



# Taller de Programación



# AGENDA

Pasaje de mensajes - RECEPCION

Ejemplos



# COMUNICACIÓN – Pasaje de Mensajes -CMRE

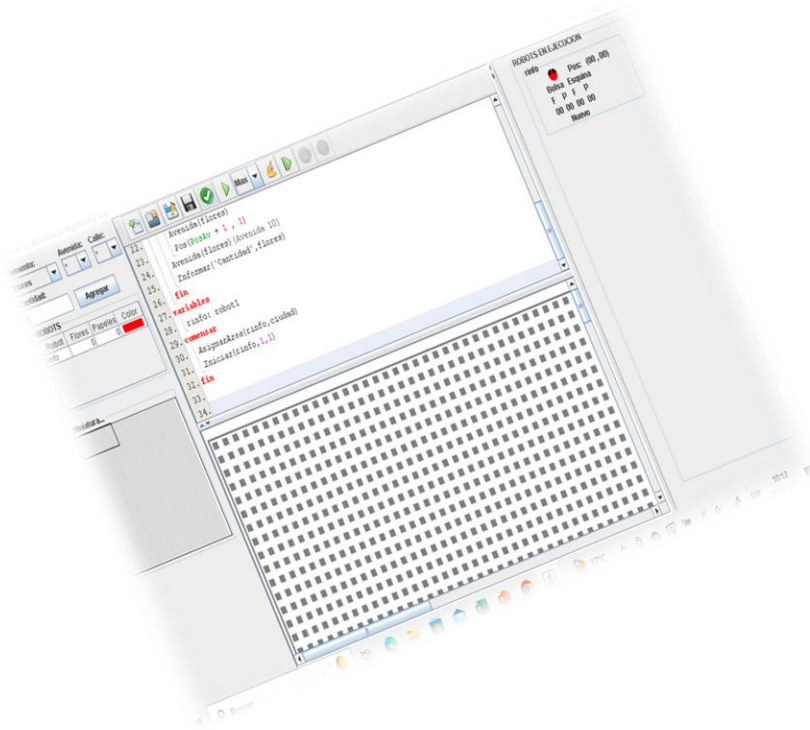
## ENVÍO DE MENSAJES

El envío de mensajes es **asincrónico**, es decir, el robot que envía el mensaje lo hace y sigue procesando sin esperar que el robot receptor lo reciba.

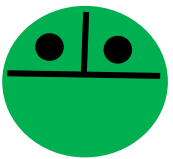
## RECEPCIÓN DE MENSAJES

La recepción de mensajes es **sincrónica**, es decir, el robot que espera un mensaje **NO** sigue procesando hasta que recibe el mensaje.

*Cómo es la sintaxis?*



Instrucción 1



Instrucción 2

Sentencia de comunicación

Instrucción 3



# COMUNICACIÓN – Pasaje de Mensajes - Recepción

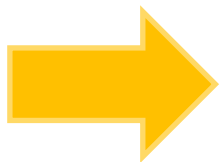
**RecibirMensaje(variable, variableRobot)**

*Cómo  
escribimos el  
programa?*

```
programa ejemploRecepcion
areas
  area1: AreaPC(1,1,20,20)
robots
  robot tipo1
  comenzar
  ...
fin

robot tipo2
  comenzar
  ...
fin
```

```
variables
robot1:tipo1
robot2: tipo1
robot3:tipo2
```



Supongamos que el **robot 1**, y  
el **robot2** quieren recibir un  
mensaje del **robot3**



# COMUNICACIÓN – Pasaje de Mensajes - Recepción

programa envio

**areas**

area1: AreaPC(1,1,20,20)

**Procesos**

....

