

## Laboratorio M3-13

Realizar los siguientes ejercicios en tu Editor de Código preferida, adjuntar tu archivo py.

Enviar laboratorio a: [jpruiz@itgcorp.co](mailto:jpruiz@itgcorp.co)

*Recuerda adjuntar el número de grupo al que perteneces en el asunto del email*

### Ejercicio No.13 - La clase Timer

#### Tiempo Estimado

30 - 60 minutos

#### Nivel de Dificultad

Fácil / Medio

#### Objetivos

- Mejorar las habilidades del estudiante para definir clases desde cero.
- Definir y usar variables de instancia.
- Definir y usar métodos.

#### Escenario

Necesitamos una clase capaz de contar segundos. ¿Fácil? No es tan fácil como podrías pensar, ya que tendremos algunos requisitos específicos.

Léelos con atención, ya que la clase sobre la que escribes se utilizará para lanzar cohetes en misiones internacionales a Marte. Es una gran responsabilidad. ¡Contamos contigo!

Tu clase se llamará **Timer** (temporizador en español). Su constructor acepta tres argumentos que representan **horas** (un valor del rango [0..23]; usaremos tiempo militar), minutos (del rango [0..59]) y **segundos** (del rango [0..59]).

Cero es el valor predeterminado para todos los parámetros anteriores. No es necesario realizar ninguna comprobación de validación.

La clase en sí debería proporcionar las siguientes facilidades:

- Los objetos de la clase deben ser "imprimibles", es decir, deben poder convertirse implícitamente en cadenas de la siguiente forma: "hh:mm:ss", con ceros a la izquierda agregados cuando cualquiera de los valores es menor que 10.
- La clase debe estar equipada con métodos sin parámetros llamados **next\_second()** y **previous\_second()**, incrementando el tiempo almacenado dentro de los objetos en +1/-1 segundos respectivamente.

Emplea las siguientes sugerencias:

- Todas las propiedades del objeto deben ser privadas.
- Considera escribir una función separada (¡no un método!) para formatear la cadena con el tiempo.

Completa la plantilla que te proporcionamos en el editor. Ejecuta tu código y comprueba si el resultado es el mismo que el nuestro.

## Salida Esperada

23:59:59

00:00:00

23:59:59 Cola vacía

## Código base

```
1 class Timer:
2     def __init__( ??? ):
3         #
4         # Escribir código aquí.
5         #
6
7     def __str__(self):
8         #
9         # Escribir código aquí.
10        #
11
12    def next_second(self):
13        #
14        # Escribir código aquí.
15        #
16
17    def prev_second(self):
18        #
19        # Escribir código aquí.
20        #
21
22
23 timer = Timer(23, 59, 59)
24 print(timer)
25 timer.next_second()
26 print(timer)
27 timer.prev_second()
28 print(timer)
```