

APUNTES DE LOGO

Eugenio Roanes Macías¹ y Eugenio Roanes Lozano²

Sección Departamental de Álgebra³
Facultad de Educación
Universidad Complutense de Madrid

¹ **roanes@mat.ucm.es**, Despacho 3005.

² **eroanes@mat.ucm.es**, Despacho 3005.

³ **www.ucm.es/info/secdealg/**

Hoja 1: INICIACION A LA GEOMETRIA DE LA TORTUGA

<u>Descripción de la primitiva o comando</u>	<u>castellano</u>	<u>inglés</u>
AVanzar n pasos (pixels aprox.)	AV n	FD n
REtroceder n pasos	RE n	BK n
Girar Derecha n grados (sentido horario)	GD n	RT n
Girar Izquierda n grados (sentido antihorario)	GI n	LT n
Borrar Pantalla (y tortuga al centro)	BP	CS
No todas las primitivas son necesarias, p.ej.: GI 90 equivale a: GD -90 y: RE 100 equicale a: AV -100		
Subir Lápiz (no dejar traza al moverse)	SL	PU
Bajar Lápiz (dejar traza al moverse)	BL	PD
Ocultar la Tortuga	OT	HT
Mostrar la Tortuga	MT	ST
Cambiar a color n (n=1,2,...,16) (n=0..255 en MSW-Logo)	PONCL n	SETPC n
Borrar por donde va pasando	GOMA	PE
Dejar de borrar por donde va pasando (en MSW-Logo: o ⁴ :	BL LAPIZNORMAL PONLAPIZ)	PD
Repetir n veces lo indicado entre corchetes:	REPITE n [...] REPEAT n [...]	
Para llenar el interior de una línea simple cerrada, previamente trazada, se lleva la tortuga a un punto interior y se ordena:	RELLENA	FILL
(en la mayoría de los Logos el relleno se realiza en el color del lápiz y puede “escaparse” si el borde no es del mismo color; en MSW-Logo el color de relleno se cambia con:	PONCOLORRELLENO n	

EJERCICIOS:

1. Dibujar el borde de un cuadrado en sentido antihorario.
2. Dibujar el borde de un rectángulo en sentido horario.
3. Dibujar el borde de un triángulo equilátero en sentido antihorario.
4. Dibujar el borde de un cuadrado de centro el centro de pantalla.
5. Dibujar el borde de un hexágono regular.
6. Dibujar el borde de un cuadrado, haciendo uso de Repite.
7. Dibujar el borde de un triángulo equilátero, haciendo uso de Repite.
8. Dibujar el borde de un hexágono regular, haciendo uso de Repite.
9. Dibujar un rectángulo llenando su interior.
10. Dibujar los radios de una rueda con 24 radios.
11. *⁵ Dibujar el borde de un hexágono regular y las diagonales de extremos dos vértices opuestos del hexágono.

⁴ En algunos Logos españoles, las primitivas se tienen que acentuar con las reglas usuales del castellano. En MSWLogo funcionan con o sin tilde.

⁵ Los ejercicios marcados con un asterisco tienen cierta dificultad.

Hoja 2: PROCEDIMIENTOS Y SUBPROCEDIMIENTOS

Procedimiento titulado CUADRADO, que dibuje un cuadrado de lado fijo:

PARA CUADRADO

REPITE 4 [AV 77 GD 90]

FIN

TO CUADRADO

REPEAT 4 [FD 77 RT 90]

END

Procedimiento titulado RECTANG, que dibuje un rectángulo de lados fijos:

PARA RECTANG

REPITE 2 [AV 50 GD 90 AV 100 GD 90]

FIN

Programa que dibuje una colección de cuadrados con un vértice común, cada uno de ellos girado 36 grados respecto del anterior (llamaremos COLECUAD al procedimiento y CUADRADO al subprocedimiento):

PARA COLECUAD

REPITE 10 [CUADRADO GI 36]

FIN

PARA CUADRADO

REPITE 4 [AV 35 GD 90]

