

(4.5/10 puntos) EJERCICIO1 (45 MINUTOS)

ORDENACIÓN Y PUNTEROS A FUNCIONES

Dada la siguiente estructura:

```
#define LENGTH 100
struct nomina
{
    int codigo; //identificador único para cada empleado
    char nombre[LENGTH];
    float retenciones; //ej. 15.13
    float salarioBruto; //ej. 2548.5
};
```

Ordenación: Implementa una función para ordenar un vector de *struct nomina*. La ordenación será:

- ascendente por salario neto. **Salario neto** = **salario bruto** – **salario bruto*(retenciones/100)**.
 - Ejemplo: salario bruto:2548.5, retenciones: 15.13 => salario neto =2162.91.
- descendente por el campo nombre.

Punteros a funciones: Para evaluar este apartado, tu función de ordenación recibirá como parámetro un puntero a una función de comparación.

Nota: Se puede utilizar el algoritmo Quicksort de C para realizar los dos apartados de este ejercicio.

En Moodle tienes disponibles tres ficheros:

- *main.c*. Contiene un *main* de prueba para el ejercicio. NO CAMBIES NADA en este fichero, en caso contrario el ejercicio no se evaluará.
- *funciones.h*. Contiene la definición del *struct* para realizar el ejercicio y el prototipo de la función ***ejercicio1***. Añade aquí los prototipos de las funciones que necesites.
- *funciones.c*. Contiene la implementación vacía de la función ***ejercicio1***. Completa el código de esta función con el código fuente necesario para resolver el ejercicio. Puedes añadir las funciones que consideres oportunas.

```
void ejercicio1(int nEle, struct nomina* Vector, int criterio)
```

- *nEle*: número de elementos del vector
- *Vector*: el vector a ordenar. Ya ha sido reservado y rellenado con valores en el *main*.
- *Criterio de ordenación de Vector*:
 - 0: ascendente por salario neto
 - 1: descendente por nombre

ENTREGA:

- **Rellena al inicio de *funciones.c* tu nombre, apellidos y DNI.**
- Sube, como FICHEROS SEPARADOS (no comprimir), los ficheros *main.c*, *funciones.c* y *funciones.h* a la tarea del ejercicio, en caso contrario no se evaluará.
- Una vez expirada la hora tope para entregar la tarea no se aceptarán entregas.
- Todos los archivos se pasarán por un software detector de copias específico para códigos fuente. Se considerarán copiados todos los implicados, independiente de quién copie de quién y supondrá una calificación de cero en todos los ejercicios del examen.
- El ejercicio deberá compilar y solucionar el problema propuesto. En caso contrario, no se evaluará.
- Si no vas a entregar este ejercicio, no subas ningún fichero a Moodle.

COMPETENCIAS EVALUADAS

CU2: Conocer y perfeccionar el nivel de usuario en el ámbito de las TIC.

CEB4: Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería