

Pseudocodigo Champions League

Estructura Equipos

numero	int
gol	int
nombre	string
liga	string
estadio	string
capacidad	int
plantillaTitular	[11]jugadores
estadistica	estadisticas
izquierda	*equipos
derecha	*equipos

Estructura jugadores

nombre	string
numero	int
nacionalidad	string
posicion	string
golesAnotados	int

Estructura estadisticas

jugados	int
ganados	int
perdidos	int
golesFavor	int8
golesContra	int8
diferenciaGoles	int8

raiz tipo apuntador a estructura
siguienteFase [] int
contadores

Crear nodo raiz vacio

variable nuevo puntero a equipos
creacion del nuevo nodo
insertar valor de 0 a identificador
hacer nulos a sus dos hijos
pasar la direccion de el nodo a raiz

Creacion de los nodos

variable nuevo puntero a equipos
creacion del nuevo nodo
insertar valor a identificador dependiendo su posicion

```

llenar datos del equipo
for i=0;i<11;i++){
    pedir datos de jugador i
}

```

```

hacer nulos a sus dos hijos
    recorrer hasta que
        Si el valor del identificador actual es menor al anterior
            nodo=recorrer-izquierdo
        FinSi
    si no
        nodo=recorrer-derecho
    FinSi
    si valor identificador es menor a actual
        anterior-izquierdo= nuevo
    FinSi
    si no
        anterior-derecho= nuevo
    FinSi
fin

```

```

para i:=0;i<tamaño de equipos+1;i++
    Partido(recibe direccion de memoria de nodo posicion i)
        busqueda del equipo
        raiz = nodo
        si arbol es nulo
            retorna raiz
        FinSi
        si identificador es menor arbol-identificador
            buscar por la izquierda
        si no si identificador arbol-identificador
            buscar por la derecha
        si no
            si son iguales
                retorna raiz

        fin busqueda

```

```

busqueda=rivalUno
busqueda=rivalDos

```

```

si golLocal es mayor a golVis
    rivalUno es quien ha ganado
FinSi
si no si golLocal==golVis
    tiempos extra
FinSi

```

```

        si golLocal>golVis
            rivalUno ha ganado
        FinSi
        si no
            rival dos ha ganado
        FinSi

    si no
        tanda de penales
        random de goles
        si randomLocal==randomVis
            si golLocal==1
                golLocal=0
            FinSi
            si no
                golLocal=0
            FinSi
        FinSi
        si no si golVis==1
            golVis=0
            FinSi
            si no
                golVis=1
            FinSi
        FinSi

    FinSi

    elimina equipo perdedor

    si nodo es nulo
        retorna nodo
    FinSi
    si no si identificador < identificador raiz
        busca por la izquierda
    FinSi
    si no si identificador < identificador raiz
        busca por la derecha
    FinSi

    si no
        si raiz-izquierda es nulo && raiz-derecha es nulo
            raiz se iguala a nulo
    FinSi

    si no si raiz-izquierda es nulo tiene un hijo
        auxiliar es de tipo raiz
        raiz = raiz-derecha
        auxiliar se iguala a nulo

```

FinSi

si no si raiz-derecha es nulo tiene un hijo

temp := root

root = root.derecha

temp = nil

if temp == nil

FinSi

si no tiene dos hijos

 auxiliar es igual a raiz-derecha

para auxiliar-izquierda sea diferente de nulo

auxiliar es igual a temp.izquierda

FinPara

raiz-identificador es igual a auxiliar-identificador

raiz-derecha es igual a eliminaNodo(raiz-derecha, auxiliar-identificador

retorna raiz

Equipo eliminado

EquipoVencedor= siguiente fase[i]

i++

FinPara

Encuentro cuartosdeFinal

Encuentro semifinal

Encuentro Final

Imprimir campeon

FinPseudocodigo