

Actividad Obligatoria 2. Paradigma Orientado a Objetos

Realizar este ejercicio como actividad autónoma del alumno si no diese tiempo a realizarla en el laboratorio de prácticas.

La actividad **deberá concluirse antes del próximo laboratorio** puesto que será utilizada.

Ejercicio 1:

Partiendo de la clase `Lista` de la práctica anterior, amplíese para implementar una lista (simplemente enlazada) polimórfica capaz de coleccionar **cualquier objeto**, implementando **al menos** los miembros:

- `Añadir`
- `Borrar`
- `Contiene`
- `ToString`
- `GetElemento`
- La propiedad `NumeroElementos`

Piense detenidamente la signatura de los métodos y si las propiedades son de lectura y/o escritura.

Valídense su funcionamiento con la herramienta de *testing* del Visual Studio. Pruebe la lista con elementos de tipo `String`, `Angulo`, `Persona`, `int` y `double`.

Utilice correctamente todos los elementos de programación aprendidos hasta ahora.

Ejercicio 2:

La realización autónoma de esta actividad es opcional pero muy aconsejable para la correcta comprensión de los conceptos del primer bloque teórico de la asignatura.

Utilizando la implementación anterior, impleméntese una nueva clase `Set` (conjunto) en un *assembly* distinto. La característica de un conjunto es que, si intentamos añadir un objeto que ya está en el conjunto, éste no se insertaría (no colecciona elementos repetidos).

Utilizando sobrecarga de operadores, implemente los métodos u operadores:

- `+` para añadir elementos
- `-` para borrar elementos
- `[]` para saber si existe un elemento en el conjunto
- `|` para unión de conjuntos
- `&` para la intersección de conjuntos
- `-` para la diferencia de conjuntos
- `^` para saber si un elemento está en el conjunto
- La propiedad `NumeroElementos`
- `ToString`

Como referencia de cómo deberían funcionar las operaciones, usar la teoría de conjuntos: http://es.wikipedia.org/wiki/Teoría_de_conjuntos