

**Trabajo Práctico N° 13:**

**Módulo Concurrente (Realizar programas en R-info con distintos tipos de robots. Utilizar memoria compartida para la comunicación y sincronización entre robots. Combinar problemas con memoria compartida y pasaje de mensajes).**

**Ejercicio 1.**

*Realizar un programa con 2 robots recolectores de flores (floreros) y 2 robots recolectores de papeles (papeleros). Los floreros comparten área y tienen 5 intentos cada uno para juntar las flores de una esquina dentro de dicha área, elegida al azar en cada intento. Del mismo modo, los papeleros comparten área y tienen 3 intentos cada uno para juntar los papeles. En cada intento, cada robot va a la esquina al azar, junta todos los elementos (flores o papeles, según le corresponda) y vuelve a su esquina original. Al finalizar sus intentos, cada robot debe acceder a la esquina (10,10) y depositar los elementos recogidos de a uno.*

- Área de floreros: (1,1) a (5,10).
- Área de papeleros: (6,1) a (10,9).
- Esquinas de inicio de floreros: (6,10) y (7,10).
- Esquinas de inicio de papeleros: (8,10) y (9,10).

programa TP13\_E1

procesos

proceso juntarFloresEsquina (ES flores : numero)

comenzar

mientras (HayFlorEnLaEsquina)

tomarFlor

flores := flores + 1

fin

proceso juntarPapelesEsquina (ES papeles : numero)

comenzar

mientras (HayPapelEnLaEsquina)

tomarPapel

papeles := papeles + 1

fin

areas

areaD : AreaC(10,10,10,10)

areaF : AreaPC(1,1,5,10)

areaP : AreaPC(6,1,10,9)

areaF1 : AreaP(6,10,6,10)

areaF2 : AreaP(7,10,7,10)

areaP1 : AreaP(8,10,8,10)

areaP2 : AreaP(9,10,9,10)

robots

robot florero

**variables**

avIni, caIni, av, ca, flores : numero

**comenzar**

avIni := PosAv

caIni := PosCa

flores := 0

repetir 5

Random(av,1,5)

Random(ca,1,10)

BloquearEsquina(av,ca)

Pos (av,ca)

juntarFloresEsquina(flores)

Pos (avIni,caIni)

LiberarEsquina(av,ca)

repetir flores

BloquearEsquina(10,10)

Pos (10,10)

depositarFlor

Pos (avIni,caIni)

LiberarEsquina(10,10)

**fin**

robot papelero

**variables**

avIni, caIni, av, ca, papeles : numero

**comenzar**

avIni := PosAv

caIni := PosCa

papeles := 0

repetir 3

Random(av,6,10)

Random(ca,1,9)

BloquearEsquina(av,ca)

Pos (av,ca)

juntarPapelesEsquina(papeles)

Pos (avIni,caIni)

LiberarEsquina(av,ca)

repetir papeles

BloquearEsquina(10,10)

Pos (10,10)

depositarPapel

Pos (avIni,caIni)

LiberarEsquina(10,10)

**fin****variables**

robotF1, robotF2 : florero

robotP3, robotP4 : papelero

**comenzar**

AsignarArea(robotF1,areaD)

AsignarArea(robotF1,areaF)

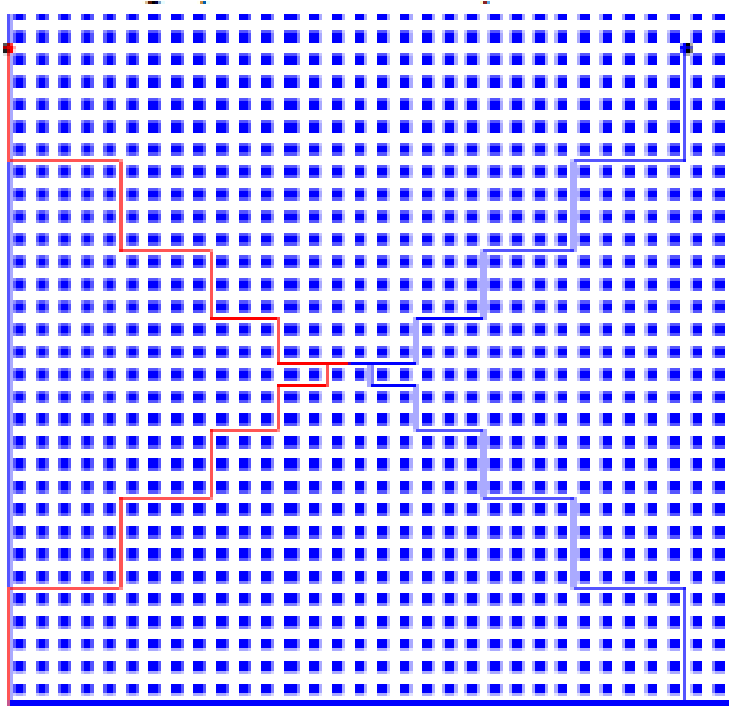
AsignarArea(robotF1,areaF1)

```
AsignarArea(robotF2,areaD)
AsignarArea(robotF2,areaF)
AsignarArea(robotF2,areaF2)
AsignarArea(robotP3,areaD)
AsignarArea(robotP3,areaP)
AsignarArea(robotP3,areaP1)
AsignarArea(robotP4,areaD)
AsignarArea(robotP4,areaP)
AsignarArea(robotP4,areaP2)
Iniciar(robotF1,6,10)
Iniciar(robotF2,7,10)
Iniciar(robotP3,8,10)
Iniciar(robotP4,9,10)
```

