

# Examen de Programación Concurrente y Distribuida

3º Curso de Grado en Ingeniería Informática

**Febrero. Curso 2019-20**

## CUESTIONES

1. Describa brevemente las consideraciones de diseño para los sistemas distribuidos (No extenderse más de un folio por una cara). **(0,5 Puntos)**
2. Justifique si el siguiente algoritmo para el control de la concurrencia cumple las condiciones requeridas. **(0,75 Puntos)**

```

process P0
repeat
  c0 := 1;
  while c1=1 do c0=0;
  Sección Crítica
  c0 := 0;
  Resto0
forever

process P1
repeat
  c1 := 1;
  while c0=1 do;
  Sección Crítica
  c1 := 0;
  Resto1
forever
  
```

Donde inicialmente  $c0=0$  y  $c1=0$ .

3. Usando la instrucción hardware *exchange*, garantice la exclusión mutua para los procesos P1 y P2. **(0,75 Puntos)**

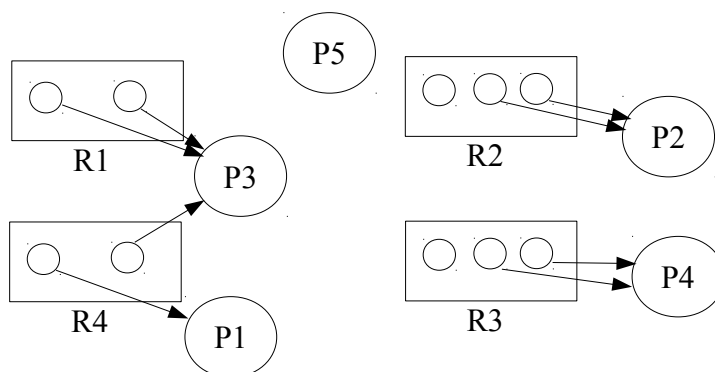
```

process P1
repeat
  Sección Crítica
  Resto1
forever

process P2
repeat
  Sección Crítica
  Resto2
forever
  
```

4. **(1 Punto)**. Tenemos un sistema operativo con 5 procesos, que en un momento dado presenta el siguiente estado:

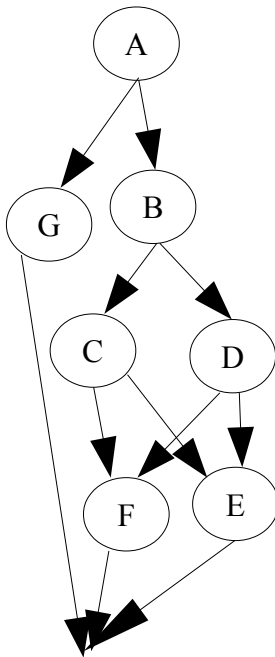
	N. Máximas			
	R1	R2	R3	R4
P1	1	1	0	1
P2	0	2	0	1
P3	2	1	0	1
P4	0	1	2	0
P5	1	0	1	0



Usando la técnica para evitar los interbloqueos, ¿debería concederse a P3 un ejemplar del recurso de R2?. Justifique la respuesta.

## PROBLEMAS

5. Dado el grafo de precedencias de la figura. Complete el programa que aparece a continuación, únicamente con los semáforos que están definidos (y con la inicialización que se define) para que cumpla dicho grafo de precedencias. **(1,5 Puntos)**

**Program P**

```

var
  s1: semaphore;
  s2: semaphore;
  s3: semaphore;
  s4: semaphore;

```

**process P1**

```

begin
  A;
  G;
end

```

**process P2**

```

begin
  C;
  E;
end

```

**process P3**

```

begin
  B;
  D;
  F;
end

begin
  initial(s1,1);
  initial(s2,1);
  initial(s3,0);
  initial(s4,0);
  conbegin
    P1; P2; P3;
  coend
end

```

6. Un concesionario de venta de vehículos dispone de dos comerciales y un gerente. Los clientes llegan al concesionario para ver los modelos o para realizar la recogida de vehículos. Tanto los comerciales como el gerente atienden a ambos tipos de clientes, pero los comerciales atienden prioritariamente a los clientes que van a ver los modelos, mientras que el gerente atiende prioritariamente a los que van a recoger vehículos.

Solucionar el problema anterior usando **monitores**. Se asume una semántica de la operación `resume` tipo “desbloquear y espera urgente” (la habitual de *Pascal-FC*). **(2,5 Puntos)**

7. Solucionar el problema anterior usando **canales**. **(3 Puntos)**

**ANEXO 1. Estructura de los procesos para los problemas 6 y 7**

```
program concesionario;  
  
const  
    np1=50;  
    np2=50;  
  
process type cliente1(id:integer);  
begin  
    repeat  
        { PROTOCOLO OCUPACION }  
        writeln('Proceso tipo P1 ',id,' accediendo');  
        { PROTOCOLO LIBERACION }  
    forever  
end;  
  
process type cliente2(id:integer);  
begin  
    repeat  
        { PROTOCOLO OCUPACION }  
        writeln('Proceso tipo P2 ',id,' accediendo');  
        { PROTOCOLO LIBERACION }  
    forever  
end;  
  
var  
    i,j: integer;  
    CL1: array[1..np1] of clienteT1;  
    CL2: array[1..np2] of clienteT2;  
  
begin  
    cobegin  
        for i := 1 to np1 do CL1[i](i);  
        for j := 1 to np2 do CL2[j](j);  
    coend  
end.
```