**robots**

robot tipo1

variables

valor:numero

comenzar

**recibirMensaje (valor, robot3)**

fin

robot tipo2

variables

x:numero

comenzar

x:= 8

**EnviarMensaje (5,robot1)**

**EnviarMensaje (x,robot2)**

**fin**

**variables**

robot1: tipo1

robot2: tipo1

robot3: tipo2

**comenzar**

AsignarArea(robot1,ciudad)

Iniciar(robot1, 2, 2)

AsignarArea(robot2,ciudad)

Iniciar(robot2, 2, 2)

AsignarArea(robot3,ciudad)

Iniciar(robot3, 2, 2)

**fin**



# COMUNICACIÓN – Pasaje de Mensajes - Recepción

## CONSIDERACIONES EN LA RECEPCION DE MENSAJES

La recepción es SIEMPRE sobre una variable (entero o booleano)

La recepción SIEMPRE debe incluir el nombre de una variable robot declarado (no el tipo)

*Se puede recibir  
de cualquier  
robot?*



# COMUNICACIÓN – Pasaje de Mensajes – Recepción

Realizar un programa donde existen **dos robots juntadores**. El robot 1 debe realizar un rectángulo de 5 (alto) x 3 (ancho) juntando flores y el robot 2 un rectángulo de 8 (alto) x 2 (ancho) juntando flores. Luego un tercer robot **jefe** debe informar la cantidad de flores juntadas por cada robot. El tamaño de los rectángulos debe enviárselo el robot jefe a los robots juntadores.

## ROBOT juntadores

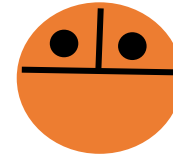


Recibe el alto y ancho del rectángulo

Invoca al proceso rectángulo (le envía el tamaño y le devuelve las flores juntadas)

Envía la cantidad juntada en el rectángulo al **jefe**

## ROBOT jefe



Envía el alto y ancho del rectángulo a cada robot juntador

Recibe la cantidad juntada por el robot 1

Informa la cantidad recibida

Recibe la cantidad juntada por el robot 2

Informa la cantidad recibida



# COMUNICACIÓN – Pasaje de Mensajes – Recepción

```
programa Recepcion
areas
  area1: AreaPC(1,1,20,20)
Procesos
  proceso rectángulo (E alto:numero;
                      E: ancho:numero;
                      ES flores: numero);

  comenzar
  ....
  fin
robots
  robot juntador
  variables
    altura,ancho,f:numero
  comenzar
    RecibirMensaje(altura,robot3)
    RecibirMensaje(ancho,robot3)
    rectángulo (altura,ancho,f)
    EnviarMensaje (f,robot3)
  fin
```

```
robot jefe
variables
  f:numero
comenzar
  EnviarMensaje (5,robot1)
  EnviarMensaje (3,robot1)
  EnviarMensaje (8,robot2)
  EnviarMensaje (2,robot2)
  RecibirMensaje(f,robot1)
  Informar (f)
  RecibirMensaje(f,robot2)
  Informar (f)
fin
```

```
variables
  robot1: juntador
  robot2: juntador
  robot3: jefe
comenzar
  ....
fin
```

¿Qué ocurre si el  
robot 2 termina de  
juntar sus flores  
primero?





# COMUNICACIÓN – Pasaje de Mensajes – Recepción

...

robot jefe

variables

f:numero

comenzar

.....

EnviarMensaje (5,robot1)

EnviarMensaje (3,robot1)

EnviarMensaje (8,robot2)

EnviarMensaje (2,robot2)

**RecibirMensaje(f,\*)**

Informar (f)

**RecibirMensaje(f,\*)**

Informar (f)

fin

variables

robot1:juntadores

robot2: juntadores

robot3: jefe

comenzar

....

fin

**RecibirMensaje(variable,\*)**

Cuando se utiliza \* no implica que en \* esté almacenado el número del robot que hizo el envío



# EJERCICIOS PARA ANALIZAR

Realizar un programa donde existen **dos robots juntadores**. El robot 1 debe realizar un rectángulo de 5 (alto) x 3 (ancho) juntando flores y el robot 2 un rectángulo de 8 (alto) x 2 (ancho) juntando flores. Luego un tercer robot **jefe** debe informar la cantidad de flores **TOTALES** juntadas por ambos robots. El tamaño de los rectángulos debe enviárselo el robot jefe a los robots juntadores.

