

315623 /
2oj

INSTITUTO UNIVERSITARIO DE CIENCIAS DE LA
EDUCACION INCORPORADO A LA
UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

**APRENDIZAJE POR COMPUTADORA
CON EL LENGUAJE LOGO**

TESIS CON
FALLA DE ORIGEN

TESIS PROFESIONAL

QUE PARA OBTENER EL TITULO DE:
LICENCIADO EN PEDAGOGIA

P R E S E N T A :

OCTAVIO SIGILFREDO ALANIS CHANES

ASESOR :

Lic. Ma. Margarita Ramírez Flores



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE

	PAG.
INTRODUCCION	1
CAPITULO I.- EL LENGUAJE LOGO	
1.1.- ¿ PORQUE APRENDER CON COMPUTADORA ?.....	1
1.2.- ANTECEDENTES PARA EL LENGUAJE LOGO.....	7
1.3.- EL MARCO HISTORICO DE LOGO	11
1.4.- TEORIAS PEDAGOGICAS EN QUE SE APOYA LOGO	15
CAPITULO II.- CARACTERISTICAS PRINCIPALES DE LOGO	
2.1.- LAS CARACTERISTICAS DEL LENGUAJE LOGO	23
2.2.- EL VALOR PEDAGOGICO DE LOGO	37
CAPITULO III.- APRENDIZAJE CON EL LENGUAJE LOGO	
3.1.- APRENDIENDO CON EL LENGUAJE LOGO	40
3.2.- PROCEDIMIENTOS	47
3.3.- VARIABLES	51
3.4.- ELABORANDO FIGURAS	54

CAPITULO IV.- ALGUNAS APLICACIONES DEL LENGUAJE LOGO

4.1.- APLICACIONES DE LOGO EN LA ESCUELA	60
4.2.- JUGANDO AL RELOJ	69
4.3.- DIBUJOS A ESCALA	73
4.4.- PLANO CARTESIANO	76
4.5.- EJERCICIO DE ESTADISTICA	82

CAPITULO V.- OTRA ALTERNATIVA DEL LENGUAJE LOGO

5.1.- PLATICANDO CON LA COMPUTADORA	88
---	----

CAPITULO VI.- LOGO EN LA ESCUELA

6.1.- EXPERIENCIAS EN EL CAMPO DOCENTE.....	99
---	----

CONCLUSIONES	109
--------------------	-----

BIBLIOGRAFIA	114
--------------------	-----

ANEXO.- COMANDOS DE ORDENES RAPIDAS

A.- COMANDOS DE LOGO EN LA IBM.....	116
-------------------------------------	-----

B.- EDICION.....	119
------------------	-----

INTRODUCCION

El tiempo se deslizo en un suspiro. Hace mucho tiempo que terminé mis estudios de Licenciatura y me parece que apenas dejamos nuestros salones en la Universidad.

No estoy seguro si la distancia y mi nueva situación personal, fueran la causa de que hiciera caso omiso de titularme; en cuanto concluí mi preparación universitaria, se presentó la oportunidad de irme a radicar a la provincia e iniciar y descubrir otros derroteros.

Pero cosa curiosa, la invitación en aquellos lugares era para atender materias de tipo profesional en una Escuela Normal Rural, que yo ni siquiera sabía que existieran; lo cual significaba que seguiría en la docencia y en la preparación profesional de nuevos maestros.

Lo más irónico de todo, es que durante muchos años fui asesor de tesis profesionales e incluso ocupé la presidencia de los jurados de exámenes de titulación para Maestros de Educación Primaria; y yo sólo había concluido mi carrera y mi servicio social, pero faltaba mi título. Si he de ser sincero, no encontraba algún tema que me incitara a hacerlo, ya había oído hablar de muchos tópicos, pero a mí particularmente se me hacían tribiales.

Sin embargo, el tiempo me abrió el camino; la oportunidad llegó años más tarde cuando me comisionaron a tomar un curso de computación en el Politécnico Nacional, puesto que las materias que se impartían en la normal, fueron sustituidas por otras distintas en el cambio que se

dio al crearse la Licenciatura para Educación Primaria, y una de las materias sería "Tecnología y Computación Educativa", que precisamente yo atendería, y así encontré con un mundo lleno de posibilidades y grandes alcances aún no explorados.

Así pues, el trabajo que hoy presento ante todos ustedes, lleva el único propósito de contribuir con un grano de arena a mejorar la calidad de la enseñanza, y que si bien no es un trabajo total, en el sentido de que la tecnología no puede darse como acabada, sino como un proceso que cada día tiene nuevos avances, sí pretende que aquellas personas o maestros que lo lean, puedan comprender que hoy en día existen nuevas formas de educar a los jóvenes.

Este trabajo que hoy expongo, lo dividí en partes específicas y que enseguida enumero:

En primer lugar los dos capítulos iniciales, son una investigación de carácter bibliográfico, donde doy los fundamentos del por qué propongo la utilización del Lenguaje LOGO para propiciar una nueva forma de aprendizaje y el valor pedagógico que tiene en su forma de presentación. No se discute que existen otros lenguajes de computación que pueden ser también útiles, quizás hasta más vistosos; sin embargo creo que este Lenguaje presenta otro tipo de bondades que permiten al usuario tener una participación más dinámica.

En el capítulo tercero muestro cómo se aplicaría el Lenguaje Logo desde el primer encuentro con la computadora y trato de iniciarlo en forma inductiva hasta llevarlo a operaciones más complejas, pero siempre permitiendo que la iniciativa y la creatividad del usuario

encuentre otras formas de trabajo con los mismos resultados.

En el capítulo cuarto hago referencia a algunas aplicaciones de tipo práctico para la escuela primaria y dejo camino libre para que se investigue mucho más en las actividades que se presentan.

En el capítulo quinto intento establecer comunicación de manera "dialogada" con la computadora, y expongo ejemplos que pueden servir de base para que maestros o alumnos elaboren sus propios "coloquios" con la máquina.

El capítulo sexto narra las experiencias que he tenido con diferentes tipo de alumnos, desde los de ocho y nueve años y mis propios alumnos de la Licenciatura y los resultados obtenidos de estos encuentros.

Finalmente elaboro conclusiones que permiten tener una idea general de la aplicación de este forma de enseñanza y las enormes posibilidades que tenemos los maestros hoy en día.

Quisiera resaltar que este trabajo que presento, tiene como finalidad el convertirse en un apoyo efectivo para lograr que el programa de Modernización Educativa tenga verdaderos efectos en la preparación de los niños mexicanos, pero sobre todo en la reformulación de contenidos que conduzcan a mejorar la calidad de la educación y que los servicios educativos puedan ofrecerse con mayor eficacia.

Luego de concluir mi tesis, me siento satisfecho de los logros alcanzados, pero sé también que es el inicio de una etapa que presenta retos fuertes, pues el avance tecnológico esta abarrotando todas nuestras formas de vida y sobre todo, entendí que nunca se es demasiado grande para emprender nuevas tareas.

