

Concatenar dos listas doblemente enlazadas circulares

Estructuras de Datos
Facultad de Informática - UCM

Supongamos una implementación del TAD Lista mediante listas doblemente enlazadas circulares con nodo fantasma:

```
class ListLinkedDouble {  
    // ...  
private:  
    struct Node { int value; Node *next; Node *prev; };  
    Node *head;  
    int num_elems;  
}
```

Escribe un método `append` con la siguiente especificación:

```
{  l1 = [x0, ..., xn-1], l2 = [y0, ..., ym-1]  }  
l1.append(l2);  
{  l1 = [x0, ..., xn-1, y0, ..., ym-1], l2 = []  }
```

Es decir, `append` debe añadir los elementos de la lista recibida como parámetro en la lista `this`. La lista recibida como parámetro debe quedar vacía al final de la llamada a `append`.

Importante: El coste del método `append` debe de ser $O(1)$.

Solución

```
void append(ListLinkedDouble &l2) {  
    if (!l2.empty()) {  
        head->prev->next = l2.head->next;  
        l2.head->next->prev = head->prev;  
        head->prev = l2.head->prev;  
        l2.head->prev->next = head;  
  
        num_elems += l2.num_elems;  
  
        // Dejamos la lista l2 vacia  
        l2.head->next = l2.head;  
        l2.head->prev = l2.head;  
        l2.num_elems = 0;  
    }  
}
```