fin

**Ejercicio 2.**

Realizar un programa en el cual 2 robots corren una carrera. El recorrido realizado por cada uno es el que se muestra en la siguiente figura. Durante el recorrido, el robot 1 debe juntar todas las flores que encuentre en los vértices de cada escalón, mientras que el robot 2 debe juntar todos los papeles que encuentre en los vértices de cada escalón. Al finalizar, deben informar la cantidad de elementos recogidos. El robot 1 debe iniciar su recorrido en la esquina (1,1) y el robot 2 debe iniciar su recorrido en la esquina (31,1). Al finalizar la carrera, un robot jefe (inicializado en la esquina (15,1)) debe informar qué robot llegó primero a la esquina central de su recorrido.



programa TP13\_E2

procesos

proceso izquierda

comenzar

repetir 3

derecha

fin

proceso juntarFloresEsquina (ES flores : numero)

comenzar

mientras (HayFlorEnLaEsquina)

tomarFlor

flores := flores + 1

fin

proceso juntarPapelesEsquina (ES papeles : numero)

comenzar

mientras (HayPapelEnLaEsquina)

tomarPapel

papeles := papeles + 1

fin

proceso escaleraDecF (ES flores : numero)

variables

pasos : numero

comenzar

pasos := 5

juntarFloresEsquina(flores)

repetir 4

repetir pasos

mover

juntarFloresEsquina(flores)

derecha

repetir pasos

mover

juntarFloresEsquina(flores)

izquierda

pasos := pasos - 1

fin

proceso escalonCompF (ES flores : numero ; E rob : numero)

comenzar

mover

juntarFloresEsquina(flores)

derecha

BloquearEsquina(16,16)

mover

EnviarMensaje(rob,robotJ)

juntarFloresEsquina(flores)

repetir 2

derecha

mover

LiberarEsquina(16,16)

fin

proceso escaleraCrecF (ES flores : numero)

variables

pasos : numero

comenzar

pasos := 2

repetir 4

repetir pasos

mover

juntarFloresEsquina(flores)

derecha

repetir pasos

mover

juntarFloresEsquina(flores)

izquierda

pasos := pasos + 1

fin

proceso escaleraDecP (ES papeles : numero)

variables

```
pasos : numero
comenzar
pasos := 5
juntarPapelesEsquina(papeles)
repetir 4
  repetir pasos
  mover
juntarPapelesEsquina(papeles)
izquierda
repetir pasos
  mover
juntarPapelesEsquina(papeles)
derecha
pasos := pasos - 1
fin
proceso escalonCompP (ES papeles : numero ; E rob : numero)
comenzar
  mover
  juntarPapelesEsquina(papeles)
  izquierda
  BloquearEsquina(16,16)
  mover
  EnviarMensaje(rob,robotJ)
  juntarPapelesEsquina(papeles)
  repetir 2
    derecha
  mover
  LiberarEsquina(16,16)
fin
proceso escaleraCrecP (ES papeles : numero)
variables
pasos : numero
comenzar
pasos := 2
repetir 4
  repetir pasos
  mover
  juntarPapelesEsquina(papeles)
  izquierda
  repetir pasos
  mover
  juntarPapelesEsquina(papeles)
  derecha
pasos := pasos + 1
fin
areas
areaPC : AreaPC(16,16,16,16)
areaF : AreaPC(1,1,15,30)
areaP : AreaP(17,1,31,30)
robots
```

robot tipo1

variables

flores, rob : numero

comenzar

flores := 0

RecibirMensaje(rob,robotJ)

escaleraDecF(flores)

escalonCompF(flores,rob)

escaleraCrecF(flores)

Informar('Flores',flores)

EnviarMensaje(V,robotJ)

fin

robot tipo2

variables

papeles, rob : numero

comenzar

papeles := 0

RecibirMensaje(rob,robotJ)

escaleraDecP(papeles)

escalonCompP(papeles,rob)

escaleraCrecP(papeles)

