

## Hoja de ejercicios voluntarios

### Preparación previa

Antes de resolver esta hoja de problemas, se recomienda:

- Revisar las transparencias de la asignatura. Esta hoja revisa gran parte de los conceptos de programación en C vistos a lo largo de la asignatura

### Entrega

Los ficheros de código fuente (extensión .c) desarrollados por el alumno para resolver los ejercicios de la hoja.

- Recordad poner *comentarios* adecuados y ejecutar la *herramienta de estilo* antes de realizar la entrega
- Es obligatorio documentar todas las funciones desarrolladas

## Estadísticas de las carreras del mundial de Fórmula 1

### Descripción

La Federación Internacional del Automóvil (FIA) está valorando modificar el sistema de puntuación utilizado en las carreras del mundial de Fórmula 1. Para facilitar su trabajo, precisa de un programa que calcule una serie de estadísticas a partir de los datos obtenidos durante el mundial. Para cada una de las carreras del mundial, la FIA dispone de los siguientes datos de interés por cada piloto participante:

- El identificador del piloto
- La escudería a la que pertenece.
- La posición obtenida por el piloto en la carrera. La posición se representa como un número entero en el que el ganador se identifica mediante el número 1.
- El tiempo de la vuelta rápida obtenida por el piloto en la carrera. La vuelta rápida se expresa en segundos y se representa como un número real.

Para facilitar el acceso a estos datos, la FIA nos proporciona la librería *libformula1.a*. Entre otros elementos, esta librería define en su fichero de cabecera (*formula1.h*) una estructura de datos denominada **piloto\_t** que representa los datos de interés de un piloto en una carrera. Asimismo, también define una función para recuperar los datos de las carreras del mundial. En este caso, una carrera del mundial se identifica con un *número entero* entre 0 y MAX\_CARRERAS - 1. El fichero de cabecera con el prototipo de la función se muestra a continuación:

```
#define MAX_CARRERAS      6
#define MAX_PILOTOS      8
#define MAX_ESCUDERIAS   4
#define MAX_STRING       20

typedef enum nombre_escuderia {FERRARI, MERCEDES, RED_BULL, MCLAREN} escuderia_t;

typedef struct piloto_f1 {
    char id[MAX_STRING];           // identificador único del piloto
    escuderia_t escuderia;         // escuderia donde corre
    int posicion;                  // posicion en la carrera
    float vuelta_rapida;           // tiempo de su vuelta rapida en la carrera (en segundos)
} piloto_t;

/**
 * @brief Recupera los datos del mundial de formula 1 y los almacena en datos_mundial
 * @param datos_mundial: array bidimensional donde almacenar las estadísticas de cada piloto para cada
 * carrera del mundial. Para cada carrera, se incluyen las estadísticas de todos los pilotos
 * @returns true si se han podido recuperar los datos, false en caso de error
 */
bool recupera_datos_mundial (piloto_t datos_mundial[MAX_CARRERAS][MAX_PILOTOS]);
```

Se pide implementar un programa **f1.c** que implemente las siguientes funcionalidades:

- Una función denominada **obten\_ganador\_carrera** que retorne los datos del piloto ganador de la carrera indicada.
- Una función denominada **obten\_vuelta\_rapida\_carrera** que retorne los datos del piloto que ha realizado la vuelta rápida en la carrera indicada.

## Introducción al software, Curso 2019-2020

- Una función denominada ***numero\_carreras\_perfectas*** que retorne el número de *carreras perfectas* obtenidas en el mundial por la escudería indicada. Una *carrera perfecta* se define como aquella en la que coinciden la escudería del piloto ganador de la carrera y la del piloto que ha conseguido la vuelta rápida.

Además de implementar las funcionalidades requeridas, se requiere implementar una función *main* que compruebe que las funcionalidades desarrolladas son correctas. La función *main* deberá solicitar por pantalla el número de carrera que se quiere consultar, así como gestionar los errores que el usuario pueda cometer al introducir esta información (*nota: podemos asumir que el usuario siempre va a introducir un número*). Por sencillez, no es necesario solicitar al usuario que introduzca la escudería, sino que ésta puede escribirse directamente en el código.

### Ayudas para la implementación

En la función para calcular el número de *carreras perfectas* de una escudería hay que tener en cuenta que una escudería puede disponer de varios pilotos en una misma carrera.

Adicionalmente, la empresa nos ha proporcionado los resultados esperados para los datos del mundial proporcionados:

Índice de la carrera	Ganador	Vuelta rápida
2	Bottas	Hamilton
4	Gasly	Norris

Escudería	Número de carreras perfectas
RED_BULL	2