FIN

Programa que dibuje una fila horizontal de cinco cuadrados:

PARA FILACUAD

REPITE 5 [CUADRADO SL GD 90 AV 45 GI 90 BL]

FIN

EJERCICIOS

1. Procedimiento TRIANG, que dibuje el borde de un triángulo equilátero.
 2. Procedimiento HEXAG, que dibuje el borde de un hexágono regular.
 3. Procedimiento que dibuje un cuadrado de centro el centro de pantalla.
 4. Procedimiento que dibuje los 24 radios de una rueda de bicicleta.
 5. Programa que dibuje una colección de hexágonos regulares con un vértice común, cada uno de ellos girado 60 grados respecto del anterior.
 6. * Programa que dibuje una fila horizontal de cinco triángulos equiláteros, cuyas bases están contenidas en la misma recta.
 7. *. En el procedimiento anterior, si consideramos un triángulo y los adyacentes a él, pueden compartir un vértice, cortarse o estén separados. Pensar cuando se da cada caso.
-

Hoja 3: VARIABLES. PROCEDIMIENTOS CON ENTRADAS

Para guardar en la variable AB el número 32.7:

HAZ "AB 32.7

MAKE "AB 32.7

Para guardar en la variable G la palabra GATO:

HAZ "G "GATO

MAKE "G "GATO

Para guardar en la variable F la FRASE *el gato gris*:

HAZ "F [EL GATO GRIS]

MAKE "F [EL GATO GRIS]

Para pedir lo almacenado en la variable V: **:V**

y para escribirlo⁶: **ES :V** **PR :V**

Si lo almacenado es un número, se puede hacer con ello lo mismo que con un número:

AV :AB

FD :AB

GD :AB

RT :AB

Un procedimiento poco elegante⁷ que dibuja un cuadrado cuyo lado mide el número almacenado en la variable L es:

PARA CUADRADO

REPITE 4 [AV :L GD 90]

FIN

Por ejemplo, para dibujar dos cuadrados, de lados 77 y 82, se procede así:

HAZ "L 77

CUADRADO

HAZ "L 82

CUADRADO

Pero es más breve y elegante⁸, construirlo como sigue (procedimiento con una entrada):

PARA CUADRADO :L

REPITE 4 [AV :L GD 90]

FIN

en cuyo caso se ejecuta de este modo:

CUADRADO 77

CUADRADO 82

⁶ Detallaremos más adelante cómo escribir números, palabras y frases.

⁷ Haciéndolo así la variable sería *global* (habría que reservarla para uso exclusivo de este procedimiento).

⁸ Varios procedimientos pueden tener una entrada con el mismo nombre (son variables *locales*) y además no interferirían con una variable externa a los procedimientos que tuviera el mismo nombre.

Hoja 3: VARIABLES. PROCEDIMIENTOS CON ENTRADAS (Continuación)

Un procedimiento con entradas "pasa" los valores de las variables a un subprocedimiento suyo.

```
PARA CUADRADOS :L  
REPITE 8 [CUADRADO :L GD 45]  
FIN
```

Ejecución: **CUADRADOS 50**

Procedimiento que dibuje un rectángulo cuyos lados midan los números almacenados en las variables BAS y ALT:

```
PARA RECTANG :BAS :ALT  
REPITE 2 [AV :ALT GD 90 AV :BAS GD 90]  
FIN
```

Ejecución de este procedimiento: **RECTANG 100 60**

EJERCICIOS

1. Procedimiento TRIANG, que dibuje un triángulo equilátero de lado L.
 2. Procedimiento RUEDA, que dibuje los 36 radios de longitud L de una rueda.
 3. Procedimiento BANDERA, que dibuje una bandera consistente en un mástil de longitud L y cuya tela sea rectangular de lados A y B.
 4. Programa que dibuje una fila horizontal de M baldosas cuadradas de lado L
 5. * Programa que dibuje una fila de N ruedas (de 36 radios de longitud L), de modo que la distancia entre los centros de dos contiguas sea D.
-

Hoja 4: GEOMETRIA DE LA TORTUGA 3D

Nota: Las primitivas de tortuga 3D sólo existen en algunos Logos. Las de esta hoja corresponden al dialecto MSW-Logo⁹ y se describen considerando a la tortuga como si fuera un avión.