Informar('Papeles',papeles)

EnviarMensaje(V,robotJ)

fin

robot jefe

variables

termino : boolean

rob, robGanador : numero

comenzar

EnviarMensaje(1,robot1)

EnviarMensaje(2,robot2)

RecibirMensaje(rob,\*)

robGanador := rob

RecibirMensaje(rob,\*)

repetir 2

RecibirMensaje(termino,\*)

Informar ('RobotGanador',robGanador)

fin

variables

robot1 : tipo1

robot2 : tipo2

robotJ : jefe

comenzar

AsignarArea(robot1,areaPC)

AsignarArea(robot1,areaF)

AsignarArea(robot2,areaPC)

AsignarArea(robot2,areaP)

AsignarArea(robotJ,areaF)

Iniciar(robot1,1,1)

Iniciar(robot2,31,1)

Iniciar(robotJ,15,1)

fin



**Ejercicio 3.**

Realizar un programa donde 4 robots colaboren para recoger todas las flores de una esquina indicada por un robot jefe, seleccionada, de manera aleatoria, dentro del cuadrante (2,2) y (10,10). Para ello, el jefe determina, inicialmente, una esquina y los robots deben accederla, tomar de a una las flores y volver a su posición inicial. Cuando los robots terminan, el jefe deberá informar cuál de ellos logró recoger más flores. Las esquinas de inicio de los robots deberán ser jefe (1,1) y robots (2,1), (3,1), (4,1) y (5,1).

programa TP13\_E3

procesos

proceso accederEsquina(ES flores : numero ; E av : numero ; E ca : numero)

variables

avIni, caIni : numero

comenzar

avIni := PosAv

caIni := PosCa

BloquearEsquina(av,ca)

Pos (av,ca)

mientras (HayFlorEnLaEsquina)

tomarFlor

flores := flores + 1

Pos (avIni,caIni)

LiberarEsquina(av,ca)

BloquearEsquina(av,ca)

Pos (av,ca)

Pos (avIni,caIni)

LiberarEsquina(av,ca)

fin

proceso actualizarMax (E flores : numero ; ES floresMax : numero ; E rob : numero ;

ES robMax : numero)

comenzar

si (flores > floresMax)

floresMax := flores

robMax := rob

fin

areas

areaPC: AreaPC(2,2,10,10)

area1: AreaP(2,1,2,1)

area2: AreaP(3,1,3,1)

area3: AreaP(4,1,4,1)

area4: AreaP(5,1,5,1)

areaJ: AreaP(1,1,1,1)

robots

robot recolector

variables

av, ca, flores, rob : numero

comenzar

flores := 0

```

RecibirMensaje(rob,robotJ)
RecibirMensaje(av,robotJ)
RecibirMensaje(ca,robotJ)
accederEsquina(flores,av,ca)
EnviarMensaje(rob,robotJ)
EnviarMensaje(flores,robotJ)

```

fin

robot jefe

variables

av, ca, flores, floresMax, rob, robMax : numero

comenzar

```

floresMax := 0
robMax := 0
EnviarMensaje(1,robot1)
EnviarMensaje(2,robot2)
EnviarMensaje(3,robot3)
EnviarMensaje(4,robot4)
Random(av,2,10)
Random(ca,2,10)
EnviarMensaje(av,robot1)
EnviarMensaje(ca,robot1)
EnviarMensaje(av,robot2)
EnviarMensaje(ca,robot2)
EnviarMensaje(av,robot3)
EnviarMensaje(ca,robot3)
EnviarMensaje(av,robot4)
EnviarMensaje(ca,robot4)
repetir 4
  RecibirMensaje(rob,*)
  si (rob = 1)
    RecibirMensaje(flores,robot1)
  sino
    si (rob = 2)
      RecibirMensaje(flores,robot2)
    sino
      si (rob = 3)
        RecibirMensaje(flores,robot3)
      sino
        RecibirMensaje(flores,robot4)
  actualizarMax(flores,floresMax,rob,robMax)
  Informar ('RobotGanador',robMax)

```

fin

variables

robot1, robot2, robot3, robot4 : recolector  
robotJ : jefe

comenzar

```

AsignarArea(robot1,areaPC)
AsignarArea(robot1,area1)
AsignarArea(robot2,areaPC)
AsignarArea(robot2,area2)

```

```
AsignarArea(robot3,areaPC)
AsignarArea(robot3,area3)
AsignarArea(robot4,areaPC)
AsignarArea(robot4,area4)
AsignarArea(robotJ,areaJ)
Iniciar(robot1,2,1)
Iniciar(robot2,3,1)
Iniciar(robot3,4,1)
Iniciar(robot4,5,1)
Iniciar(robotJ,1,1)
```

**fin**

**Ejercicio 4.**

(a) Realizar un programa en el que 4 robots mueven de a una todas las flores de la esquina (10,10) a la esquina (11,11). Para ello, cada robot que toma una flor de la esquina (10,10) la deposita en la esquina (11,11) y, luego, retorna a su esquina inicial. Cada robot que finaliza (o sea que detecta que la esquina (10,10) se ha vaciado) deberá avisar al robot coordinador que ha finalizado. Cuando todos los robots finalizaron, el robot coordinador deberá informar qué robot finalizó último y, a continuación, deberá recolectar todas las flores de la esquina (11,11).

