

## Trabajo Práctico N° 1: Resolución de Problemas.

### Ejercicio 1.

*Esta noche Juan se encuentra haciendo zapping sabiendo que hay un canal de televisión que está transmitiendo la película “30 años de felicidad”. Luego de terminar de ver la película debe apagar el televisor. Analizar las siguientes soluciones:*

#### Solución 1:

*Encender el televisor.  
Cambiar de canal hasta encontrar la película.  
Ver la película.  
Apagar el televisor.*

#### Solución 2:

*Encender el televisor.  
si (está transmitiendo “30 años de felicidad”)  
ver la película.  
Apagar el televisor.*

#### Solución 3:

*Encender el televisor.  
repetir 20  
    cambiar de canal.  
    Ver la película “30 años de felicidad”.  
Apagar el televisor.*

#### Solución 4:

*Encender el televisor.  
mientras (no se transmite en el canal actual “30 años de felicidad”)  
    cambiar de canal.  
    Ver la película.  
Apagar el televisor.*

**(a)** Comparar las soluciones 1 y 4.

La solución 1 no tiene todos los pasos elementales necesarios y es ambigua, ya que no se especifica el nombre de la película. En cambio, la solución 4 tiene todos los pasos elementales necesarios y no es ambigua, ya que especifica el nombre de la película.

**(b)** Explicar por qué las soluciones 2 y 3 son incorrectas.

Las solución 2 es incorrecta porque sólo analiza la condición de que se esté transmitiendo la película en el canal sincronizado apenas se enciende el televisor y, si no se está transmitiendo, no hay acciones suficientes para resolver el algoritmo, dado que la

precondición es que “hay una canal de televisión que está transmitiendo la película”. La solución 3, en cambio, es incorrecta porque realiza una repetición de 20 veces el proceso, pero presuponiendo que va a encontrar la película; sin embargo, *a priori*, no se sabe si la película se va a encontrar en alguno de esos canales (no es una precondición); si no es encontrada, no hay acciones suficientes para resolver el algoritmo.

**(c)** *¿Qué ocurriría con la solución 4 si ningún canal estuviera transmitiendo la película?*

Si ningún canal estuviera transmitiendo la película, la solución 4 se quedaría iterando al infinito. Para resolver este problema, se debe contar con la evaluación de alguna condición dentro de la iteración que, si es verdadera, la dé por terminada. Como ejemplo, puede considerarse la condición: “si la cantidad de canales cambiados es mayor o igual a la cantidad total de canales, salir de la iteración”. Para esto, se debería contar, por un lado, con una precondición adicional, que es la cantidad total de canales y, por otro lado, también se debería contar con una variable que indique la cantidad de canales cambiados (inicializada en 1 antes de la iteración, dado el canal que se tiene al encender el televisor), que vaya aumentando en una unidad cada vez que el algoritmo ejecuta la iteración.

## Ejercicio 2.

Ud. desea comprar la revista “Crucigramas” que cada mes tiene reservada en el puesto de revistas que se encuentra en la esquina de su casa, al otro lado de la calle. Verificar que no pasen autos antes de cruzar. Indicar, para cada uno de los siguientes algoritmos, si representa la solución a este problema. Justificar la respuesta.

Algoritmo 1:

Caminar hasta la esquina.  
mientras (no pasen autos)  
    Cruzar la calle  
    Comprar la revista “Crucigramas”.

Algoritmo 2:

mientras (no llegue a la esquina)  
    dar un paso  
mientras (pasen autos)  
    esperar 1 segundo  
Cruzar la calle.  
Llegar al puesto de revistas.  
Comprar la revista “Crucigramas”.

Algoritmo 3:

mientras (no llegue a la esquina)  
    dar un paso.  
mientras (pasen autos)  
    esperar 1 segundo  
mientras (no llegue a la otra vereda)  
    dar un paso.  
Llegar al puesto de revistas.  
Comprar la revista “Crucigramas”.

Algoritmo 4:

repetir 10  
    dar un paso.  
Cruzar la calle.  
Llegar al puesto de revistas.  
Comprar la revista “Crucigramas”.

El algoritmo 1 no representa la solución a este problema, ya que es poco claro en cuanto a la acción “caminar hasta la esquina” e, igualmente, en cuanto a la iteración. El algoritmo 2 representa la solución a este problema, pero, sin embargo, sigue sin ser claro respecto a la acción de cómo “cruzar la calle”, se puede descomponer en pasos elementales. El algoritmo 3 también representa la solución a este problema, ya que tiene la mayoría de los pasos elementales necesarios (se podría agregar uno más). El algoritmo 4 no representa la solución a este problema, ya que presupone que, con 10 pasos, va a llegar a la esquina; sin embargo, *a priori*, no se sabe con cuántos pasos llega a la esquina (no es una precondición).

### Ejercicio 3.

*Utilizando las estructuras de control vistas resolver:*

**(a)** *Un algoritmo que, en caso de ser necesario, permita cambiar el filtro de papel de una cafetera. Considerar que se está frente a la cafetera y que se dispone de un filtro suplente.*

Algoritmo 3a:

```
Abrir la cafetera
Mirar el filtro
si (el filtro no se encuentra limpio)
    sacar el filtro de la cafetera
    colocar el filtro suplente en la cafetera
cerrar la cafetera
```

**(b)** *Modificar la solución anterior para que, cuando se encuentre que el filtro de la cafetera está limpio, se guarde el filtro suplente en el lugar correspondiente.*

Algoritmo 3b:

```
Abrir la cafetera
Mirar el filtro
si (el filtro no se encuentra limpio)
    sacar el filtro de la cafetera
    colocar el filtro suplente en la cafetera
sino
    guardar el filtro suplente en el lugar correspondiente
Cerrar la cafetera
```

## Ejercicio 4.

*Escribir un algoritmo que permita trasladar 70 cajas de 30 kilos cada una, desde la Sala A hasta la Sala B. Considerar que sólo se llevará una caja a la vez porque el contenido es muy frágil. Para realizar el trabajo, debe ponerse un traje especial y quitárselo luego de haber realizado el trabajo.*

### Algoritmo 4:

Colocarse traje especial

repetir 70

```
mientras (no llegue a la Sala A)
    dar un paso en dirección a la Sala A
    agarrar una caja en la Sala A
    mientras (no llegue a la Sala B)
        dar un paso en dirección a la Sala B
        dejar una caja en la Sala B
    Quitar traje especial
```

## Ejercicio 5.

*Modificar el algoritmo 4 suponiendo que puede trasladar 60 kilos a la vez.*

### Algoritmo 5:

```
Colocarse traje especial
repetir 35
    mientras (no llegue a la Sala A)
        dar un paso en dirección a la Sala A
    repetir 2
        agarrar una caja en la Sala A
        mientras (no llegue a la Sala B)
            dar un paso en dirección a la Sala B
        repetir 2
            dejar una caja en la Sala B
    Quitarse traje especial
```

## Ejercicio 6.

*Escribir un algoritmo que permita guardar fotos en un álbum familiar. El álbum está compuesto por 150 páginas y se encuentra vacío. En cada página, entran 10 fotos. El álbum se completa por páginas. Una vez que el álbum está completo, debe guardarse en la biblioteca. Considerar que cuenta con fotos suficientes para completar el álbum. Para mayor simplicidad, las páginas se completan de un solo lado.*

### Algoritmo 6a:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
repetir 150  
    repetir 10  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
        pasar de página  
    Guardar el álbum

### Algoritmo 6b:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
repetir 149  
    repetir 10  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
        pasar de página  
    repetir 10  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
Cerrar el álbum  
Guardar el álbum

## Ejercicio 7.

*Modificar el algoritmo anterior si, ahora, no se conoce la cantidad de fotos que entran en una página. Se cuentan con fotos suficientes para completar el álbum.*

### Algoritmo 7a:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
repetir 150  
    mientras (haya lugar en la página)  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
        pasar de página  
    Guardar el álbum

### Algoritmo 7b:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
repetir 149  
    mientras (haya lugar en la página)  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
        pasar de página  
    mientras (haya lugar en la página)  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
    Cerrar el álbum  
Guarda el álbum

## Ejercicio 8.

*Modificar el algoritmo del Ejercicio 6 pero suponiendo, ahora, que no se sabe la cantidad de páginas que tiene el álbum. Se sabe que, en cada página, entran 10 fotos. Se cuentan con fotos suficientes para completar el álbum.*

### Algoritmo 8a:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
mientras (haya páginas vacías)  
    repetir 10  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
        mientras (haya páginas siguientes)  
            cambiar de página  
    Cerrar el álbum  
    Guardar el álbum

### Algoritmo 8b:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
repetir 10  
    agarrar una foto  
    colocar la foto en la página en un espacio vacío  
    mientras (haya páginas vacías)  
        pasar de página  
    repetir 10  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
    Cerrar el álbum  
    Guardar el álbum

## Ejercicio 9.

*Modificar el algoritmo del Ejercicio 6 pero suponiendo, ahora, que no se sabe la cantidad de páginas que tiene el álbum ni la cantidad de fotos que entran en cada página. Se cuentan con fotos suficientes para completar el álbum.*

### Algoritmo 9a:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
mientras (haya lugar en la página)  
    agarrar una foto  
    colocar la foto en la página en un espacio vacío  
    mientras (haya páginas vacías)  
        pasar de página  
        mientras (haya lugar en la página)  
            agarrar una foto  
            colocar la foto en la página en un espacio vacío  
Cerrar el álbum  
Guardar el álbum

### Algoritmo 9b:

Abrir el álbum familiar  
Ubicarse en la primera página del álbum familiar  
mientras (haya páginas siguientes)  
    mientras (haya lugar en la página)  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
        pasar la página  
    mientras (haya lugar en la página)  
        agarrar una foto  
        colocar la foto en la página en un espacio vacío  
Cerrar el álbum  
Guardar el álbum

## Trabajo Práctico N° 2: **Algoritmos y Lógica. Introducción al Lenguaje del Robot.**

### Ejercicio 1.

*Escribir un programa que le permita al robot recoger una flor de la esquina (2,84) si existe.*

```
programa TP2_E1
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (2,84)
        si (HayFlorEnLaEsquina)
            tomarFlor
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 2.

*Escribir un programa que le permita al robot recorrer la calle 50 desde la avenida 65 hasta la avenida 23 depositando un papel en cada esquina. Debe avanzar hasta el final aunque durante el recorrido se quede sin papeles.*

```
programa TP2_E2
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (65,50)
        repetir 3
            derecha
        repetir 42
            si (HayPapelEnLaBolsa)
                depositarPapel
                mover
            si (HayPapelEnLaBolsa)
                depositarPapel
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

### Ejercicio 3.

*Escribir un programa que le permita al robot recorrer el perímetro del cuadrado determinado por (1,1) y (2,2).*

programa TP2\_E3

areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

    comenzar

        repetir 4

            mover

            derecha

    fin

variables

    R-info: robot1

    comenzar

        AsignarArea(R-info,ciudad)

        Iniciar(R-info,1,1)

    fin

## Ejercicio 4.

Modificar el Ejercicio 3 para que, además, recoja, de ser posible, un papel en cada esquina.

```
programa TP2_E4
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
comenzar
    repetir 4
        si (HayPapelEnLaEsquina)
            tomarPapel
        mover
        derecha
    fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 5.

*Escribir un programa que le permita al robot dejar todas las flores que lleva en su bolsa en la esquina (50,50).*

programa TP2\_E5

areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

comenzar

    Pos (50,50)

    mientras (HayFlorEnLaBolsa)

        depositarFlor

    fin

variables

    R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

## Ejercicio 6.

*Escribir un programa que le permita al robot recorrer la avenida 75 desde la calle 45 hasta la calle 15 recogiendo todas las flores que encuentre.*

programa TP2\_E6

areas

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

robots

robot robot1

comenzar

Pos (75,45)

repetir 2

derecha

repetir 30

mientras (HayFlorEnLaEsquina)

    tomarFlor

    mover

    mientras (HayFlorEnLaEsquina)

        tomarFlor

    fin

variables

R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

## Ejercicio 7.

*Escribir un programa que le permita al robot recorrer la avenida 10, depositando una flor en cada esquina. Si, en algún momento del recorrido, se queda sin flores en la bolsa, debe seguir caminando (sin depositar) hasta terminar la avenida.*

```
programa TP2_E7
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (10,1)
        repetir 99
            si (HayFlorEnLaBolsa)
                depositarFlor
                mover
            si (HayFlorEnLaBolsa)
                depositarFlor
            fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 8.

