

# DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

## PROYECTO FINAL DE FUNDAMENTOS DE INGENIERÍA DE SOFTWARE

### **NOMBRE DEL PROYECTO:** ACCESO EMPLEADOS

Una empresa de ingeniería desea tener una aplicación para el control de acceso de sus empleados. Para ello dispone de un fichero de texto denominado **empleados.txt**, el cual contiene los datos personales de sus empleados, dispuestos con el siguiente formato:

La primera línea contiene el número de empleados.

La segunda línea y sucesivas son los datos de los empleados, siendo una línea para cada empleado, compuesta de:

- **Nombre:** cadena de 60 caracteres( nombres y apellidos separados por guion bajo)
- **Dirección :** Cadena de 100 caracteres (separados por guion bajo)
- **Teléfono:** Vector de 9 enteros
- **E-mail:** cadena de 20 caracteres.

### **Ejemplo de línea de fichero**

Emilia\_Stefania\_Gómez\_Carrillo Av.Ameria\_310\_Mañosca 032820585 [esgomez@espe.edu.ec](mailto:esgomez@espe.edu.ec)

Por otro lado se dispone de otro fichero llamado **horas.txt**, el cual registra todas las entradas y salidas de los empleados de un año. Cada entrada o salida de un empleado se detalla en cada línea del fichero, con los siguientes campos separados por espacios:

- Identificador del empleado (numérico).
- Registro de entrada o salida (Valor 0: entrada, valor 1: salida).
- Hora con formato 99:99 (por ejemplo las 23:45).
- Fecha con formato : 99/99/9999

**Por ejemplo la siguiente línea de fichero corresponde a una entrada del empleado 2 a las 8:45 el día 3 de Diciembre de 2019:**

2 0 8:45 03/12/2019

Con estos dos ficheros de datos la empresa quiere tener un control sobre la puntualidad y horas de trabajo dedicadas, almacenando para cada empleado el número de faltas cometidas por impuntualidad y el cómputo total de horas trabajadas en el año. Se considera una falta de puntualidad al entrar en el trabajo a una hora mayor de las 07:30 de la mañana. El cómputo total de horas trabajadas se calcula, acumulando las diferencias entre la hora de entrada y salida de cada empleado.

1. Defina la estructura necesaria para almacenar de cada empleado, sus datos personales, faltas de puntualidad cometida y horas totales trabajadas. Defina un nuevo tipo llamado **EMPLEADO** y úselo a lo largo del todo el ejercicio.
2. Realice la función **registrar\_datos\_personales**, la cual reserva de forma dinámica la tabla de estructuras de empleados y almacena en ella el contenido del fichero **empleados.txt**. Los empleados se irán almacenando en la tabla en el orden en que aparecen en el fichero. No es necesario añadir un campo más en la estructura para la identificación del empleado, ya que será el índice en la tabla. Esta función debe devolver el número de empleados existentes en la empresa, así como la tabla de empleados completada.
3. Realice la función **carga\_horas** tal que, dada la estructura de empleados ya rellena, actualice, para cada empleado, el número de faltas de puntualidad y el cómputo de horas trabajadas. El

# DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

## PROYECTO FINAL DE FUNDAMENTOS DE INGENIERÍA DE SOFTWARE

identificador de cada empleado corresponde al índice en la tabla donde está almacenado (empezando por cero).

4. Realice la función ***imprime\_mejores\_empleados***, la cual recibe la tabla de empleados rellena con los datos de empleados, faltas, horas trabajadas. Esta función debe imprimir en pantalla los mejores empleados (si los hubiese) ordenados en forma ascendente por faltas. Un empleado será considerado mejor que otro si tiene menos faltas de puntualidad y su número total de horas es mayor.
5. Escribir el programa principal un menú para acceder a las opciones de: Registrar Datos Personales, Cargar Horas e Imprimir Mejores empleados, cada una de estas opciones deber llamar a las a las funciones correspondientes.

### DESCRIPCIÓN DE ENTREGABLES:

1. *Identifique los requisitos Funcionales y no Funcionales.*
2. *Priorice los requisitos Funcionales con los siguientes parámetros: requerido, opcional, conveniente.*
3. *Diseñe las interfaces de entrada y navegación correspondiente.*
4. *Realice pruebas unitarias para la función que elija desarrollar de acuerdo al proceso de pruebas, <https://www.browserstack.com/guide/how-to-write-test-cases>, revise este vinculo y realice el caso de prueba.*
5. *Programe la funcionalidad que considere requerido.*
6. *El proyecto debe estar en una carpeta con el siguiente formato: **G#\_PFINAL\_EMPLEADOS**, por ejemplo, **G1\_PFINAL\_EMPLEADOS**.*
7. *El archivo **empleado.txt**, con una carga de 10 registros como mínimo.*
8. *En esta carpeta debe estar en formato pdf los numerales 1,2,3,4 y el numeral 5 debe entregar el código fuente .c , debidamente documentado y comentado, comprima, archivo txt , comprima la carpeta y suba el archivo comprimido a la plataforma.*