
Listas iguales circularmente**V55523_es**

Dadas dos listas de enteros, queremos saber si una es igual circularmente a la otra. Es decir, si se puede obtener una lista a partir de la otra rotando sus elementos.

Por ejemplo: $l_1 = [1, 2, 3]$ es igual circularmente a $l_2 = [3, 1, 2]$, porque al colocar el 3 de l_2 al final, obtenemos l_1 . Otro ejemplo: $[2, 7, 1, 0]$ no es igual a $[0, 1, 7, 2]$. Un caso especial es que consideraremos que dos listas vacías son iguales circularmente.

El ejercicio es implementar la función siguiente:

```
/**
 * @pre  cierto
 * @post se devuelve cierto si l1 y l2 son iguales circularmente,
 *       falso en caso contrario.
 */
bool circularly_equal(const list<int>& l1, const list<int>& l2)
```

Solo hay que enviar esta función.

Ejemplo de entrada

```
# iguales
1 2 3
1 2 3
# rotación 1
1 2 3
3 1 2
# diferentes reverso
1 2 3
3 2 1
# diferentes
1 2 1 2
1 1 2 1
# rotación 4
2 3 4 5 1
5 1 2 3 4
```

Ejemplo de salida

```
Iguals
Iguals
Diferentes
Diferentes
Iguals
```

Observación

Enviad un fichero solo con la función; si lo enviáis con el `main`, da CE.

Información del problema

Autoría: PRO2

Generación: 2026-03-25T11:54:10.454Z

© Jutge.org, 2006–2026.

<https://jutge.org>