



Taller de Programación



AGENDA



Ambiente CMRE



PROGRAMA CONCURRENTE – AMBIENTE CMRE

R-info | VERSIÓN BETA (release 2.9.5)

Elemento: Flores Avenida: A Calle: A
Cantidad: 0

ROBOTS

Robot	Flores	Papeles	Color
rinfo	0	0	Red

Miniatura...

Código del programa

```
22. Avenida(flores)
23. Pos(PosAv + 1, 1)
24. Avenida(flores) {Avenida 10}
25. Informar('Cantidad', flores)
26. fin
27. variables
28. rinfo: robot1
29. comenzar
30. AsignarArea(rinfo, ciudad)
31. Iniciar(rinfo, 1, 1)
32. fin
33.
34.
```

Información de cada robot

ROBOTS EN EJECUCION

rinfo Pos: (00, 00)
Bolsa Esquina
F P F P
00 00 00 00
Nuevo

Ciudad

Cómo aplica CMRE los conceptos de concurrencia?

Windows taskbar: Buscar, 17°C, 10:12 29/9/2023



PROGRAMA CONCURRENTE – AMBIENTE CMRE

Comunicación

Recursos Compartidos

Sincronización

Heterogeneidad de
procesadores





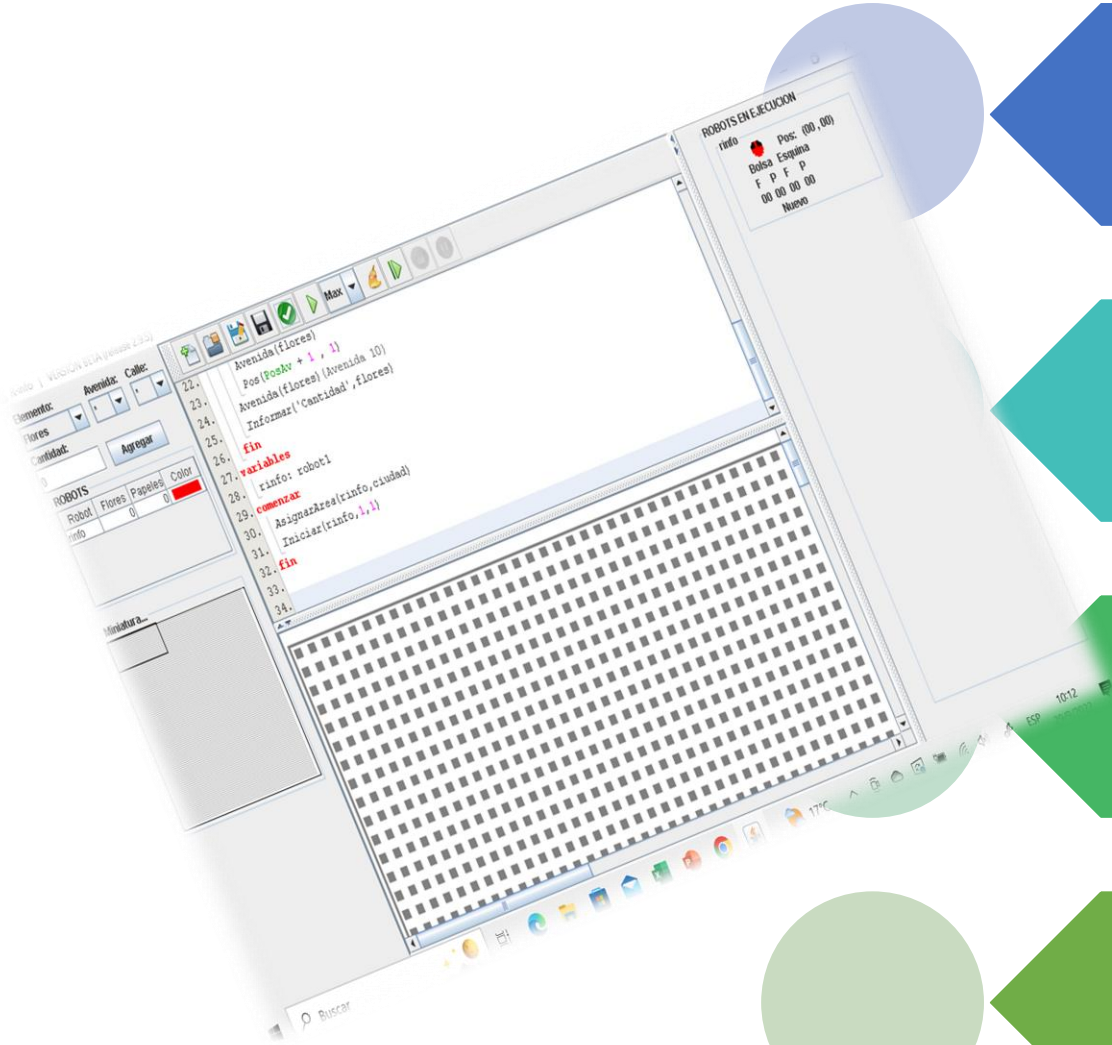
PROGRAMA CONCURRENTE – AMBIENTE CMRE

ROBOTS: se permite declarar más de un robot

AREAS: existen distintos tipos de áreas (privadas, compartidas, parcialmente compartidas)