Profr. Octavio S. Alanís Chanes

CAPITULO I

CAPITULO I: EL LENGUAJE LOGO

1.1.- ¿POR QUE APRENDER CON COMPUTADORA?

La informática, día a día avanza de manera vertiginosa, la información no puede parar ni un sólo segundo, se renuevan los avances científicos y tecnológicos, la comunicación depende de los nuevos sistemas de satélites que controlan cada vez mejor nuestro sistema de vida. Todo esto, tiene un poderoso auxiliar: LA COMPUTADORA.

Si esto ocurre con la mayor parte de los sistemas, nosotros nos preguntamos ? y la Educación? ¿ acaso no tiene que ver con el desarrollo de la vida social, material y económica de la gente? creemos que sí, y de manera muy importante.

En la medida que aumenta el número de aplicaciones de las computadoras, éstas dejan de ser herramientas para hacer operaciones transformándose en procesadoras de información. En la enseñanza, también crecen las expectativas de los nuevos cambios informativos y resulta necesario apoyarnos con los equipos que nos permitan introducirnos a estos cambios informáticos. Estamos viviendo en un mundo en el que las computadoras se han vuelto "parte de nuestras vidas".

Las relaciones que se daban anteriormente empiezan a cambiar debido a necesidad de buscar, producir, almacenar y transferir información.

APRENDIZAJE POR COMPUTADORA

Este cambio se ha dado de manera repentina pero con fuerza, lo que ha provocado en muchas ocasiones que dicho cambio sea imprevisto y muchas veces no planeado.

La enseñanza en casi todos los niveles ha sido lenta en responder al reto de los cambios tecnológicos. Las computadoras han aparecido en las escuelas de nivel superior, medio superior, medio y en contadas escuelas elementales, pero de una manera poco eficiente, ya que más bien han estado enfocadas al Área administrativa, no así a la enseñanza.

Los usos que en educación le podemos dar a la computación son muy variados y diversos. Podemos enumerar algunas aplicaciones que se pueden dar en diferentes disciplinas y en distintos niveles:

- a.- Como instrumento de diagnóstico
- b.- Como programas Tutoriales
- c.- Para prácticas de destrezas
- d.- Para manejar y organizar la instrucción
- e.- Como paquetes (procesadores de palabras, bases de datos, hojas electrónicas)
- f.- Para simulaciones y modelaje matemático
- g.- Para aplicar métodos numéricos
- h.- Para la resolución de problemas
- i.- Como auxiliar didáctico

Podemos dar una breve explicación de cada uno de ellos:

LA COMPUTADORA COMO INSTRUMENTO DE DIAGNOSTICO

Un sistema diagnóstico computarizado tiene un gran potencial como apoyo a la docencia. Idealmente se trata de un paquete de programas interactivos de exámenes que cambian de manera dinámica, según los resultados que produce el alumno como respuesta a las preguntas.

Un paquete computarizado de exámenes diagnósticos interactivos se puede aplicar al principio de un curso y también durante el semestre con el fin de obtener datos para un seguimiento. A partir del desempeño de cada estudiante en los exámenes, es posible obtener un perfil de aprendizaje de cada alumno.

LA COMPUTADORA COMO PROGRAMA TUTORIAL

El propósito de programas tutoriales es el desarrollo de conocimientos y destrezas nuevas en el estudiante. La computadora realmente desempeña el papel de maestro y el estudiante, se supone, va a adquirir habilidades nuevas. Los programas tutoriales presentan por lo regular el material en porciones pequeñas, cuidadosamente secuenciadas, que requieren la participación activa del alumno. Muchos tópicos en matemáticas (como, por ejemplo, la resolución de ecuaciones en álgebra, el cálculo de áreas y perímetros en geometría), se prestan para este tipo de uso de una computadora.

LA COMPUTADORA COMO INSTRUMENTO DE PRACTICA

Hay programas que se diseñaron para ofrecer al estudiante una oportunidad de practicar ciertas destrezas y habilidades, como, por ejemplo, operaciones básicas con números enteros, factorización de

polinomios, etcétera. Estos programas, si están bien hechos, normalmente empiezan con un examen de diagnóstico y, de acuerdo con el resultado, eligen el nivel de ejercicios apropiados para el alumno. Al final presentan un resumen de los resultados o una calificación, que incluso puede ser registrada en una impresora o un disco.

LA COMPUTADORA COMO ORGANIZADORA DE LA ENSEÑANZA

Los términos que se usan para describir las aplicaciones de las computadoras en la organización de la enseñanza son varios.

CAI (*computer-assisted instruction*)

Instrucción con ayuda de computadora.

CAL (*computer-aided learning*)

Aprendizaje facilitado con computadora.

CBI (*computer-based instruction*)

Instrucción basada en computadora.

CMI (*computer-managed instruction*)

Instrucción manejada por computadora.

Estos términos muchas veces se confunden porque tienen significados comunes con respecto a las estrategias educativas, las configuraciones de equipo, los tipos de *software* implantado y los usuarios, entendiéndose como *software* todos los programas o programaje que es intangible y consiste en un conjunto de instrucciones al equipo, que se usa para resolver un problema o efectuar algún tipo de proceso. Hablar de *hardware*, es referirse a los componentes físicos, esto es el equipo de trabajo, es todo lo que puede verse o tocarse.

LA COMPUTADORA CON PAQUETERIA

El uso de procesadores de palabras, bases de datos y hojas electrónicas es cada día más común y en la enseñanza estos paquetes también tienen su lugar. Sirvan los siguientes ejemplos como prueba de ello. El estudiante se comunica a través del procesador de palabras, con el cual escribe sus trabajos. Las bases de datos que almacenan y organizan información tienen muchas aplicaciones. Una hoja de cálculo puede servir para proyectos con temas numéricos, ya que permite la programación de algoritmos y el despliegue de información en forma tabular.

LA COMPUTADORA PARA SIMULACIONES Y MODELAJE MATEMATICO

Las computadoras son muy útiles para el análisis de grandes cantidades de datos numéricos, como los resultados de algún experimento. Con su ayuda es posible representar gráficamente los datos de distintas maneras, calcular parámetros estadísticos, transformar los datos con diferentes escalas, buscar una curva ajustada y probar diferentes modelos para interpretar los datos.

Pero no sólo el manejo de datos numéricos, sino también la manipulación de símbolos puede simplificarse con las computadoras. Existen para ello paquetes utilísimos con sistemas algebraicos y simbólicos.

LA COMPUTADORA EN LA RESOLUCION DE PROBLEMAS

Hay una gran variedad de problemas que se pueden resolver con más facilidad con la ayuda de una computadora, sobre todo si la solución requiere muchos cálculos numéricos y experimentación. En general, podemos afirmar que el proceso de escribir un programa representa ya en sí la solución de un problema. Pero también es posible que el alumno resuelva problemas con software que no desarrolló él mismo.