*Escribir un programa que le permita al robot recorrer la avenida 23 buscando una esquina sin papeles que seguro existe. Al encontrarla, debe depositar, en esa esquina, todos los papeles que lleva en su bolsa. Informar en qué calle dejó los papeles.*

```
programa TP2_E8
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (23,1)
        mientras (HayPapelEnLaEsquina)
            mover
            mientras (HayPapelEnLaBolsa)
                depositarPapel
                Informar (PosCa)
            fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 9.

*Escribir un programa que le permita al robot recorrer la calle 17 depositando un papel en las avenidas impares. El recorrido termina cuando el robot llega a la esquina (100,17).*

```
programa TP2_E9
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (1,17)
        derecha
        repetir 49
            si (HayPapelEnLaBolsa)
                depositarPapel
            repetir 2
                mover
            si (HayPapelEnLaBolsa)
                depositarPapel
            mover
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 10.

*Programar al robot para que recorra las 5 primeras avenidas juntando, en cada esquina, todas las flores y papeles.*

```
programa TP2_E10
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
comenzar
    repetir 4
        repetir 99
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
            mover
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
            Pos (PosAv+1,1)
        repetir 99
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
            mover
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 11.

*Programar al robot para que recorra el perímetro de la ciudad recogiendo todas las flores y papeles que encuentre y dejando, en cada vértice, sólo un papel. Puede ocurrir que, algún vértice, quede vacío si el robot no tiene papeles en su bolsa para depositar.*

```
programa TP2_E11
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        repetir 4
            repetir 99
                mover
                mientras (HayFlorEnLaEsquina)
                    tomarFlor
                mientras (HayPapelEnLaEsquina)
                    tomarPapel
                si (HayPapelEnLaBolsa)
                    depositarPapel
                derecha
            fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 12.

*Programar al robot para que recorra todas las calles depositando, en cada esquina vacía, un papel. En caso de no tener más papeles, debe continuar el recorrido (sin depositar).*

```
programa TP2_E12
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        derecha
        repetir 99
            repetir 99
                si (~HayFlorEnLaEsquina & ~HayFlorEnLaEsquina & HayPapelEnLaBolsa)
                    depositarPapel
                mover
                si (~HayFlorEnLaEsquina & ~HayFlorEnLaEsquina & HayPapelEnLaBolsa)
                    depositarPapel
                Pos (1,PosCa+1)
            repetir 99
                si (~HayFlorEnLaEsquina & ~HayFlorEnLaEsquina & HayPapelEnLaBolsa)
                    depositarPapel
                mover
                si (~HayFlorEnLaEsquina & ~HayFlorEnLaEsquina & HayPapelEnLaBolsa)
                    depositarPapel
    fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Trabajo Práctico N° 3: Datos.

### Ejercicio 1.

Indicar qué hacen los siguientes programas considerando las diferentes situaciones que podrían presentarse:

- (a) i. todas las esquinas de la avenida 6 tienen, al menos, 1 flor.  
ii. sólo la esquina (6,20) tiene flor.  
iii. ninguna esquina de la avenida 6 tiene flor.

programa TP3\_E1a  
areas  
ciudad: AreaC(1,1,100,100)  
robots  
robot robot1  
comenzar  
Pos (6,1)  
mientras (HayFlorEnLaEsquina & (PosCa<100))  
    mover  
    tomarFlor  
fin  
variables  
R-info: robot1  
comenzar  
AsignarArea(R-info,ciudad)  
Iniciar(R-info,1,1)  
fin

- i. Si todas las esquinas de la avenida 6 tienen, al menos, 1 flor, este programa analiza si hay flor en la esquina (6,1), se mueve a la calle siguiente (6,2) y toma una flor de esta esquina. Luego, esta iteración continúa mientras que haya flor en la esquina y el robot no se encuentre en la calle 100, lo cual va a dejar de suceder cuando, en alguna esquina, haya sólo una flor o se encuentre en la calle 100.
- ii. Si sólo la esquina (6,20) tiene flor, el robot nunca se mueve de la esquina (6,1), ya que, en esa posición, no se cumple la condición de que haya flor en la esquina.
- iii. Si ninguna esquina de la avenida 6 tiene flor, el robot nunca se mueve de la esquina (6,1), ya que, en esa posición, no se cumple la condición de que haya flor en la esquina.

- (b) i. todas las esquinas de la avenida tienen, al menos, 1 flor y 1 papel.  
ii. sólo la esquina (6,20) tiene flor y ningún papel, las demás están vacías.  
iii. sólo la esquina (6,20) tiene papel y no tiene ninguna flor, las demás están vacías.  
iv. ninguna esquina de la avenida 1 tiene flor ni papel.

programa TP3\_E1b  
areas  
ciudad: AreaC(1,1,100,100)  
robots  
robot robot1  
variables  
nro: numero  
comenzar  
nro:=0  
repetir 10  
  si (~ (HayFlorEnLaEsquina | HayPapelEnLaEsquina))  
    mover  
    nro:=nro+1  
  Informar (nro)  
fin  
variables  
R-info: robot1  
comenzar  
AsignarArea(R-info,ciudad)  
Iniciar(R-info,1,1)  
fin

- i. Si todas las esquinas de la avenida tienen, al menos, 1 flor y 1 papel, este programa informa nro= 0.
- ii. Si sólo la esquina (6,20) tiene flor y ningún papel y las demás están vacías, este programa informa nro= 10.
- iii. Si sólo la esquina (6,20) tiene papel y no tiene ninguna flor y las demás están vacías, este programa informa nro= 10.
- iv. Si ninguna esquina de la avenida 1 tiene flor ni papel, este programa informa nro= 10.

## Ejercicio 2.

*Programar al robot para que informe la cantidad de flores que hay en la calle 44.*

**(a) Recogiendo todas las flores.**

```
programa TP3_E2a
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    variables
        flores: numero
    comenzar
        flores:=0
        Pos (1,44)
        derecha
        repetir 99
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                flores:=flores+1
                mover
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                flores:=flores+1
            Informar (flores)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

**(b) Sin modificar el contenido de cada esquina.**

```
programa TP3_E2b
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    variables
        flores_calle, flores_esq: numero
    comenzar
        flores_calle:=0
        Pos (1,44)
        derecha
```

```
repetir 99
flores_esq:=0
mientras (HayFlorEnLaEsquina)
    tomarFlor
    flores_esq:=flores_esq+1
repetir flores_esq
    depositarFlor
    flores_calle:=flores_calle+flores_esq
    mover
    flores_esq:=0
mientras (HayFlorEnLaEsquina)
    tomarFlor
    flores_esq:=flores_esq+1
repetir flores_esq
    depositarFlor
    flores_calle:=flores_calle+flores_esq
    Informar (flores_calle)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

### Ejercicio 3.

*Programar al robot para que informe la cantidad de esquinas vacías que hay en la ciudad.*

```
programa TP3_E3
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
variables
    esquinas: numero
comenzar
    esquinas:=0
    repetir 99
        repetir 99
            si (~HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)
                esquinas:=esquinas+1
            mover
            si (~HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)
                esquinas:=esquinas+1
            Pos (PosAv+1,1)
            repetir 99
                si (~HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)
                    esquinas:=esquinas+1
                mover
                si (~HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)
                    esquinas:=esquinas+1
            Informar (esquinas)
fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 4.

*Escribir un programa que le permita al robot caminar por la calle 7 hasta encontrar 20 flores. Hay como máximo una flor por esquina. Seguro existen 20 flores.*

programa TP3\_E4

areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

variables

    flores: numero

comenzar

    flores:=0

    Pos (1,7)

    derecha

    mientras (flores<20)

        si (HayFlorEnLaEsquina)

            flores:=flores+1

        si (flores<20)

            mover

    fin

variables

    R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

## Ejercicio 5.

*Escribir un programa que le permita al robot caminar por la calle 7 hasta encontrar 20 flores. Hay como máximo una flor por esquina. Puede no haber 20 flores.*

programa TP3\_E5  
areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

variables

    flores: numero

comenzar

    flores:=0

    Pos (1,7)

    derecha

    mientras ((flores<20) & (PosAv<100))

        si (HayFlorEnLaEsquina)

            flores:=flores+1

        si (flores<20)

            mover

        si (HayFlorEnLaEsquina)

            flores:=flores+1

    fin

variables

    R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

## Ejercicio 6.

*Escribir un programa que le permita al robot caminar por la calle 7 hasta encontrar 20 flores. Puede haber más de una flor por esquina. Seguro existen 20 flores.*

programa TP3\_E6

areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

variables

    flores: numero

comenzar

    flores:=0

    Pos (1,7)

    derecha

    mientras (flores<20)

        mientras (HayFlorEnLaEsquina)

            tomarFlor

            flores:=flores+1

        si (flores<20)

            mover

    fin

variables

    R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

## Ejercicio 7.

*El robot debe limpiar de papeles la calle 34. Al terminar el recorrido debe informar cuantas esquinas tenían, originalmente, exactamente, 6 papeles.*

programa TP3\_E7

areas

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

robots

robot robot1

variables

esquinas, papeles: numero

comenzar

esquinas:=0

Pos (1,34)

derecha

repetir 99

papeles:=0

mientras (HayPapelEnLaEsquina)

tomarPapel

papeles:=papeles+1

si (papeles=6)

esquinas:=esquinas+1

mover

papeles:=0

mientras (HayPapelEnLaEsquina)

tomarPapel

papeles:=papeles+1

si (papeles=6)

esquinas:=esquinas+1

Informar (esquinas)

fin

variables

R-info: robot1

comenzar

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

fin

## Ejercicio 8.

Programar al robot para que recorra la calle 2 hasta encontrar, al menos, 10 papeles. Puede no haber 10 papeles.

