

Eliminar todas las apariciones de un elemento en una lista

Se pide implementar una función (externa al TAD) que dada una lista (TAD *List*) elimine todas las apariciones de un elemento (identificado a partir del operador `==` del tipo con el que se instancie la lista). La función no debe crear nueva memoria, es decir, no debe llamar a los métodos *push_back* ni *push_front*. Si el elemento no se encuentra en la lista la operación simplemente no tendrá efecto.

Entrada

Primero se solicitará el número de casos de prueba a procesar. Cada caso consistirá en dos líneas. La primera contiene una serie de números positivos separados por espacios, que se van introduciendo uno a uno en la lista (llamando a `push_back`), y acabada en `-1` (que no se introduce en la lista). La segunda línea solo contiene el número a eliminar.

Salida

Para cada caso de prueba se imprimirá la lista original introducida y después la lista resultante tras invocar a la operación.

Entrada de ejemplo

```
4
1 2 1 2 1 -1
2
1 2 1 2 1 -1
1
1 1 1 1 -1
1
1 2 3 4 -1
5
```

Salida de ejemplo

```
1 2 1 2 1
1 1 1
1 2 1 2 1
2 2
1 1 1 1

1 2 3 4
1 2 3 4
```