



Taller de Programación



AGENDA

Ambiente CMRE

Memoria Compartida

Ejemplos

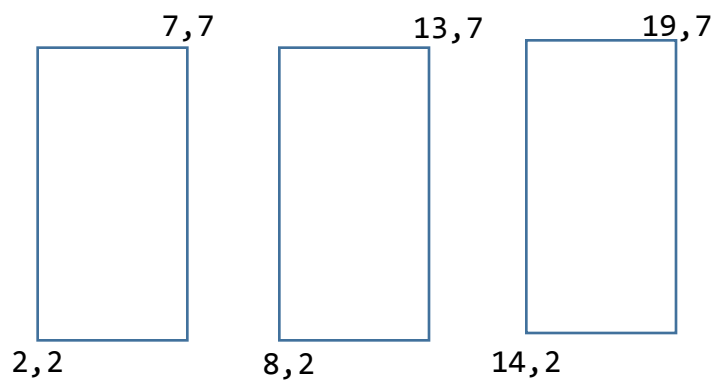


PENDIENTE DE LA CLASE

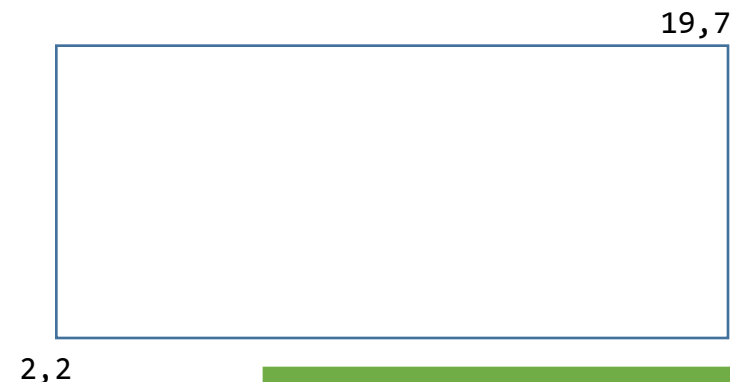


Ejercicio 3: Implemente el siguiente juego. Existen áreas privadas para cada uno de tres robots delimitadas por las esquinas $(2,2)$ $(7,7)$; $(8,2)$ $(13,7)$; $(14,2)$ $(19,7)$ respectivamente y un robot fiscalizador.

Cada robot jugador debe tratar de juntar la mayor cantidad de flores posible, para esto tiene tres intentos. En cada intento se posiciona en una esquina determinada al azar (dentro de su área) y junta todas las flores de esa esquina y vuelve a su esquina original. El robot fiscalizador determinará cuántas flores juntaron entre los 3 robots. Los robots se posicionan inicialmente en $(2,2)$, $(8,2)$ y $(15,2)$ y el robot fiscalizador en la esquina $(1,1)$.



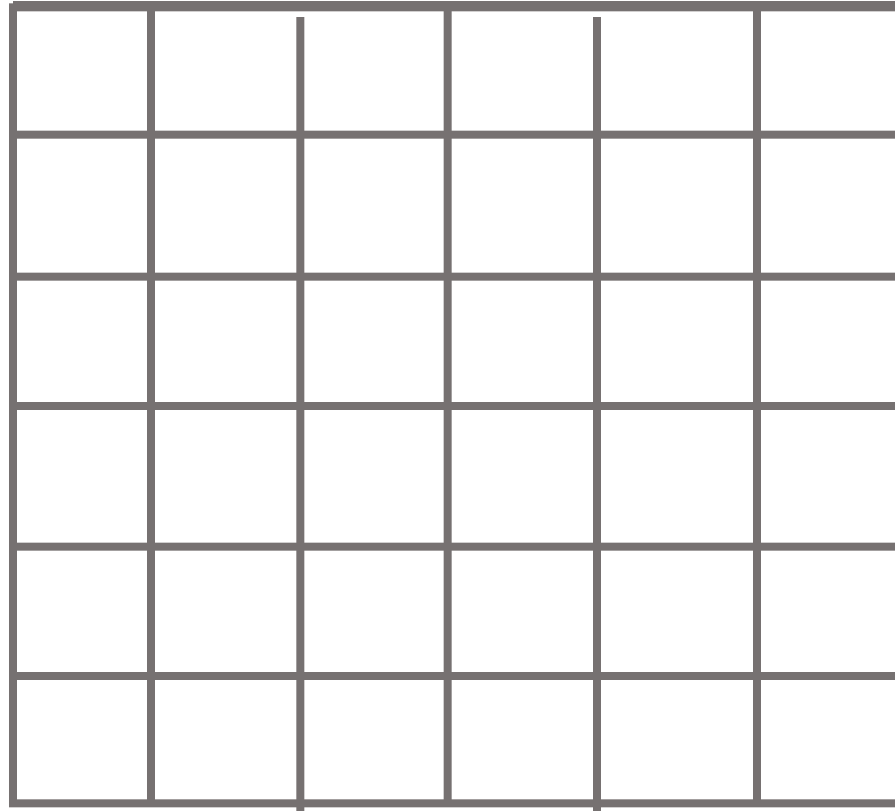
¿Qué ocurre si el área es compartida?



