

## **Trabajo Práctico Número 2**

*Programación I – Laboratorio I*

*Tecnicatura Superior en Programación*

**UTN-FRA**

**Autores: Taboada Ezequiel.**

**Revisores: Scarafilo German, Cardozo Marina, Maggiori Gianni, Luchetta Giovanni.**

## Índice de contenido

### Guia de contenidos

1. Enunciado.....	
3	
2. Menu.....	
... 4	
3. Aclaraciones.....	5
4. Cómo realizar y entregar el trabajo práctico.....	6

## 1. ENUNCIADO:

La FIFA requiere un programa que administre los jugadores de cada confederación de fútbol.

Por el momento solo existen 6 confederaciones y se sabe que la lista de jugadores asociados a la FIFA no puede sobrepasar los 3000 jugadores.

```
typedef struct{  
  
    int id;  
    char nombre[50];  
    char region[50];  
    int anioCreacion;  
  
}eConfederacion;
```

```
typedef struct{  
    int id;  
    char nombre[50];  
    char posicion[50];  
    short numeroCamiseta;  
    int idConfederacion;  
    float salario;  
    short aniosContrato;  
    short isEmpty;  
  
}eJugador;
```

**Nota:** Se debe respetar el formato y tipos de datos.

## **2. MENÚ:**

El sistema deberá tener el siguiente menú de opciones:

1. **ALTA DE JUGADOR:** Se debe permitir ingresar un jugador calculando automáticamente el número de Id. El resto de los campos se le pedirá al usuario, al momento de ingresar la confederación se deberá mostrar la lista completa de confederaciones disponibles y su elección será mediante número de id.

**\*Nota:** el id del jugador debe ser autoincremental , único y autónomo, es decir, no debe depender de la posición de un array ni calcularse en base a buscar el mayor id que tenga un jugador dentro del array.

2. **BAJA DE JUGADOR:** Se deberá mostrar la lista completa de jugadores (ordenada por ID) con todos sus datos y se podrá elegir uno ingresando el Número de Id y se eliminará el jugador del sistema.

3. **MODIFICACIÓN DE JUGADOR:** Se deberá mostrar la lista completa de jugadores (ordenada por ID) con todos sus datos y se podrá elegir uno ingresando el Número de Id, permitiendo modificar: nombre, posicion, numero de camiseta, confederación, salario,años de contrato.  
Este proceso debe contar con menú propio permitiendo elegir qué campo se desea modificar.

4. **INFORMES:**

- ★ **Listado de los jugadores ordenados alfabéticamente por nombre de confederación y nombre de jugador.** (Es ordenamiento por doble criterio, se ordena por nombre de la confederación y en caso de igualdad se ordena por nombre del jugador).
- ★ **Listado de confederaciones con sus jugadores.**
- ★ **Total y promedio de todos los salarios y cuántos jugadores cobran más del salario promedio.**
- ★ **Informar la confederación con mayor cantidad de años de contratos total.**
- ★ **Informar porcentaje de jugadores por cada confederación.**
- ★ **Informar cual es la región con más jugadores y el listado de los mismos.**

Este proceso debe contar con menú propio permitiendo elegir qué informe se desea ver.

5. **SALIR:** Se deberá pedir confirmación al usuario antes de cerrar el programa.

### 3. ACLARACIONES

Al momento de imprimir se deberá informar el nombre correspondiente a cada campo en formato tabla con su correspondiente cabecera y se debe mostrar **el nombre de la confederación.**

Por ejemplo:

\*imagen 01

ID	NOMBRE	POSICION	Nº CAMISETA	SUELDO	CONFEDERACION	AÑOS de CONTRATO
1	Lionel Messi	Delantero	10	200000.00	UEFA	4
2	Givanildo Vieira(Hulk)	Delantero	7	100000.00	CONMEBOL	3
3	Mohamed El Shenawy	Arquero	1	80000.00	CAF	2
4	Enzo Copetti	Delantero	9	86000.00	CONMEBOL	4

\*imagen 02

ID	NOMBRE	REGION	AÑO CREACION
100	CONMEBOL	SUDAMERICA	1916
101	UEFA	EUROPA	1954
102	AFC	ASIA	1954
103	CAF	AFRICA	1957
104	CONCACAF	NORTE Y CENTRO AMERICA	1961
105	OFC	OCEANIA	1966

**Nota 1:** Se deberá realizar el menú de opciones y las validaciones a través de funciones. Tener en cuenta que no se podrá ingresar a los casos 2, 3 y 4 sin antes haber realizado la carga de algún jugador.

**Nota 2:** Cada entidad deberá contar con su biblioteca correspondiente con las funciones necesarias para el funcionamiento correcto del programa .

**Nota 3:** Se deberá contar con bibliotecas de funciones para entradas y salidas de datos, verificando y validando los mismos.

**Nota 4:** La entidad eConfederaciones deberá contar con sus funciones de alta, baja y modificación, pero para trabajar en el main solo usaremos los valores hardcodedos que se ven en la \*imagen02.

### 4. Cómo realizar la entrega

El trabajo práctico deberá ser entregado en el repositorio de GIT **tps\_Laboratorio\_1** (el mismo que fue reportado para el TP\_1), dentro de la carpeta **TP\_2**, el proyecto en github deberá incluir los archivos **.cproject** y **.project**, caso contrario no se corregirá y contará como desaprobado.

Recordar que el repositorio **tps\_Laboratorio\_1** que contendrán todos los TPs **deberá ser privado**.

El mismo consistirá en el proyecto de Eclipse con el programa funcionando y comentado, respetando las reglas de estilo de la cátedra.

La compilación no deberá arrojar mensajes de error ni de warnings.