

## Ejercicios tema 3. Estructuras básicas y representaciones lineales.

1. Implementar los siguientes TADs con los métodos descritos en los apuntes del tema 3 de la asignatura. Implementar cada TAD en dos versiones: utilizando un array como estructura de datos y utilizando una lista enlazada como estructura de datos.
  - a. TAD Cola
  - b. TAD Cola con Prioridad
  - c. TAD Lista
  - d. TAD Conjunto
  - e. TAD Multiconjunto
  - f. TAD Diccionario
2. Añadir un método llamado **obtenerPrioridad** al TAD Cola con Prioridad. Este método recibirá como parámetro un número correspondiente a una prioridad y devolverá una cola con todos los elementos de la cola original que tengan la prioridad igual a la recibida como parámetro por el método.
3. Implementar un subalgoritmo **evaluarSufija** que reciba como parámetro un array de cadenas que represente una expresión en forma sufija. Este subalgoritmo deberá utilizar una pila para evaluar la expresión y devolver el valor. La expresión solo contendrá los operadores binarios '+', '-', '\*', y '/'. Por ejemplo, para el array ['3', '10', '-', '4', '\*'] el programa devolverá el valor -28 correspondiente a evaluar la expresión  $(3-10)*4$ .
4. Añadir los siguientes métodos al TAD Lista:
  - a. Método **contiene** que recibe como parámetro un valor y devuelve verdadero o falso dependiendo de si la lista contiene o no un elemento igual a ese valor.
  - b. Método **eliminarValor** que recibe como parámetro un valor y elimina la primera aparición de ese valor en la lista.
  - c. Método **eliminarTodo** que recibe como parámetro un valor y elimina todas las apariciones de ese valor en la lista.
  - d. Método **invertir** que invierte el orden de los elementos en la lista.
6. Añadir al TAD conjunto el método **diferenciaSimétrica** que recibe como parámetro un conjunto y devuelve la diferencia simétrica entre los elementos del conjunto sobre el que se llama al método y el conjunto recibido como parámetro.
7. Añadir al TAD Multiconjunto un método **toConjunto** que devuelve el conjunto de los elementos en el multiconjunto sin considerar la multiplicidad de cada uno.
8. Implementar el TAD Pila utilizando el TAD Lista como estructura de datos para almacenar los datos de la pila.
9. Implementar el TAD Conjunto utilizando el TAD Lista como estructura de datos para almacenar los elementos del conjunto.
10. Implementar el TAD Diccionario utilizando el TAD Lista como estructura de datos para almacenar los pares claves, valor del diccionario.

11. Implementar el TAD Lista ordenada. Este TAD tiene las mismas operaciones que el TAD lista para consultar los valores o el tamaño de la lista. Sin embargo, solo tiene una operación para almacenar un valor en la lista. Esta operación colocará el nuevo valor en una posición de tal forma que la lista permanezca ordenada.

12. Implementar el TAD Matriz. El TAD Matriz tiene las mismas operaciones que el TAD Lista pero, en lugar de un solo valor de la posición, las operaciones reciben una coordenada (dos valores).