programa TP3\_E8

areas

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

robots

robot robot1

variables

papeles: numero

comenzar

papeles:=0

Pos (1,2)

derecha

mientras ((papeles<10) & (PosAv<100))

  mientras (HayPapelEnLaEsquina & (papeles<10))

    tomarPapel

    papeles:=papeles+1

  mover

  mientras (HayPapelEnLaEsquina & (papeles<10))

    tomarPapel

    papeles:=papeles+1

fin

variables

R-info: robot1

comenzar

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

fin

## Ejercicio 9.

Programar al robot para que recorra la calle 2 hasta encontrar 10 papeles y 4 flores. Seguro existen dichas cantidades.

```
programa TP3_E9
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
variables
    papeles, flores: numero
comenzar
    papeles:=0
    flores:=0
    Pos (1,2)
    derecha
    mientras ((papeles<10) | (flores<4))
        mientras (HayPapelEnLaEsquina & (papeles<10))
            tomarPapel
            papeles:=papeles+1
        mientras (HayFlorEnLaEsquina & (flores<4))
            tomarFlor
            flores:=flores+1
        si ((papeles<10) | (flores<4))
            mover
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 10.

*Programar al robot para que recorra el perímetro de la ciudad e informe la cantidad de papeles recogidos en cada lado.*

Suponiendo que cada vértice corresponde sólo a un lado:

```
programa TP3_E10
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
variables
    papeles, papeles_vertice: numero
comenzar
    repetir 4
        papeles:=0
        repetir 99
            mientras (HayPapelEnLaEsquina)
                tomarPapel
                papeles:=papeles+1
            mover
            derecha
            Informar (papeles)
fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

Suponiendo que cada vértice corresponde a dos lados:

```
programa TP3_E10
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
papeles, papeles_vertice: numero
comenzar
papeles:=0
mientras (HayPapelEnLaEsquina)
    tomarPapel
    papeles:=papeles+1
    repetir papeles
        depositarPapel
    repetir 98
        mover
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
        papeles_vertice:=0
        mover
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
            papeles_vertice:=papeles_vertice+1
        repetir papeles_vertice
            depositarPapel
        derecha
        Informar (papeles)
    repetir 2
        papeles:=0
    repetir 99
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
            mover
        papeles_vertice:=0
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
            papeles_vertice:=papeles_vertice+1
        repetir papeles_vertice
            depositarPapel
        derecha
        Informar (papeles)
    papeles:=0
    repetir 99
        mientras (HayPapelEnLaEsquina)
```

```
tomarPapel
papeles:=papeles+1
mover
mientras (HayPapelEnLaEsquina)
    tomarPapel
    papeles:=papeles+1
    derecha
    Informar (papeles)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Trabajo Práctico N° 4: Repasso.

### Ejercicio 1.

*Indicar qué hacen los siguientes programas considerando las diferentes situaciones que podrían presentarse:*

(a)

```
programa TP4_E1a
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
comenzar
    Pos (4,3)
    si (HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)
        tomarFlor
        Informar (V)
    sino
        Informar (F)
fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

Este programa, si hay flor en la esquina y no hay papel en la esquina, toma flor e informa V, sino informa F.

(b)

```
programa TP4_E1b
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
comenzar
    Pos (6,1)
    mientras (HayFlorEnLaEsquina & (PosCa<100))
        mover
        tomarFlor
fin
```

**variables**

R-info: robot1

**comenzar**

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

**fin**

Este programa se posiciona en (6,1) y, mientras hay flor en la esquina y se encuentra en una calle menor a 100, se mueve y toma flor; cuando esto deja de ser cierto, deja de realizar estas acciones.

**(c)**

**programa TP4\_E1c**

**areas**

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

**robots**

robot robot1

**comenzar**

repetir 99

mientras (HayFlorEnLaEsquina)

    tomarFlor

    mover

    mientras (HayFlorEnLaEsquina)

        tomarFlor

**fin**

**variables**

R-info: robot1

**comenzar**

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

**fin**

Este programa recorre toda la avenida 1 y, mientras hay flor en las esquinas correspondientes, toma flor.

**(d)**

**programa TP4\_E1d**

**areas**

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

**robots**

robot robot1

**variables**

nro: numero

**comenzar**

```
nro:=0
si (~ (HayFlorEnLaEsquina | HayPapelEnLaEsquina))
    mover
    nro:=nro+1
    Informar (nro)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

Este programa, si no hay flor ni papel en la esquina, se mueve y suma 1 en la variable nro; si no es cierto, informa el valor de la variable nro.

(e)

```
programa TP4_E1e
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
SinFlor: numero
comenzar
SinFlor:=0
Pos (1,20)
derecha
mientras (HayFlorEnLaEsquina & (PosAv<100))
    tomarFlor
    si (~HayFlorEnLaEsquina)
        SinFlor:=SinFlor+1
    mover
    Informar (SinFlor)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

Este programa se posiciona en (1,20), gira a la derecha y, mientras hay flor en la esquina y se encuentra en una calle menor a 100, toma flor y, si ahora no hay flor en la esquina, suma 1 en la variable SinFlor y, luego, se mueve; cuando esto deja de ser cierto, informa el valor de la variable SinFlor.

(f)

```
programa TP4_E1f
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
variables
    cant: numero
comenzar
    cant:=0
    mientras (HayFlorEnLaEsquina & HayPapelEnLaEsquina)
        tomarFlor
        tomarPapel
        cant:=cant+1
    fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

Este programa, mientras hay flor y papel en la esquina, toma flor y papel y suma 1 en la variable cant; cuando esto de ser cierto, deja de realizar estas acciones.

## Ejercicio 2.

*Programar al robot para que recorra la calle 3 desde la avenida 5 hasta la avenida 20 depositando un papel en cada esquina. Si durante el recorrido se queda sin papeles para depositar, debe detenerse.*

```
programa TP4_E2
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (5,3)
        derecha
        repetir 15
            si (HayPapelEnLaBolsa)
                depositarPapel
                mover
            fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

### Ejercicio 3.

Suponiendo que el robot cuenta con suficiente cantidad de flores y papeles en su bolsa, escribir un programa que le permita recorrer la calle 45 dejando en las avenidas pares sólo una flor y en las impares sólo un papel.

```
programa TP4_E3
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        Pos (1,45)
        derecha
        repetir 49
            depositarPapel
            mover
            depositarFlor
            mover
            depositarPapel
            mover
            depositarFlor
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 4.

Programar al robot para que recorra la calle 20 e informe cuántas esquinas tienen sólo flores y cuántas esquinas tienen sólo papeles.

programa TP4\_E4

areas

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

robots

robot robot1

variables

flores, papeles: numero

comenzar

Pos (1,20)

derecha

repetir 99

  si (HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)

    flores:=flores+1

  si (~HayFlorEnLaEsquina & HayPapelEnLaEsquina)

    papeles:=papeles+1

  mover

  si (HayFlorEnLaEsquina & ~HayPapelEnLaEsquina)

    flores:=flores+1

  si (~HayFlorEnLaEsquina & HayPapelEnLaEsquina)

    papeles:=papeles+1

  Informar (flores,papeles)

fin

variables

R-info: robot1

comenzar

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

fin

## Ejercicio 5.

*Programar al robot para que recorra el perímetro de la ciudad dejando un papel en aquellas esquinas que sólo tienen papel y una flor en las esquinas que tienen sólo flores. El recorrido debe finalizar al terminar de recorrer el perímetro.*

```
programa TP4_E5
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
    repetir 4
        repetir 99
            si (HayFlorEnLaEsquina & ~HayPapelEnLaEsquina & HayFlorEnLaBolsa)
                depositarFlor
            si (~HayFlorEnLaEsquina & HayPapelEnLaEsquina & HayPapelEnLaBolsa)
                depositarPapel
            mover
            derecha
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 6.

Programar al robot para que recorra el perímetro de la ciudad buscando una esquina con, exactamente, 3 flores y 3 papeles, suponiendo que esta esquina existe. Debe informar cuál es la esquina encontrada.

```
programa TP4_E6
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
variables
    flores, papeles, pasos, lados: numero
comenzar
    flores:=0
    papeles:=0
    lados:=1
    mientras ((~ ((flores=3) & (papeles=3))) & (lados<5))
        pasos:=1
        lados:=lados+1
        mientras ((~ ((flores=3) & (papeles=3))) & (pasos<100))
            flores:=0
            papeles:=0
            pasos:=pasos+1
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                flores:=flores+1
            mientras (HayPapelEnLaEsquina)
                tomarPapel
                papeles:=papeles+1
            si ((flores=3) & (papeles=3))
                Informar (PosAv,PosCa)
            sino
                mover
            si (~ ((flores=3) & (papeles=3)))
                derecha
fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 7.

*Ídem 6. pero no se puede asegurar que tal esquina existe. En caso de encontrarla, informar cuál es esa esquina.*

programa TP4\_E7

areas

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

robots

robot robot1

variables

flores, papeles: numero

esquina: boolean

comenzar

esquina:=F

repetir 4

si (esquina=F)

repetir 99

si (esquina=F)

flores:=0

papeles:=0

mientras (HayFlorEnLaEsquina)

tomarFlor

flores:=flores+1

mientras (HayPapelEnLaEsquina)

tomarPapel

papeles:=papeles+1

si ((flores=3) & (papeles=3))

esquina:=V

Informar (PosAv,PosCa)

si (esquina=F)

mover

si (esquina=F)

derecha

fin

variables

R-info: robot1

comenzar

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

fin

## Ejercicio 8.

Indicar si son verdaderas o falsas las siguientes afirmaciones de acuerdo al programa “ejercicio8”. Justificar cada respuesta.

```
programa TP4_E8
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
comenzar
    repetir 5
        mover
        derecha
        mientras ((HayFlorEnLaEsquina | HayPapelEnLaEsquina) & (PosAv<100))
            mover
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
        fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

(a) *Se puede asegurar que el robot pasará por la esquina (10,6).*

Esta afirmación es FALSA.

(b) *Se puede asegurar que el robot pasará por la esquina (1,6).*

Esta afirmación es VERDADERA.

(c) *El robot se puede caer de la ciudad.*

Esta afirmación es FALSA.

(d) *En todas las esquinas por las que pasó el robot, hay flores o papeles.*

Esta afirmación es FALSA.

(e) *Al detenerse, se puede asegurar que el robot levantará flores y papeles.*

Esta afirmación es FALSA.

(f) *Al finalizar el recorrido, el robot tiene flores y papeles en la bolsa.*

Esta afirmación es FALSA.