LA COMPUTADORA COMO AUXILIAR DIDACTICO

La mayor parte del software educativo disponible para la enseñanza se elaboró teniendo en mente la interacción entre un estudiante y una computadora. Casi todos los usos propuestos en el inciso b son de esta naturaleza; la configuración básica consiste en una computadora y un alumno, y deja al maestro el papel de auxiliar de la máquina. No obstante, existe otra función que pueden desempeñar la computadora en la educación: en vez de asignarle al maestro el papel de auxiliar de las computadoras, deberían usarse las máquinas como auxiliares de los maestros, para que éstos pudieran ofrecer instrucción de calidad en el salón de clases.

La configuración para este tipo de aplicación es triangular, en el sentido de crear un ambiente que permite la interacción entre tres elementos: el maestro, el grupo de alumnos y una computadora. Un software adecuado como auxiliar didáctico puede facilitar la labor del maestro en la preparación de sus clases, realizar su capacidad para presentar nuevos temas y animarlo a probar nuevas estrategias pedagógicas.

1.2.- ANTECEDENTES PARA EL LENGUAJE LOGO

Algunos educadores creen que las computadoras en el aula son otra distracción más.

Otros temen que el uso continuado de las computadoras en la escuela produzca una población que no sepa sumar ni restar.

Otros creen que las computadoras son otra pompa de jabón, como los laboratorios de idiomas o el uso de la TV en clase y que no modificarán realmente la educación.

El recelo con que se ha acogido la introducción de las computadoras en el ámbito educativo se comprende perfectamente porque hasta hace pocos años la mayoría de las experiencias conocidas nos llegaban envueltas en un aura de mecanicismo y conductismo, que las hacía profundamente sospechosas de reducir al alumno a un papel pasivo, y también limitadas en cuanto al espectro de actividades intelectuales que proclamaban ayudar a desarrollar.

Los siguientes puntos tratan de poner de manifiesto las limitaciones de la implantación de la informática en la escuela o que han constituido un factor negativo en las experiencias desarrolladas:

1.- El haberse limitado a objetivos pobres, tales como la enseñanza de la programación con una metodología incorrecta o utilizando un lenguaje no estructurado tal como el BASIC.

2.- El haber comenzado a utilizar programas de enseñanza asistida por computadora mal diseñados, de ejercitación mecánica, que han desprestigiado el tema ante el profesorado.

Entre los problemas que están retrasando la aplicación de la tecnología informática a la enseñanza, en cuanto actividad educativa, se cuentan, por lo menos dos importantes:

- 1.- La inversión necesaria para dotar un aula con un equipo suficiente de computadoras apropiadas.
- 2.- La disponibilidad de paquetes de aplicaciones útiles.

Dentro de este segundo punto es donde cabe un debate más revelador sobre qué esperamos y exigimos del uso de las computadoras en el aula. Sin embargo, aunque el avance en este sentido es más lento que en el primero, ya se está empezando a sentir la necesidad de que estas aplicaciones se adapten lo mejor posible a nuestro sistema educativo, a nuestra cultura e idiosincrasia, con unos niveles mínimos de calidad.

Los maestros pedimos demasiado, pero ¿nos referimos a la computadora o al software? Las objeciones al uso de las computadoras en el aula muchas veces son esencialmente críticas no al equipo informático - hardware- sino a los programas de aplicación -software-. Es de ellos de los que se esperan maravillas y no el sustituir a los maestros. Esperamos y confiamos que esta nueva herramienta posea una versatilidad y unos recursos suficientes para cooperar en el aprendizaje, sin pretender que desarrolle capacidades por encima de lo que puede ofrecer un buen profesor en el aula. Sin embargo, una reflexión en el enfoque del uso de la computadora en el aula puede suministrar nuevas perspectivas de trabajo.

La computadora entró en la escuela en primer lugar orientada hacia el profesor, inconscientemente pensando en que podría sustituirlo, tal como entró en las fábricas; sin embargo, tampoco en las fábricas consigue amenazar la existencia de los obreros -aunque en la práctica sí reduzca su número, sino que puede sustituirle en la realización de procesos difíciles o imposibles anteriormente. También podría estar destinado a mejorar los procesos de aprendizaje de los alumnos y entrar en las aulas tal como entró la calculadora, el video o el bolígrafo para sustituir a la escritura con plumilla. Probablemente en este campo es donde no ha fracasado todavía: cuando se ha intentado convertir la computadora no en el centro del aula, sino en una extensión instrumental de la mente del alumno.

En definitiva, se trata de replantear el enfoque. Dirigir el uso de la computadora no sólo en los maestros sino fundamentalmente a los alumnos, dedicando en este sentido los recursos económicos, no a informatizar la parte administrativa, sino la atención personal del alumno. Usar la computadora no sólo para enseñar asignaturas, sino como una herramienta útil para un aprendizaje cuyo eje no tiene por qué ser necesariamente la computadora, ni convertir ésta en la administradora de la enseñanza.

Retomando la idea primitiva de Papert, de hacer accesible la informática a la educación, podemos proporcionar a los estudiantes otro instrumento que también tenga en cuenta los factores emotivos, que, en lugar de hacer que se sientan inferiores, les ayude en lo que se hace en las escuelas: aprender. Intentar nuevos enfoques creativos de cómo utilizar la informática en la escuela y aprovechar como motivación, por lo menos en los primeros estadios, el "poder de captación" de la computadora como una valiosa herramienta educativa.

APRENDIZAJE POR COMPUTADORA

En este contexto nace LOGO como una alternativa, con la pretensión de familiarizar a los niños con el uso de la computadora, acercarlos al mundo de la informática, favorecer su progreso en el campo de las matemáticas desde una óptica completamente nueva y en cierta manera, abrir nuevas perspectivas que permitan emprender el camino de aprender en serio, es decir, aprender a pensar y aprender a aprender.

Para este nuevo uso habrá que modificar el planteamiento general: cambiar la relación que se establece entre el niño y la computadora de manera que no se le utilice para programar al niño, sino que sea el niño quien programe la computadora.

Se trata de permitir que el niño se convierta en el constructor de sus propias estructuras intelectuales, aprovechando su facilidad para aprender a usar las computadoras y que al hacerlo, modifique su manera de aprender todo lo demás. Si esto fuera posible, la computadora se habría convertido en una poderosa ayuda intelectual cuya efectividad, una vez que los estudiantes hubieran asimilado los métodos de trabajo, podría aplicarse a los demás campos de su aprendizaje sin necesidad de soporte material de la computadora.

Por lo tanto no sabemos cuál va ser el resultado de las líneas de trabajo que se ha venido dando hasta ahora y todavía menos a largo plazo. Pero en el campo informático no se trata de aplicar fórmulas, sino de descubrir soluciones; con LOGO se pretende crear un ambiente que sea dominado por el niño.

