Ejercicios de práctica de programación orientada a objetos en Arduino:

Ejemplo:

Enlace proyecto de ejemplo:

https://www.tinkercad.com/things/4eTZJAPKYxB

https://www.tinkercad.com/things/dfZESWkw7gU

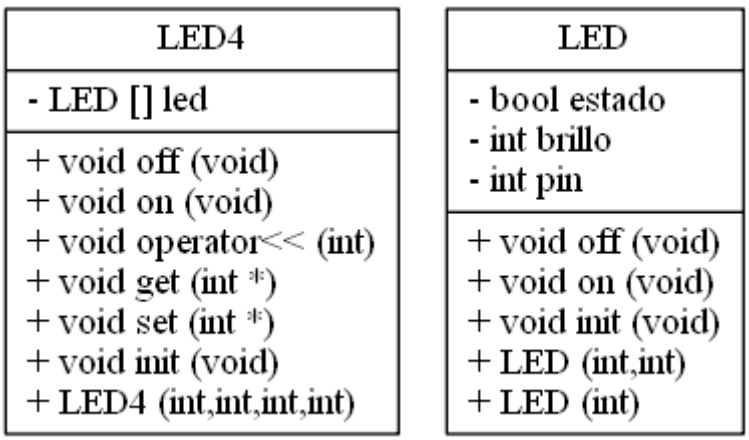
https://www.tinkercad.com/things/5EkCakxaZlW

Clase LED: permite controlar un led, detalle de miembros:

- Brillo: define el brillo que debe tener el led al ser encendido (0 a 100)
- Pin: el numero de pin en el que se conecta el led, solo algunos nros son validos
- Estado: indica si el led esta encendido o no
- Off(): apaga el led
- On(): enciende el led, si esta conectado a una salida PWM podrá graduar el brillo
- Init(): inicializa el pin como salida
- Contructores: reciben en nro de pin y valor inicial de brillo

Clase LED4: permite controlar un grupo de 4 leds como una barra lineal. Detalle de miembros:

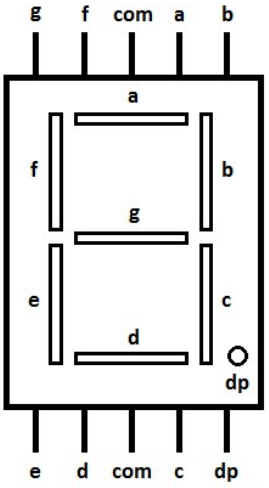
- Led: arreglo de 4 objetos LED
- Contructor: recibe los 4 pines donde se conectan los leds
- On() y off(): permiten encender y apagar los leds, respetando el valor de brillo de cada uno
- Get() y set(): permiten establecer y obtener el valor individual de brillo de cada led en un arreglo de 4 enteros que podrán tomar valor 0 o 1
- Operator<<: permite realizar un desplazamiento a izquierda del valor de los leds para generar un efecto visual



Ejercitación de práctica:

1) Implementar una clase que permita controlar un display de led de 7 segmentos. Al igual que la clase LED4, podrá tener un arreglo de 8 LEDs. Recordar que en un display

de 7 segmentos cada sección del número es un led individual distribuido como indica la figura. Además existe un 8º led que permite indicar el punto decimal.



LED7S
- LED [] led
+ void dpOff () + void dpOn (void) + void off (void) + void on (void) + void operator<< (int) + void operator+ (int &) + void operator= (int &) + void init (void)

Detalle de los métodos solicitados:

- dpOn y dpOff: prende y apaga el punto decimal
- on y off: prende y apaga el display mostrando el dígito correspondiente
- operator<<: permite asignar un valor entero (hezadecimal entre 0 y F) al display
- operator+: permite sumar un valor entero al valor actual del display. Si el resultado de la suma supera F, muestra F, si es menor a 0, muestra 0. Recordar que se puede sumar un numero negativo.
- Operator= permite asignarle un valor entero, funciona similar al operator<<

2) Implementar una clase DCMOTOR para controlar un motor de dc. Ajustar la interface de cada método según corresponda. Recordar que la velocidad del motor se controla igual que el brillo del led mediante el método AnalogWrite().

DCMOTOR
- int rpm
+ void getRPM () + void setRPM () + void off () + void on () + DCMOTOR ()