## **Trabajo Práctico N° 5:** **Programación Estructurada.**

### **Ejercicio 1.**

*Escribir un proceso que le permita al robot realizar un cuadrado de lado 2 girando en la dirección de las agujas del reloj.*

```
programa TP5_E1
procesos
    proceso cuadrado
    comenzar
        repetir 4
            repetir 2
                mover
                derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            cuadrado
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

## Ejercicio 2.

Utilizar el proceso desarrollado en 1 para realizar un programa para cada uno de los recorridos de la figura 5.9.

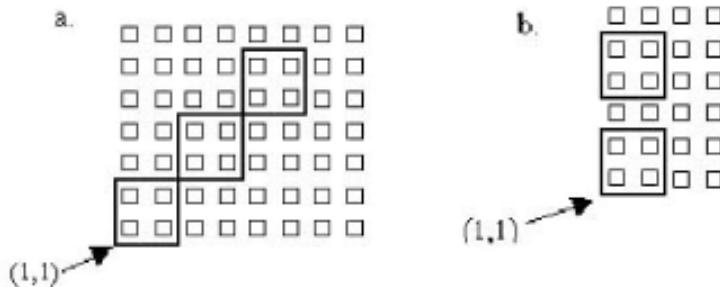


Figura 5.9: Recorridos usando cuadrados de lado 2

(a)

```
programa TP5_E2a
procesos
    proceso cuadrado
    comenzar
        repetir 4
            repetir 2
                mover
                derecha
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            repetir 3
                cuadrado
                Pos(PosAv+2,PosCa+2)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

(b)

```
programa TP5_E2b
procesos
```

```
proceso cuadrado
comenzar
    repetir 4
        repetir 2
            mover
            derecha
    fin
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
        repetir 2
            cuadrado
            Pos (PosAv,PosCa+3)
    fin
variables
    R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

### Ejercicio 3.

*Escribir un proceso que le permita al robot realizar un rectángulo de base 5 y altura 3 girando en la dirección de las agujas del reloj a partir de la posición (1,1).*

```
programa TP5_E3
procesos
    proceso rectangulo
        comenzar
            repetir 2
                repetir 3
                    mover
                    derecha
                repetir 5
                    mover
                    derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            comenzar
                rectangulo
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

## Ejercicio 4.

Programar al robot para que realice los recorridos de la figura 5.10 utilizando el proceso desarrollado en 3.

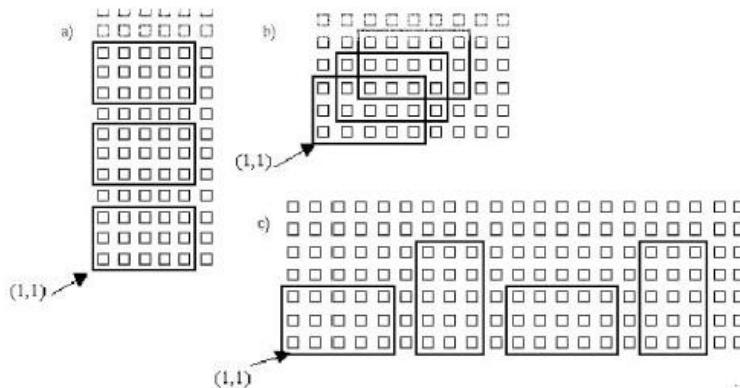


Figura 5.10: Recorridos usando rectángulos de 5x3.

(a)

```
programa TP5_E4a
procesos
    proceso rectangulo
    comenzar
        repetir 2
            repetir 3
                mover
                derecha
            repetir 5
                mover
                derecha
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    comenzar
        repetir 3
            rectangulo
            Pos (PosAv,PosCa+4)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

(b)

```
programa TP5_E4b
procesos
    proceso rectangulo
    comenzar
        repetir 2
            repetir 3
                mover
                derecha
            repetir 5
                mover
                derecha
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            repetir 3
                rectangulo
                Pos (PosAv+1,PosCa+1)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

(c)

```
programa TP5_E4c
procesos
    proceso rectangulo
    comenzar
        repetir 2
            repetir 3
                mover
                derecha
            repetir 5
                mover
                derecha
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
```

```
robot robot1
comenzar
    repetir 2
        rectangulo
        Pos (PosAv+10,1)
        Pos (7,6)
        derecha
    repetir 2
        rectangulo
        Pos (PosAv+10,6)
    fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 5.

Rehacer el recorrido del Ejercicio 4.c trasladando los papeles de cada esquina correspondientes a un lado del rectángulo al vértice siguiente en el recorrido. Por ejemplo, para el rectángulo con vértice en (1,1), los papeles de (1,2) y (1,3) deben ser trasladados a (1,4); los de la calle 4 entre las avenidas 2 y 5 deben ser reubicados en (6,4); y así siguiendo.

```
programa TP5_E5
procesos
    proceso tomar_papeles
        comenzar
            mientras (HayPapelEnLaEsquina)
                tomarPapel
            fin
    proceso depositar_papeles
        comenzar
            mientras (HayPapelEnLaBolsa)
                depositarPapel
            fin
    proceso rectangulo
        comenzar
            repetir 2
                repetir 2
                    mover
                    tomar_papeles
                mover
                depositar_papeles
            derecha
            repetir 4
                mover
                tomar_papeles
            mover
            depositar_papeles
            derecha
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            repetir 2
                rectangulo
                Pos (PosAv+10,1)
            Pos (7,6)
            derecha
            repetir 2
                rectangulo
                Pos (PosAv+10,6)
```

```
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 6.

(a) Escribir un proceso que le permita al robot realizar un rectángulo de base 5 y altura 3 girando en la dirección contraria a la de las agujas del reloj.

programa TP5\_E6a

procesos

    proceso izquierda

        comenzar

            repetir 3

                derecha

            fin

    proceso rectangulo

        comenzar

            derecha

            repetir 2

                repetir 5

                    mover

                    izquierda

                repetir 3

                    mover

                    izquierda

                izquierda

            fin

areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

    comenzar

        rectangulo

    fin

variables

    R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

(b) Indicar si se produce alguna modificación en los procesos de los Ejercicios 4 y 5 si se reemplaza el módulo realizado en 3 por el implementado en 6.a.

programa TP5\_E6b

procesos

    proceso izquierda

        comenzar

            repetir 3

                derecha

```
fin
proceso tomar_papeles
comenzar
    mientras (HayPapelEnLaEsquina)
        tomarPapel
    fin
proceso depositar_papeles
comenzar
    mientras (HayPapelEnLaBolsa)
        depositarPapel
    fin
proceso rectangulo
comenzar
    derecha
    repetir 2
        repetir 5
            mover
            tomar_papeles
        izquierda
        depositar_papeles
    repetir 3
        mover
        tomar_papeles
    izquierda
    depositar_papeles
    izquierda
fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
comenzar
    repetir 2
        rectangulo
        Pos (PosAv+10,1)
    Pos (7,6)
    derecha
    repetir 2
        rectangulo
        Pos (PosAv+10,6)
fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

## Ejercicio 7.

(a) Escribir el proceso LimpiarEsquina que le permita al robot recoger todas las flores y todos los papeles de la esquina donde se encuentra parado.

```
programa TP5_E7a
procesos
    proceso LimpiarEsquina
        comenzar
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            comenzar
                LimpiarEsquina
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

(b) Escribir un programa que le permita al robot recoger todas las flores y papeles de la avenida 89, utilizando los procesos implementados en 7.a.

```
programa TP5_E7b
procesos
    proceso LimpiarEsquina
        comenzar
            mientras (HayFlorEnLaEsquina)
                tomarFlor
            mientras (HayPapelEnLaEsquina)
                tomarPapel
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            comenzar
                Pos (89,1)
                repetir 99
```

```
LimpiarEsquina
mover
LimpiarEsquina
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

(c) Modificar el proceso 6.a para que el robot realice el rectángulo indicado dejando a su paso todas las esquinas vacías. Para hacerlo, debe utilizar el proceso LimpiarEsquina.

```
programa TP5_E7c
procesos
    proceso izquierda
    comenzar
        repetir 3
            derecha
        fin
    proceso LimpiarEsquina
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
        mientras (HayPapelEnLaEsquina)
            tomarPapel
        fin
    proceso rectangulo
    comenzar
        derecha
        repetir 2
            repetir 5
                mover
                LimpiarEsquina
            izquierda
            repetir 3
                mover
                LimpiarEsquina
            izquierda
            izquierda
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            rectangulo
```

```
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

(d) Rehacer el recorrido 4.b utilizando el proceso definido en 7.c.

```
programa TP5_E7d
procesos
    proceso izquierda
    comenzar
        repetir 3
            derecha
        fin
    proceso LimpiarEsquina
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
        mientras (HayPapelEnLaEsquina)
            tomarPapel
        fin
    proceso rectangulo
    comenzar
        derecha
        repetir 2
            repetir 5
                mover
                LimpiarEsquina
            izquierda
            repetir 3
                mover
                LimpiarEsquina
            izquierda
            izquierda
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            repetir 3
                rectangulo
                Pos (PosAv+1,PosCa+1)
            fin
    variables
```

R-info: robot1

**comenzar**

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

**fin**

## Ejercicio 8.

Programar al robot para que recorra la ciudad de la siguiente manera: primero debe recorrer la avenida 1 juntando todas las flores que encuentre, luego debe recorrer la calle 1 juntando todos los papeles que encuentre. Luego, recorre la avenida 2 y la calle 2 de la misma manera y así siguiendo. Implementar un módulo para recorrer la avenida y otro módulo para recorrer la calle.

```
programa TP5_E8
procesos
    proceso izquierda
        comenzar
        repetir 3
            derecha
        fin
    proceso tomar_flores
        comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
        fin
    proceso tomar_papeles
        comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
        fin
    proceso recorrer_avenida
        comenzar
        repetir 99
            tomar_flores
            mover
        fin
    proceso recorrer_calle
        comenzar
        repetir 99
            tomar_papeles
            mover
        fin
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
    comenzar
    repetir 99
        recorrer_avenida
        Pos (1,PosAv)
        derecha
        recorrer_calle
        Pos (PosCa+1,1)
        izquierda
```

```
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 9.

(a) Implementar un proceso para que el robot recorra una avenida juntando flores y se detenga cuando haya juntado 30 flores (seguro existe dicha cantidad).

```
programa TP5_E9a
procesos
    proceso recorrer_avenida
    variables
        flores: numero
    comenzar
        flores:=0
        mientras (flores<30)
            mientras (HayFlorEnLaEsquina & (flores<30))
                tomarFlor
                flores:=flores+1
            si (flores<30)
                mover
            fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    comenzar
        recorrer_avenida
    fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

(b) Modificar el proceso implementado en (a) sabiendo que las 30 flores pueden no existir.

```
programa TP5_E9b
procesos
    proceso recorrer_avenida
    variables
        flores: numero
    comenzar
        flores:=0
        mientras ((flores<30) & (PosCa<100))
            mientras (HayFlorEnLaEsquina & (flores<30))
                tomarFlor
                flores:=flores+1
```

```
si (flores<30)
    mover
fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
comenzar
    recorrer_avenida
fin
variables
R-info: robot1
comenzar
    AsignarArea(R-info,ciudad)
    Iniciar(R-info,1,1)
fin
```

(c) Implementar un programa que recorra todas las avenidas de la ciudad, utilizando el proceso implementado en (a).

```
programa TP5_E9c
procesos
    proceso recorrer_avenida
    variables
        flores: numero
    comenzar
        flores:=0
        mientras (flores<30)
            mientras (HayFlorEnLaEsquina & (flores<30))
                tomarFlor
                flores:=flores+1
            si (flores<30)
                mover
            fin
        areas
        ciudad: AreaC(1,1,100,100)
    robots
    robot robot1
    comenzar
        repetir 99
            recorrer_avenida
            Pos (PosAv+1,1)
            recorrer_avenida
        fin
    variables
    R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
```

Iniciar(R-info,1,1)  
**fin**

## Ejercicio 10.

(a) Implementar un proceso para que el robot recorra una calle y se detenga cuando encuentre un papel (seguro existe). Este proceso debe informar la cantidad de pasos dados hasta encontrar el papel.

```
programa TP5_E10a
procesos
    proceso recorrer_calle
    variables
        pasos: numero
    comenzar
        pasos:=0
        mientras (~HayPapelEnLaEsquina)
            mover
            pasos:=pasos+1
            Informar (pasos)
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    comenzar
        derecha
        recorrer_calle
    fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

(b) Modificar el proceso implementado en (a) sabiendo que el papel puede no existir y, en dicho caso, debe informar 999.

