

Módulo POO – Actividades de repaso

1. La UNLP desea un sistema que le permita administrar a sus proyectos, investigadores y subsidios. Un proyecto tiene un nombre, un código, un director y los investigadores que participan en el proyecto (hasta 50 como máximo). De cada investigador se desea saber su nombre, apellido, categoría (1 a 5) y su especialidad. Cualquier investigador puede pedir hasta un máximo de 5 subsidios. De cada subsidio se desea saber el monto pedido, el motivo y se fue otorgado o no.

- a. Implemente el modelo de clases teniendo en cuenta:

- i. Un proyecto solo debería poder construirse con el nombre y el código
- ii. Un investigador solo debería poder construirse con nombre y apellido, categoría y especialidad
- iii. Un subsidio solo debería poder construirse con el monto solicitado y el motivo. Un subsidio siempre se crea en estado no-otorgado.

- b. Implemente métodos (en las clases donde corresponda) que permitan:

- i. `void agregarInvestigador(unInvestigador);`
// agrega un investigador a un proyecto.
- ii. `void agregarSubsidio(unSubsidio);`
// agrega un subsidio a un investigador.
- iii. `double dineroTotalOtorgado();`
// devuelve la cantidad de dinero de todos los subsidios otorgados a todos los investigadores de un proyecto.
- iv. `int cantidadDeSubsidios(String nombre_y_apellido);`
// devuelve la cantidad de subsidios (otorgados o no) solicitados por el investigador llamado "nombre_y_apellido".
- v. `void otorgarTodos(String nombre_y_apellido);`
// otorga todos los subsidios pendientes que tiene el investigador // llamado "nombre_y_apellido"
- vi. `String toString();`
// Devuelve un string que tiene el nombre del proyecto, su código, el nombre y apellido del director, el total de dinero otorgado y el nombre y apellido de cada investigador. Para cada investigador, // además, se debe agregar la categoría del mismo y el dinero de sus // subsidios otorgados.

NOTA: puede crear todos los métodos auxiliares que considere necesario.

- c. Escriba un programa principal que instancie un proyecto con un director y dos investigadores. Asigne dos subsidios a cada investigador y otorgue los subsidios del primero de ellos, luego imprima todos los datos del `toString` pedido.

2. Una escuela de artes musicales arma coros con algunos de sus alumnos para participar de ciertos eventos. Un coro tiene un director (un profesor de la escuela) y alumnos como coristas. De los coristas se conoce el nombre, el dni, la edad y el tono fundamental (un número entero). Del director se conoce el nombre, el dni, la edad y la antigüedad (un número entero). Los coros poseen un nombre y están formados por un director y una serie de coristas. Asimismo pueden formarse de dos formas: o bien los coristas se colocan en el escenario uno al lado del otro formando un semicírculo, o bien conforman hileras de la misma cantidad de coristas.



- a. Implemente el modelo de clases teniendo en cuenta que los coros deberían crearse con un director y sin ningún corista, pero sí sabiendo cuantos coristas va a tener el coro.
- b. Implemente métodos (en las clases donde corresponda) que permitan:
 - i. agregar un corista a un coro.
 - En el coro semicircular los coristas se deben ir agregando de izquierda a derecha
 - En el coro por hileras los coristas se deben ir agregando de izquierda a derecha completando la hilera antes de pasar a la siguiente, comenzando por la hilera de adelante.
 - ii. Determinar si un coro está lleno o no. Devuelve true si el coro tiene a todos sus coristas asignados o false en caso contrario.
 - iii. Determinar si un coro (se supone que está lleno) está bien formado.

Un coro está bien formado si:

 - En el caso del coro semicircular, de izquierda a derecha los coristas están ordenados de mayor a menor en cuanto a tono fundamental.
 - En el caso del coro orquestal, desde adelante hacia atrás los coristas están ordenados de mayor a menor en cuanto a tono fundamental y todos los miembros de una misma hilera tienen el mismo tono fundamental.
 - iv. Devolver la representación de un coro formada por el nombre del coro, todos los datos del director y todos los datos de todos los coristas.
- c. Cree una función main que instancie cuatro coros y que los vaya almacenando en un arreglo. El tipo de cada uno de los cuatro coros es leído por teclado. Una vez leído el tipo de coro se deberá leer o bien la cantidad de coristas (en el caso del coro semicircular) o la cantidad de hileras e integrantes por hilera (en el caso del coro orquestal), luego se deberá crear la cantidad de coristas necesarios (leyendo sus datos) y almacenándolos en el coro. Finalmente imprima toda la información de los cuatro coros indicando si están bien formados o no.