

Explicación Práctica de Monitores – Ejercicio adicional

Ejercicios

EJERCICIO de la práctica

En un entrenamiento de futbol hay 20 jugadores que forman 4 equipos (cada jugador conoce el equipo al cual pertenece llamando a la función `DarEquipo()`). Cuando un equipo está listo (han llegado los 5 jugadores que lo componen), debe enfrentarse a otro equipo que también esté listo (los dos primeros equipos en juntarse juegan en la cancha 1, y los otros dos equipos juegan en la cancha 2). Una vez que el equipo conoce la cancha en la que juega, sus jugadores se dirigen a ella. Cuando los 10 jugadores del partido llegaron a la cancha comienza el partido, juegan durante 50 minutos, y al terminar todos los jugadores del partido se retiran (no es necesario que se esperen para salir).



EJERCICIO de la práctica

Process Jugador[id: 0..19]

```
{ Cancha.llegada();  
}
```

¿Como sabe a qué cancha ir?

Process Partido [id: 0..1]

```
{ Cancha[id].Iniciar();  
  delay (90minutos); // se juega el partido  
  Cancha[id].Terminar();  
}
```

Monitor Cancha [id: 0..1]

```
{ int cant = 0;  
  cond espera, inicio;
```

Procedure llegada ()

```
{ cant ++;  
  if (cant == 10) signal (inicio);  
  wait (espera);  
}
```

Procedure Iniciar ()

```
{ if (cant < 10) wait (inicio);  
}
```

Procedure Terminar ()

```
{ signal_all(espera);  
}
```

```
}
```



EJERCICIO de la práctica

Primero cada equipo se debe juntar (independientemente del resto), y luego debe ver a que cancha ir. Se necesita un monitor para cada equipo.

Process Jugador[id: 0..19]

```
{ int miEquipo = ...;
  int numeroC;

  Equipo[miEquipo].llegada(numeroC);
}
```

Monitor Equipo [id: 0..3]

```
{ int cant = 0;
  cond espera;

  Procedure llegada (cancha: OUT int)
  { cant ++;
    if (cant < 4) wait (espera)
    else { cancha = Determinar la cancha
          signal_all (espera);
        }
  }
}
```

¿Cómo se hace, porque no depende sólo de este grupo?



EJERCICIO de la práctica

Se requiere un monitor para administrar las canchas

Process Jugador[id: 0..19]

```
{ int miEquipo = ...;
  int numeroC;

  Equipo[miEquipo].llegada(numeroC);
}
```

Monitor Admin

```
{ int cant = 0;

  Procedure DarCancha (cancha: OUT int)
  { cant ++;
    if (cant <= 2) cancha = 0
    else cancha = 1;
  }
}
```

Monitor Equipo [id: 0..3]

```
{ int cant = 0;
  cond espera;

  Procedure llegada (cancha: OUT int)
  { cant ++;
    if (cant < 4) wait (espera)
    else { Admin.DarCancha(cancha);
          signal_all (espera);
        }
  }
}
```

Sólo lo conocerá uno de los integrantes del grupo (el último en llegar)



EJERCICIO de la práctica

Process Jugador[id: 0..19]

```
{ int miEquipo = ...;
  int numeroC;

  Equipo[miEquipo].llegada(numeroC);
}
```

Monitor Admin

```
{ int cant = 0;

  Procedure DarCancha (cancha: OUT int)
  { cant ++;
    if (cant <= 2) cancha = 0
    else cancha = 1;
  }
}
```

Monitor Equipo [id: 0..3]

```
{ int cant = 0, numCancha;
  cond espera;

  Procedure llegada (cancha: OUT int)
  { cant ++;
    if (cant < 4) wait (espera)
    else { Admin.DarCancha(numCancha);
          signal_all (espera);
        };
    cancha = numCancha;
  }
}
```

Ahora si cada persona puede ir a la cancha que le corresponde y jugar el partido.



EJERCICIO de la práctica

Process Jugador[id: 0..19]

```
{ int miEquipo = ...;
  int numeroC;

  Equipo[miEquipo].llegada(numeroC);
  Cancha[numeroC].llegada();
}
```

Process Partido [id: 0..1]

```
{ Cancha[id].Iniciar();
  delay (90minutos); // se juega el partido
  Cancha[id].Terminar();
}
```

Monitor Cancha [id: 0..1]

```
{ int cant = 0;
  cond espera, inicio;

  Procedure llegada ()
  { cant ++;
    if (cant == 10) signal (inicio);
    wait (espera);
  }

  Procedure Iniciar ()
  { if (cant < 10) wait (inicio);
  }

  Procedure Terminar ()
  { signal_all(espera);
  }
}
```



EJERCICIO de la práctica

Process Jugador[id: 0..19]

```
{ int miEquipo = ...;
  int numeroC;

  Equipo[miEquipo].llegada(numeroC);
  Cancha[numeroC].llegada();
}
```

Monitor Equipo [id: 0..3]

```
{ int cant = 0, numCancha;
  cond espera;
```

Procedure llegada (cancha: OUT int)

```
{ cant ++;
  if (cant < 4) wait (espera)
  else { Admin.DarCancha(numCancha);
        signal_all (espera);
      };
  cancha = numCancha;
}
}
```

Process Partido [id: 0..1]

```
{ Cancha[id].Iniciar();
  delay (90 minutos); // se juega el partido
  Cancha[id].Terminar();
}
```

Monitor Admin

```
{ int cant = 0;
```

Procedure DarCancha (num: OUT int)

```
{ cant ++;
  if (cant <= 2) num = 0
  else num = 1;
}
}
```

Monitor Cancha [id: 0..1]

```
{ int cant = 0;
  cond espera, inicio;
```

Procedure llegada ()

```
{ cant ++;
  if (cant == 10) signal (inicio);
  wait (espera);
}
```

Procedure Iniciar ()

```
{ if (cant < 10) wait (inicio);
}
```

Procedure Terminar ()

```
{ signal_all(espera);
}
}
```