1.3.- EL MARCO HISTORICO DE LOGO

Durante la breve historia de la informática se han ido diseñando diferentes lenguajes de programación de alto nivel respondiendo, en cierta manera, a la evolución tecnológica a las crecientes demandas surgidas de la actividad humana.

Para situar al Logo en su contexto, dentro del espectro de lenguajes de alto nivel, vamos a trazar una panorámica de la evolución de los mismos. Contando con los peligros que supone hacer simplificaciones, pero con el intento de clarificar la situación del Logo en el panorama de la informática en general, vamos a distinguir, históricamente, tres generaciones que a grandes rasgos corresponden con las tres décadas de los años cincuenta, sesenta y setenta.

A finales de los años cincuenta aparecen los lenguajes de alto nivel que, aunque hoy en día hayan quedado en gran medida reducidos a un papel fundamentalmente histórico, en su momento permitieron a los programadores comunicarse con la computadora de una manera más "humana"; probablemente el FORTRAN (1956), diseñado para el cálculo científico, sea el primer lenguaje de alto nivel de amplia difusión, relegando al FLOWMATIC, SPEED CODING y SHORTCODE a ensayos. En 1958 aparece el ALGOL y en 1959 el COBOL, promovido por un comité estatal norteamericano, CODASIL, y diseñado especialmente para fines administrativos.

En estos tres lenguajes está el embrión de lo que serían las líneas del futuro desarrollo en la década siguiente. En primer lugar, la especialización en campos de aplicación preferente y, en segundo lugar, el papel precursor del ALGOL como inspirador de los demás

APRENDIZAJE POR COMPUTADORA

lenguajes de tipo estructurado. El ALGOL fue desarrollado a partir de los trabajos de P. Naur y en su diseño, en principio orientado a las implicaciones de cálculo científico, se introdujeron ideas tan importantes para el futuro como la estructuración y la recursividad. El FORTRAN, diseñado por J. Backus, es preferido para las aplicaciones científicas y de cálculo con una amplia aplicación todavía en la actualidad y el COBOL en el campo de la gestión y administración, aún muy utilizado hoy en día en este terreno, a pesar de tratarse de un lenguaje ya superado.

A primera vista puede parecer extraño que en lugar de existir una tendencia general hacia la unificación de las maneras de comunicarse con las computadoras, la realidad sea la contraria: cada vez existen más lenguajes con el contraste evidente de que las computadoras, por su parte, siempre utilizan el mismo código: el binario. Esta diversificación, para muchos usuarios de la informática, supone un inconveniente más que hay que superar para aprovechar sus posibilidades.

Los años sesentas ven aparecer una gran multitud de lenguajes nuevos, en los que las dos líneas de desarrollo (especialización y estructuración) se mezclarán en muy diverso grado. Del ALGOL surgirá el PASCAL (N. Wirth, 1970), para aplicaciones de tipo general y como modelo de lenguaje estructurado. El LIPS, desarrollado a principios de los años sesenta en un departamento del Massachusetts Institute Technology (M.I.T), el LOGO y PROLOG, aunque el segundo tenga, además, aplicación en el sistema educativo, serán lenguajes especializados en problemas de inteligencia artificial. El FORTRAN

tendrá como heredero directo al BASIC, de uso general, implementado en la mayoría de las pequeñas computadoras y muy accesible al usuario-programador, diseñado en 1964. Otros, sin embargo, intentan ser de aplicación universal, como el PLI, desarrollado en 1960, descendencia del ALGOL y COBOL.

Para terminar, sobre la década de los años setenta, citaremos solamente el más representativo, el ADA. Aparece en 1978 y recoge todos los avances anteriores, estando dotado de una gran potencia y rapidez. Fue el resultado del trabajo de Jean D. Ichbiah y su diseño patrocinado por el Departamento de Defensa de Estados Unidos, una acción institucional como anteriormente lo había sido el COBOL. Se trata de un lenguaje estructurado, con "packages", proceso paralelo, tratamiento de excepciones, etc., capaz de ser utilizado en un amplio espectro de aplicaciones, con aspiraciones de llegar a ser de uso universal a pesar de su sofisticación.

En el proceso de desarrollo del LOGO participaron tres elementos que ya se han observado en los orígenes de muchos otros lenguajes: algún organismo estatal que suele promover, patrocinar o coordinar el trabajo que se realiza en alguna universidad, con la colaboración de su personal y dirigido por algún departamento y la implicación de la empresa privada en el proyecto, aportando parte del capital con el propósito de comercializar el producto obtenido.

Uno de los precursores del desarrollo del LOGO fue H. Abelson, nos relata que este lenguaje nació en 1968 como iniciativa de la National Science Foundation, y se encargó a una empresa con arraigo en el campo de la informática. Sin embargo, la parte esencial de su estudio

y diseño se realizó en el Massachusetts Institute of Technology (M.I.T.) bajo la dirección de Seymour Papert y se efectuaron los primeros ensayos experimentales en escuelas piloto. También colaboraron otras universidades como la Universidad de Edimburgo, Massachusetts, entre otras. Como antecesor del LOGO suele citarse el LIPS, también desarrollado en el M.I.T. y se le podría considerar una consecuencia informática del mismo, aunque profundamente transformada para hacerlo accesible a un espectro de usuarios mucho más amplio -el LIPS ha estado mucho tiempo en manos de investigadores que trabajan en el campo de la inteligencia artificial- quizás el aspecto que más diferencia a los dos lenguajes desde el punto de vista externo es el que se le ha añadido la posibilidad de comandar el movimiento de un objeto (la "tortuga") por la pantalla o en el mundo real.

LOGO surge como resultado de la investigación de un equipo de trabajo del M.I.T. que se proponía crear un lenguaje de utilidad específica en el campo de la educación, teniendo como antecesor directo al LIPS y como lejano al ALGOL. En principio se implementó únicamente en grandes ordenadores, pero desde su creación, en 1968, ha estado en continua evolución y como se trata de un lenguaje de alto nivel complejo y sofisticado, la mayoría de la investigación con él en la década de los años setenta se hizo con grandes computadoras. Sólo recientemente, al aparecer en el mercados las pequeñas computadoras, se han hecho implementaciones accesibles al tipo de máquinas que se utilizan en las escuelas, de bajo costo y mediante las cuales se han podido desarrollar experiencias que podríamos denominar de "laboratorio" típicas de los primeros tiempos del LOGO. De estas implementaciones en computadoras chicas, históricamente las primeras

fueron realizadas por el *Logo Group* del M.I.T. para dos de las máquinas más difundidas en los centros de enseñanza de Estados Unidos: el *Apple II* y el *Texas Instruments 99/4*.

1.4.- TEORIAS PEDAGOGICAS EN QUE SE APOYA LOGO

Sin intentar personalizar en exceso en S. Papert, el proceso de la creación del Logo podemos decir que en el director del equipo Logo del M.I.T. confluye la influencia de dos importantes pensadores del campo de la psicología y la pedagogía: John Dewey y Jean Piaget.

