

LISTAS

Opción Múltiple

1. Se define una clase Lista de Enteros de la siguiente manera:

```
public class Lista {  
    private Nodo primero;  
    // Métodos  
}
```

Donde la clase Nodo contiene un entero y una referencia al nodo siguiente. Si cada entero ocupa 4 bytes y cada referencia ocupa 8 bytes, entonces, un objeto de la clase Lista al que se le ingresaron 3 enteros ocupará:

- a. Ninguna de las otras es correcta
- b. 8 bytes
- c. 44 bytes
- d. 36 bytes

Verdadero / Falso

1. El TDA Lista puede implementarse en un vector.
Verdadero
Falso
2. Cuando se define una clase Pila, con nodos simplemente enlazados, conviene tener como atributo un puntero al primer nodo y al último.
Verdadero
Falso

Desarrollo: Implementación

1. Explicá cómo sería la implementación del alta en una lista circular de máximo N nodos. Podés escribir en la caja que está debajo o subir un archivo.
2. Dada una lista circular de N nodos, en donde se mantienen los últimos N datos recibidos, escribí el método

```
public void alta(T dato);
```

Tené en cuenta que, si hay menos de N nodos, el alta debe ir al final del último nodo.

3. Dada una lista doblemente enlazada implementá el método

```
public void alta(T dato);
```

Coloca al dato en la lista de forma de que la lista quede ordenada.

Asumí que el resto de los métodos básicos ya están implementados y que T implementa

Comparable, es decir, tiene un método `compareTo(Dato d)` que devuelve

- Un negativo si el dato es menor al que se pasa por parámetro,
- Un positivo, si es mayor o
- 0 si son iguales.

4. Dada una lista doblemente enlazada implementá el método

```
public void baja(T dato);
```

Elimina el dato de la lista junto al nodo en el que se encuentra.

Asumí que el resto de los métodos básicos ya están implementados y que T implementa

Comparable, es decir, tiene un método `compareTo(Dato d)` que devuelve

- Un negativo si el dato es menor al que se pasa por parámetro,
- Un positivo, si es mayor o
- 0 si son iguales.

5. Estás implementando la clase Lista. Esta clase ya tiene implementados los métodos básicos: alta, baja, consulta.

Tenés que agregar (implementar) el siguiente método:

```
/*  
PRE: agregar  
POST: modifica la lista que llama al método: le une lista al final.  
*/  
public void unir(Lista lista);
```

Nota 1: si necesitás agregar algún otro método que no sea básico, podés hacerlo.

Nota 2: agregá las precondiciones

6. Estás implementando la clase Lista. Esta clase ya tiene implementados los métodos básicos: alta, baja, consulta.

Tenés que agregar (implementar) el siguiente método:

```
/*  
PRE: agregar  
POST: crea una nueva Lista formada por la unión de la que llama al método  
con lista (parámetro), que va al final. Devuelve una referencia a la nueva  
lista.  
*/  
public Lista unir(Lista lista);
```

Nota 1: si necesitás agregar algún otro método que no sea básico podés hacerlo.

Nota 2: agregá las precondiciones

DESARROLLO: USO DE LISTAS

1. La clase Lista ya tiene implementados los métodos básicos: alta, baja, consulta, y tiene una referencia actual con tres métodos: reiniciar(), haySiguiente(), siguiente().

Considerar como implementada la clase Alimento a partir de la siguiente interfaz:

```
public class Alimento {
    /* Crea un alimento con su nombre, la cantidad de calorías y una lista de
    los ingredientes que lo conforman */
    public Alimento (String nombre, int calorías, Lista<String> ingredientes);

    /* devuelve el nombre del alimento */
    public String obtenerNombre();

    /* devuelve la cantidad de calorías */
    public int obtenerCalorias ();

    /* devuelve una referencia a la lista de ingredientes */
    public Lista<string> obtenerIngredientes ();
};
```

Implementar el método comidasParaCeliacos de la clase BuscadorComidas, según:

```
public class BuscadorComidas {
    /*
    Post: busca en "comidas" aquellas que tienen algún ingrediente de la lista
    "ingredientesPermitidos" y ninguno de la lista "ingredientesNoPermitidos" y
    que tengan una caloría menor a "caloríaMaxima".
    Devuelve una referencia a una lista con los alimentos que cumplen con estas
    características.
    */
    public Lista<Alimento> comidasParaCeliacos (
        Lista<Alimento> comidas,
        Lista<String> ingredientesPermitidos,
        Lista<String> ingredientesNoPermitidos,
        int caloríaMaxima);
};
```