```
programa TP5_E10b
procesos
    proceso recorrer_calle
    variables
        pasos: numero
    comenzar
        pasos:=0
        mientras (~HayPapelEnLaEsquina & (PosAv<100))
            mover
            pasos:=pasos+1
            si (HayPapelEnLaEsquina)
```

```
    Informar (pasos)
    sino
        Informar (999)
    fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
comenzar
derecha
recorrer_calle
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

(c) Implementar un programa que recorra todas las calles de la ciudad, utilizando el proceso implementado en (b).

```
programa TP5_E10c
procesos
    proceso recorrer_calle
    variables
        pasos: numero
    comenzar
        pasos:=0
        mientras (~HayPapelEnLaEsquina & (PosAv<100))
            mover
            pasos:=pasos+1
            si (HayPapelEnLaEsquina)
                Informar (pasos)
            sino
                Informar (999)
            fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
comenzar
derecha
repetir 99
    recorrer_calle
    Pos (1,PosCa+1)
    recorrer_calle
fin
```

**variables**

R-info: robot1

**comenzar**

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

**fin**

## Ejercicio 11.

Programar al robot para que realice el recorrido de la figura 5.11 utilizando un proceso que permita hacer un escalón.

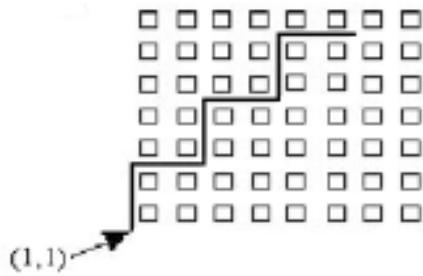


Figura 5.11: Recorrido en escalera de tres escalones.

```
programa TP5_E11
procesos
    proceso izquierda
    comenzar
        repetir 3
            derecha
        fin
    proceso escalon
    comenzar
        repetir 2
            mover
            derecha
        repetir 2
            mover
            izquierda
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            repetir 3
                escalon
            fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

## Trabajo Práctico N° 6: Parámetros de Entrada.

### Ejercicio 1.

*Escribir un proceso que le permita al robot realizar un cuadrado a partir de la esquina donde está parado, girando en la dirección de las agujas del reloj y recibiendo como dato la longitud del lado.*

```
programa TP6_E1
procesos
    proceso cuadrado (E lado: numero)
        comenzar
            repetir 4
                repetir lado
                    mover
                    derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            comenzar
                cuadrado(2)
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

## Ejercicio 2.

Utilizar el proceso de 1 para realizar los recorridos de la figura 6.5 a partir de (1,1).

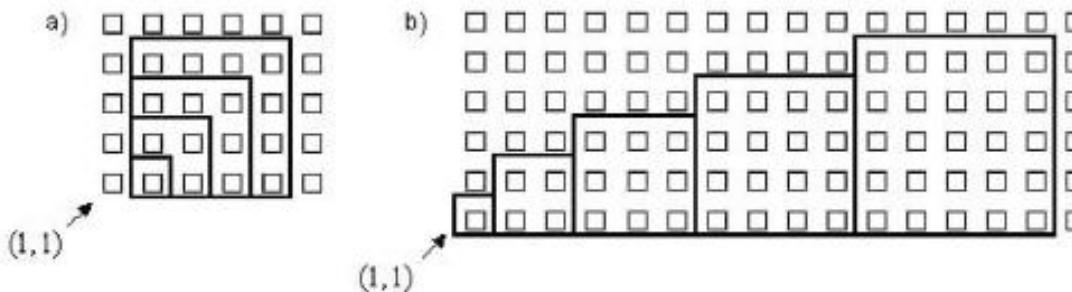


Figura 6.5: Recorridos con cuadrados

(a)

```
programa TP6_E2a
procesos
    proceso cuadrado (E lado: numero)
    comenzar
        repetir 4
            repetir lado
                mover
                derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
        variables
            tamaño: numero
        comenzar
            tamaño:=1
            Pos (2,1)
            repetir 4
                cuadrado(tamaño)
                tamaño:=tamaño+1
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

(b)

```
programa TP6_E2b
procesos
    proceso cuadrado (E lado: numero)
        comenzar
            repetir 4
                repetir lado
                    mover
                    derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
        variables
            tamaño: numero
        comenzar
            tamaño:=1
            repetir 5
                cuadrado(tamaño)
                Pos (PosAv+tamaño,1)
                tamaño:=tamaño+1
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

### Ejercicio 3.

*Escribir un proceso que le permita al robot realizar un rectángulo a partir de la esquina donde está parado, cuyas dimensiones, alto y ancho, se reciben.*

programa TP6\_E3

procesos

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

        comenzar

            repetir 2

                repetir alto

                    mover

                    derecha

                repetir ancho

                    mover

                    derecha

            fin

        areas

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

        robots

            robot robot1

            comenzar

                rectangulo(2,4)

            fin

        variables

            R-info: robot1

        comenzar

            AsignarArea(R-info,ciudad)

            Iniciar(R-info,1,1)

    fin

### Ejercicio 4.

Utilizar el proceso realizado en 3 para que el robot efectúe los recorridos de la figura 6.6 a partir de (1,1).

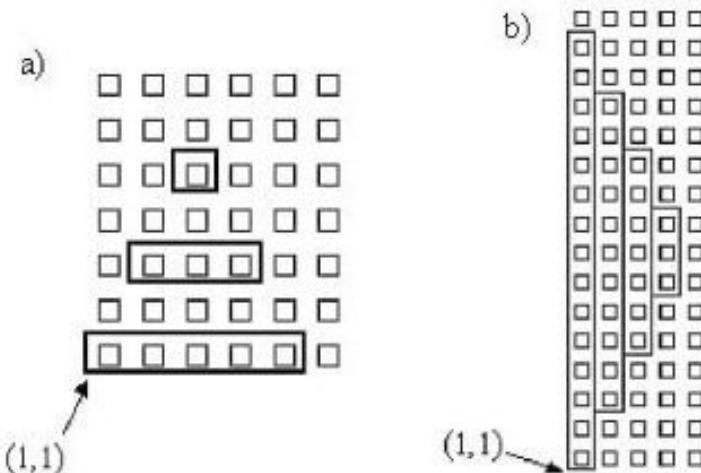


Figura 6.6: Recorridos con rectángulos

(a)

```
programa TP6_E4a
procesos
    proceso rectangulo (E alto: numero; E ancho: numero)
        comenzar
            repetir 2
                repetir alto
                    mover
                    derecha
                repetir ancho
                    mover
                    derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
        variables
            altura, base: numero
        comenzar
            altura:=1
            base:=5
            repetir 3
                rectangulo(alto,ancho)
                Pos (PosAv+1,PosCa+2)
```

```
base:=base-2
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

(b)

```
programa TP6_E4b
procesos
    proceso rectangulo (E alto: numero; E ancho: numero)
    comenzar
        repetir 2
            repetir alto
                mover
                derecha
            repetir ancho
                mover
                derecha
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        variables
            altura, base: numero
        comenzar
            altura:=15
            base:=1
            repetir 4
                rectangulo(altura,base)
                Pos (PosAv+1,PosCa+2)
                altura:=altura-4
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

### Ejercicio 5.

Programar al robot para que realice cada uno de los cuatro recorridos de la figura 6.7.

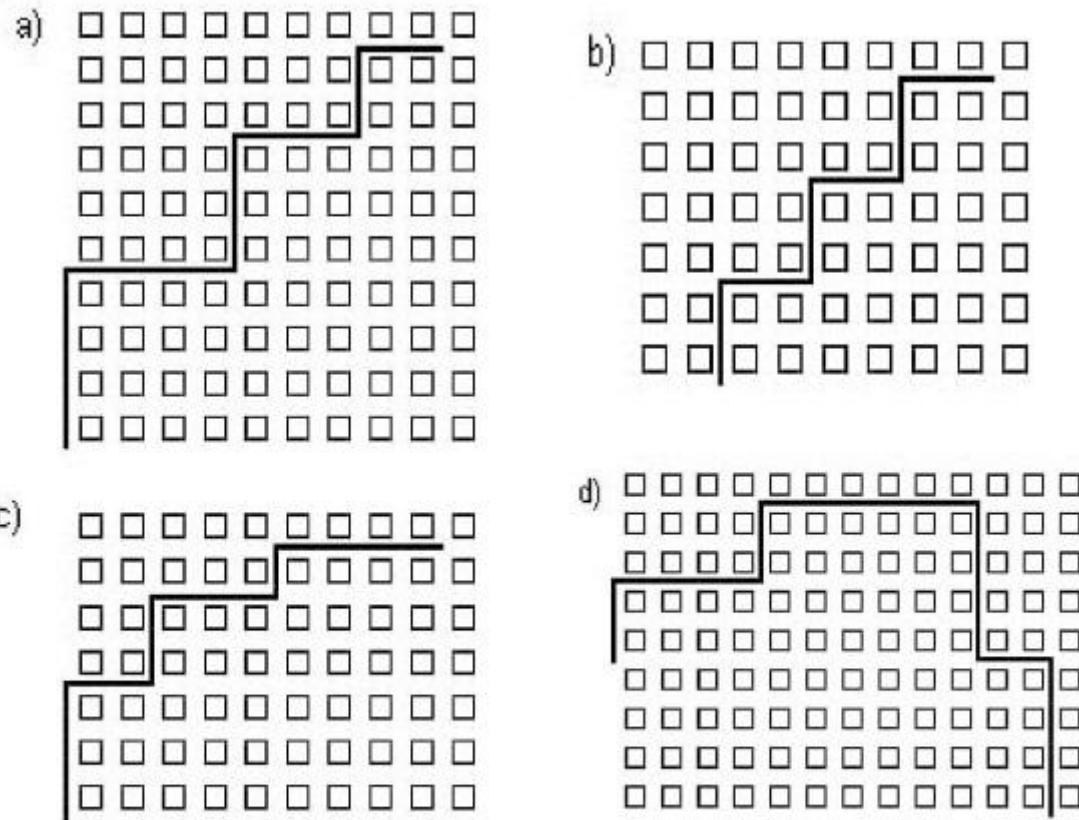


Figura 6.7: Recorrido con escalones.