## ROBOT juntadores

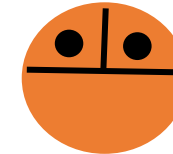


Recibe el alto y ancho del rectángulo

Invoca al proceso rectángulo (le envía el tamaño y le devuelve las flores juntadas)

Envía la cantidad juntada en el rectángulo al **jefe**

## ROBOT jefe



Envía el alto y ancho del rectángulo a cada robot juntador

Repite 2 veces

Recibe la cantidad juntada por algún robot

Suma la cantidad al total

Informa la cantidad recibida



# EJERCICIOS PARA ANALIZAR

```
programa RecepcionIndistinta
```

```
areas
```

```
    area1: AreaPC(1,1,20,20)
```

```
Procesos
```

```
    proceso rectángulo (E alto:numero;  
                        E: ancho:numero;  
                        ES flores: numero);
```

```
        comenzar
```

```
        ....
```

```
    fin
```

```
robots
```

```
    robot juntador
```

```
    variables
```

```
        altura,ancho,f:numero
```

```
    comenzar
```

```
        RecicibirMensaje(altura,robor3)
```

```
        RecibirMensaje(ancho,robot3)
```

```
        rectángulo (altura,ancho,f)
```

```
        EnviarMensaje (f,robot3)
```

```
fin
```

```
robot jefe
```

```
    variables
```

```
        total,f:numero
```

```
    comenzar
```

```
        EnviarMensaje (5,robot1)
```

```
        EnviarMensaje (3,robot1)
```

```
        EnviarMensaje (8,robot2)
```

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

```
        Repetir 2
```

```
            RecibirMensaje(f,*)
```

```
            total:= total + f
```

```
            Informar (total)
```

```
    fin
```

```
variables
```

```
    robot1:juntador
```

```
    robot2: juntador
```

```
    robot3: jefe
```

```
    comenzar
```

```
        ....
```

```
fin
```

*Y si el jefe quiere  
informar que robot  
juntó mas?*



# EJERCICIOS PARA ANALIZAR

```
programa RecepcionIndistinta
areas
  area1: AreaPC(1,1,20,20)
Procesos
  proceso rectángulo (E alto:numero;
                      E: ancho:numero;
                      ES flores: numero);

  comenzar
  ....
  fin
robots
  robot juntador
  variables
    altura,ancho,f:numero
  comenzar
    RecicibirMensaje(altura,robor3)
    RecibirMensaje(ancho,robot3)
    rectángulo (altura,ancho,f)
    EnviarMensaje (f,robot3)
  fin
```

Clase 2 -3- Módulo Concurrente

```
robot jefe
variables
  max,robotMax,f:numero
comenzar
  EnviarMensaje (5,robot1)
  EnviarMensaje (3,robot1)
  EnviarMensaje (8,robot2)
  EnviarMensaje (2,robot2)
  Repetir 2
    RecibirMensaje(f,*)
    si (f > max) entonces
      max:= f
      robotMax:= *
  Informar (robotMax)
  fin
variables
  robot1:juntador
  robot2: juntador
  robot3: jefe
comenzar
  ....
fin
```





# EJERCICIOS PARA ANALIZAR

Realizar un programa donde existen **dos robots juntadores**. El robot 1 debe realizar un rectángulo de 5 (alto) x 3 (ancho) juntando flores y el robot 2 un rectángulo de 8 (alto) x 2 (ancho) juntando flores. Luego un tercer robot **jefe** debe informar que robot juntador juntó la mayor cantidad de flores. El tamaño de los rectángulos debe enviárselo el robot jefe a los robots juntadores.