- El robot coordinador inicia en la esquina (1,1).
- Los robots inician en las esquinas (9,9) (9,10) (9,11) y (9,12), respectivamente.

programa TP4\_E4a

procesos

proceso trasladarFlor

variables

avIni, caIni : numero

comenzar

avIni := PosAv

caIni := PosCa

BloquearEsquina(10,10)

Pos (10,10)

mientras (HayFlorEnLaEsquina)

tomarFlor

BloquearEsquina(11,11)

Pos (11,11)

LiberarEsquina(10,10)

depositarFlor

Pos (avIni,caIni)

LiberarEsquina(11,11)

BloquearEsquina(10,10)

Pos (10,10)

Pos (avIni, caIni)

LiberarEsquina(10,10)

fin

proceso juntarFloresEsquina

comenzar

mientras (HayFlorEnLaEsquina)

tomarFlor

fin

proceso recolectarFlores

comenzar

Pos (11,11)

juntarFloresEsquina

Pos (1,1)

fin

areas

areaPC1 : AreaPC(10,10,10,10)

```
areaPC2 : AreaPC(11,11,11,11)
area1 : AreaP(9,9,9,9)
area2 : AreaP(9,10,9,10)
area3 : AreaP(9,11,9,11)
area4 : AreaP(9,12,9,12)
areaC : AreaP(1,1,1,1)
```

robots

```
robot recolector
```

variables

```
rob : numero
```

comenzar

```
RecibirMensaje(rob,robotC)
```

```
trasladarFlor
```

```
EnviarMensaje(rob,robotC)
```

fin

```
robot coordinador
```

variables

```
rob : numero
```

comenzar

```
EnviarMensaje(1,robot1)
```

```
EnviarMensaje(2,robot2)
```

```
EnviarMensaje(3,robot3)
```

```
EnviarMensaje(4,robot4)
```

```
repetir 4
```

```
RecibirMensaje(rob,*)
```

```
Informar ('RobotFinalizóÚltimo',rob)
```

```
recolectarFlores
```

fin

variables

```
robot1, robot2, robot3, robot4 : recolector
```

```
robotC : coordinador
```

comenzar

```
AsignarArea(robot1,areaPC1)
```

```
AsignarArea(robot1,areaPC2)
```

```
AsignarArea(robot1,area1)
```

```
AsignarArea(robot2,areaPC1)
```

```
AsignarArea(robot2,areaPC2)
```

```
AsignarArea(robot2,area2)
```

```
AsignarArea(robot3,areaPC1)
```

```
AsignarArea(robot3,areaPC2)
```

```
AsignarArea(robot3,area3)
```

```
AsignarArea(robot4,areaPC1)
```

```
AsignarArea(robot4,areaPC2)
```

```
AsignarArea(robot4,area4)
```

```
AsignarArea(robotC,areaPC2)
```

```
AsignarArea(robotC,areaC)
```

```
Iniciar(robot1,9,9)
```

```
Iniciar(robot2,9,10)
```

```
Iniciar(robot3,9,11)
```

```
Iniciar(robot4,9,12)
```

Iniciar(robotC,1,1)

fin

**(b)** Implementar una variante en la cual los robots, luego de tomar cada flor de la esquina (10,10), vuelvan a su esquina inicial, pasen a la esquina (11,11) a depositarla y, finalmente, vuelvan a la esquina inicial.

programa TP4\_E4b

procesos

proceso trasladarFlor

variables

avIni, caIni : numero

comenzar

avIni := PosAv

caIni := PosCa

BloquearEsquina(10,10)

Pos (10,10)

mientras (HayFlorEnLaEsquina)

tomarFlor

Pos (avIni,caIni)

LiberarEsquina(10,10)

BloquearEsquina(11,11)

Pos (11,11)

depositarFlor

Pos (avIni,caIni)

LiberarEsquina(11,11)

BloquearEsquina(10,10)

Pos (10,10)

Pos (avIni, caIni)

