
Intersecció de llistes ordenades

T58713_ca

Fes la funció següent, que calcula la intersecció de dues llistes de persones en **ordre ascendent** de DNI:

```
/**
 * @brief Calcula la intersecció de dues llistes ordenades.
 *
 * @param a Llista en ordre ascendent de DNI.
 * @param b Llista en ordre ascendent de DNI.
 *
 * @return Llista en ordre ascendent de DNI amb les persones comunes d'a i b.
 */
list<Persona> interseccio_ordenada(const list<Persona>& a, const list<Persona>& b)
```

La intersecció conté les persones que apareixen a les dues llistes alhora. Dues persones es consideren iguals si tenen el mateix DNI.

Les llistes estan ordenades ascendentment per DNI (comparació lexicogràfica de strings).

La declaració de la classe `Persona` és la següent:

```
class Persona {
public:
    Persona(string dni, string nom); // Construeix una persona
    string dni() const; // retorna el DNI
    string nom() const; // retorna el nom
};
```

Observació

Has d'implementar la funció `interseccio_ordenada` en el fitxer `interseccio.cc`, compilar (està preparat per poder compilar i debuggar amb VSCode), i finalment **enviar només el fitxer `interseccio.cc`**.

Cal que la funció sigui eficient i que no visiti elements de forma innecessària.

Els fitxers públics (icona del gatet) contenen:

<code>main.cc</code>	el programa principal, amb la entrada/sortida feta
<code>persona.hh</code>	la classe <code>Persona</code> , amb els mètodes esmentats
<code>Makefile</code>	per compilar amb <code>make</code> còmodament
<code>.vscode</code>	carpeta per compilar i debuggar amb VSCode

Entrada

La funció rep dues llistes de persones no modificables, ordenades ascendentment per DNI. (De la lectura de les llistes, que estan separades per una línia buida, ja se n'encarrega el programa principal.)

Sortida

La funció retorna la llista amb les persones comunes de les dues llistes, preservant l'ordre ascendent original. (De l'escriptura de la llista resultant també se n'encarrega el programa principal.)

Exemple d'entrada

```
12345678Z Anna
34567890K Clara
45678901X David
56789012T Elena
78901234B Gerard
```

```
23456789D Bernat
34567890K Clara
56789012T Elena
67890123L Ferran
89012345C Helena
```

Exemple de sortida

```
34567890K Clara
56789012T Elena
```

Informació del problema

Autoria: Pau Fernández

Generació: 2026-02-25T09:47:34.108Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>