se borra del DOMINGO (MEDIODIA)
Amigo 2 se pasa del DOMINGO (MEDIODIA) al SABADO (NOCHE)
Amigos registrados para El Asadito
Amigo 1
Amigo 2
Amigo 3
Amigo 4
Amigo 5
Propuestas registradas para El Asadito
Hay 3 amigos anotados para el SABADO (NOCHE)
Amigo 1
Amigo 3
Amigo 2
Amigos no registrados en ninguna propuesta:
Amigo 4
Amigo 5
```