LiberarEsquina(10,10)

fin

proceso juntarFloresEsquina

comenzar

mientras (HayFlorEnLaEsquina)

tomarFlor

fin

proceso recolectarFlores

comenzar

Pos (11,11)

juntarFloresEsquina

Pos (1,1)

fin

areas

areaPC1 : AreaPC(10,10,10,10)

areaPC2 : AreaPC(11,11,11,11)

area1 : AreaP(9,9,9,9)

area2 : AreaP(9,10,9,10)

area3 : AreaP(9,11,9,11)

```
area4 : AreaP(9,12,9,12)
areaC : AreaP(1,1,1,1)
robots
robot recolector
variables
  rob : numero
comenzar
  RecibirMensaje(rob,robotC)
  trasladarFlor
  EnviarMensaje(rob,robotC)
fin
robot coordinador
variables
  rob : numero
comenzar
  EnviarMensaje(1,robot1)
  EnviarMensaje(2,robot2)
  EnviarMensaje(3,robot3)
  EnviarMensaje(4,robot4)
  repetir 4
    RecibirMensaje(rob,*)
  Informar ('RobotFinalizóÚltimo',rob)
  recolectarFlores
fin
variables
robot1, robot2, robot3, robot4 : recolector
robotC : coordinador
comenzar
  AsignarArea(robot1,areaPC1)
  AsignarArea(robot1,areaPC2)
  AsignarArea(robot1,area1)
  AsignarArea(robot2,areaPC1)
  AsignarArea(robot2,areaPC2)
  AsignarArea(robot2,area2)
  AsignarArea(robot3,areaPC1)
  AsignarArea(robot3,areaPC2)
  AsignarArea(robot3,area3)
  AsignarArea(robot4,areaPC1)
  AsignarArea(robot4,areaPC2)
  AsignarArea(robot4,area4)
  AsignarArea(robotC,areaPC2)
  AsignarArea(robotC,areaC)
  Iniciar(robot1,9,9)
  Iniciar(robot2,9,10)
  Iniciar(robot3,9,11)
  Iniciar(robot4,9,12)
  Iniciar(robotC,1,1)
fin
```

**(c) Analizar:** *¿Cuál de las 2 soluciones maximiza la concurrencia? ¿Se podría resolver este problema sin que los robots deban regresar a su esquina inicial?*

La solución del inciso (b) maximiza la concurrencia. No se puede resolver este problema sin que los robots deban regresar a su esquina inicial.



**Ejercicio 5.**

Realizar un programa en el que 4 robots juegan una carrera por avenidas diferentes: 4, 6, 8 y 10, respectivamente. Todos los robots inician en la calle 1. Para poder avanzar, cada robot debe juntar un papel de una fuente de papeles localizada en la esquina (11,11), colocarlo en la esquina actual de su avenida y avanzar un paso. Cuando la esquina fuente ya no tiene más papeles o cuando se haya completado la avenida, deberán avisar al robot coordinador y éste determinará el robot que llegó más lejos.

**programa TP\_E5****procesos**

**proceso** juntarPapelyAvanzar (E av : numero ; ES ca : numero)

**comenzar**

BloquearEsquina(11,11)

Pos (11,11)

mientras ((HayPapelEnLaEsquina) & (PosCa < 100))

tomarPapel

Pos (av,ca)

LiberarEsquina(11,11)

depositarPapel

mover

ca := PosCa

BloquearEsquina(11,11)

Pos (11,11)

Pos (av,ca)

LiberarEsquina(11,11)

**fin**

**proceso** actualizarMax (E ca : numero ; ES caMax : numero ; E rob : numero ; ES robMax : numero)

**comenzar**

si (ca > caMax)

caMax := ca

robMax := rob

**fin****areas**

areaPC : AreaPC(11,11,11,11)

area1 : AreaP(4,1,4,100)

area2 : AreaP(6,1,6,100)

area3 : AreaP(8,1,8,100)

area4 : AreaP(10,1,10,100)

areaC : AreaP(1,1,1,1)

**robots**

robot corredor

**variables**

av, ca, rob : numero

**comenzar**

av := PosAv

ca := PosCa

RecibirMensaje(rob,robotC)

```
juntarPapelyAvanzar(av,ca)
EnviarMensaje(rob,robotC)
EnviarMensaje(ca,robotC)
fin
robot coordinador
variables
  ca, caMax, rob, robMax : numero
comenzar
  caMax := 0
  robMax := 0
  EnviarMensaje(1,robot1)
  EnviarMensaje(2,robot2)
  EnviarMensaje(3,robot3)
  EnviarMensaje(4,robot4)
  repetir 4
    RecibirMensaje(rob,*)
    si (rob = 1)
      RecibirMensaje(ca,robot1)
    sino
      si(rob = 2)
        RecibirMensaje(ca,robot2)
      sino
        si (rob = 3)
          RecibirMensaje(ca,robot3)
        sino
          RecibirMensaje(ca,robot4)
    actualizarMax(ca,caMax,rob,robMax)
  Informar ('RobotQueLlegóMásLejos',robMax)
fin
variables
  robot1, robot2, robot3, robot4 : corredor
  robotC : coordinador
comenzar
  AsignarArea(robot1,areaPC)
  AsignarArea(robot1,area1)
  AsignarArea(robot2,areaPC)
  AsignarArea(robot2,area2)
  AsignarArea(robot3,areaPC)
  AsignarArea(robot3,area3)
  AsignarArea(robot4,areaPC)
  AsignarArea(robot4,area4)
  AsignarArea(robotC,areaC)
  Iniciar(robot1,4,1)
  Iniciar(robot2,6,1)
  Iniciar(robot3,8,1)
  Iniciar(robot4,10,1)
  Iniciar(robotC,1,1)
fin
```

