

Intercambiar de dos en dos los elementos de una lista doblemente enlazada

En este ejercicio partimos de la clase `ListLinkedDouble`, que implementa el TAD lista mediante listas doblemente enlazadas circulares con nodo fantasma. Queremos añadir un nuevo método, llamado `swap2by2()`:

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

public:
    ...
    void swap2by2();
};
```

Por ejemplo, si la lista fuera $xs = [1, 2, 3, 4, 5, 6]$ el resultado de `xs.swap2by2()` sería $[2, 1, 4, 3, 6, 5]$. Esta es la especificación informal si la longitud de la lista es par

```
{ lista =  $[x_0, x_1, \dots, x_{n-1}]$ ,  $n$  par }
lista.swap2by2();
{ lista =  $[x_1, x_0, x_3, x_2, \dots, x_{n-1}, x_{n-2}]$  }
```

y si la longitud es impar el último elemento se queda en la misma posición

```
{ lista =  $[x_0, x_1, \dots, x_{n-1}]$ ,  $n$  impar }
lista.swap2by2();
{ lista =  $[x_1, x_0, x_3, x_2, \dots, x_{n-2}, x_{n-3}, x_{n-1}]$  }
```

Se pide:

1. Implementar el método `swap2by2()`.
2. Indicar su coste con respecto al tamaño de la lista de entrada.

Importante: Para la implementación del método no pueden crearse, directa o indirectamente, nuevos nodos mediante `new` ni borrar nodos mediante `delete`; han de reutilizarse los nodos de las listas de entrada. Tampoco se permite copiar valores de un nodo a otro. El coste de la operación ha de ser lineal con respecto a la longitud de la lista.

Entrada

La entrada consta de una serie de casos de prueba. Cada caso se muestra en dos líneas. La primera contiene el número N de elementos de la lista. En la segunda se muestran esos N elementos. La entrada termina con un 0, que no debe ser procesado.

Salida

Para cada caso de prueba se escribirá en una línea la lista modificada tras intercambiar los elementos de dos en dos.

Entrada de ejemplo

```
1
6
2
2 4
6
2 4 3 6 7 1
5
3 7 2 5 3
0
```

Salida de ejemplo

```
[6]
[4, 2]
[4, 2, 6, 3, 1, 7]
[7, 3, 5, 2, 3]
```

Autores

Isabel Pita y Alberto Verdejo