Norteamericano el primero, casi un personaje del siglo pasado, fundamentalmente un filósofo universitario, y europeo el segundo, un psicólogo cuyas investigaciones en el campo del proceso de aprendizaje han transformado profundamente la pedagogía contemporánea, Dewey y Piaget son claves fundamentales para comprender la filosofía subyacente en el Logo.

Seymour Papert siguió la tradición pedagógica de Dewey y fue colaborador de Piaget en Europa, y tanto el enfoque de los objetivos de la educación del primero como el análisis del proceso del aprendizaje del segundo, impregnan los conceptos esenciales de sus propuestas sobre el uso del Logo en la escuela. Sin embargo, Papert hace un intento de definición del papel de la informática en la educación que tiene tanto de asimilación de la epistemología genética como de aplicación de las investigaciones que, juntamente con Marvin Minsky, realizó en el campo de la inteligencia artificial.

Además, aplica todo esto a una propuesta netamente original: la concepción de que el desarrollo de la inteligencia infantil puede acelerarse mediante la utilización de modelos u objetos que le sirvan al sujeto como ayudas, puntos de referencia y contraste en la

construcción y generación de sus estructuras mentales, a la vez que presupone que el esquema corporal puede servir de soporte del aprendizaje.

1.4.1.- Dewey

De larga vida (93 años), John Dewey (1859-1952) fue un filósofo y pedagogo que trabajó en el campo de la reforma de la educación en Norteamérica, proponiendo uno de los muchos intentos de renovación pedagógica que a finales del siglo XIX y principios del XX fueron sentando las bases de un nuevo espíritu en el quehacer escolar. De su aportación a la moderna pedagogía se deben seleccionar, para el tema que nos ocupa, los siguientes considerandos:

Para Dewey, obrar científicamente es obrar experimentalmente. Se llega a considerar que cualquier acto del pensamiento no es sino un tipo especial de acción o práctica. Como consecuencia lógica, la validez de sus ideas radica en su capacidad de ser para la acción. Su ideal de la formación en la escuela es el "aprender obrando".

No era suficiente con que el educador expusiera un conjunto de conocimientos al alcance del niño, sino que para que el proceso de aprendizaje fuera eficaz era imprescindible que los contenidos objeto del aprendizaje contaran previamente con el interés y el asentimiento del estudiante, por lo que el proceso educativo debía empezar primero por crearlo en el sujeto. En este proceso de selección de los contenidos es donde el educador habría de intervenir para que cupiera la posibilidad de relacionar íntimamente el aprender con el hacer, ya que sólo se aprende si se hace y se aprende sólo lo que se hace.

El proceso educativo debe crear primero el interés en el niño y el papel del profesor es decisivo en este estadio, pues debe procurar

que sus alumnos se motiven hacia temas en lo que exista de una manera evidente la posibilidad de mezclar el pensar con el hacer, o de pensar mientras se hace, o de hacer para pensar.

En este contexto de la pedagogía de la actividad en el que se enmarcar la idea de que el educador no tiene por qué participar en el proceso de "hacer" del niño, sino que es algo que éste debe realizar por sí mismo; el educador se convierte en cotrabajador con él, es el que organiza desde "abajo" y no la figura que ordena, manda y dirige desde arriba. El profesor pierde esa superioridad que caracterizaba los peores ejemplos del sistema escolar de su época, muy criticado por él, y se convierte en un colaborador del alumno.

1.4.2.- PIAGET

No es necesario presentar extensamente las aportaciones de Jean Piaget a la psicopedagogía en general, pero su trabajo en el campo del desarrollo de la inteligencia en el niño ha pasado a informar de tal manera el lenguaje Logo a través de Papert, que es necesario recordar cuáles son sus fundamentos, sobre todo sus teorías respecto a que el proceso de aprendizaje no se puede producir en condiciones de acto reflejo, de comportamiento automático, o en situación de aprendizaje pasivo y memorístico, ya que sin la manipulación no se puede realizar el proceso de acomodación y asimilación de las ideas; todo este tipo de actividades se ven reducidas a ejercicios puramente gimnásticos, sin ningún aporte intelectual, o como señala J. Palacios "la escuela tradicional embute en el alumno una considerable cantidad de conocimientos, "amuebla" su inteligencia y la somete a una "gimnasia mental" que le facilita un pretendido fortalecimiento y desarrollo". (1)

Un elemento fundamental de la aportación de Piaget a la pedagogía moderna es la pedagogía operatoria, en la que se trata de dejar que el niño llegue a la maduración del proceso lógico poniéndole en situaciones en las que necesite ese proceso.

El niño va incorporando a sus estructuras lógicas mentales los resultados de las experiencias que continuamente va adquiriendo al manipular los objetos de su alrededor. "Conocer un objeto es operar sobre él y transformarlo para captar los mecanismos de esta transformación en relación con las acciones transformadoras" (2)

Es cualidad importante de esta adquisición el poder realizar la operación al derecho y al revés, comprobar que se ha adquirido el concepto mediante su verbalización con uno u otro medio de expresión (oral, escrita, por medio de dibujos...) y realizando el mismo proceso en ambas direcciones, desarrollo y reconstrucción, con lo que se demuestra que se conocen todos los elementos del proceso y todas las relaciones que existen entre ellos. Solamente se puede decir que se ha adquirido el proceso lógico cuando se comprueba esta dualidad. La sintonicidad del mundo de la tortuga es el equivalente de la importancia del tema de la psicomotricidad en el proceso de aprendizaje, de la importancia de adquirir el esquema corporal de la operación, de efectuar las operaciones con el cuerpo, de dominar los conceptos de adelante, atrás y la lateralidad.

(1).- J. Palacios. "La cuestión escolar", México 1984 Pág. 492

(2).- J. Piaget. "Psicología del Niño", Madrid 1984 Pág. 133

Hacer pedagogía operatoria no es algo que pueda realizarse de una forma uniformizada - " todos los alumnos resuelvan el cuestionario de la página x "- sino que ha de creársele la necesidad antes de presentar el material.

De otro modo, en el caso de que el material no consiga una respuesta adecuada por parte del alumno , realmente no sirve de nada intentar corregir o cambiar el resultado de su trabajo, ni hacerle borrar los errores de su libreta. Habría que aprender a respetar las conclusiones de nuestros alumnos. No decir "está mal" sino "continúa escribiendo..." hasta que él mismo, al descubrir el concepto que se está tratando, se dé cuenta de su falsa interpretación y modifique sus conclusiones. Hacérselo borrar no es que sea intrínsecamente malo, simplemente es inútil. El error desaparece de su libreta, pero no por eso ha adquirido el concepto, ni lo que está escrito le sirve intelectualmente para nada. Sólo se conseguirá que pueda , en todo caso, llegar a aplicar mecánicamente fórmulas que para él no significan nada. En el caso de que no demuestre haber asimilado un concepto, siempre se le pueden suministrar otros ejemplos o materiales, que impliquen la misma operación o estructura y continuar haciéndolo hasta que pueda verbalizarlo y utilizarlo en ambos sentidos (los objetos se pueden tratar reversiblemente).