**Ejercicio 6.**

(a) Tres robots deben recorrer el perímetro de su cuadrante, como se indica a continuación:

- El robot 1 comienza la esquina (2,2) y debe realizar un cuadrante de 6x6 juntando todas las flores que encuentre.
- El robot 2 comienza en la esquina (5,5) y debe realizar un cuadrante de 10x10 juntando todas las flores y los papeles que encuentre.
- El robot 3 comienza en la esquina (9,9) y debe realizar un cuadrante de 7x7 juntando todos los papeles que encuentre.

Cada robot que finalice su cuadrante deberá avisar al robot fiscalizador. Al recibir el aviso, el robot fiscalizador indicará, inmediatamente, una calle a la que deberá dirigirse el robot recolector, considerando que el robot que finalizó primero irá a la calle 20, el segundo a la 21 y el tercero a la 22.

Cuando los robots recolectores reciben un número de calle, deberán posicionarse en la avenida 1 de dicha calle y avanzar a lo largo de la calle depositando, en cada esquina, un papel, una flor o ambos, según lo que cada robot haya juntado. El recorrido finalizará al completar la calle o vaciarse las bolsas.

programa TP13\_E6a

procesos

proceso juntarFloresPapeles (E rob : numero ; ES flores : numero ; ES papeles : numero)

comenzar

```
si ((rob = 1) | (rob = 2))
  mientras (HayFlorEnLaEsquina)
    tomarFlor
    flores := flores + 1
si ((rob = 2) | (rob = 3))
  mientras (HayPapelEnLaEsquina)
    tomarPapel
    papeles := papeles + 1
```

fin

proceso recorrerCalle (E rob : numero ; E flores : numero ; E papeles : numero)