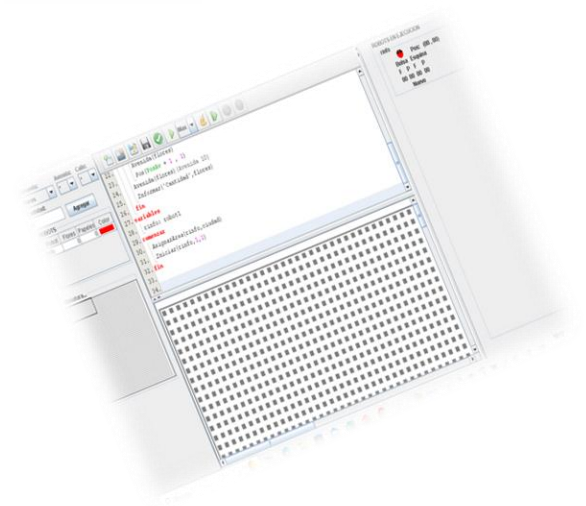
COMUNICACIÓN: permite el intercambio de mensajes entre robots

SINCRONIZACIÓN: permite bloquear y desbloquear recursos compartidos (esquinas)





PROGRAMA CONCURRENTE – AMBIENTE CMRE



programa nombre

procesos

// Procesos utilizados por los robots

areas

// Áreas de la ciudad

robots

// Robots del programa

variables

// Variables robots

comenzar

// Asignación de áreas

// Inicialización de robots

fin



PROGRAMA CONCURRENTE – AMBIENTE CMRE

programa nombre

procesos

// Procesos utilizados por los robots

areas

// Áreas de la ciudad

robots

// Robots del programa

variables

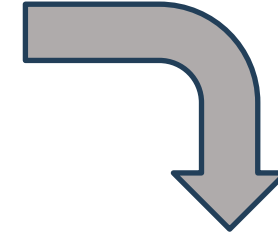
// Variables robots

comenzar

// Asignación de áreas

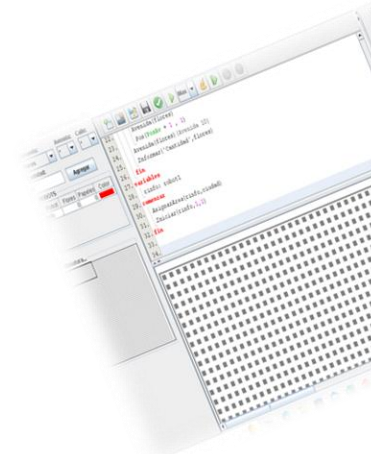
// Inicialización de robots

fin



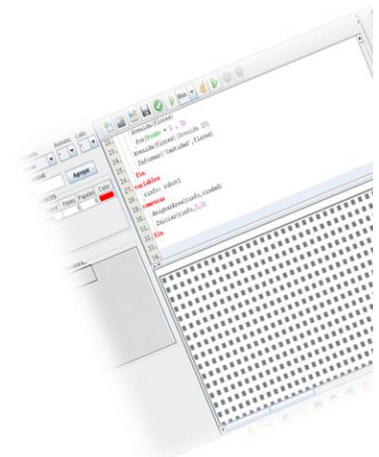
```
proceso nombre (ES flores:numero;E valor:boolean)
variables
    nombre : tipo

comenzar
    //código del proceso
fin
```