1.4.3.- MINSKY

En el intento de enseñar a "pensar" a los ordenadores se va adquiriendo experiencia que arroja luz sobre el conocimiento de nuestra propia inteligencia, pudiéndose aprovechar lo aprendido para mejorar nuestra manera de pensar.

Esta es la idea implícita en Logo, donde al reflexionar acerca de un modelo informático se adquieren unos mecanismo de conocimiento que mejoran cualitativamente nuestra manera previa de pensar.

1.4.4.- S. PAPER

El trabajo de Papert puede enjuiciarse desde dos puntos de vista: como propulsor de una aplicación radicalmente renovadora del uso de la informática en la escuela y como creador y diseñador, conjuntamente con su equipo del M.I.T., de una "ayuda para pensar": el Logo y en particular la tortuga gráfica.

Papert, discípulo de Piaget, amplió y concretó su línea de pensamiento en dos temas fundamentales:

- que el aprendizaje es un proceso que se realiza a partir de la manipulación de objetos y

- que el proceso de aprendizaje es genético: lo que el niño puede aprender depende de las estructuras que ya posee y que el crecimiento y aparición de las mismas se desarrollan una a partir de otra; en el proceso de aprendizaje los nuevos elementos deben basarse en otros anteriores, por lo que el aprendizaje debe ser constructivo y acumulativo.

Logo es el resultado de una reflexión, tal vez radical, sobre las ideas educativas en las que debería basarse una utilización renovadora de la computadora en la escuela. Este lenguaje constituye una herramienta intelectual mediante la que su praxis resulta lo más próxima posible al funcionamiento real del pensamiento del niño, ajustándose al máximo a la estructura de su inteligencia, respetándose los presupuestos de Piaget y permitiendo usar la

computadora en una extensión de su mente, que le sirva para pensar y que le ayude a este fin. Para ello precisa de un instrumento que sea sencillo de utilizar. Se trata de convertir la computadora en un lápiz inteligente, en un instrumento de uso diario, corriente y accesible a todos.

Papert se inclina por el constructivismo. El niño es el sujeto que, lejos de adquirir pasivamente los conocimientos, participa activamente en su propio proceso de aprendizaje y se constituye en el conductor de su propio razonamiento.

Los objetivos fundamentales que inicialmente perseguía Papert con la utilización del lenguaje Logo eran el aprendizaje de las matemáticas y luego la familiarización de los niños con la informática.

Para cubrir estos objetivos iniciales nace el Logo, un lenguaje informático fácilmente dominable, potente y próximo al lenguaje cotidiano. El análisis de las pruebas que se llevaron a cabo con él en el aula mostró que su uso cubría también otra serie de objetivos pedagógicos ya que, al haber incorporado al mismo los nuevos elementos de la psicopedagogía, se convertía en un ejercicio intelectual excelente por sí solo, capaz de ser aplicado a cualquier campo del aprendizaje.

Papert propone la tortuga, los gráficos que proporciona la tortuga, como una ayuda para el desarrollo intelectual del niño, poniendo el ejemplo del papel que desempeñaron los engranajes del diferencial del automóvil en su propio desarrollo intelectual: un objeto transaccional.

Logo y las computadoras pueden convertirse en un atajo para que los niños de la cultura contemporánea, al menos en las sociedades avanzadas, puedan acelerar su desarrollo intelectual. El niño puede

utilizar la tortuga, en su formación, como un modelo de aplicación, un objeto al que se puede amar y superar así las múltiples dificultades que a nivel afectivo impone la sociedad y el ambiente escolar al aprendizaje de ciertas materias, especialmente las matemáticas. Un objeto que, para terminar, dispone de la suficiente flexibilidad para que con él se puedan realizar, incluso, complicadas operaciones intelectuales y, además, usarlo de una manera individualizada.

Como descubrimiento y aportación original de Papert podemos citar la idea de que el esquema corporal puede servir en gran medida de soporte al aprendizaje, es decir, que la sintonía facilita la investigación y la experimentación.

CAPITULO II

CAPITULO II: CARACTERISITICAS PRINCIPALES DE LOGO

2.1.- LAS CARACTERISTICAS DEL LENGUAJE LOGO

El lenguaje Logo goza de las siguientes características:

Características fundamentales:

1.- Fácil

Transparente

Lenguaje Natural

Sencillo

2.- Basado en objetos

3.- Extensible

Recursivo

Analitico

Combinatorio

Características complementarias:

4.- Interactivo

2.1.1.- Fácil

Es importante que sea muy fácil de usar para que, desde el primer momento, la gran mayoría de los usuarios a los que va destinado, encuentre factible realizar con él tareas de investigación o experimentación, en su campo de interés personal. Ser fácil quiere decir que ya en la primera sesión con la computadora, el alumno se sienta capaz de empezar a hacer cosas y poder realmente hacerlo.

En la mayoría de los lenguajes que tienen un vocabulario fijo, aprender a programar empieza cuando ya te has familiarizado con todos los vocabularios.

Para utilizar el lenguaje en pruebas y ensayos y manejar ideas con facilidad, es necesario que éstas se manipulen de una manera ágil y evidente y que, una vez comprobadas, puedan desecharse en caso de que no sirvan, o modificar las que tengan algún elemento aprovechable y, sobre todo, que sea un instrumento que permita al usuario concentrarse en la tarea creadora sin preocuparse de los elementos que podríamos llamar "técnicos". La comodidad del lenguaje tendría como objetivo ofrecer un campo abonado para el desarrollo de la creatividad.

Cuanto más fácil sea modificar y cambiar elementos de esas ideas, en este caso los procedimientos, más fácil será trabajar con ese instrumento. Para ello existe un editor integrado en el Logo, para facilitar el escribir y modificar todos los procedimientos de modo que, una vez realizado, éstos se incorporan de manera automática sin tener que utilizar compiladores u otros artificios informáticos.

Para que un lenguaje informático se convierta en un instrumento natural y cómodo de usar y con él se pueda dar rienda suelta a la creatividad, es necesario que el umbral de partida sea mínimo para, desde allí, poder progresar y desarrollar el conocimiento y uso del mismo.