comenzar

```
si ((rob = 1) | (rob = 2))
  mientras ((flores > 0) & (PosAv < 100))
    depositarFlor
    flores := flores - 1
    mover
  si (flores > 0)
    depositarFlor
si ((rob = 2) | (rob = 3))
  mientras ((papeles > 0) & (PosAv < 100))
    depositarPapel
    papeles := papeles - 1
    mover
```

```
    si (papeles > 0)
        depositarPapel
fin
areas
areaPC1 : AreaPC(2,2,16,16)
areaPC2 : AreaPC(1,20,100,22)
areaF : AreaP(1,1,1,1)
robots
robot recolector1
variables
    flores, papeles, rob, ca : numero
comenzar
    flores := 0
    papeles := 0
    rob := 1
    repetir 6
        juntarFloresPapeles(rob,flores,papeles)
        mover
        derecha
        repetir 2
            juntarFloresPapeles(rob,flores,papeles)
            mover
            juntarFloresPapeles(rob,flores,papeles)
            BloquearEsquina(5,8)
            mover
            juntarFloresPapeles(rob,flores,papeles)
            mover
            LiberarEsquina(5,8)
        repetir 2
            juntarFloresPapeles(rob,flores,papeles)
            mover
            derecha
            repetir 2
                juntarFloresPapeles(rob,flores,papeles)
                mover
                juntarFloresPapeles(rob,flores,papeles)
                BloquearEsquina(8,5)
                mover
                juntarFloresPapeles(rob,flores,papeles)
                mover
                LiberarEsquina(8,5)
            repetir 2
                juntarFloresPapeles(rob,flores,papeles)
                mover
                derecha
            repetir 6
                juntarFloresPapeles(rob,flores,papeles)
                mover
            EnviarMensaje(rob,robotF)
        repetir 2
```

```
derecha
RecibirMensaje(ca,robotF)
Pos (1,ca)
recorrerCalle(rob,flores,papeles)
fin
robot recolector2
variables
  flores, papeles, rob, ca : numero
comenzar
  flores := 0
  papeles := 0
  rob := 2
  repetir 2
    juntarFloresPapeles(rob,flores,papeles)
    mover
  juntarFloresPapeles(rob,flores,papeles)
  BloquearEsquina(5,8)
  mover
  juntarFloresPapeles(rob,flores,papeles)
  mover
  LiberarEsquina(5,8)
  repetir 6
    juntarFloresPapeles(rob,flores,papeles)
    mover
  derecha
  repetir 3
    juntarFloresPapeles(rob,flores,papeles)
    mover
  juntarFloresPapeles(rob,flores,papeles)
  BloquearEsquina(9,15)
  mover
  juntarFloresPapeles(rob,flores,papeles)
  mover
  LiberarEsquina(9,15)
  repetir 5
    juntarFloresPapeles(rob,flores,papeles)
    mover
  derecha
  repetir 5
    juntarFloresPapeles(rob,flores,papeles)
    mover
  juntarFloresPapeles(rob,flores,papeles)
  BloquearEsquina(15,9)
  mover
  juntarFloresPapeles(rob,flores,papeles)
  mover
  LiberarEsquina(15,9)
  repetir 3
    juntarFloresPapeles(rob,flores,papeles)
    mover
```

```
derecha
repetir 6
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(8,5)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(8,5)
repetir 2
  juntarFloresPapeles(rob,flores,papeles)
  mover
EnviarMensaje(rob,robotF)
repetir 2
  derecha
RecibirMensaje(ca,robotF)
Pos (1,ca)
recorrerCalle(rob,flores,papeles)
fin
robot recolector3
variables
  flores, papeles, rob, ca: numero
comenzar
  flores := 0
  papeles := 0
  rob := 3
  repetir 5
    juntarFloresPapeles(rob,flores,papeles)
    mover
  juntarFloresPapeles(rob,flores,papeles)
  BloquearEsquina(9,15)
  mover
  juntarFloresPapeles(rob,flores,papeles)
  mover
  LiberarEsquina(9,15)
  derecha
  repetir 7
    juntarFloresPapeles(rob,flores,papeles)
    mover
  derecha
  repetir 7
    juntarFloresPapeles(rob,flores,papeles)
    mover
  derecha
  juntarFloresPapeles(rob,flores,papeles)
  BloquearEsquina(15,9)
  mover
  juntarFloresPapeles(rob,flores,papeles)
  mover
```

```

LiberarEsquina(15,9)
repetir 5
  juntarFloresPapeles(rob,flores,papeles)
  mover
EnviarMensaje(rob,robotF)
repetir 2
  derecha
RecibirMensaje(ca,robotF)
Pos (1,ca)
recorrerCalle(rob,flores,papeles)
fin
robot fiscalizador
variables
  ca, rob : numero
comenzar
  ca := 20
  repetir 3
    RecibirMensaje(rob,*)
    si (rob = 1)
      EnviarMensaje(ca,robot1)
    sino
      si (rob = 2)
        EnviarMensaje(ca,robot2)
      sino
        EnviarMensaje(ca,robot3)
    ca := ca + 1
  fin
variables
  robot1 : recolector1
  robot2 : recolector2
  robot3 : recolector3
  robotF : fiscalizador
comenzar
  AsignarArea(robot1,areaPC1)
  AsignarArea(robot1,areaPC2)
  AsignarArea(robot2,areaPC1)
  AsignarArea(robot2,areaPC2)
  AsignarArea(robot3,areaPC1)
  AsignarArea(robot3,areaPC2)
  AsignarArea(robotF,areaF)
  Iniciar(robot1,2,2)
  Iniciar(robot2,5,5)
  Iniciar(robot3,9,9)
  Iniciar(robotF,1,1)
fin

```

(b) *Analizar (no es necesario implementar): ¿Cómo debería modificarse el ejercicio anterior si los robots recolectores no conocen, de antemano, el tamaño de su cuadrante (por ejemplo, porque lo calcula el fiscalizador de manera aleatoria)?*

Si los robots recolectores no conocen, de antemano, el tamaño de su cuadrante (por ejemplo, porque lo calculo el fiscalizador de manera aleatoria), el ejercicio anterior debería modificarse...

(c) *Modificar el ejercicio anterior (a) para que, ahora, el robot fiscalizador espere a que todos los robots recolectores hayan completado sus cuadrantes antes de indicarles la calle que deberán recorrer.*

### programa TP13\_E6c

#### procesos

proceso juntarFloresPapeles (E rob : numero ; ES flores : numero ; ES papeles : numero)

#### comenzar

```

si ((rob = 1) | (rob = 2))
  mientras (HayFlorEnLaEsquina)
    tomarFlor
    flores := flores + 1
si ((rob = 2) | (rob = 3))
  mientras (HayPapelEnLaEsquina)
    tomarPapel
    papeles := papeles + 1

```

#### fin

proceso recorrerCalle (E rob : numero ; E flores : numero ; E papeles : numero)

