

# Problema B

Entremezclar dos listas consiste en formar otra tomando alternativamente elementos de cada una de esas listas. Por ejemplo, si entremezclamos la lista 1 3 5 7 con la lista 2 4 6 8, obtenemos la lista 1 2 3 4 5 6 7 8. Cuando una de las dos listas se queda vacía, se cogen todos los elementos de la otra.

Queremos extender *mediante herencia* la clase `deque<int>`, que implementa las colas dobles de enteros mediante listas de nodos dinámicos, doblemente enlazados, circular y con nodo fantasma, con una nueva operación **entremezclar** que mezcle una lista enlazada doble (la del objeto *this*) con otra (recibida como argumento), comenzando a coger elementos de la primera lista. La lista recibida como argumento pasará a ser vacía.

En la implementación del nuevo método no pueden hacerse nuevos **news**, ni *copiar* los enteros de un nodo a otro. También hará falta un método **print** para mostrar por pantalla los elementos de la lista, de inicio a fin.

No modifiques ni subas al juez el fichero `deque_eda.h` cuya clase `deque` debes extender.

## Entrada

La entrada consta de una serie de casos de prueba. En la primera línea aparece el número de casos de prueba que vendrán a continuación.

Cada caso se muestra en cuatro líneas. La primera contiene el número  $N$  de elementos de la lista principal (un número entre 0 y 100.000). En la segunda línea se muestran esos  $N$  elementos, números entre 1 y 1.000.000, separados por espacios. La tercera línea contiene el número  $M$  de elementos de la segunda lista (un número entre 0 y 100.000) y la cuarta línea contiene esos  $M$  elementos, números entre 1 y 1.000.000.

## Salida

Para cada caso de prueba se escribirá una línea que contendrá los elementos de la lista modificada tras entremezclarla con los elementos de la segunda lista.

## Entrada de ejemplo

```
3
4
1 3 5 7
4
2 4 6 8
3
7 3 9
0

2
2 1
4
5 3 4 6
```

## Salida de ejemplo

```
1 2 3 4 5 6 7 8
7 3 9
2 5 1 3 4 6
```