Quizás el elemento más novedoso que Logo aporta al mundo educativo sea el empleo de la tortuga gráfica, puesto que provee al niño de un objeto fácilmente manipulado por la pantalla. Aunque antes de Logo ya existían otras herramientas pedagógicas que permitían que el alumno investigara y experimentara a su ritmo, éstas podían resultar más complicadas que el trabajo que hace la tortuga, el umbral de conocimientos previos se reduce, ya que puede experimentarse con ella desde el momento en que se distinguen unas pocas teclas del teclado

(evidentemente esto requiere de un microuniverso de experimentación adecuada), y podríamos imaginar una situación mucho más idónea si en el intento de acercar el Logo al usuario, éste pudiera dar las órdenes de viva voz en lugar de comunicarse a través del teclado, eliminando así un medio artificioso de hacerlo. No sería de extrañar que, en sucesivas versiones de este u otro lenguaje parecido, apareciera ya implementada esta modalidad, lo cual acercaría mucho más al individuo, sobre todo niños pequeños e incluso a aquellos que por causa de algún impedimento físico les resulta prácticamente imposible el manejo de un teclado estándar.

En la misma línea de experimentación sensorial que el Logo gráfico cabría nombrar el Logo musical, aunque este último no esté tan extendido como el primero. Puesto que esta familia de lenguajes está en continuo desarrollo, es posible imaginar una evolución del mismo de manera que el acercamiento de la computadora al niño se haga extensiva a otras áreas de su formación.

2.1.1.1.- Transparente

Técnicamente hablando, se trata de un lenguaje que puede usarse sin que previamente se deba conocer el funcionamiento interno de las computadoras o se tenga que ir profundizando obligatoriamente en esos temas conforme se va avanzando para poder sacar partido de sus posibilidades. El Logo soluciona todos los problemas de comunicación y relación con la computadora sin que el usuario tenga que pararse a pensar en cómo organizar la máquina por dentro.

Creemos que se trata de una característica fundamental y básica del lenguaje, puesto que permite utilizarlo sin tener que profundizar en los temas hardware ni sumergirse en los complicados temas del manejo de la memoria de la computadora o memorizar gruesos manuales.

2.1.1.2.- Lenguaje natural

La mayoría de los lenguajes informáticos toman su vocabulario del inglés, lo cual, para personas de otra lengua, constituye una dificultad añadida al ya difícil campo de la programación. Ciertamente, para adultos esto puede representar un mal menor, pero el problema se agrava cuando pretendemos que sean los niños los que dominen la máquina.

De aquí la importancia que reviste la posibilidad de usar el Logo en lengua materna, evitando de esta manera el problema de la traducción a otro idioma. Esto ayuda a aproximar el Logo a un lenguaje natural, haciendo que las órdenes y comandos tomen su vocabulario del entorno cotidiano y por lo tanto, resulten más cercanas, más fáciles de recordar y sobre todo, más inteligibles.

2.1.1.3.- Sencillo

La sencillez del manejo del lenguaje tiene dos elementos que merecen la pena resaltar:

- a) que utiliza un sólo tipo de variables.
 - b) que manipula listas.
- a) Aunque este renglón ha sido criticado desde el punto de vista informático, resulta de gran ayuda para aquéllos que se inician. Se simplifica el trabajo con las variables todo lo posible, lo que

APRENDIZAJE POR COMPUTADORA

permite ceñirse exclusivamente al uso que de ellas se va a hacer, diferenciando tan sólo el nombre del contenido.

Esta sencillez allana el camino para los usuarios sin conocimientos matemáticos y evita que se asocie la informática con las matemáticas, un tema que tradicionalmente inspira a los alumnos las ideas de difícil, abstracto, teórico y fracaso. Por otro lado incluye una fuente de dificultades debido, primordialmente, al mal uso de las normas sintácticas, pero resulta de gran valor por su similitud con el manejo de los símbolos que se requieren en otras disciplinas del currículum escolar, tales como matemáticas o física, y sobre todo porque inicia de una manera natural en el hábito de la simbolización.

b) Es decir, que permite tratarlas como un todo constituido de elementos individuales (las palabras), diferenciándose así de la forma en que lo hacen otros lenguajes informáticos de alto nivel. En Logo pueden tratarse listas enteras o elementos de estas listas, de forma similar a lo que se hace con las frases y palabras del lenguaje cotidiano, y no como un conjunto de caracteres encadenados.

El manejo de las listas en el Logo es una de las herencias del LISP, del cual incorpora los avances fundamentales, pero sin embargo, su uso suele presentar dificultades de las cuales se hablará más adelante.

2.1.2.- Basado en el uso de objetos

Para que el aprendizaje sea un proceso operatorio, el niño tiene que manipular y experimentar y para ello necesita algo, objetos, aún cuando sea capaz de manipular representaciones, ya que las acciones

APRENDIZAJE POR COMPUTADORA

mentales se crean mediante la manipulación de objetos y se sostienen y desarrollan mediante una continuación de tal contacto.

El lenguaje Logo propone un conjunto de objetos a disposición del usuario para que puedan producirse situaciones propicias al aprendizaje por medio de su manipulación, y para que el uso de la computadora se acomode todo lo posible al mismo esquema que sigue el proceso de aprendizaje que podríamos llamar natural, del que, en muchas ocasiones, el proceso de manipulación de los objetos de su entorno, de reconstrucción de secuencias, de reordenación de los mismos usando nuevos criterios, de reagrupación de forma distinta es un primer paso, previo a la acomodación y asimilación en las estructuras lógicas del individuo.

Sin ellos no sería posible aplicar el principio de que el aprendizaje se realiza por medio de la manipulación y de la acción, de que se aprende lo que se hace, lo que se experimenta, tal como viene reflejado en la cita de Dewey:

"El pensamiento fecundo integral no es posible sino donde, en alguna manera, se emplea el método experimental." (3)

La manipulación de los objetos va proporcionando un caudal de resultados que se incorporan a las estructuras mentales del sujeto y que van completando sus conceptos. Este proceso de racionalización se perfecciona con la simbolización, algo que el mismo uso de la informática fomenta de manera natural. Este proceso de simbolización es previo y necesario para que la adaptación de las estructuras sea completa.

Los objetos son elementos que el usuario puede manejar, transformar, mover, manipular en un espacio informático abstracto, que se pueden describir como una formalización informática o una simbolización de un objeto real. En Logo se puede experimentar con ellos porque tienen unas cualidades que se modifican como resultado de las manipulaciones, el status de la tortuga, por ejemplo, y porque es posible referirse a ellos por medio de variables.

Logo ofrece tres objetos para experimentar: la palabra, la lista y el número, aunque, en otro orden, también se podría considerar como objetos a los procedimientos creados o incluso a los gráficos de la tortuga.

2.1.3.- Extensible

La tercera característica fundamental incorporada en el Logo es que se trata de un lenguaje extensible. Entendemos por extensible que permite crear nuevas órdenes, que son procedimientos (listas de órdenes que se ejecutan secuencialmente) creados por el usuario, funcionalmente idénticos a las primitivas del lenguaje y que se pueden usar, por lo tanto, en otros procedimientos como si aparentemente se hubieran incorporado al lenguaje. Esta extensibilidad del vocabulario es uno de los factores que le dan potencia y flexibilidad.