(a)

```
programa TP6_E5a
procesos
    proceso izquierda
    comenzar
        repetir 3
            derecha
        fin
    proceso escalon (E alto: numero; E ancho: numero)
    comenzar
        repetir alto
            mover
            derecha
        repetir ancho
            mover
            izquierda
        fin
```

```
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
altura, base: numero
comenzar
altura:=4
base:=4
repetir 3
escalon(altura,base)
altura:=altura-1
base:=base-1
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

(b)

```
programa TP6_E5b
procesos
    proceso izquierda
    comenzar
        repetir 3
            derecha
        fin
    proceso escalon (E alto: numero; E ancho: numero)
    comenzar
        repetir alto
            mover
        derecha
        repetir ancho
            mover
        izquierda
    fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
comenzar
Pos (2,1)
repetir 3
    escalon(2,2)
fin
```

variables

R-info: robot1

comenzar

AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

fin

(c)

```
programa TP6_E5c
procesos
    proceso izquierda
        comenzar
            repetir 3
                derecha
            fin
        proceso escalon (E alto: numero; E ancho: numero)
        comenzar
            repetir alto
                mover
                derecha
            repetir ancho
                mover
                izquierda
            fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    variables
        altura, base: numero
    comenzar
        altura:=3
        base:=2
        repetir 3
            escalon(altura,base)
            altura:=altura-1
            base:=base+1
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

(d)

```
programa TP6_E5d
procesos
    proceso izquierda
        comenzar
            repetir 3
                derecha
            fin
        proceso escalon (E alto: numero; E ancho: numero)
        comenzar
            repetir alto
                mover
                derecha
            repetir ancho
                mover
                izquierda
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            variables
                altura, base: numero
            comenzar
                altura:=2
                base:=4
                Pos (1,5)
                repetir 2
                    escalon(altura,base)
                    base:=base+4
                repetir 2
                    derecha
                repetir 4
                    mover
                    izquierda
                escalon(altura,4)
            fin
            variables
                R-info: robot1
            comenzar
                AsignarArea(R-info,ciudad)
                Iniciar(R-info,1,1)
            fin
```

## Ejercicio 6.

(a) Escribir un proceso que le permita al robot recorrer una avenida cuyo número se ingresa como parámetro de entrada.

```
programa TP6_E6a
procesos
    proceso avenida (E avenida: numero)
        comenzar
            Pos (avenida,1)
            repetir 99
                mover
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            comenzar
                avenida(2)
            fin
        variables
            R-info: robot1
            comenzar
                AsignarArea(R-info,ciudad)
                Iniciar(R-info,1,1)
            fin
```

(b) Utilizar el proceso de 6.a para recorrer todas las avenidas de la ciudad.

```
programa TP6_E6b
procesos
    proceso avenida (E avenida: numero)
        comenzar
            Pos (avenida,1)
            repetir 99
                mover
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
        variables
            av: numero
        comenzar
            av:=1
            repetir 99
                avenida(av)
```

```
av:=av+1
Pos(av,1)
avenida(av)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

(c) Utilizar el proceso de 6.a para recorrer las avenidas 5, 6, 7, ... , 15.

```
programa TP6_E6c
procesos
    proceso avenida (E avenida: numero)
        comenzar
            Pos(avenida,1)
            repetir 99
                mover
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            variables
                av: numero
            comenzar
                av:=5
                repetir 10
                    avenida(av)
                    av:=av+1
                    Pos(av,1)
                    avenida(av)
                fin
            variables
            R-info: robot1
            comenzar
                AsignarArea(R-info,ciudad)
                Iniciar(R-info,1,1)
            fin
```

(d) Utilizar el proceso de 6.a para recorrer las avenidas pares de la ciudad.

```
programa TP6_E6d
procesos
```

proceso avenida (E avenida: numero)

comenzar

Pos (avenida,1)

repetir 99

    mover

fin

areas

    ciudad: AreaC(1,1,100,100)

robots

    robot robot1

variables

    av: numero

comenzar

    av:=2

    repetir 49

        avenida(av)

        av:=av+2

        Pos (av,1)

        avenida(av)

fin

variables

    R-info: robot1

comenzar

    AsignarArea(R-info,ciudad)

    Iniciar(R-info,1,1)

fin

## Ejercicio 7.

Programar al robot para que realice un módulo CalleFlor que recorra una calle cuyo número se ingresa como parámetro, hasta juntar tantas flores como lo indica otro parámetro de entrada que este módulo recibe. La cantidad de flores seguro existe.

```
programa TP6_E7
procesos
    proceso CalleFlor (E calle: numero; E flores: numero)
    variables
        tot_flores: numero
    comenzar
        Pos (1,calle)
        derecha
        tot_flores:=0
        mientras (tot_flores<flores)
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                tot_flores:=tot_flores+1
                si (tot_flores<flores)
                    mover
            fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        variables
            ca, tot_flores: numero
        comenzar
            ca:=10
            tot_flores:=10
            CalleFlor(ca,tot_flores)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

## Ejercicio 8.

Programar al robot para que realice un módulo Avenida que recorra una avenida, cuyo número se ingresa como parámetro, hasta dar tantos pasos como los indicados por otro parámetro de entrada que este módulo recibe. Es decir, si recibe los valores 3 y 1, debe dar 1 paso en la avenida 3; si recibe 12 y 5, debe dar 5 pasos en la avenida 12; y así sucesivamente. En cambio, si recibe algún valor negativo no debe dar pasos. Considerar que la cantidad máxima de pasos que podrá dar es 99, cualquier valor que reciba mayor que 99 implicará realizar sólo hasta 99 pasos. Los números de avenida seguro son entre 1 y 100.

```
programa TP6_E8
procesos
    proceso Avenida (E avenida: numero; E pasos: numero)
        comenzar
            Pos (avenida,1)
            si (pasos>0)
                si (pasos<100)
                    repetir pasos
                        mover
                sino
                    repetir 99
                        mover
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
        variables
            av, tot_pasos: numero
        comenzar
            av:=10
            tot_pasos:=10
            Avenida(av,tot_pasos)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

## Trabajo Práctico N° 7: **Parámetros de Entrada/Salida.**

### Ejercicio 1.

*Escribir un programa que le permita al robot informar la cantidad total de flores y la cantidad total de papeles que hay en toda la ciudad. Para hacerlo, utilizar un proceso que recorra una calle cuyo número recibe como parámetro y devuelva la información correspondiente.*

```
programa TP7_E1
procesos
    proceso contar_flores (ES flores: numero)
        comenzar
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                flores:=flores+1
            fin
        proceso contar_papeles (ES papeles: numero)
        comenzar
            mientras (HayPapelEnLaEsquina)
                tomarPapel
                papeles:=papeles+1
            fin
        proceso recorrer_calle (E calle: numero; ES flores: numero; ES papeles: numero)
        comenzar
            Pos (1,calle)
            repetir 99
                contar_flores(flores)
                contar_papeles(papeles)
                mover
                contar_flores(flores)
                contar_papeles(papeles)
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
            variables
                ca, tot_flores, tot_papeles: numero
            comenzar
                ca:=1
                tot_flores:=0
                tot_papeles:=0
                derecha
                repetir 100
                    recorrer_calle(ca,tot_flores,tot_papeles)
                    ca:=ca+1
```

```
Informar (tot_flores,tot_papeles)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 2.

El robot debe limpiar de flores las calles impares de la siguiente forma: toda flor que se encuentre en una calle impar debe ser trasladada a la calle par siguiente sobre la misma avenida. Por ejemplo, si en (4,1) hay una flor, debe llevarse a (4,2). Al terminar el recorrido, debe informar la cantidad total de flores que trasladó.

Recorriendo por calle:

```
programa TP7_E2
procesos
    proceso tomar_flores (ES flores_esq: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores_esq:=flores_esq+1
        fin
    proceso trasladar_flores (E flores_esq: numero; ES flores: numero)
    comenzar
        Pos (PosAv,PosCa+1)
        repetir flores_esq
            depositarFlor
        Pos (PosAv,PosCa-1)
        flores:=flores+flores_esq
    fin
    proceso recorrer_calle_impar (ES flores: numero)
    variables
        tot_flores_esq: numero
    comenzar
        repetir 99
            tot_flores_esq:=0
            tomar_flores(tot_flores_esq)
            si (tot_flores_esq>0)
                trasladar_flores(tot_flores_esq,flores)
            mover
            tot_flores_esq:=0
            tomar_flores(tot_flores_esq)
            si (tot_flores_esq>0)
                trasladar_flores(tot_flores_esq,flores)
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    variables
        tot_flores: numero
    comenzar
        tot_flores:=0
        derecha
```

```
repetir 49
    recorrer_calle_impar(tot_flores)
    Pos (1,PosCa+2)
    recorrer_calle_impar(tot_flores)
    Informar (tot_flores)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

Recorriendo por avenida:

```
programa TP7_E2
procesos
    proceso tomar_flores (ES flores_esq: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores_esq:=flores_esq+1
        fin
    proceso trasladar_flores (E flores_esq: numero; ES flores: numero)
    comenzar
        Pos (PosAv,PosCa+1)
        repetir flores_esq
            depositarFlor
            flores:=flores+flores_esq
        Pos (PosAv,PosCa-1)
        fin
    proceso recorrer_avenida (ES flores: numero)
variables
    tot_flores_esq: numero
comenzar
    repetir 49
        tot_flores_esq:=0
        tomar_flores(tot_flores_esq)
        si (tot_flores_esq>0)
            trasladar_flores(tot_flores_esq,flores)
        Pos (PosAv,PosCa+2)
        tot_flores_esq:=0
        tomar_flores(tot_flores_esq)
        si (tot_flores_esq>0)
            trasladar_flores(tot_flores_esq,flores)
    fin
areas
    ciudad: AreaC(1,1,100,100)
robots
    robot robot1
```

```
variables
tot_flores: numero
comenzar
tot_flores:=0
repetir 99
    recorrer_avenida(tot_flores)
    Pos(PosAv+1,1)
    recorrer_avenida(tot_flores)
    Informar(tot_flores)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

### Ejercicio 3.

*Escribir un programa para que el robot recorra la avenida 9 depositando en cada esquina lo que haga falta para que la cantidad de flores supere en 1 a la cantidad de papeles. Si no tiene en su bolsa lo necesario para hacerlo, debe detener recorrido. Al finalizar, debe informar la cantidad de esquinas que pudo completar adecuadamente. Si el recorrido quedo incompleto, debe retornar a (9,1).*

programa TP7\_E3

procesos

  proceso contar\_flores (ES flores: numero)

    comenzar

      mientras (HayFlorEnLaEsquina)

        tomarFlor

        flores:=flores+1

      fin

  proceso contar\_papeles (ES papeles: numero)

    comenzar

      mientras (HayPapelEnLaEsquina)

        tomarPapel

        papeles:=papeles+1

      repetir papeles

        depositarPapel

      fin

  proceso comparar\_flores\_papeles (E flores: numero; E papeles: numero; ES diferencia: numero)

    comenzar

      diferencia:=flores-papeles

    fin

  proceso depositar\_flores (E flores: numero; E diferencia: numero; ES esquinas: numero; ES termino: boolean)

    variables

      conteo: numero

    comenzar

      conteo:=0

      si (diferencia=1)

        repetir flores

          depositarFlor

          esquinas:=esquinas+1

      si (diferencia>1)

        repetir (flores-(diferencia-1))

          depositarFlor

          esquinas:=esquinas+1

      si (diferencia<1)

        repetir (flores+(1-diferencia))

          si (HayFlorEnLaBolsa)

            depositarFlor

            conteo:=conteo+1

          si (conteo=(flores+(1-diferencia)))

```
esquinas:=esquinas+1
sino
    termino:=V
fin
proceso recorrer_avenida (ES esquinas: numero; ES termino: boolean)
variables
    tot_flores, tot_papeles, tot_diferencia: numero
comenzar
    mientras ((termino=F) & (PosCa<100))
        tot_flores:=0
        tot_papeles:=0
        tot_diferencia:=0
        contar_flores(tot_flores)
        contar_papeles(tot_papeles)
        comparar_flores_papeles(tot_flores,tot_papeles,tot_diferencia)
        depositar_flores(tot_flores,tot_diferencia,esquinas,termino)
        si (termino=F)
            mover
            si (PosCa=100)
                tot_flores:=0
                tot_papeles:=0
                tot_diferencia:=0
                contar_flores(tot_flores)
                contar_papeles(tot_papeles)
                comparar_flores_papeles(tot_flores,tot_papeles,tot_diferencia)
                depositar_flores(tot_flores,tot_diferencia,esquinas,termino)
fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
tot_esquinas: numero
tot_termino: boolean
comenzar
tot_esquinas:=0
tot_termino:=F
Pos (9,1)
recorrer_avenida(tot_esquinas,tot_termino)
si (tot_termino=V)
    Pos (9,1)
    Informar (tot_esquinas)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 4.

Programar al robot para que recorra las calles de la ciudad. Por cada calle, determinar si debe depositar una flor o un papel en cada esquina, dependiendo si el total de flores de la calle es mayor o igual que el total de papeles (deposita una flor por cada esquina) o si el total de flores es menor al total de papeles (deposita un papel por cada esquina). Al terminar el recorrido de todas las calles, debe informar cuántas de las calles fueron completadas con flores.

