| **Algoritmos y Estructura de Datos – 1er Parcial** | **29/08/2023 – Martes Noche** | |
| --- | --- | --- |
| Nombre y Apellido: Franco Marcolin | Curso: K1094 | Legajo: 2087753 |

El área de cobranza de una empresa de servicios de telefonía necesita unificar las llamadas de sus usuarios de un período en una sola estructura. Cuenta con:

A) Un archivo **Usuarios.dat** que contiene como máximo **10000 usuarios distintos**. Cada registro tiene la siguiente estructura:

| * Id de Usuario: 6 dígitos | * Nombre de Usuario: 100 caracteres |
| --- | --- |

B) Un archivo **Llamadas.dat** que posee un registro por cada llamada realizada por un usuario en un determinado período**.** Cada registro tiene la siguiente estructura:

| * Id de Usuario: 6 dígitos | * Duración en minutos de la llamada: float |
| --- | --- |
|  |  |

C) Cuenta con la siguiente función que puede invocar sin desarrollar:

***float obtenerPrecio(float minutos)***

La función devuelve el precio que se debe abonar por la llamada según el total de minutos.

Se requiere cargar un **array de totales por usuario** que contenga la información de todos los usuarios existentes sumarizando para cada uno el total de llamadas realizadas, el total de minutos hablados y el total a abonar. Cada elemento del array debe tener el siguiente formato:

| * Id de Usuario: 6 dígitos | * Total de minutos: float |
| --- | --- |
| * Nombre de Usuario: 100 caracteres | * Cantidad total de llamadas realizadas: unsigned * Total a abonar: float |
|  |  |

**Resuleva los siguientes puntos:**

**1)** Codifique la definición de los **tipos de datos** a utilizar (registros de los archivos y array de usuarios).

**2)** Codifique y luego utilice en la solución de su examen:

**2.1)** ***agregar***: recibe el array de totales por usuario, su len y una estructura del mismo tipo del array. Agrega esta estructura luego del último elemento útil que contenga el array.

**2.2)** ***buscar***: recibe el array de totales por usuario, su len y un Id de Usuario. Retorna la posición del usuario en el array si es que lo encuentra, -1 en caso contrario.

**3)** Realice el procedimiento ***cargarUsuarios*** que recibe por parámetro el array de totales vacio y su len ya inicializada y carga los usuarios del archivo detallado en el punto A.

**4)** Realice el procedimiento ***cargarLlamadas*** que recibe por parámetro el array de totales y su len luego de haberse cargado los usuarios en el punto 3. El procedimiento debe utilizar el archivo del punto B para sumarizar en cada usuario el total de minutos, la cantidad total de llamadas realizadas y el total a abonar. Para este último puede hacer uso de la función *obtenerPrecio* detallada en C.

**5)** Realice el procedimiento ***mostrarLlamadas*** que recibe por parámetro el array de totales y su len luego de haberse cargado los totales de las llamadas en el punto 4. El procedimiento muestra por pantalla el Id de Usuario y el Total a abonar para todos los usuarios que hayan realizado llamadas (pueden existir usuarios que no hayan realizado llamdas en el período analizado).