



## Tema 2 - Estructura de Datos Lineales

### Guía Docente

Un tipo abstracto de datos lineal nos permite definir secuencias de elementos (por ejemplo, la secuencia de los números primos, lista de estudiantes matriculados en un curso, etc). Los arrays (o listas de Python) ya nos permiten representar este tipo de estructuras lineales, almacenando sus elementos en posiciones contiguas de memoria. Aunque los arrays tienen varias ventajas como el acceso rápido y directo a los elementos, sin embargo sus operaciones de inserción y borrado son menos eficientes debido a que los elementos se almacenan en posiciones contiguas.

Para superar este problema, se implementan listas enlazadas que permiten almacenar los elementos de una secuencia en posiciones no contiguas, consiguiendo que las operaciones de inserción y borrado sean más eficientes.

Además, en este tema se estudian los principales tipos abstractos de datos lineales, en concreto, las pilas, colas y listas. Para cada uno de estos tipos, se presenta su especificación junto con varias implementaciones (basadas en arrays o en listas enlazadas) en Python. Para el tipo abstracto lista, se estudian dos implementaciones: listas simplemente enlazadas simples y listas doblemente enlazadas.

Durante el tema, también se plantean algunos problemas donde estos tipos de datos pueden ser útiles para su resolución. El material del tema se completa con dos hojas de problemas que permitirá al estudiantes practicar o bien extendiendo las estructuras de datos con nuevos métodos o bien utilizando dichas estructuras en la resolución de problemas.

Al final del tema, los estudiantes deberían ser capaces de:

1. Comprender el principio LIFO (last-in, first-out) en el que se basa las pilas.
2. Ser capaz de definir y explicar el tipo abstracto de datos de Pila y sus operaciones.
3. Implementar una pila usando una lista de Python.
4. Implementar una pila usando una lista enlazada.

5. Resolver problemas como invertir una secuencia de caracteres o comprobar si los paréntesis de una expresión aritmética o lógica están correctamente balanceados, mediante el uso de pilas.
6. Comprender el principio FIFO (first-in, first-out) en el que se basa las colas.
7. Ser capaz de definir y explicar el tipo abstracto de datos de Cola y sus operaciones.
8. Implementar una cola usando una lista de Python.
9. Implementar una cola usando una lista enlazada.
10. Resolver problemas como el problema de Josephus usando una cola.
11. Ser capaz de definir y explicar el tipo abstracto de datos de Lista y sus operaciones.
12. Implementar una lista utilizando una lista simplemente enlazada.
13. Comprender las ventajas y desventajas de las implementación basada en array y basada en lista enlazada.
14. Implementar una lista utilizando una lista simplemente enlazada.
15. Comprender las ventajas de la implementación de lista doble frente a la de lista simple.
16. Ser capaces de implementar nuevas operaciones en las estructuras de datos estudiadas.
17. Ser capaces de extender y combinar las diferentes estructuras lineales para proponer nuevas estructuras de datos.
18. Resolver problemas utilizando listas.