

Mezclar listas enlazadas ordenadas

En este ejercicio se trata de practicar las listas enlazadas dobles (utilizaremos las vistas en clase para implementar el TAD `deque`, aunque la nueva operación no sea una operación de colas dobles). Queremos una operación que inserte en una lista enlazada ordenada los elementos de otra lista enlazada ordenada recibida como argumento, de tal forma que la lista resultante quede también ordenada. La lista recibida como argumento pasará a ser vacía.

Requisitos de implementación.

En la resolución del problema, se extenderá *mediante herencia* la clase `deque` con un método que implemente esta operación de mezcla ordenada.

El coste de la operación debe ser lineal con respecto a la suma del número de elementos en las listas que se mezclan. No pueden hacerse nuevos `news`; deben reutilizarse los nodos que ya existen en las dos listas enlazadas.

También se añadirá un método que permita mostrar el contenido de la lista en una línea, separando sus elementos por espacios.

Entrada

La entrada consta de una serie de casos de prueba. La primera línea contiene el número de casos de prueba que vendrán a continuación. Cada caso ocupa dos líneas. Cada una de estas líneas representa una de las listas, y contiene sus elementos ordenados de menor a mayor, una serie de números entre 1 y 1.000.000, seguidos de un 0, que marca el final de la descripción de la lista, sin pertenecer a ella.

Salida

Para cada caso de prueba se escribirá en una línea la lista modificada tras mezclar de forma ordenada los elementos en ambas listas.

Entrada de ejemplo

```
6
1 3 5 7 0
2 4 6 8 0
2 4 6 8 0
1 3 5 7 0
1 2 3 0
6 7 8 0
1 10 20 0
2 6 8 12 20 22 0
0
1 2 3 0
1 1 2 2 0
0
```

Salida de ejemplo

```
1 2 3 4 5 6 7 8
1 2 3 4 5 6 7 8
1 2 3 6 7 8
1 2 6 8 10 12 20 20 22
1 2 3
1 1 2 2
```

Autor: Alberto Verdejo