¿Qué problema debemos considerar?



PENDIENTE DE LA CLASE

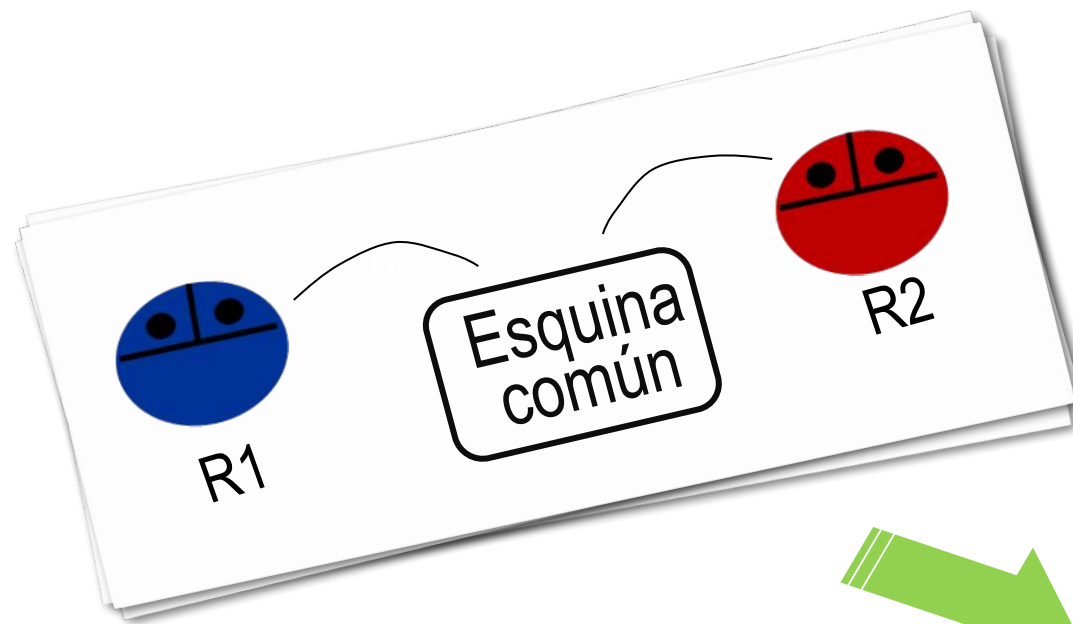


Dos o más robots
podrían pararse en la
misma esquina



COMUNICACIÓN – Memoria Compartida

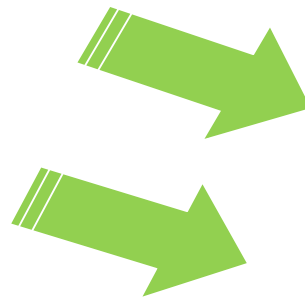
MEMORIA COMPARTIDA



proceso 1



proceso 2

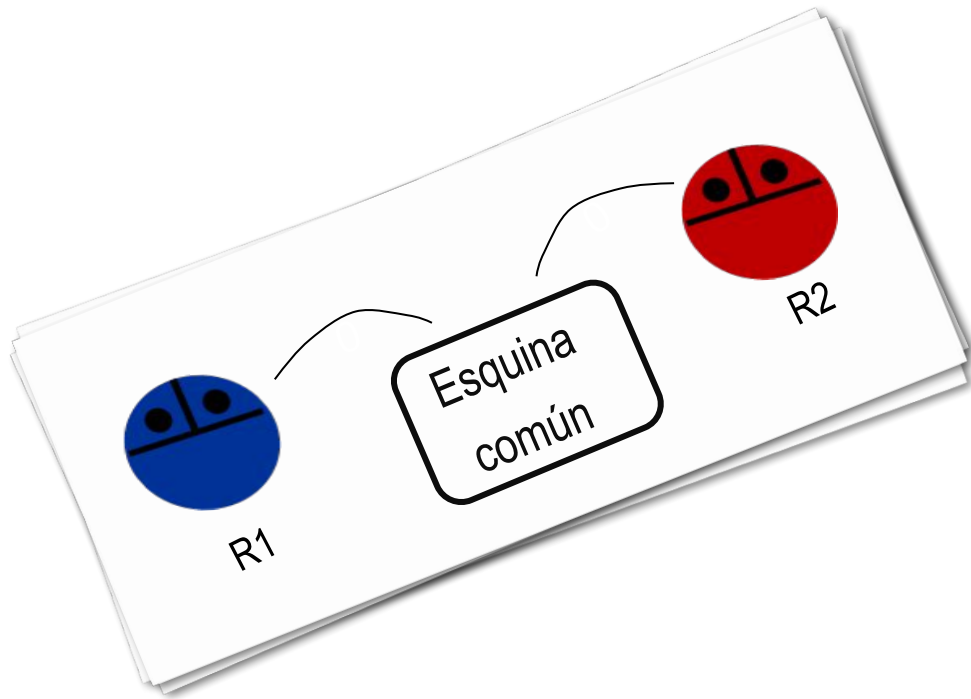


Bloquear un recurso – Desbloquear un recurso

A cargo del programador – A cargo del lenguaje



COMUNICACIÓN – Memoria Compartida



MANEJO DE RECURSOS- **BLOQUEAR**

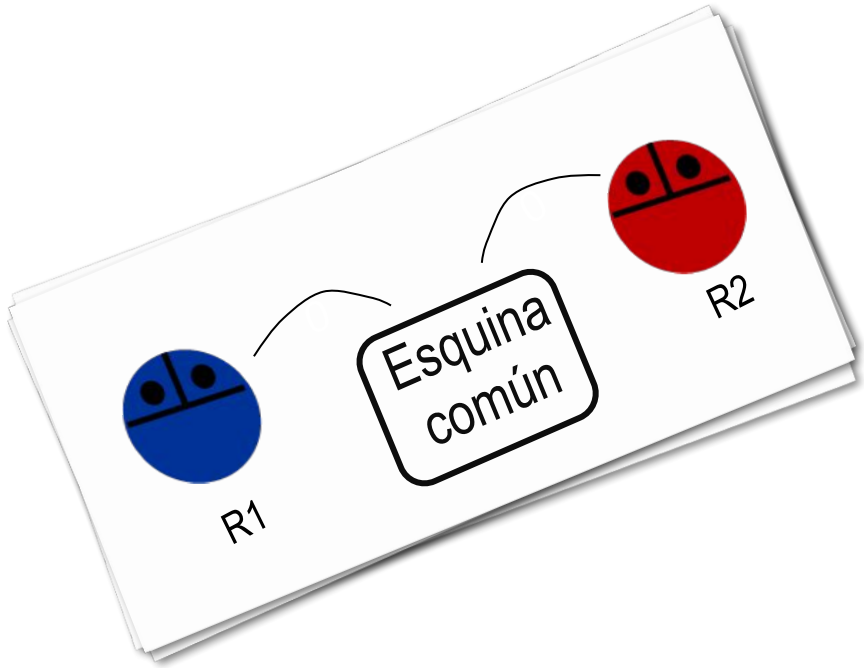
Dado un recurso **DISPONIBLE** el programador bloquea dicho recurso para que ningún otro proceso pueda acceder.

MANEJO DE RECURSOS- **DESBLOQUEAR**

Dado un recurso **BLOQUEADO** el programador libera dicho recurso para que cualquier proceso pueda bloquearlo.



COMUNICACIÓN – Memoria Compartida



MANEJO DE RECURSOS- **PROGRAMADOR**

El control de los accesos al recurso se encuentra a cargo del código desarrollado por el programador.

MANEJO DE RECURSOS- **SIST. OPERATIVO**

El control de los accesos al recurso se encuentra a cargo del sistema operativo.

¿Cómo es en
CRME?



COMUNICACIÓN – Memoria Compartida

Sintaxis

BloquearEsquina(avenida, calle)



BloquearEsquina(10, 15)

BloquearEsquina(posAv, posCa)

BloquearEsquina(av, ca)

COMO FUNCIONA?

Instrucción 1

Instrucción 2

BloquearEsquina(av, ca)

Instrucción 3



Si la esquina **NO** está bloqueada, se bloquea y el proceso sigue su ejecución.

Si la esquina está bloqueada, el proceso “queda esperando” para poder ejecutar la instrucción de bloqueo hasta que la esquina **NO** esté bloqueada.