```
programa TP7_E4
procesos
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores:=flores+1
        fin
    proceso contar_papeles (ES papeles: numero)
    comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
        fin
    proceso recorrer_calle (ES calles: numero)
variables
    tot_flores, tot_papeles, esquinas: numero
comenzar
    tot_flores:=0
    tot_papeles:=0
    esquinas:=0
    repetir 99
        contar_flores(tot_flores)
        contar_papeles(tot_papeles)
        mover
        contar_flores(tot_flores)
        contar_papeles(tot_papeles)
        Pos (1,PosCa)
        si (tot_flores>=tot_papeles)
            mientras (HayFlorEnLaBolsa & (PosCa<100))
                depositarFlor
                mover
                esquinas:=esquinas+1
            si (HayFlorEnLaBolsa)
                depositarFlor
                esquinas:=esquinas+1
            sino
                mientras (HayPapelEnLaBolsa & (PosCa<100))
                    depositarPapel
                    mover
```

```
si (HayPapelEnLaBolsa)
    depositarPapel
    si (esquinas=100)
        calles:=calles+1
    fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
tot_calles: numero
comenzar
tot_calles:=0
derecha
repetir 99
    recorrer_calle(tot_calles)
    Pos (1,PosCa+1)
    recorrer_calle(tot_calles)
    Informar (tot_calles)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 5.

*Escribir un programa que le permita al robot recorrer las calles impares de la ciudad. Cada calle debe recorrerse sólo hasta encontrar una esquina con alguna flor o algún papel o ambos, que seguro existe. Al finalizar cada calle, debe informar cuántos pasos se ha dado hasta encontrar la esquina.*

```
programa TP7_E5
procesos
    proceso recorrer_calle_impar
    variables
        pasos: numero
    comenzar
        pasos:=0
        mientras (~ (HayFlorEnLaEsquina | HayPapelEnLaEsquina))
            mover
            pasos:=PosAv-1
            Informar (pasos)
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
        comenzar
            derecha
            repetir 49
                recorrer_calle_impar
                Pos (1,PosCa+2)
                recorrer_calle_impar
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
            Iniciar(R-info,1,1)
        fin
```

## Ejercicio 6.

Escribir un programa que le permita al robot recorrer cuadrados hasta encontrar un cuadrado con, exactamente, 3 flores y 2 papeles (seguro existe). El primer cuadrado es de lado 99 y los siguientes van decrementando en uno el tamaño del lado (98, 97 y así sucesivamente).

```
programa TP7_E6
procesos
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores:=flores+1
        fin
    proceso contar_papeles (ES papeles: numero)
    comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
        fin
    proceso cuadrado (E lado: numero; ES flores: numero; ES papeles: numero)
    comenzar
        repetir 4
            repetir lado
                contar_flores(flores)
                contar_papeles(papeles)
                mover
                derecha
            fin
        areas
            ciudad: AreaC(1,1,100,100)
        robots
            robot robot1
        variables
            tot_flores, tot_papeles, lado: numero
        comenzar
            tot_flores:=0
            tot_papeles:=0
            lado:=99
            mientras (~ ((tot_flores=3) & (tot_papeles=2)))
                cuadrado(lado,tot_flores,tot_papeles)
                si (~ ((tot_flores=3) & (tot_papeles=2)))
                    lado:=lado-1
            fin
        variables
            R-info: robot1
        comenzar
            AsignarArea(R-info,ciudad)
```

Iniciar(R-info,1,1)  
**fin**

## **Trabajo Práctico N° 8:** **Ejercicios Adicionales.**

### **Ejercicio 1.**

*Escribir un programa que le permita al robot recorrer todas las avenidas de la ciudad. Cada avenida debe recorrerse sólo hasta encontrar una esquina vacía (sin flor ni papel) que seguro existe. Además, a medida que se recorre cada avenida, debe informar si la misma tuvo, a lo sumo, 45 flores (hasta que se encontró la esquina). Nota: Se debe usar Modularización.*

```
programa TP8_E1
procesos
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores:=flores+1
        fin
    proceso recorrer_avenida (ES flores_avenida: numero)
    comenzar
        mientras (HayFlorEnLaEsquina | HayPapelEnLaEsquina)
            contar_flores(flores_avenida)
            mover
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    variables
        tot_flores_avenida: numero
    comenzar
        repetir 99
            tot_flores_avenida:=0
            recorrer_avenida(tot_flores_avenida)
            Informar (tot_flores_avenida<=45)
            Pos (PosAv+1,1)
            tot_flores_avenida:=0
            recorrer_avenida(tot_flores_avenida)
            Informar (tot_flores_avenida<=45)
        fin
    variables
        R-info: robot1
    comenzar
        AsignarArea(R-info,ciudad)
        Iniciar(R-info,1,1)
    fin
```

## Ejercicio 2.

*Escribir un programa que le permita al robot recorrer todas las avenidas de la ciudad. Al finalizar el recorrido, debe informar la cantidad de esquinas con, exactamente, 20 flores y la cantidad de avenidas con menos de 60 papeles. Nota: Se debe usar Modularización y no modificar la cantidad de papeles/flores de las esquinas.*

programa TP8\_E2

**procesos**

**proceso** contar\_flores (ES flores: numero)

**comenzar**

      mientras (HayFlorEnLaEsquina)

        tomarFlor

        flores:=flores+1

      repetir flores

        depositarFlor

**fin**

**proceso** contar\_papeles (ES papeles: numero)

**comenzar**

      mientras (HayPapelEnLaEsquina)

        tomarPapel

        papeles:=papeles+1

      repetir papeles

        depositarPapel

**fin**

**proceso** recorrer\_avenida (ES esquinas: numero; ES avenidas: numero)

**variables**

      tot\_flores, tot\_papeles, papeles\_avenida: numero

**comenzar**

      papeles\_avenida:=0

      repetir 99

        tot\_flores:=0

        tot\_papeles:=0

        contar\_flores(tot\_flores)

        contar\_papeles(tot\_papeles)

        si (tot\_flores=20)

          esquinas:=esquinas+1

        papeles\_avenida:=papeles\_avenida+tot\_papeles

        mover

      tot\_flores:=0

      tot\_papeles:=0

      contar\_flores(tot\_flores)

      contar\_papeles(tot\_papeles)

      si (tot\_flores=20)

        esquinas:=esquinas+1

      papeles\_avenida:=papeles\_avenida+tot\_papeles

      si (papeles\_avenida<60)

        avenidas:=avenidas+1

**fin**

**areas**

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

**robots**

robot robot1

**variables**

tot\_esquinas, tot\_avenidas: numero

**comenzar**

tot\_esquinas:=0

tot\_avenidas:=0

repetir 99

    recorrer\_avenida(tot\_esquinas,tot\_avenidas)

    Pos (PosAv+1,1)

    recorrer\_avenida(tot\_esquinas,tot\_avenidas)

    Informar (tot\_esquinas,tot\_avenidas)

**fin**

**variables**

R-info: robot1

**comenzar**

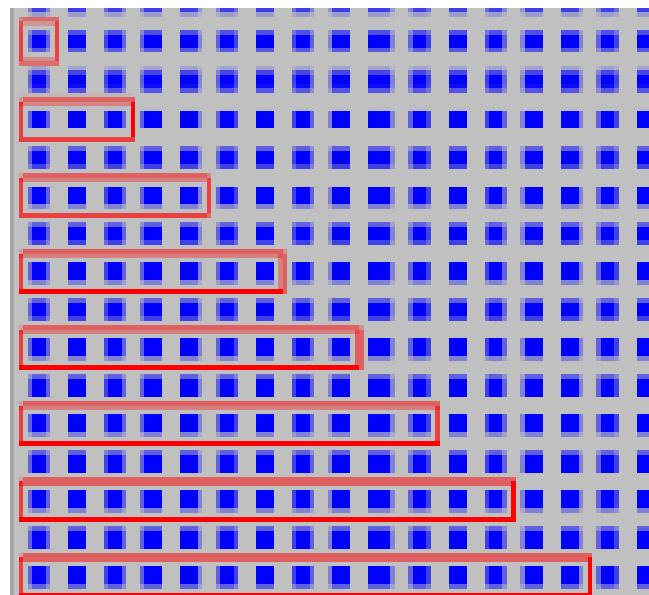
AsignarArea(R-info,ciudad)

Iniciar(R-info,1,1)

**fin**

### Ejercicio 3.

Escribir un programa que le permita al robot realizar el siguiente recorrido, comenzando en la esquina (1,1) juntando todas las flores y papeles de cada esquina. Al finalizar el recorrido, debe informar la cantidad total de flores y de papeles que tiene en la bolsa.  
Nota: Se debe usar Modularización.