## ROBOT juntadores

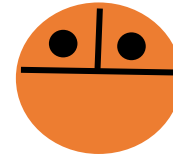


Recibe el alto y ancho del rectángulo

Invoca al proceso rectángulo (le envía el tamaño y le devuelve las flores juntadas)

Envía la cantidad juntada en el rectángulo al **jefe** y **ademas** quien es

## ROBOT jefe



Envía el alto y ancho del rectángulo a cada robot juntador

Repite 2 veces

**Recibe la cantidad juntada por algún robot**

Si la (cantidad es máxima)

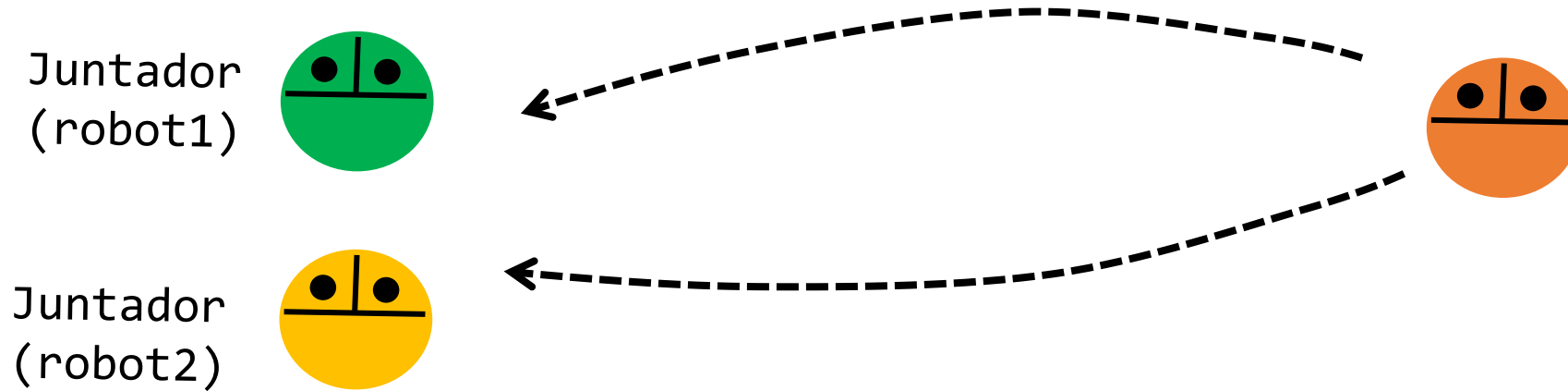
Actualiza el máximo y el número de robot máximo

Informa el robot máximo

Los robots  
juntadores  
**NO conocen**  
su  
**identificación**



# COMUNICACIÓN – Pasaje de Mensajes – Paso 1



Juntador (robot1)

```
RecibirMensaje(quienSoy, robot3)
```

Juntador (robot2)

```
RecibirMensaje(quienSoy, robot3)
```

Jefe (robot3)

```
EnviarMensaje(1, robot1)  
EnviarMensaje(2, robot2)
```

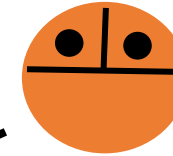
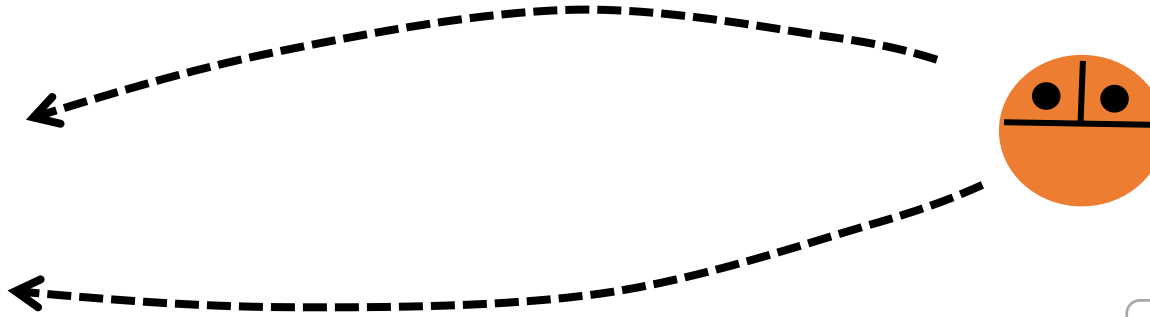