Pasar al modo 3D:

PERSPECTIVA

Baja el morro n grados:

CABECEA n

Sube el morro n grados:

BAJANARIZ n

Sube ala izquierda y baja ala derecha n grados:

BALANCEA n

Sube ala derecha y baja ala izquierda n grados:

BALANCEAIZQUIERDA n

Borrar Pantalla (y tortuga al centro, orientada “hacia arriba”, como el transbordador espacial en el despegue, con el timón de cola hacia nosotros)

BP

Como ocurre en 2D, no todas las primitivas son necesarias, p.ej.: **BAJANARIZ 90**

es equivalente a: **BALANCEA 90 GD 90 BALANCEAIZQUIERDA 90**

Podemos dibujar los semiejes en la posición habitual tecleando, por ejemplo:

semieje z+: **BP AV 200 RE 200**

semieje x+: **BAJANARIZ 90 RE 200 AV 200**

semieje y+: **GD 90 AV 200 RE 200**

Las siguientes líneas de comandos dibujan cuadrados en distintos planos verticales:

BP REPITE 4 [AV 100 GD 90]

BP REPITE 4 [AV 100 CABECEA 90]

Podemos dibujar 8 planos verticales, cada uno girado respecto del anterior 45 grados, así:

BP REPITE 8 [BALANCEA 45 REPITE 4 [AV 100 GD 90]]

Pedir la lista de las coordenadas de la tortuga en el espacio: **POS3D**

EJERCICIOS:

1. Dibujar un taburete cuadrado con una pata en cada esquina. Idea: repetir 4 veces “lado – pata”.
 2. A partir del taburete anterior, dibujar una silla (con el respaldo inclinado).
 3. Reformar la silla anterior para dibujar una tumbona.
 4. Dibujar una pirámide cuadrangular regular. Idea: proceder como si fuera un taburete invertido, con las patas inclinadas 45 grados respecto del plano horizontal.
 5. Dibujar un octaedro a partir de la pirámide anterior.
-

⁹ La implementación de primitivas 3D en WinLogo es completamente distinta. Por ejemplo, no existe la primitiva **PERSPECTIVA** (las primitivas 3D funcionan directamente); los nombres de algunas de las primitivas 3D son distintos; y **AV** ha sido sustituido en 3D por **ANDA**.

Hoja 5: OPERACIONES

Operaciones binarias

Para sumar 13, 7 y 2.5:	13 + 7 + 2.5	
y para escribir su resultado:	ES 13 + 7 + 2.5	PR 13 + 7 + 2.5
o para avanzar esa distancia:	AV 13 + 7 + 2.5	FD 13 + 7 + 2.5
Para multiplicar 4 por 2.5:	4 * 2.5	
Para restar 13 menos 7:	13 - 7	
Para dividir 15.7 entre 10:	15.7/10	
Cociente entero de 17 entre 5:	COCIENTE 17 5	INTQUOTIENT 17 5
Resto de división entera 17 entre 5:	RESTO 17 5	REMAINDER 17 5

Como es habitual, los paréntesis indican la prioridad entre operaciones: **3* (2+4)**

Si no se ponen paréntesis, se sigue el orden de prioridades interno de Logo. Por ejemplo: **3*2+4** se interpreta como **3* (2+4)**, puesto que, para Logo, el producto es prioritario frente a la suma.