COMUNICACIÓN – Memoria Compartida

Sintaxis

`LiberarEsquina(avenida, calle)`



`LiberarEsquina(10,15)`

`LiberarEsquina(posAv,posCa)`

`LiberarEsquina(av,ca)`

COMO FUNCIONA?

Instrucción 1

Instrucción 2

`LiberarEsquina(av,ca)`

Instrucción 3



Si la esquina está bloqueada, se desbloquea y el proceso sigue su ejecución.

Si la esquina está **NO** está bloqueada, no hace nada. No debería desbloquearse una esquina que no ha sido bloqueada anteriormente.



COMUNICACIÓN – Memoria Compartida - Ejemplos



Supongamos que tenemos un programa en el cual están declarados dos tipos robots y dos variables robot, una correspondiente a cada tipo. El robot 1 (de tipo 1) inicia su trabajo en la esquina (1,1) camina 5 cuadras y luego se posiciona en la esquina (10,10). El robot2 (de tipo 2) inicia su trabajo en la esquina (2,2) junta las flores de la esquina y luego se posiciona en la esquina (10,10). Ambos robots después de su trabajo vuelven a su esquina original.



Analicemos las siguientes soluciones

...



COMUNICACIÓN – Memoria Compartida - Ejemplos

SOLUCION 1: Es correcta?

```
programa problema1
areas
  area1: AreaPC(1,1,100,100)
robots
  robot tipo1
  comenzar
  ...
  fin
  robot tipo2
  comenzar
  ...
  fin

variables
  robot1: tipo1
  robot2: tipo2
```

```
robot tipo1
comenzar
  repetir 5
    mover
  Pos (10,10)
  Pos (1,1)
fin
```

```
robot tipo2
comenzar
  mientras (hayFlorEnLaEsquina)
    tomarFlor
  Pos (10,10)
  Pos (2,2)
fin
```

Los robots podrían chocarse
al querer posicionarse en la
esquina (10,10)



COMUNICACIÓN – Memoria Compartida - Ejemplos

SOLUCIÓN 1: ¿Es correcta?

```
programa problema1
areas
  area1: AreaPC(1,1,100,100)
robots
  robot tipo1
  comenzar
  ...
  fin
  robot tipo2
  comenzar
  ...
  fin

variables
  robot1: tipo1
  robot2: tipo2
```

```
robot tipo1
comenzar
  repetir 5
    mover
    BloquearEsquina(10,10)
    Pos (10,10)
    LiberarEsquina(10,10)
    Pos (1,1)
  fin
```

```
robot tipo2
comenzar
  mientras (hayFlorEnLaEsquina)
    tomarFlor
    Pos (10,10)
    Pos (2,2)
  fin
```

Los robots podrían chocarse al querer posicionarse en la esquina (10,10)



COMUNICACIÓN – Memoria Compartida - Ejemplos

SOLUCIÓN 2: ¿Es correcta?

```
programa problema1
areas
  area1: AreaPC(1,1,100,100)
robots
  robot tipo1
  comenzar
  ...
  fin
  robot tipo2
  comenzar
  ...
  fin

variables
  robot1: tipo1
  robot2: tipo2
```

```
robot tipo1
comenzar
  repetir 5
    mover
  BloquearEsquina(10,10)
  Pos (10,10)
  LiberarEsquina(10,10)
  Pos (1,1)
fin
```

```
robot tipo2
comenzar
  mientras (hayFlorEnLaEsquina)
    tomarFlor
  BloquearEsquina(10,10)
  Pos (10,10)
  LiberarEsquina(10,10)
  Pos (2,2)
fin
```

Los robots podrían chocarse al querer posicionarse en la esquina (10,10)



COMUNICACIÓN – Memoria Compartida - Ejemplos

SOLUCIÓN 3: ¿Es correcta?

```
programa problema1
areas
  area1: AreaPC(1,1,100,100)
robots
  robot tipo1
  comenzar
  ...
fin
robot tipo2
comenzar
...
fin

variables
  robot1: tipo1
  robot2: tipo2
```

robot tipo1

comenzar

repetir 5

mover

BloquearEsquina(10,10)

Pos (10,10)

LiberarEsquina(10,10)

Pos (1,1)

fin

robot tipo2

comenzar

BloquearEsquina(10,10)

mientras (hayFlorEnLaEsquina)

tomarFlor

Pos (10,10)

LiberarEsquina(10,10)

