

## TALLER 1 PROBLEMA 3

### Métodos Computacionales 2

Para el ejercicio pruebe que corra en el Binder del curso.

En la cafetería de su trabajo se encuentra un microondas (clase **Microwave**), el cual se puede controlar a partir de un control remoto (clase **RemoteControl**). El control remoto tiene los siguientes comandos:

- `void set_time(int time_sec_param, int time_min_param)`, esta función recibe en primer lugar un `int` de tiempo en segundos, y un `int` de tiempo en minutos, lo cual inicializa el tiempo en el microwave.
- `void add_time_sec(int add_time_sec_param)`, esta función recibe un `int` de tiempo en segundos que quiere agregar al tiempo del microwave.
- `void add_time_min(int add_time_min_param)`, esta función recibe un `int` de tiempo en minutos que quiere agregar al tiempo del microwave.
- `void del_time_sec(int del_time_sec_param)`, esta función recibe un `int` de tiempo en segundos que quiere restar al tiempo microwave.
- `void del_time_min(int del_time_min_param)`, esta función recibe un `int` de tiempo en minutos que quiere restar al tiempo microwave.
- `void show_time()`, esta función muestra el tiempo actual del microwave.

Por defecto el microondas empieza en un tiempo de 0 segundos y 0 minutos. Y el tiempo no puede ser menor a 0 sec, 0 min, ni mayor a 0 seg, 90 min (no se van a probar estos casos de prueba límites).

Su tarea es crear la clase **Microwave** y la clase **RemoteControl** como se describen arriba. La clase **RemoteControl** tendrá como parámetro una instancia de la clase **Microwave** (por ejemplo, `RemoteControl(microwave_1)`, donde **Microwave microwave1**) lo cual indica que el **RemoteControl** controlara esa instancia del **Microwave**. A través del control remoto se debe poder visualizar el tiempo actual del microondas con ayuda de la función `void show_time()` y debe aparecer como se muestra a continuación:

03:25

Donde los primeros dos dígitos son los minutos y los siguientes dos dígitos son los segundos. Si el número de minutos o segundos es menor que 10 debe mostrar un cero a la izquierda.

El programa debe incluir los siguientes archivos:

- `main.cpp`: este archivo debe incluir la función `main` tal cual como se encuentra abajo.
- `microwave.h`: el header `microwave` debe tener la clase `Microwave` y declarar su constructor, sus atributos y sus métodos.
- `microwave.cpp`: este archivo debe implementar las funciones declaradas en el archivo `microwave.h`.
- `remotecontrol.h` el header `remotecontrol` debe tener la clase `RemoteControl` y declarar su constructor, sus atributos y sus métodos, en particular debe tener los siguientes métodos explicados arriba:  

```
void set_time(int time_sec_param, int time_min_param),
void add_time_sec(int add_time_sec_param),
void add_time_min(int add_time_min_param),
void del_time_sec(int del_time_sec_param),
void del_time_min(int del_time_min_param),
void show_time().
```
- `remotecontrol.cpp`: este archivo debe implementar las funciones declaradas en el archivo `remotecontrol.h`.
- `Makefile` el archivo `makefile` debe poder compilar los tres archivos `main.cpp`, `microwave.cpp` y `remotecontrol.cpp` con solo aplicar el comando `make all` en la terminal debe quedar el ejecutable `output`. Además agregue en el archivo `MakeFile` el comando `clean`: para borrar archivos de `output`.

Todos los archivos deben estar guardados en una carpeta llamada `ApellidoNombreApellidoNombre_Taller1_ParteB`, y se debe comprimir la carpeta en un archivo llamado `ApellidoNombreApellidoNombre_Taller1_ParteB.zip`.

La calificación de su ejercicio se realizara con la siguiente serie de comandos en `bind`.

```
unzip ApellidoNombreApellidoNombre_Taller1_ParteB.zip
cd ApellidoNombreApellidoNombre_Taller1_ParteB
make all
.\output
make clean
```

Copie la siguiente función `main()` en su archivo `main.cpp`, y entregue la función `main` tal como aparece,

```
int main ()
{
Microwave microwave_1;
RemoteControl rc_1 = RemoteControl(microwave_1);
rc_1.set_time(33, 5);
rc_1.del_time_sec(30);
rc_1.del_time_min(2);
```

```

rc_1.show_time();

Microwave microwave_2;
RemoteControl rc_7 = RemoteControl(microwave_2);
rc_7.set_time(0, 15);
rc_7.add_time_sec(90);
rc_7.add_time_min(12);
rc_7.show_time();

Microwave microwave_3;
RemoteControl rc_8 = RemoteControl(microwave_3);
rc_8.add_time_sec(10);
rc_8.add_time_sec(15);
rc_8.add_time_min(12);
rc_8.show_time();

Microwave microwave_4;
RemoteControl rc_6 = RemoteControl(microwave_4);
rc_6.set_time(20,9);
rc_6.del_time_sec(35);
rc_6.add_time_sec(40);
rc_6.show_time();

return 0;
}

```

El resultado esperado de poner en la terminal `make all` seguido de `.\output` debe ser:

```

03:03
28:30
12:25
09:25

```

Puede leer más sobre el problema original en el siguiente link [microwaveovens](#).

**Nota:** El programa debe ser llamado `ApellidoNombreApellidoNombre_Taller1_ParteB.zip` donde `Apellido` y `Nombre` debe reemplazarlos con el primer apellido y nombre de la pareja. El programa debe correr en binder, si el programa demora más de 10 segundos en correr se considerará incorrecto.