Procedimiento que calcula el área de un triángulo de base B y altura H:

```
PARA AREA :B :H
HAZ "AR (:B * :H) / 2
ES :AR
FIN
```

Para ejecutarlo se procede así: **AREA 5 3**

Pero es más elegante sustituir la penúltima línea por: **(ES [SU AREA ES] :AR)**

Operaciones unarias:

Para calcular la raíz cuadrada de 64:	RC 64	SQRT 64
Para calcular la parte entera de 5.13:	ENTERO 5.13	INT 5.13
Para redondear el número 5.13:	REDONDEA 5.13	ROUND 5.13
Para número natural aleatorio ≥ 0 y < 15 :	AZAR 15	RANDOM 15

Procedimiento que calcula el cociente y el resto de la división entera del número A entre el B (sin usar las primitivas **COCIENTE** y **RESTO**):

```
PARA DIVENT :A :B
HAZ "C ENTERO (:A / :B)
ES :C
ES :A - :B * :C
FIN
```

Procedimiento que simula la tirada de un dado:

```
PARA DADO
ES 1 + AZAR 6
FIN
```

Hoja 5: OPERACIONES (Continuación)

Otras operaciones binarias:¹

Para calcular 3 elevado a 5 (sólo algunos Logos, por ejemplo MSW-Logo):

POTENCIA 3 5

Otras operaciones unarias:

Funciones trigonométricas directas:
(ángulo expresado en grados)

SEN 45

SIN 45

COS 45

COS 45

TAN 45

TAN 45

e inversa:

ARCTAN 1

ARCTAN 1

(en MSW-Logo y WinLogo existen también **ARCSEN** y **ARCCOS**).

EJERCICIOS

- 1 Reformar el procedimiento DIVENT de modo que devuelva los resultados indicando con mensajes cuál es el cociente entero y el resto.
- 2 Procedimiento PITAG, que calcule la medida de la hipotenusa de un triángulo rectángulo cuyos catetos midan X e Y.
- 3 Procedimiento RAYOS, que dibuje los N radios de longitud L de una rueda (sólo los radios).
- 4 Procedimiento POLREG, que dibuje un polígono regular de N lados de longitud L (cada lado).
- 5 * Procedimiento SUERTE, que saque al azar uno de los números siguientes¹⁰: 20, 25, 30, 35, 40, 45, 50.
- 6 * Procedimiento DADOS que simule la suma de la tirada de dos dados.
- 7 * Procedimiento POTEN :X :N que calcule x^n (supuesto que n es un número natural mayor que 1), sin usar la primitiva **POTENCIA**.
Nota: Usar **REPITE**, una variable "contador" y otra variable para ir guardando los resultados parciales.
- 8 * Procedimiento FACTORIAL :N que calcule la factorial de n.
Nota: Usar **REPITE**, una variable "contador" y otra variable para ir guardando los resultados parciales.
- 9 * Procedimiento SUMA :N que calcule $2+4+8+\dots+2^n$ usando la primitiva **POTENCIA**.

(implementaremos de otro modo más elegante los ejercicios 7, 8 y 9 en la hoja 7)

¹⁰ Obsérvese que **AZAR 5 + 1000** es lo mismo que: **AZAR (5 + 1000)** y que: **AZAR 1005**, pues la suma es prioritaria frente a **AZAR**. Son equivalentes: **1000 + AZAR 5** y: **(AZAR 5) + 1000**

Hoja 6: COORDENADAS Y RUMBO

Para pedir la posición de la tortuga (abscisa, ordenada), siendo el origen de coordenadas el centro de pantalla gráfica y la unidad de longitud el lado del pixel:

	POS	POS
Para pedir sólo su abscisa:	COORX	XCOR
Y para pedir sólo su ordenada:	COORY	YCOR

Para que la tortuga se traslade (en línea recta) al punto (100,60):

PONPOS [100 60] **SETPOS [100 60]**

Para que se traslade al punto de abscisa 30 (sin cambiar la ordenada):

PONX 30 **SETX 30**

Para que se traslade al punto de ordenada 25 (sin cambiar la abscisa):

PONY 25 **SETY 25**

(estas primitivas no alteran la orientación de la tortuga).