PROGRAMA CONCURRENTE – AMBIENTE CMRE



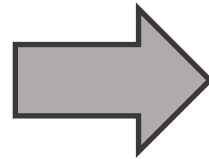
programa nombre

procesos

// Procesos utilizados por los robots

areas

// Áreas de la ciudad



```
ciudad1: areaC(1,1,10,10)
ciudad2: areaP(15,15,20,20)
ciudad3: areaPC(30,32,50,51)
```

robots

// Robots del programa

areaC: área compartida (pueden acceder todos los robots declarados)

variables

// Variables robots

areaP: área privada (pueden acceder sólo un robot de los declarados)

comenzar

// Asignación de áreas

// Inicialización de robots

areaPC: área parcialmente compartida (pueden acceder más de un robot pero no todos los robots declarados)

fin



PROGRAMA CONCURRENTE – AMBIENTE CMRE

R-info | VERSIÓN BETA (release 2.9)

Elemento: Flores Avenida: Calle: Cantidad: 0 Agregar

ROBOTS

Robot	Flores	Papeles	Color
robot1	0	0	Red
robot2	0	0	Blue
robot3	0	0	Magenta

Miniatura...

```
1. programa areasEjemplo
2. areas
3.   area1: AreaC(1,1,10,10)
4.   area2: AreaP(12,1,15,10)
5.   area3: AreaPC(17,1,30,10)
6. robots
7.   robot florero
8.   variables
9.     avenida:numero
10.    calle:numero
11. comenzar
12.   avenida2:=PosAv
13.   calle2:=PosCa
14. fin
15. variables
16. robot1:florero
```

AREA PRIVADA

AREA COMPARTIDA

AREA PARCIALMENTE COMPARTIDA

ROBOTS EN EJECUCION

robot1 Pos: (00,00)
Bolsa Esquina
F P F P
00 00 00 00
Nuevo

robot2 Pos: (00,00)
Bolsa Esquina
F P F P
00 00 00 00
Nuevo

robot3 Pos: (00,00)
Bolsa Esquina
F P F P
00 00 00 00
Nuevo



PROGRAMA CONCURRENTE – AMBIENTE CMRE

programa nombre

procesos

// Procesos utilizados por los robots

areas

// Áreas de la ciudad

robots

// Robots del programa

variables

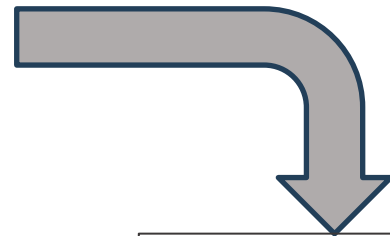
// Variables robots

comenzar

// Asignación de áreas

// Inicialización de robots

fin



```
robot tipo1
variables
...
comenzar
    // Código del robot 1
fin
```



PROGRAMA CONCURRENTE – AMBIENTE CMRE

programa nombre

procesos

// Procesos utilizados por los robots

areas

// Áreas de la ciudad

robots

// Robots del programa

variables

// Variables robots

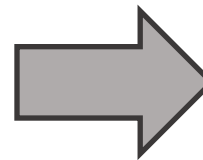
comenzar

// Asignación de áreas

// Inicialización de robots

fin

```
robot tipo1
variables
...
comenzar
    // Código del robot 1
fin
```



```
robot1: tipo1
robot2: tipo1
```

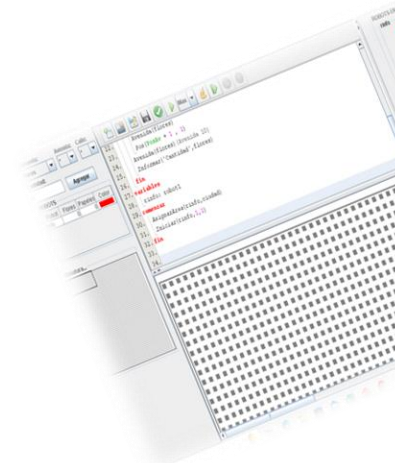
```
robot tipo1
variables
...
comenzar
    // Código del robot 1
fin
```

```
robot tipo2
variables
...
comenzar
    // Código del robot 2
fin
```

```
robot1: tipo1
robot2: tipo2
```



PROGRAMA CONCURRENTE – AMBIENTE CMRE



programa nombre

procesos

// Procesos utilizados por los robots

areas

// Áreas de la ciudad

robots

// Robots del programa

variables

// Variables robots

comenzar

// Asignación de áreas

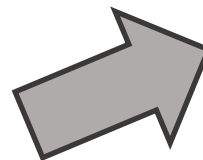
// Inicialización de robots

fin

Todos los robots
declarados deben
estar asignados al
menos a un area

Un robot puede
estar asignado a 1
o más de un áreas
del programa

//AsignarArea(variableRobot,nombreArea)
AsignarArea(robot1,ciudad1)
iniciar(robot1, 5, 5)