# COMUNICACIÓN – Pasaje de Mensajes – Paso 1

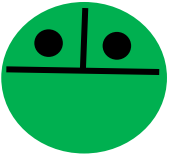
Juntador  
(robot1)



Juntador  
(robot2)

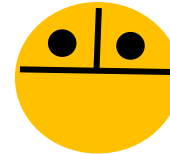


Juntador  
(robot 1)



```
RecibirMensaje(quienSoy, robot3)  
RecibirMensaje(alto, robot3)  
RecibirMensaje(ancho, robot3)  
  
... .
```

Juntador  
(robot2)



```
RecibirMensaje(quienSoy, robot3)  
RecibirMensaje(alto, robot3)  
RecibirMensaje(ancho, robot3)  
  
... .
```

Jefe  
(robot3)

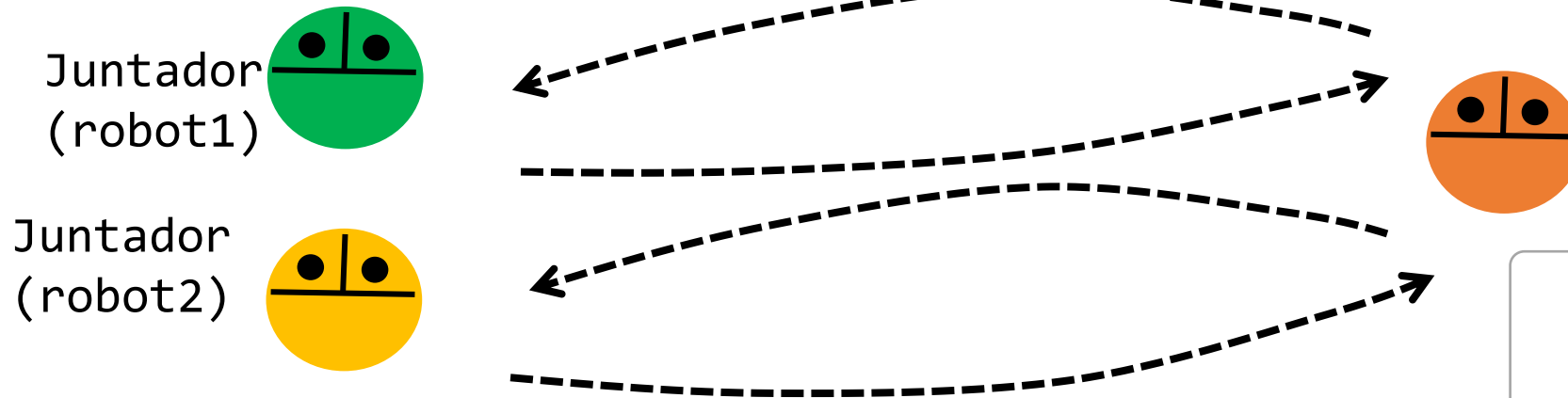


```
EnviarMensaje(1, robot1)  
EnviarMensaje(5, robot1)  
EnviarMensaje(3, robot1)
```

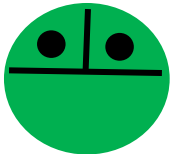
```
EnviarMensaje(2, robot2)  
EnviarMensaje(8, robot2)  
EnviarMensaje(2, robot2)  
  
... .
```



