

Explicación Monitores

Ejercicios

SINTAXIS

Monitor nombre;

Variables

Cond colaEspera;

cont: integer:=0;

variables que sólo se conocen en el monitor

Procedure uno ()

Var

Begin

....

End;

Begin

Inicialización variables. *Es lo primero en ejecutarse*

End.



*Operaciones con la variable condición:
wait(colaEspera);
signal(colaEspera);
signal_all (colaEspera);*

Ejercicio:

Existen N personas que desean utilizar un cajero automático. No es necesario tener en cuenta el orden de llegada de las personas al cajero.

Monitor Cajero;

cond cola;
int cant := 0;

Procedure usarCajero;

Begin

if (cant > 0) then wait (cola);
cant = cant + 1;

End;

Procedure salirCajero

Begin

cant = cant - 1;
signal(cola);

End;

End Monitor;

Process Persona[a:1..N]:

Cajero.usarCajero;

“usa el cajero t minutos”

Cajero.salirCajero;

End Process;

El if testea una sólo vez la condición.



Ejercicio:

Existen N personas que desean utilizar un cajero automático. No es necesario tener en cuenta el orden de llegada de las personas al cajero.

Monitor Cajero;

cond cola;
int cant := 0;

Procedure usarCajero;

Begin

while (cant > 0) wait (cola);
cant = cant + 1;

End;

Procedure salirCajero

Begin

cant = cant - 1;
signal(cola);

End;

End Monitor;

Process Persona[a:1..N]:

Cajero.usarCajero;

“usa el cajero t minutos”

Cajero.salirCajero;

End Process;

*Cómo se puede
aprovechar la exclusión
mutua?*



Ejercicio: Existen N personas que desean utilizar un cajero automático. No es necesario tener en cuenta el orden de llegada de las personas al cajero.

Monitor Cajero;

Procedure usarCajero (t :integer);

Begin

 delay (t);

End;

End Monitor;

Process Persona[$a:1..N$]:

 Cajero.usarCajero (t);

End Process;



Ejercicio:

Existen N personas que desean utilizar una sala que permite ingresar hasta tres personas al mismo tiempo. No es necesario tener en cuenta el orden.

Monitor Sala;

cond espera;

int cant:=0;

Procedure usarSala;

Begin

While (cant == 3) wait (espera);

Cant++;

End;

Procedure SalirSala;

Begin

Cant--;

Signal (espera);

End;

End Monitor;

Process Persona[a:1..N]:

Sala.usarSala;

//Esta un tiempo t en la sala

Sala.salirSala;

End Process;



signal_all(espera);



Ejercicio:

Existen N personas que desean utilizar un cajero automático. Se debe tener en cuenta el orden de llegada de las personas al cajero.

Monitor Cajero;

cond cola;
int cant := 0;

Procedure usarCajero;

Begin

while (cant > 0) wait (cola);
cant = cant + 1;

End;

Procedure salirCajero

Begin

cant = cant - 1;
signal(cola);

End;

End Monitor;

Process Persona[a:1..N]:

Cajero.usarCajero;

“usa el cajero t minutos”

Cajero.salirCajero;

End Process;

*Se respeta la cantidad
de personas pero **no se**
respeta el orden*



Ejercicio:

Existen N personas que desean utilizar un cajero automático. Se debe tener en cuenta el orden de llegada de las personas al cajero.

Monitor Cajero;

```
cond cola;  
boolean libre:= true;  
cant:integer:=0;
```

Procedure usarCajero;

Begin

```
  if (not libre) then
```

```
    cant:= cant + 1;
```

```
    wait (cola);
```

```
  else
```

```
    libre:= false;
```

End;

Procedure salirCajero

Begin

```
  if (cant > 0) then
```

```
    signal (cola);
```

```
    cant:= cant - 1;
```

```
  else
```

```
    libre:= true;
```

End;

End Monitor;

Process Persona[a:1..N]:

```
  Cajero.usarCajero;
```

```
  “usa el cajero t minutos”
```

```
  Cajero.salirCajero;
```

End Process;