EJERCICIOS PARA ANALIZAR

programa Ejercicio-1-a

areas

ciudad : AreaC(1,1,100,100)

robots

robot tipo1

comenzar

Pos (20,20)

fin

robot tipo2

comenzar

Pos(20,20)

fin

variables

robot1: tipo1

robot2: tipo2

comenzar

AsignarArea(robot1,ciudad)

AsignarArea(robot2,ciudad)

Iniciar(robot1, 10, 10)

Iniciar(robot2, 13, 15)

fin

programa Ejercicio-1-b

areas

ciudad : AreaC(1,1,100,100)

robots

robot tipo1

comenzar

mover

Pos (20,20)

fin

robot tipo2

comenzar

repetir 5

mover

Pos(20,20)

fin

variables

robot1: tipo1

robot2: tipo2

comenzar

AsignarArea(robot1,ciudad)

...

fin



EJERCICIOS PARA ANALIZAR

programa Ejercicio-1-c

areas

ciudad : AreaC(1,1,100,100)

robots

robot tipo1

comenzar

mover

Pos (20,20)

Pos(40,40)

fin

robot tipo2

comenzar

repetir 5

mover

Pos(20,20)

fin

variables

robot1: tipo1

robot2: tipo2

comenzar

...

fin

programa Ejercicio 1-d

areas

parte1ciudad : AreaC(1,1,50,50)

parte2ciudad : AreaP(51,51,99,99)

robots

robot tipo1

comenzar

mover

Pos (20,20)

Pos(40,40)

Pos(75,78)

fin

robot tipo2

comenzar

repetir 5

mover

Pos(80,80)

fin

variables

robot1: tipo1

robot2: tipo2

comenzar

AsignarArea(robot1,parte1ciudad)

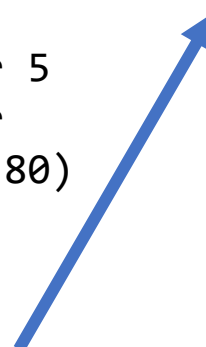
AsignarArea(robot2,parte1ciudad)

AsignarArea(robot2,parte2ciudad)

Iniciar(robot1, 10, 10)

Iniciar(robot2, 62, 62)

fin





EJERCICIOS PARA ANALIZAR

Ejercicio: Realice un programa donde un robot recorra el perímetro de un rectángulo de un tamaño 5 (alto) x 3 (ancho) juntando flores.

Al finalizar el robot debe informar las flores juntadas. Inicialmente el robot se encuentra en la esquina (2,2).

Debe modularizar el rectángulo. El rectángulo debe recibir alto y ancho y devolver las flores.

programa Rectangulos

procesos

proceso juntar (ES flor:numero)

comenzar

mientras (HayFlorEnLaEsquina)

tomarFlor

flor:= flor + 1

fin

proceso rectangulo (E alto:numero;
E ancho:numero;
ES flores:numero)

comenzar

flores:= 0

repetir 2

repetir alto

juntar(flores)

mover

derecha

repetir ancho

juntar(flores)

mover

derecha

fin

areas

ciudad : AreaC(1,1,100,100)

robots

robot tipo1

variables

f:numero

comenzar

rectangulo (5,3,f)

Informar (f)

fin

variables

robot1: tipo1

comenzar

AsignarArea(robot1,ciudad)

Iniciar(robot1, 2, 2)

fin

Modifique el ejercicio para que exista otro robot que comience en (8,8)



EJERCICIOS PARA ANALIZAR

programa Rectangulos

Procesos

proceso juntar (ES flor:numero)

comenzar

 mientras (HayFlorEnLaEsquina)

 tomarFlor

 flor:= flor + 1

fin

proceso rectangulo (E alto:numero;
 E ancho:numero;
 ES flores:numero)

comenzar

 flores:= 0

 repetir 2

 repetir alto

 juntar(flores)

 mover

 derecha

 repetir ancho

 juntar(flores)

 mover

 derecha

fin

areas

 ciudad : AreaC(1,1,100,100)

robots

 robot tipo1

variables

 f:numero

comenzar

 rectangulo (5,3,f)

 Informar (f)

fin

variables

 robot1: tipo1

 robot2: tipo1

comenzar

 AsignarArea(robot1,ciudad)

 Iniciar(robot1, 2, 2)

 AsignarArea(robot2,ciudad)

 Iniciar(robot2, 8, 8)

fin