# COMUNICACIÓN – Pasaje de Mensajes – Paso 2



Juntador  
(robot1)



```
RecibirMensaje(quienSoy, robot3)  
....
```

```
EnviarMensaje(quienSoy, robot3)  
EnviarMensaje(flores, robot3)
```

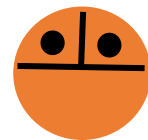
Juntador  
(robot2)



```
RecibirMensaje(quienSoy, robot3)  
....
```

```
EnviarMensaje(quienSoy, robot3)  
EnviarMensaje(flores, robot3)
```

Jefe  
(robot3)



```
EnviarMensaje(1, robot1)  
EnviarMensaje(5, robot1)  
EnviarMensaje(3, robot1)
```

```
EnviarMensaje(2, robot2)  
EnviarMensaje(8, robot2)  
EnviarMensaje(2, robot2)
```

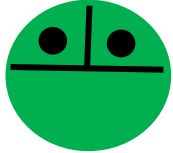
```
RecibirMensaje (quien,*)  
Si (quien = 1)  
    RecibirMensaje (quien,quien)
```





# COMUNICACIÓN – Pasaje de Mensajes – TODO

Juntador  
(robot1)



```
RecibirMensaje(quienSoy, robot3)  
...
```

```
EnviarMensaje(quienSoy, robot3)  
EnviarMensaje(flores, robot3)
```

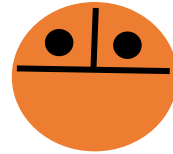
Juntador  
(robot2)



```
RecibirMensaje(quienSoy, robot3)  
...
```

```
EnviarMensaje(quienSoy, robot3)  
EnviarMensaje(flores, robot3)
```

Jefe  
(robot3)



```
EnviarMensaje(1, robot1)  
EnviarMensaje(2, robot2)  
...
```

```
RecibirMensaje(numRobot, *)
```

```
Si numRobot = 2
```

```
    RecibirMensaje(valor, robot2)
```

```
Sino
```

```
    RecibirMensaje(valor, robot1)
```

Jefe  
(robot3)



```
EnviarMensaje(1, robot1)  
EnviarMensaje(2, robot2)
```

```
RecibirMensaje(numRobot, *)
```

```
Si * = 2
```

```
    RecibirMensaje(valor, robot2)
```

Jefe  
(robot3)



```
EnviarMensaje(1, robot1)  
EnviarMensaje(2, robot2)
```

```
RecibirMensaje(numRobot, *)
```

```
RecibirMensaje(valor, *)
```



# EJERCICIOS PARA ANALIZAR

```
programa RecepcionMaximo
areas
  area1: AreaPC(1,1,20,20)
Procesos
  proceso rectángulo (E alto:numero;
                      E: ancho:numero;
                      ES flores: numero);

  comenzar
  ....
  fin
robots
  robot juntador
  variables
    altura,ancho,f,quien: numero
  comenzar
    RecicibirMensaje(quien,robor3)
    RecicibirMensaje(altura,robor3)
    RecibirMensaje(ancho,robot3)
    rectángulo (altura,ancho,f)
    EnviarMensaje (quien,robot3)
    EnviarMensaje (f,robot3)
  fin
```

```
robot jefe
variables
  max,rmax,f:numero
comenzar
  max:=0
  EnviarMensaje (1,robot1)
  EnviarMensaje (5,robot1)
  EnviarMensaje (3,robot1)

  EnviarMensaje (2,robot2)
  EnviarMensaje (8,robot2)
  EnviarMensaje (2,robot2)
  Repetir 2
    RecibirMensaje(quien,*)
    Si(quien = 1)
      RecibirMensaje(f,robot1)
    Sino
      RecibirMensaje(f,robot2)
    Si (f> = max)
      max:= f
      rmax:= quien
  Informar (rmax)
fin
```