Nota: Como se habrá observado, las coordenadas del punto (30,20) son dadas a Logo en la forma [30 20]. Pero si el valor de la abscisa está guardado en la variable A y el de la ordenada en la variable B, entonces se han de dar así: **FR :A :B** **SE :A :B**
 (lo que se justificará más adelante, al manejar listas).

Para pedir el rumbo de la tortuga (en grados, respecto de la vertical hacia arriba y en sentido horario, es decir, respecto del semieje y+):

RUMBO	HEADING
-------	---------

Para orientar la tortuga con rumbo 45 grados:

PONRUMBO 45 **SETH 45**

Para orientar la tortuga hacia el punto de coordenadas (60,40):

PONRUMBO HACIA [60 40] **SETH TOWARDS [60 40]**

(estas primitivas no alteran la posición de la tortuga -sus coordenadas-, sólo su orientación).

EJERCICIOS

- 1 Dibujar un rectángulo usando sólo las primitivas **PONX** y **PONY**.
 - 2 Dibujar un triángulo rectángulo isósceles usando sólo **PONPOS**.
 - 3 Construir el procedimiento SEGMENTO, cuyas 4 entradas sean las coordenadas de 2 puntos y que dibuje el segmento de extremos dichos puntos.
 - 4 Definir el procedimiento CUADRILA, cuyas 8 entradas sean las coordenadas de 4 puntos y dibuje el cuadrilátero de vértices dichos puntos.
 - 5 Definir el procedimiento DISTAN, cuyas 4 entradas sean las coordenadas de 2 puntos, que calcule y devuelva la distancia entre ellos.
 - 6 Definir unos procedimientos N, S, E, W, NE, NW, SE, SW, que pongan a la tortuga rumbo a los puntos cardinales.
 - 7 * Definir el procedimiento TRIAN, cuyas 6 entradas sean las coordenadas de 3 puntos y que dibuje el triángulo de vértices dichos puntos y lo rellene.
-

Asignatura optativa (y libre configuración) **INFORMATICA Y MATEMATICA**
Hoja 7: CONDICIONALES Y OPERACIONES LÓGICAS

Vimos como escribir un número: **ES 7** **PR 7**
pero también se pueden escribir palabras: **ES "HOLA** **PR "HELLO**
y frases: **ES [Hola Pepe]** **PR [Hi Joe]**

Procedimiento que compara un número dado con 4 y contesta *MAYOR* si el número es mayor que 4:

```
PARA MAY :X
SI :X > 4 [ES "MAYOR]
FIN
```

Procedimiento que compara un número con 4, para ver si es mayor que 4 o no lo es:

```
PARA COMPARA :X
SI :X > 4 [ES "SI"] [ES "NO"]
FIN
```

Nota: En MSW-Logo el condicional, cuando va seguido de dos corchetes, es **SISINO**, en lugar de **SI** (si va seguido de un sólo corchete, es el habitual **SI**).

Procedimiento que decide si un número es cero, o no lo es:

```
PARA DECIDE :N
SI :N = 0 [ES "SI"] [ES "NO"]
FIN
```

Procedimiento que decide si la edad no es suficiente para poder entrar:

```
PARA EDAD :E
SI NO :E < 18 [ES [PUEDES ENTRAR]]
FIN
```

Procedimiento que indica si la temperatura es agradable:

```
PARA TEMP :T
SI Y :T > 18 :T < 25 [ES "AGRADABLE"]
FIN
```

Procedimiento que informa si la edad permite asistir gratis a cierto espectáculo:

```
PARA NOPAGAR :E
SI O :E > 70 :E < 6 [ES "GRATIS"]
FIN
```

Para parar la ejecución de un procedimiento (saltar directamente al **FIN** o **END**), se puede incluir en él la orden:

ALTO **STOP**

pero es mejor evitar su uso (es poco elegante).

Los equivalentes en los Logos ingleses de **SI**, **NO**, **Y**, **O**, son, respectivamente: **IF**, **NOT**, **AND**, **OR**

Para que al ejecutar un procedimiento, éste devuelva el valor alojado en la variable X, basta incluir en el la orden: **DEV :X**. Así el procedimiento se comporta exactamente igual que una operación.

