

**FRUTN - MDP - Técnico Superior en Programación**  
**Programación III - Laboratorio III**

---

1) Construir una clase **ArrayReales** que declare una lista (elija la que corresponda) como atributo y que implemente una interfaz llamada **Estadísticas**. Realizar un main para comprobar funcionamiento implementando todos los métodos.

```
public interface Estadisticas
{
    double minimo();
    double maximo();
    double sumatorio();
}
```

2) Construir una clase **Matematicas** que declare una lista (elija la que corresponda) como atributo y que implemente una interfaz llamada **Extremos**. Realizar un main para comprobar funcionamiento implementando todos los métodos.

```
public interface Extremos
{
    double inicio();
    double fin();
}
```

**NOTA:** Investigar uso de `Math.random()` para cargar las listas de manera automática al construir las clases

3) Construir una interfaz **Relaciones** que incluya los siguientes métodos

```
// Devuelve verdadero si a es mayor que b
boolean esMayor(Object b) ;

// Devuelve verdadero si a es menor que b
boolean esMenor(Object b) ;

// Devuelve verdadero si a es igual que b
boolean esIgual(Object b) ;
```

Crear una clase **Numero** que tendrá un atributo entero e implementa dicha interfaz. Diseñar un Main que declare dos o más objetos del tipo **Numero** y usar los métodos correspondientes. Mostrar salida por pantalla.

4 ) Escribe un programa para una biblioteca que contenga libros y revistas.

- Las características comunes que se almacenan tanto para las revistas como para los libros son el código, el título, y el año de publicación. Estas tres características se pasan por parámetro en el momento de crear los objetos.
- Los libros tienen además un atributo prestado. Los libros, cuando se crean, no están prestados.
- Las revistas tienen un número. En el momento de crear. En el momento de crear las revistas se pasa el número por parámetro.
- Tanto las revistas como los libros deben tener (aparte de los constructores) un método toString() que devuelve el valor de todos los atributos en una cadena de caracteres. También tienen un método que devuelve el año de publicación, y otro el código.
- Para prevenir posibles cambios en el programa se tiene que implementar una interfaz Prestable con los métodos prestar(), devolver() .
- La clase Libro implementa esta interfaz

5) Escribe una interfaz, llamada **ColeccionInterfaz**, que declare los siguientes métodos:

1. estaVacia(): devuelve true si la colección está vacía y false en caso contrario.
2. extraer(): devuelve y elimina el primer elemento de la colección.
3. primero(): devuelve el primer elemento de la colección.
4. añadir(): añade un objeto por el extremo que corresponda, y devuelve true si se ha añadido y false en caso contrario.

Escribe una clase **Pila**, que implemente esta interfaz, utilizando para ello un ArrayList de int y un contador de objetos.

6) Escribe una clase, de nombre **PruebaPila**, en la que se implementen dos métodos:

1. rellenar(): recibe por parámetro un objeto de tipo **ColeccionInterfaz**, y añade los números del 1 al 10.
2. imprimirYVaciar(): recibe por parámetro un objeto de tipo **ColeccionInterfaz** y va extrayendo e imprimiendo los datos de la colección hasta que se quede vacía.
3. Crear un main que realice las salidas por pantalla correspondientes

7) Considérese una jerarquía de barcos, todos tienen como comportamiento común msgSocorro() y alarma(); las clases BarcoPasaje, PortaAvion y Pesquero lo implementan. El BarcoPasaje nos indica cuantas camas dispone, el PortaAvion, la cantidad de aviones y su tripulación y el Pesquero, su nombre, su cantidad de pescadores y su potencia. Codifique las clases e interfaces necesarias para probar su funcionamiento