Pos (2,2)

fin

No debo bloquear
antes los recursos



COMUNICACIÓN – Memoria Compartida - Ejemplos

SOLUCIÓN 4: ¿Es correcta?

```
programa problema1
areas
  area1: AreaPC(1,1,100,100)
robots
  robot tipo1
  comenzar
  ...
  fin
  robot tipo2
  comenzar
  ...
  fin

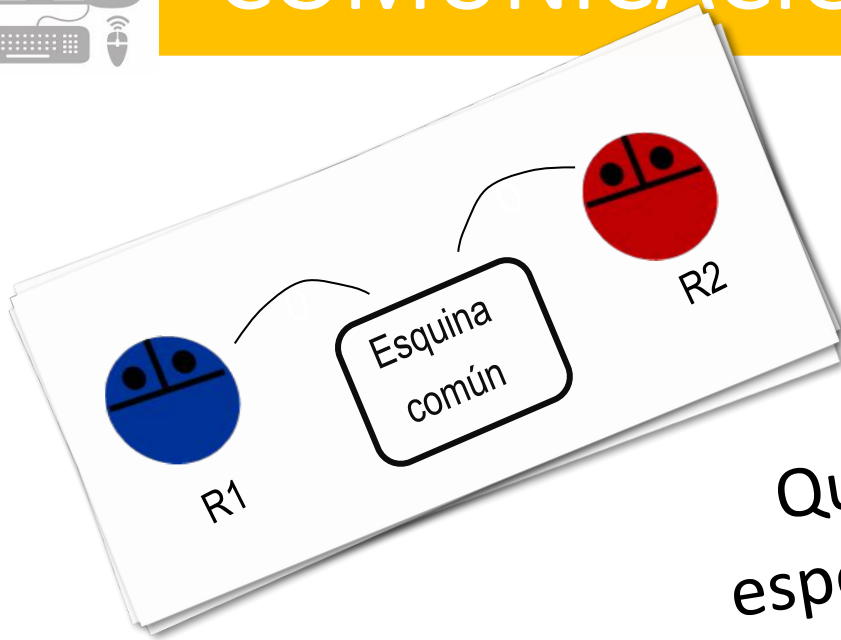
variables
  robot1: tipo1
  robot2: tipo2
```

```
robot tipo1
comenzar
  repetir 5
    mover
    BloquearEsquina(10,10)
    Pos (10,10)
    Pos (1,1)
    LiberarEsquina(10,10)
  fin

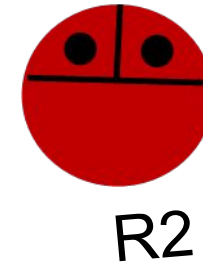
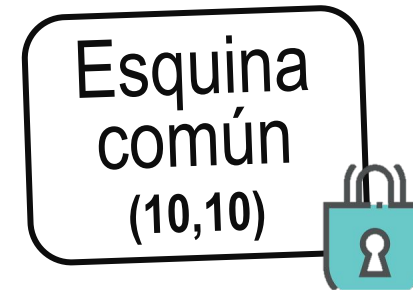
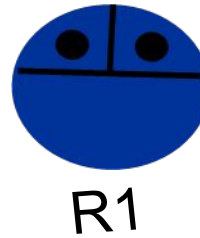
robot tipo2
comenzar
  mientras (hayFlorEnLaEsquina)
    tomarFlor
    BloquearEsquina(10,10)
    Pos (10,10)
    Pos (2,2)
    LiberarEsquina(10,10)
  fin
```



COMUNICACIÓN – Memoria Compartida - Ejemplos



Queda esperando



No debo bloquear recursos sino es necesario

Debo liberar un recurso lo más pronto posible

Bloquear la esquina(10,10)

Se posiciona en (10,10)

Vuelve a su posición inicial

Liberar la esquina (10,10)

...

Bloquear la esquina(10,10)

Se posiciona en (10,10)

Vuelve a su posición inicial

Liberar la esquina (10,10)

...



EJERCITACIÓN – Clase teórica



Para poder realizar esta actividad en el horario de teoría el alumno tiene que haber instalado el entorno en su computadora

Ejercicio 1: realice un programa para que 2 robots junten todas las flores de los perímetros $(15,15)-(30,35)$ y $(40,40)-(55,60)$ respectivamente. Luego, deben posicionarse en la esquina $(10,10)$, depositar todas las flores juntadas de a una.

Los robots se encuentran posicionados originalmente en la esquina $(15,15)$ y $(40,40)$.

Ejercicio 2: Modifique el ejercicio 1 para que cada robot deposite todas juntas las flores en la esquina $(10,10)$.

Ejercicio 3: modifique el ejercicio 1 para que exista un robot jefe que le indica a cada robot de que alto debe realizar el rectángulo y la esquina inferior izquierda donde debe comenzar el rectángulo.