(1,1)

```
programa TP8_E3
procesos
    proceso tomar_flores
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
        fin
    proceso tomar_papeles
    comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
        fin
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaBolsa)
            depositarFlor
            flores:=flores+1
        fin
    proceso contar_papeles (ES papeles: numero)
    comenzar
        mientras (HayPapelEnLaBolsa)
```

```
depositorPapel
papeles:=papeles+1
fin
proceso rectangulo (E alto: numero; E ancho: numero)
comenzar
repetir 2
    repetir alto
        tomar_flores
        tomar_papeles
        mover
        derecha
    repetir ancho
        tomar_flores
        tomar_papeles
        mover
        derecha
fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
altura, base, tot_flores, tot_papeles: numero
comenzar
tot_flores:=0
tot_papeles:=0
altura:=1
base:=15
repetir 8
    rectangulo(altura,base)
    Pos (1,PosCa+2)
    base:=base-2
    contar_flores(tot_flores)
    contar_papeles(tot_papeles)
    Informar (tot_flores,tot_papeles)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 4.

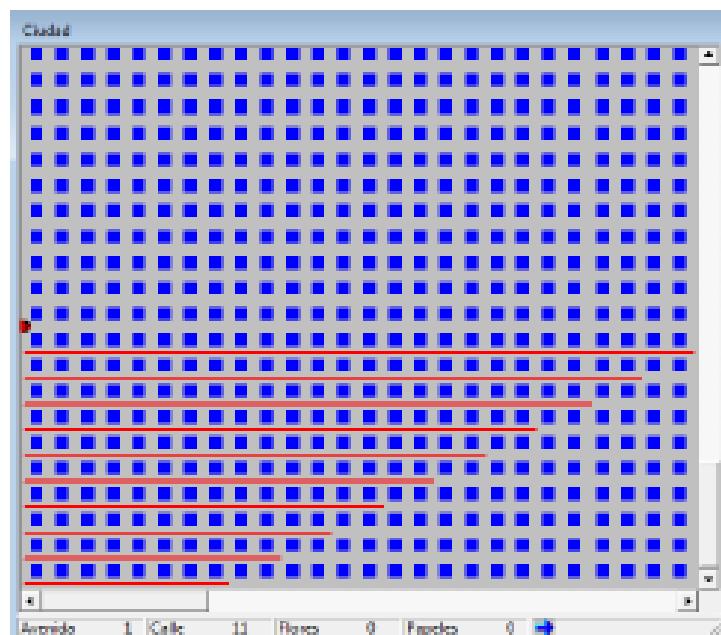
Programar al robot para que recorra la ciudad por avenidas, juntando papeles, hasta encontrar una avenida con, exactamente, 25 flores. Cuando encuentra la avenida con, exactamente, 25 flores, debe recorrer toda la calle 75 (desde la avenida 1) y dar tantos pasos como papeles juntó en todas las avenidas recorridas. Nota: La avenida con 25 flores seguro existe. La cantidad de papeles juntados (entre todas las avenidas recorridas) seguro es menor a 100. Las esquinas pueden modificarse. Modularizar. Ejemplo: Suponer que el robot encuentra que la avenida 5 tiene, exactamente, 25 flores, y durante su recorrido (avenidas 1, 2, 3, 4 y 5) juntó 62 papeles. Entonces, debe recorrer la calle 75 y dar 62 pasos.

```
programa TP8_E4
procesos
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores:=flores+1
        fin
    proceso contar_papeles (ES papeles: numero)
    comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
        fin
    proceso recorrer_avenida (ES flores_avenida: numero; ES papeles_ciudad: numero)
    comenzar
        repetir 99
            contar_flores(flores_avenida)
            contar_papeles(papeles_ciudad)
            mover
            contar_flores(flores_avenida)
            contar_papeles(papeles_ciudad)
        fin
    areas
        ciudad: AreaC(1,1,100,100)
    robots
        robot robot1
    variables
        tot_flores_avenida, tot_papeles_ciudad: numero
    comenzar
        tot_flores_avenida:=0
        tot_papeles_ciudad:=0
        mientras (tot_flores_avenida <> 25)
            tot_flores_avenida:=0
            recorrer_avenida(tot_flores_avenida,tot_papeles_ciudad)
            si (tot_flores_avenida=25)
                Pos (1,75)
```

```
derecha
repetir tot_papeles_ciudad
    mover
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 5.

Escribir un programa que le permita al robot recorrer 10 calles de la ciudad (como se muestra en la figura). En cada calle, debe juntar las flores y los papeles. Al finalizar cada calle, informar la cantidad de esquinas con el doble de flores que papeles. Al finalizar el recorrido, debe informar la cantidad total de papeles y de flores recogidas. El recorrido comienza en (1,1). En la primer calle, se deben recorrer 8 avenidas, en la siguiente 2 avenidas más (es decir, 10 avenidas) y así incrementar de a dos avenidas para las calles restantes. Nota: Se debe usar Modularización.



```
programa TP8_E5
procesos
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores:=flores+1
        fin
    proceso contar_papeles (ES papeles: numero)
    comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
        fin
    proceso recorrer_calle (E pasos: numero; ES esquinas: numero; ES flores_calles: numero; ES papeles_calles: numero)
    variables
        tot_flores, tot_papeles, aux: numero
    comenzar
        repetir pasos
```

```
tot_flores:=0
tot_papeles:=0
aux:=0
contar_flores(tot_flores)
contar_papeles(tot_papeles)
flores_calles:=flores_calles+tot_flores
papeles_calles:=papeles_calles+tot_papeles
aux:=2*tot_papeles
si ((tot_flores=aux) & (tot_papeles>0))
    esquinas:=esquinas+1
mover
tot_flores:=0
tot_papeles:=0
aux:=0
contar_flores(tot_flores)
contar_papeles(tot_papeles)
flores_calles:=flores_calles+tot_flores
papeles_calles:=papeles_calles+tot_papeles
aux:=2*tot_papeles
si ((tot_flores=aux) & (tot_papeles>0))
    esquinas:=esquinas+1
fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
tot_pasos, tot_esquinas, tot_flores_calles, tot_papeles_calles: numero
comenzar
tot_pasos:=7
tot_flores_calles:=0
tot_papeles_calles:=0
derecha
repetir 10
    tot_esquinas:=0
    recorrer_calle(tot_pasos,tot_esquinas,tot_flores_calles,tot_papeles_calles)
    Informar (tot_esquinas)
    tot_pasos:=tot_pasos+2
    Pos (1,PosCa+1)
    Informar (tot_flores_calles,tot_papeles_calles)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 6.

Realizar un programa para que el robot recorra 15 cuadrados los cuales comienzan siempre en la esquina (1,1). En cada cuadrado, debe juntar las flores y los papeles. Al finalizar los 15 cuadrados, debe informar cuántos cuadrados tenían más de 20 flores. Notas: Modularizar. Se pide que, como mínimo, exista un módulo que realice un cuadrado. El primer cuadrado debe ser de lado 1, el segundo de lado 2 y así sucesivamente hasta llegar al cuadrado 15 el cual es de lado 15.

```
programa TP8_E6
procesos
    proceso contar_flores (ES flores: numero)
    comenzar
        mientras (HayFlorEnLaEsquina)
            tomarFlor
            flores:=flores+1
        fin
    proceso contar_papeles (ES papeles: numero)
    comenzar
        mientras (HayPapelEnLaEsquina)
            tomarPapel
            papeles:=papeles+1
        fin
    proceso cuadrado (E lado: numero; ES cuadrados: numero)
variables
    tot_flores, tot_papeles: numero
comenzar
    tot_flores:=0
    tot_papeles:=0
    repetir 4
        repetir lado
            contar_flores(tot_flores)
            contar_papeles(tot_papeles)
            mover
            derecha
            si (tot_flores>20)
                cuadrados:=cuadrados+1
        fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
lado, tot_cuadrados: numero
comenzar
lado:=1
tot_cuadrados:=0
repetir 15
    cuadrado(lado,tot_cuadrados)
```

```
lado:=lado+1
Informar (tot_cuadrados)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 7.

*EXAMEN AÑO 2013: Escribir un algoritmo para que el robot recorra la avenida 8 juntando todas las flores y todos los papeles hasta encontrar una esquina vacía. Luego, debe recorrer un rectángulo que comience en (1,1), donde el alto del rectángulo es igual a la cantidad de flores juntadas en la avenida 8 y el ancho o base es igual a la cantidad de papeles juntados en la avenida 8. Ejemplo: Si, cuando el robot termina de recorrer la avenida 8 (porque encontró la esquina vacía), juntó 5 flores y 4 papeles, debe posicionarse en (1,1) y hacer un rectángulo donde el alto es 5 y el ancho es 4. Nota: Se debe usar modularización (como mínimo, debe haber un módulo para la avenida 8 y otro para el rectángulo). La esquina vacía de la avenida 8 seguro existe. El total de flores y de papeles de la avenida 8 es menor o igual a 99.*

```
programa TP8_E7
procesos
    proceso contar_flores (ES flores: numero)
        comenzar
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                flores:=flores+1
            fin
        proceso contar_papeles (ES papeles: numero)
        comenzar
            mientras (HayPapelEnLaEsquina)
                tomarPapel
                papeles:=papeles+1
            fin
        proceso recorrer_avenida (E esquina: boolean; ES flores_avenida: numero; ES
        papeles_avenida: numero)
        variables
            tot_flores, tot_papeles: numero
        comenzar
            mientras (esquina=F)
                tot_flores:=0
                tot_papeles:=0
                contar_flores(tot_flores)
                contar_papeles(tot_papeles)
                flores_avenida:=flores_avenida+tot_flores
                papeles_avenida:=papeles_avenida+tot_papeles
                si (~ ((tot_flores=0) & (tot_papeles=0)))
                    mover
                sino
                    esquina:=V
            fin
        proceso rectangulo (E alto: numero; E ancho: numero)
        comenzar
            repetir 2
                repetir alto
                    mover
```

```
derecha
repetir ancho
    mover
    derecha
fin
areas
ciudad: AreaC(1,1,100,100)
robots
robot robot1
variables
tot_flores_avenida, tot_papeles_avenida: numero
esquina: boolean
comenzar
esquina:=F
tot_flores_avenida:=0
tot_papeles_avenida:=0
Pos (8,1)
recorrer_avenida(esquina,tot_flores_avenida,tot_papeles_avenida)
Pos (1,1)
rectangulo(tot_flores_avenida,tot_papeles_avenida)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```

## Ejercicio 8.

*EXAMEN AÑO 2013: Escribir un algoritmo para que el robot recorra las calles impares de la ciudad. Cada calle debe recorrerse hasta juntar, al menos, 10 flores. Una vez que ha recorrido todas las calles, debe recorrer la avenida 10, la avenida 11 y la avenida 12 juntando todos los papeles. Al finalizar de recorrer las tres avenidas, debe informar la cantidad total de papeles juntados. Nota: Se debe usar modularización (como mínimo, debe haber un módulo para las calles y otro para las avenidas). Seguro que cada calle tiene, al menos, 10 flores.*

```
programa TP8_E8
procesos
    proceso izquierda
        comenzar
            repetir 3
                derecha
            fin
        proceso contar_flores (ES flores: numero)
        comenzar
            mientras (HayFlorEnLaEsquina)
                tomarFlor
                flores:=flores+1
            fin
        proceso contar_papeles (ES papeles: numero)
        comenzar
            mientras (HayPapelEnLaEsquina)
                tomarPapel
                papeles:=papeles+1
            fin
        proceso recorrer_calle_impar
        variables
            tot_flores: numero
        comenzar
            tot_flores:=0
            mientras (tot_flores<10)
                contar_flores(tot_flores)
                si (tot_flores<10)
                    mover
                fin
            proceso recorrer_avenida (ES papeles_avenidas: numero)
            comenzar
                repetir 99
                    contar_papeles(papeles_avenidas)
                    mover
                    contar_papeles(papeles_avenidas)
                fin
            areas
                ciudad: AreaC(1,1,100,100)
            robots
```

```
robot robot1
variables
tot_papeles_avenidas: numero
comenzar
tot_papeles_avenidas:=0
derecha
repetir 49
    recorrer_calle_impar
    Pos (1,PosCa+2)
    recorrer_calle_impar
    Pos (10,1)
izquierda
repetir 2
    recorrer_avenida(tot_papeles_avenidas)
    Pos (PosAv+1,1)
    recorrer_avenida(tot_papeles_avenidas)
    Informar (tot_papeles_avenidas)
fin
variables
R-info: robot1
comenzar
AsignarArea(R-info,ciudad)
Iniciar(R-info,1,1)
fin
```