El procedimiento siguiente devuelve el resultado de añadir al precio (P) el porcentaje de IVA (I) que se especifique:

```
PARA CONIVA :P :I
DEV :P * (1 + :I / 100)
FIN
```

y permite hacer lo siguiente (lo que no sería posible usando **ES** en lugar de **DEV**):

```
ES (CONIVA 134 7 + CONIVA 13 16)
```

Hoja 7: CONDICIONALES Y OPERACIONES LÓGICAS (Continuación)

Un procedimiento que se llama a sí mismo (esto es, que es subprocedimiento de sí mismo) se denomina *procedimiento recursivo*. Una definición *recursiva* típica en Matemáticas es la de factorial. En lugar de definir: $n!=n\cdot(n-1)\cdot\ldots\cdot2\cdot1$, se puede definir:

$$n! = \begin{cases} 1, & \text{si } n=0 \\ n\cdot(n-1)!, & \text{si } n>0 \end{cases}$$

En MSW-Logo:

```
PARA FACT :N
  SISINO :N=1 [DEV 1] [DEV :N * FACT(:N - 1)]
  FIN
```

EJERCICIOS

1. Procedimiento NOMENOR, que decida si, dados dos números, el primero es mayor o igual al segundo, respondiendo “sí” en caso afirmativo.
 2. Procedimiento EDAD, que compruebe si la edad, e , de una persona verifica la condición $17 < e < 65$, respondiendo “sí” en caso afirmativo.
 3. Procedimiento MULTIPLO, que verifique si el número A es múltiplo del B (usando negación), respondiendo “sí” en caso afirmativo y “no” en caso negativo.
Nota: Se puede hacer usando **COCIENTE** o bien usando **RESTO** o bien usando la división “:”.
 4. Procedimiento SIG, que decida si dos números no nulos son del mismo signo (comprobando el signo de su producto).
 5. * Procedimiento CALIFICA, que califique la nota x de acuerdo con el baremo usual: suspenso si $x < 5$, aprobado si $5 \leq x < 6.5$, notable si $6.5 \leq x < 8.5$, sobresaliente si $8.5 \leq x < 9.5$ y matrícula de honor si $x \geq 9.5$.
Nota: se puede hacer con las preguntas consecutivas simplemente, con las preguntas consecutivas y usando ALTO, o con las preguntas anidadas.
 6. Programa que calcule la hipotenusa de un triángulo rectángulo, dados sus dos catetos, usando un subprocedimiento que devuelva el cuadrado de un número dado.
 7. * Programa recursivo que calcule potencias de exponente natural.
 8. * Programa recursivo que calcule el término n -ésimo de la sucesión de Fibonacci. Esta sucesión es la 1,1,2,3,5,8,13,21,34,55,... (comienza: 1,1,... y cada término se obtiene sumando los dos anteriores).
 9. * Programa recursivo que devuelva la suma de los n primeros pares (excluido el cero): $2+4+6+8+\ldots+2\cdot n$ (aplicar la fórmula de la suma de términos de una progresión aritmética no es lo que se pide).
 10. *Programa CUADRADOS, que tenga una entrada, $:L$, y dibuje cuadrados de lados 15, 25, 35, 45, 55, 65 y 75, con un vértice común y lados paralelos, cuando se ordene CUADRADOS 15 (este programa debe ser recursivo –va incrementando L de 10 en 10- y usar un condicional para parar).
 11. * Programa CUA1000, que tenga una entrada, N , y escriba los números naturales que son menores que 1000 y cuadrados de un natural. Idea: no ir comprobando qué números son cuadrados de natural, sino generar con un programa recursivo los cuadrados de los sucesivos naturales y no pare mientras estos cuadrados se mantengan menores que 1000.
-

Hoja 8: LISTAS

[53 gato [7 28] 4.9] es una lista en Logo.

