

# Laboratorio 1

Con este laboratorio se espera que el estudiante adquiera competencias en la implementación de operaciones básicas, el uso básico del lenguaje **ruby**, así como reafirmar conceptos presentados en el curso hasta el momento.

También se espera que el estudiante recurra al material disponible en el grupo **Forge 2015** del curso y que recurra a Internet con espíritu crítico, identificando y corroborando fuentes confiables de información.

## Letra

Se desea implementar un sistema para el manejo de un campeonato de fútbol, el cual deberá tener un nombre que será agregado al ejecutar el sistema.

Cuando comienza el campeonato se deberá saber si el campeonato es de futbol 5, 7 u 11.

En esta primer versión del laboratorio se desea registrar los jugadores que participarán en el campeonato, de los cuales interesa el nombre y la cédula de identidad.

La cédula de identidad es lo que identifica a un jugador en el sistema, por lo que debemos asegurar que no existan dos jugadores con la misma cédula.

De los equipos interesa saber el nombre y no pueden existir dos equipos con el mismo nombre. Un jugador debe pertenecer a un solo equipo, y el equipo deberá tener la cantidad de jugadores especificada por el campeonato para poder participar.

Una vez que el equipo es creado se deberá poder agregar jugadores al mismo.

Para poder comenzar el campeonato y crear el fixture, la cantidad de equipos deberá ser par y no superar los 16 equipos.

El fixture está compuesto por un conjunto de partidos. Cuando se agrega un partido al fixture se necesitará saber los equipos que lo integran.

Para crear el fixture se deberá tener en cuenta todos los cruces posibles entre equipos, por ejemplo:

- Equipos
  - Nacional
  - Danubio
  - Defensor
  - Fenix

- Cruces
  - Nacional - Danubio
  - Defensor - Fenix
  - Nacional - Defensor
  - Danubio - Defensor
  - Fenix - Danubio
  - Nacional - Fenix

No se tomará en cuenta si el jugador es local o visitante y no será un campeonato de ida y vuelta, por lo cual el partido (Nacional - Danubio) es igual al partido (Danubio - Nacional)  
 Los partidos serán ordenados en forma aleatoria siendo el primer partido en la lista el próximo a ser jugado. Una vez que el partido es finalizado no se le podrán introducir cambios.

De los partidos también se desea poder consultar el próximo partido, mostrando los equipos involucrados. Al realizar esta consulta, el sistema permitirá ingresar el resultado final del mismo.

En todo momento se deberá tener la opción de ir viendo la tabla de posiciones, la cual será de la siguiente forma:

#### Tabla de Campeonato (Parcial)

1. Nacional | PTS: 6 | PJ: 2 | PG: 2 | PE: 0 | PP: 0
2. Danubio | PTS: 3 | PJ: 2 | PG: 1 | PE: 0 | PP: 1
3. Fenix | PTS: 1 | PJ: 1 | PG: 0 | PE: 1 | PP: 0
4. Defensor | PTS: 1 | PJ: 3 | PG: 0 | PE: 1 | PP: 2

*PTS: Puntos ( 3 si gana, 1 si es empate, 0 si pierde )*

*PJ: Partidos Jugados*

*PG: Partidos Ganados*

*PE: Partidos Empatados*

*PP: Partidos Perdidos*

La tabla de campeonato solo mostrará datos de los partidos finalizados.

Una vez que todos los partidos se han jugado, se mostrará un mensaje con el ganador del campeonato y la tabla final de resultados, por ejemplo:

- Nacional ha ganado el Campeonato !!

#### Tabla de Campeonato (Final)

1. Nacional | PTS: 9 | PJ: 3 | PG: 3 | PE: 0 | PP: 0
2. Danubio | PTS: 6 | PJ: 3 | PG: 2 | PE: 0 | PP: 1
3. Fenix | PTS: 1 | PJ: 3 | PG: 0 | PE: 1 | PP: 2
4. Defensor | PTS: 1 | PJ: 3 | PG: 0 | PE: 1 | PP: 2

## Interfaz

Junto con esta letra se proveerá de un programa principal el cual será utilizado para poner en marcha el sistema.

Se deberá implementar una interfaz con el fin de lograr una comunicación entre la misma y el programa principal, se tendrá que respetar las firmas de los métodos, incluyendo los parámetros dados.

La implementación de cada método deberá ser desarrollada como el estudiante crea conveniente, teniendo en cuenta el resultado que el método debe devolver y el/los parámetro/s de dicho método que se invocará desde el programa principal.

Dicha interfaz sera dada proximately.