

1-

Implementar la clase Pieza en los ficheros pieza.cc y pieza.h. Una pieza tiene una cadena nombre, un entero código, entero cantidad y double precio.

- Constructor vacío con los valores por defecto nombre(“Sin Nombre”) y el resto a cero.
- Constructor parametrizado cuyos únicos parámetros obligatorios son el código y el nombre (en ese orden) el resto son opcionales tomando como valores por defecto cero.
- Constructor de copia.
- Observadores y modificadores.

Nota: el observador de la variable nombre se debe implementar por referencia.

- Método getPieza. Dicho método retorna una cadena, por ejemplo: “Hay X unidades de la pieza Y.”

Siendo X la cantidad e Y el nombre de la pieza

2-

Codificar la clase Refpieza, en los ficheros, refpieza.h y refpieza.cc, que deriva de la clase Pieza. Esta clase tiene además un código de referencia de tipo string.

- Constructor parametrizado cuyos únicos parámetros, todos obligatorios, son el código de referencia, el código de pieza, y el nombre de pieza, en dicho orden.
- Constructor de copia.
- Observadores y modificadores.

3-

Implementar la clase CPieza que dispone de una lista de objetos de tipo Pieza.

- Constructor vacío. Debe asegurarse que la lista esta vacía.

- Constructor de copia.
- Observador `get()` que retorna la lista.
- Método `size()` de la clase `CPieza`, que retorna el numero de piezas de la lista.
- Método `addPieza`. Recibe un objeto `Pieza` y lo inserta en la lista si este no existe ya. Para buscar la pieza se utiliza el campo código. Retorna `true` si la pieza se ha insertado correctamente y `false` en caso contrario.
- Método `deletePieza`. Recibe un objeto `Pieza` y lo elimina de la lista. Si se ha eliminado el método retorna `true` y `false` en caso contrario. Para buscar la pieza se utiliza el campo código, como en el apartado anterior.

4-

Ejercicio de Mapas

5-

Copie los ficheros del ejercicio anterior con el nombre `cpieza2.cc` y `cpieza2.h`.

- Operador `[]`. Este recibe un entero y retorna la pieza que se encuentra en la posición de la lista que dice el entero.
- Operador `=`. Efectúa la asignación de un objeto `CPieza` en otro.