Su primer elemento es 53, el segundo *gato*, el tercero *[7 28]* y el cuarto 4.9.

Se aloja en la variable L así:

HAZ "L [53 gato [7 28] 4.9] **MAKE "L [53 gato [7 28] 4.9]**

Primer elemento de la lista L:

PRI :L

FIRST :L

Último elemento de la lista L:

UL :L

LAST :L

Elemento n-ésimo de la lista L:

ELEMENTO x :L

ITEM n :L

Número de elementos de la lista L:

CUENTA :L

COUNT :L

Eliminar el primer elemento de L:

MP :L

BF :L

Eliminar el último elemento de L:

MU :L

BL :L

Eliminar el elemento x de L **ELIMINA x :L** (elimina su primera ocurrencia)

(este comando no es habitual; por ejemplo existe en WinLogo, pero no en MSWLogo)

Añadir x como primer elemento de L:

PP x :L

FPUT x :L

Añadir x como último elemento de L:

PU x :L

LPUT x :L

Intercalar x como elemento n-ésimo de L: **INSERTA n x :L**

(este comando no es habitual; por ejemplo existe en WinLogo, pero no en MSWLogo)

Concatenar las listas L y M, ya definidas: **FR :L :M**

SE :L :M

Nota: Si la abscisa de un punto está almacenada en la variable P1 y su ordenada en la variable P2, entonces, para enviar la tortuga al punto de tales coordenadas, se ordena (de acuerdo con lo indicado anteriormente): **PONPOS FR :P1 :P2** **SETPOS SE :P1 :P2**

Si se desea que, al ejecutar un procedimiento, éste se detenga para "leer una lista", alojándola en la variable H, basta que, al definirlo, se inserte en el lugar apropiado un LEELISTA (LL):

HAZ "H LL **MAKE "H RL**

Ejemplo:

```
PARA SEGMENT
  ES [ESCRIBE LAS COORDENADAS DEL PUNTO A]
  HAZ "A LL
  ES [ESCRIBE LAS COORDENADAS DEL PUNTO B]
  HAZ "B LL
  SL PONPOS :A BL PONPOS :B
  FIN
```

se ejecuta tecleando: **SEGMENT** (y pulsando Intro) al pedir el procedimiento las coordenadas de A contestamos, p.ej., **40 50** y pulsamos Intro y al pedir las de B contestamos, p.ej., **100 0** (e Intro).

Observación: Si se recoge en la variable A un número con **LEELISTA**, por ejemplo 100, no se puede hacer:

AV :A

pues equivaldría a pedir:

AV [100]

sino que hay que ordenar:

AV PRI :A

Hoja 8: LISTAS (Continuación)

EJERCICIOS

1. ¿Cómo se puede extraer el 22 que aparece en la lista de listas **[[22 3] [4 5] [8 35]]**?
 2. Procedimiento PRIME, con una entrada, L, que devuelva el primer elemento de L, sin usar la primitiva **PRI**.
 3. Procedimiento ULTI, con una entrada, L, que devuelva el último elemento de L, sin usar la primitiva **UL**.
 4. Procedimiento TRI, que pida las coordenadas de los vértices de un triángulo (uno a uno) y lo dibuje.
 5. Procedimiento que pida la medida del lado de un cuadrado y devuelva la medida de su diagonal.
 6. * Procedimiento MENGUA que pida escribir una lista de números, y escriba las listas que se obtienen al ir eliminando un elemento cada vez (por ejemplo el último), hasta quedar vacía.
 7. * Procedimiento INV, que pida una lista de números, y la devuelva invertida (su orden).
 8. * Programa MAXIMO, que pida escribir una lista de números y devuelva el mayor de ellos.
 9. * Programa SUPRIME con una entrada, L, que devuelva la lista que se obtiene al suprimir el elemento n-ésimo de L.
 10. * Programa ADJUNTA, con dos entradas, X y L, que incorpore X como elemento n-ésimo de la lista L (sin utilizar INSERTA).
-