



# **Algoritmos y Programación II**

## **Trabajo Práctico 3**

**Trifecta**

## Ejercicio 1

Dada una lista de elementos enteros, simplemente enlazada implementada según la versión desarrollada en clase (con cursor), implementar el siguiente método:

```
// Si tu padrón es impar  
Lista* interseccion(Lista lista);
```

Devuelve un puntero a una lista que es la intersección entre la lista que invoca al método y la que se pasa por parámetro.

```
// Si tu padrón es par  
Lista* union(Lista lista);
```

Devuelve un puntero a una lista que es la unión entre la lista que invoca al método y la que se pasa por parámetro.

Implementar el método en dos versiones:

- Teniendo en cuenta que las listas no tienen orden.
- Teniendo en cuenta que las listas están ordenadas de menor a mayor, conservar el orden en la lista resultante.

Escribir en un main o una clase aparte un ejemplo de utilización de ambos métodos.

**Nota:** de la implementación de Lista no se puede cambiar nada de lo que se hizo en clase, pero se pueden agregar atributos y/o métodos si fueran necesarios.

Puntos 15

## Ejercicio 2

Adecuar la clase Lista implementada en clase (con enteros) para que sea Circular. La lista puede crecer hasta un tamaño máximo N que será una constante que vale 5 (esta constante la podemos modificar, pero no puede valer menos de 2).

Agregar y/o cambiar los métodos:

- alta:
  - Si la lista tiene menos de N elementos, el alta se hace agregando un nodo al final.
  - Si la lista tiene N elementos se reemplaza el elemento más antiguo por el nuevo, y el puntero que guarda la dirección del primer elemento, pasa al siguiente.
- mas\_antiguo: devuelve el valor más antiguo.
- promedio: devuelve el promedio de todos los elementos.

Escribir en un main o una clase aparte un ejemplo de uso.

Puntos 20

### Ejercicio 3

Implementar la clase Carrera que tiene los siguientes atributos: nombre, duración y una lista de materias, con la siguiente interfaz:

```
class Carrera {  
  
public:  
  
    /* Crea una Carrera con su nombre, la duración (en años) y una lista  
    de materias */  
  
    Carrera (string nombre, int duracion, Lista<string>* materias);  
  
  
    /* devuelve el nombre de la Carrera */  
  
    string obtener_nombre();  
  
  
    /* devuelve la duración */  
  
    int obtener_duracion ();  
  
  
    /* devuelve un puntero a la lista de materias */  
  
    Lista<string>* obtener_materias ();  
  
};
```

Implementar la clase Buscador\_Carreras con su método sugerir\_carreras, según:

```
class Buscador_Carreras {  
  
public:  
  
    /*  
    Pre: agregar las necesarias  
  
    Post: busca en "carreras" aquellas que tienen tres o más materias de  
    la lista "materias_predilectas" y una duración menor o igual que  
    duracion_maxima.  
  
    Devuelve un puntero a una lista con las carreras que cumplen con estas  
    características.  
  
    */  
  
    Lista<Carrera *>* sugerir_carreras(Lista<Carrera *>* carreras,  
                                       int duracion_maxima  
                                       Lista<string>* materias_predilectas);  
  
};
```

Escribir en un main o una clase aparte un ejemplo de uso.

Puntos 25

## Consideraciones

- No se puede utilizar ninguna clase de STL.
- El código debe estar bien modularizado
- Las variables, atributos y métodos deben tener nombres adecuados
- Es importante la eficiencia de los algoritmos

## Qué se evalúa

- Funcionalidad
- Buenas prácticas: código, modularización, comentarios, claridad
- Interfaz de usuario
- Manejo de POO – pre y poscondiciones
- Documentación
- Utilización de memoria dinámica
- Implementación de las estructuras

## Normas de entrega

Deben entregar:

- Documentación
  - Diagrama de clases UML
  - Descripción de cada TDA, indicando las pre y poscondiciones de cada una de las operaciones.
- Código fuente
  - En los archivos .h también van las pre y poscondiciones.
- Archivo makefile (todas sus rutas deben ser relativas a la carpeta del proyecto).
- Agregar a la entrega un archivo productos.txt y otro compra.txt para testeo.

**La fecha límite de entrega será el domingo 11/2 a las 23hs 55'.**