#### comenzar

```

si ((rob = 1) | (rob = 2))
  mientras ((flores > 0) & (PosAv < 100))
    depositarFlor
    flores := flores - 1
    mover
  si (flores > 0)
    depositarFlor
si ((rob = 2) | (rob = 3))
  mientras ((papeles > 0) & (PosAv < 100))
    depositarPapel
    papeles := papeles - 1
    mover
  si (papeles > 0)
    depositarPapel

```

#### fin

#### areas

```

areaPC1 : AreaPC(2,2,16,16)
areaPC2 : AreaPC(1,20,100,22)
areaF : AreaP(1,1,1,1)

```

#### robots

robot recolector1

#### variables

flores, papeles, rob, ca : numero



comenzar

```
flores := 0
papeles := 0
rob := 1
repetir 6
  juntarFloresPapeles(rob,flores,papeles)
  mover
  derecha
repetir 2
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(5,8)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(5,8)
repetir 2
  juntarFloresPapeles(rob,flores,papeles)
  mover
  derecha
repetir 2
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(8,5)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(8,5)
repetir 2
  juntarFloresPapeles(rob,flores,papeles)
  mover
  derecha
repetir 6
  juntarFloresPapeles(rob,flores,papeles)
  mover
EnviarMensaje(rob,robotF)
repetir 2
  derecha
RecibirMensaje(ca,robotF)
Pos (1,ca)
recorrerCalle(rob,flores,papeles)
```

fin

robot recolector2

variables

flores, papeles, rob, ca : numero

comenzar

```
flores := 0
papeles := 0
```

```
rob := 2
repetir 2
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(5,8)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(5,8)
repetir 6
  juntarFloresPapeles(rob,flores,papeles)
  mover
derecha
repetir 3
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(9,15)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(9,15)
repetir 5
  juntarFloresPapeles(rob,flores,papeles)
  mover
derecha
repetir 5
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(15,9)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(15,9)
repetir 3
  juntarFloresPapeles(rob,flores,papeles)
  mover
derecha
repetir 6
  juntarFloresPapeles(rob,flores,papeles)
  mover
juntarFloresPapeles(rob,flores,papeles)
BloquearEsquina(8,5)
mover
juntarFloresPapeles(rob,flores,papeles)
mover
LiberarEsquina(8,5)
repetir 2
```

```
juntarFloresPapeles(rob,flores,papeles)
  mover
EnviarMensaje(rob,robotF)
repetir 2
  derecha
RecibirMensaje(ca,robotF)
Pos (1,ca)
recorrerCalle(rob,flores,papeles)
fin
robot recolector3
variables
  flores, papeles, rob, ca: numero
comenzar
  flores := 0
  papeles := 0
  rob := 3
  repetir 5
    juntarFloresPapeles(rob,flores,papeles)
    mover
    juntarFloresPapeles(rob,flores,papeles)
    BloquearEsquina(9,15)
    mover
    juntarFloresPapeles(rob,flores,papeles)
    mover
    LiberarEsquina(9,15)
    derecha
    repetir 7
      juntarFloresPapeles(rob,flores,papeles)
      mover
      derecha
      repetir 7
        juntarFloresPapeles(rob,flores,papeles)
        mover
        derecha
        juntarFloresPapeles(rob,flores,papeles)
        BloquearEsquina(15,9)
        mover
        juntarFloresPapeles(rob,flores,papeles)
        mover
        LiberarEsquina(15,9)
      repetir 5
        juntarFloresPapeles(rob,flores,papeles)
        mover
    EnviarMensaje(rob,robotF)
    repetir 2
      derecha
    RecibirMensaje(ca,robotF)
    Pos (1,ca)
    recorrerCalle(rob,flores,papeles)
  fin
```

```
robot fiscalizador
variables
  ca, ca1, ca2, ca3, rob : numero
comenzar
  ca := 20
  repetir 3
    RecibirMensaje(rob,*)
    si (rob = 1)
      ca1 := ca
    sino
      si (rob = 2)
        ca2 := ca
      sino
        ca3 := ca
    ca := ca + 1
  EnviarMensaje(ca1,robot1)
  EnviarMensaje(ca2,robot2)
  EnviarMensaje(ca3,robot3)
fin
variables
  robot1 : recolector1
  robot2 : recolector2
  robot3 : recolector3
  robotF : fiscalizador
comenzar
  AsignarArea(robot1,areaPC1)
  AsignarArea(robot1,areaPC2)
  AsignarArea(robot2,areaPC1)
  AsignarArea(robot2,areaPC2)
  AsignarArea(robot3,areaPC1)
  AsignarArea(robot3,areaPC2)
  AsignarArea(robotF,areaF)
  Iniciar(robot1,2,2)
  Iniciar(robot2,5,5)
  Iniciar(robot3,9,9)
  Iniciar(robotF,1,1)
fin
```