

Objetivo:

Los objetivos de este ejercicio son:

1. Trabajar con las referencias
2. Tomar contacto con un nuevo tipo de nodo con dos referencias
3. Trabajar la inserción y gestión de una cadena ordenada

La clase NodeD:

Se define un nuevo tipo de nodo que consta de dos referencias, una para apuntar al siguiente nodo y otro al anterior o predecesor. Esta clase, así como sus métodos tiene visibilidad de paquete, es decir sólo puede verse y usarse dentro del paquete en el que se ha definido. Esta clase sólo tiene sentido dentro de la implementación de una cadena doblemente enlazada

Los servicios que proporciona son:

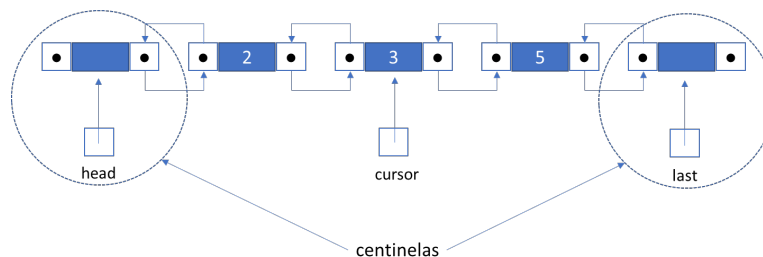
- Constructores: Dispone de dos constructores, uno que solo recibe el dato que albergará y otro que recibe el dato y las referencias siguiente y anterior
- getDato y setDato: Método para obtener y establecer el dato asociado al nodo
- getNext y setNext: métodos para obtener y establecer la referencia al nodo siguiente
- getPrev y setPrev: métodos para obtener y establecer la referencia al nodo anterior

La clase SortedDoubleLinkedList:

Esta clase define la cadena doblemente enlazada y ordenada. Esta clase genérica requiere de **dos parámetros para poder ser usada**. Un parámetro es **el tipo** (como los vistos hasta ahora) y otro es **una función de comparación** (tipo que extiende la interfaz Comparator).

Para la implementación:

- Se va a ver una implementación de este TAD usando una **cadena doblemente enlazada con centinelas**. El centinela es un **nodo sin información** que está en un extremo de la cadena. Esto hace que todas las operaciones se puedan considerar como **inserciones o borrados en el centro** de la cadena, simplificando la implementación.
- Está formada por una cadena de nodos. Estos nodos constan de dos referencias, una al nodo anterior y otra al siguiente. Esto permite avanzar y retroceder por la lista.
- Se tendrán dos referencias para poder acceder por el principio (head) y por el final (last).
- Habrá una tercera referencia (cursor) que se utilizará como un cursor interno para recorrer la lista o realizar inserciones respecto a la posición de ese cursor.



La información sobre los métodos se encuentra en la documentación y en el código relacionado con este enunciado