

## Tema 2 – Exercici recuperació 1

Volem crear una classe `Llibre` per guardar les dades dels llibres que es guarden en una biblioteca. De cada llibre hem de guardar el títol, l'autor i nº d'exemplars que hi ha a la biblioteca. De cada exemplar del llibre volem saber el seu codi i si està prestat o no. Si està prestat haurem de poder guardar el nom de la persona que ha fet el préstec i la data en què ho ha fet. Suposarem que els exemplars del llibre s'identifiquen amb un codi numèric des de 0 fins al nº total d'exemplars del llibre menys 1. Les dades dels exemplars s'hauran de guardar utilitzant un array dinàmic.

Per poder guardar les dades dels exemplars d'un llibre i dels préstecs dels exemplars, tenim creades aquestes dues classes `Exemplar` i `Prestec`:

```
class Prestec
{
public:
    Prestec(const string& nom, const string& dataPrestec)
        : m_nom(nom), m_dataPrestec(dataPrestec) {}
    string getNom() const { return m_nom; }
    string getDataPrestec() const { return m_dataPrestec; }
private:
    string m_nom;
    string m_dataPrestec;
};
```

```
class Exemplar
{
public:
    ...

private:
    int m_codi;
    Prestec* m_prestec;
};
```

A la classe `Exemplar` l'apuntador a `Prestec` serà `nullptr` si l'exemplar no està prestat i, si està prestat, guardarà l'adreça a un objecte de la classe `Prestec` que guarda les dades del préstec. Afegiu a la classe `Exemplar` tots els mètodes públics que us facin falta. Afegiu també un constructor de còpia i un operador d'assignació.

## Tema 2 – Exercici recuperació 1

A nivell d'interfície pública la classe `Llibre` ha de tenir aquests mètodes:

- Un constructor per defecte.
- Un constructor que rebi com a paràmetre el títol, l'autor i el nº d'exemplars del llibre.
- Un constructor de còpia.
- L'operador d'assignació.
- Getters i setters del títol i de l'autor del llibre.
- Un mètode per recuperar el nº d'exemplars del llibre.
- Un mètode per recuperar l'apuntador al préstec d'un exemplar (identificat pel codi).
- Un mètode per poder registrar que es fa el préstec d'un exemplar del llibre. Els préstecs sempre es fan d'un exemplar concret del llibre, que serà sempre l'exemplar amb codi més petit d'entre tots els exemplars que no estan prestats. Si tots els exemplars del llibre ja estan en préstec no es pot fer cap més préstec del llibre fins que es retorni algun dels exemplars. Aquest mètode rep com a paràmetre el nom de la persona que fa el préstec i la data en què es fa.
- Un mètode per poder registrar que es fa el retorn d'un dels exemplars del llibre que estan en préstec (identificat a partir del seu codi). Si un exemplar no està en préstec, no s'ha de poder fer el retorn d'aquell exemplar.

## Tema 2 – Exercici recuperació 1

Utilitzant la classe `Llibre` **declareu i implementeu una classe `Biblioteca`** que serveixi per guardar i gestionar tots els llibres de la biblioteca. Utilitzeu la classe `vector` per guardar tots els llibres de la biblioteca. Aquesta classe haurà de tenir la següent interfície pública:

- Un constructor per defecte
- Un mètode `afegeixLlibre` que permeti afegir un nou llibre a la biblioteca. Ha de tenir com a paràmetres un objecte de la classe `Llibre`.
- Un mètode `prestaLlibre`, que rebi com a paràmetre el títol d'un llibre i faci el préstec del llibre cridant al mètode corresponent de la classe `Llibre`. S'ha de retornar el codi de l'exemplar que s'ha prestat utilitzant un paràmetre per referència. Com a valor de retorn, ha de retornar 0 si el llibre existeix i s'ha pogut fer el préstec, -1 si el llibre existeix però no s'ha pogut fer el préstec (perquè tots els exemplars ja estan prestats), i -2 si el llibre no existeix.
- Un mètode `retornaLlibre`, que rebi com a paràmetres el títol d'un llibre i el codi de l'exemplar que es vol retornar i registri a la biblioteca el retorn de l'exemplar del llibre cridant al mètode corresponent de la classe `Llibre`. Ha de retornar 0 si el llibre existeix i s'ha pogut retornar, -1 si el llibre existeix però no s'ha pogut retornar i -2 si el llibre no existeix.