En Logo existen unas palabras especiales, comandos u operadores, que pueden usarse directamente y que constituyen el núcleo tradicional de todo lenguaje de alto nivel, pero además permite ampliar sin límite este vocabulario básico; es más, sin que sea necesario que el usuario las conozca todas al principio, es aconsejable animarle a que vaya

creando sus propias órdenes personalmente, de una manera individualizada y a medida que las necesite, ampliando el vocabulario en el campo de trabajo que más le interese. Estas órdenes Logo le resultarán fáciles y agradables de usar porque las ha creado él mismo, con sus nombres inventados y sus acciones que él conoce bien. La capacidad de ser extensible es interesante desde dos puntos de vista. En primer lugar, compatibiliza la cualidad de tener un umbral inicial bajo con la posibilidad de realizar acciones interesantes desde el primer momento y no agotarse en seguida. En segundo lugar, permite que el lenguaje pueda crecer con el niño y sea aplicable a tareas cada vez más complejas según se vaya necesitando. Se convierte en un lenguaje progresivo que permite la elaboración de algoritmos simples y complejos pudiendo llegar a poner en juego métodos de resolución de problemas muy elaborados.

El propio hecho de la extensibilidad mediante la creación de procedimientos hace que el Logo sea un lenguaje con capacidad recursiva, que permita trabajar combinatoriamente y que favorezca una metodología analítica.

2.1.3.1.- Es un lenguaje recursivo

El concepto de recursividad podríamos definir como la posibilidad de incluir un elemento constituyente de otro de la misma naturaleza un número indefinido de veces; o la posibilidad de crear un proceso prácticamente infinito por medio de un conjunto finito de reglas.

El Logo permite trabajar recursivamente porque los procedimientos utilizan variables locales, cuyos valores no se confunden con los de las variables de otros procedimientos.

En estas condiciones puede pensarse que una de las órdenes de un procedimiento, pueda llamar a otro idéntico a sí mismo, el cual por propia naturaleza irá perpetuando las llamadas sucesivas a otros iguales y así indefinidamente. Un proceso de esta naturaleza nunca concluirá a no ser que se haya previsto una condición de parar.

La recursividad es una modalidad especial de la repetición, un tipo de repetición en la que no se conoce en principio el número de veces que va a ser necesaria, que incorpora una actualización automática de los parámetros y se puede localizar en cualquier punto del proceso de acciones previstas con la limitación de que no sea la primera.

He aquí un ejemplo de proceso recursivo:

TO CIELO

FD 50

RT 90

CIELO

END

2.1.3.2.- Analítico

Otra característica del Logo relacionada con el hecho de disponer de procedimientos es el ser analítico. Su uso introduce desde muy pronto la costumbre de descomponer y analizar los problemas, dividiéndolos en sus partes constituyentes, con los que los problemas difíciles o grandes se dividen en problemas más pequeños que, no solamente en apariencia sino también en la realidad, son siempre más fáciles de solucionar.

Para realizar los trabajos de experimentación e investigación con el Logo, el usuario se verá inmerso de una manera natural en el proceso

de descomponer un programa de actuación en partes pequeñas, que le resulten manejables -proceso de análisis- y de recomponer en el orden apropiado las distintas partes en un todo -proceso de síntesis-.

Este proceso de síntesis es más complejo que la simple yuxtaposición de cada uno de los apartados, ya que requerirá un claro conocimiento de cómo deben interaccionar unas piezas con otras.

Al ejercitarse en el análisis de los problemas, en el estudio de las estrategias de descomposición, de síntesis de lo esencial de cada apartado, realizando tareas que el propio usuario se ha planteado, se está introduciendo y familiarizando con la programación modular, con el hábito de descomponer los problemas y analizar estructuradamente.

2.1.3.3.- Combinatorio

Una tercera característica derivada de ser un lenguaje procedural es la de ser combinatorio. La propia manera en que se construyen los procedimientos, combinando órdenes en conjuntos secuenciales o grupos ordenados, es combinatoria; capacidad que se amplía puesto que en el interior de los procedimientos se pueden utilizar otros procedimientos, pudiendo resultar de ello un programa de gran complejidad.

La diferencia entre programa y procedimiento no va más allá de una simple nomenclatura, pudiéndose usar indistintamente, aunque se acostumbra reservar la palabra programa para nombrar al conjunto de órdenes y procedimientos que tienen un fin en sí mismo. Un programa puede constituirse en procedimientos cuando se usa como objeto que forma parte de otro más complejo.

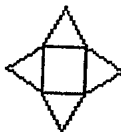
Por combinatorio entendemos que un mismo procedimiento puede utilizarse en diferentes programas y proyectos, por lo que se

constituyen en algo así como en piezas elementales, con las cuales puede contarse para construir otras estructuras se recomponen de manera diferente. Los procedimientos pueden ser piezas de una estructura y gozan de la característica de ser polivalentes.

De esta manera se convierten en objetos que pueden combinarse de varias maneras, encajar unos dentro de otros o crear superestructuras que los engloben. Con los objetos CUADRADO y TRIANGULO creados por el usuario se pueden construir otros muchos por simple combinación de ellos, por ejemplo:



Sobre



Pifata



Mantel



Casa

Los procedimientos sirven para que puedan resolverse los problemas grandes, combinando sus piezas elementales constituyentes. La adquisición de la idea de trabajar los problemas dividiéndolos en sus partes, realizando un análisis y separando los subproblemas de que consta, puede eliminar gran parte de la dificultad y es un buen método de trabajo en el campo de la programación de ordenadores y en la resolución de cualquier tipo de problemas. La ventaja del Logo es que facilita la adquisición de este método de trabajo desde sus estadios iniciales.

El trabajo de exploración en la clase de Logo irá dando por resultado una serie cada vez más larga de procedimientos, que estarán a disposición del usuario como una pequeña biblioteca de herramientas

de uso. Esta batería de piezas le sirve para crear o modificar los programas con facilidad, a partir de las piezas o partes ya construidas, probadas y conocidas.

Debido a que en el Logo hay gran facilidad para la variación, modificación y mejora de los procedimientos, éstos pueden adaptarse con rapidez a las nuevas ideas y se facilita el crecimiento rápido de los futuros proyectos.

La importancia de que sea combinatorio reside en la capacidad para utilizar las experiencias pasadas, y construir sobre, o a partir de ellas.

Como resultado de la implementación de estas ideas en el Logo obtenemos no un lenguaje redondo, terminado, cerrado, sino abierto y ampliable dentro de la dirección que le interese al usuario, quien lo va expandiendo y especializando mediante la creación de nuevas órdenes-procedimiento específicas para cada tema de investigación, con lo que pueden desarrollarse distintos "logos" según la materia que interese explorar.

2.1.4.- Interactivo

Otra característica que ayuda a convertir al Logo en algo fácil de usar en múltiples tareas es su interactividad, que deriva directamente de ser un lenguaje interpretado. En realidad, el ser interactivo tiene que ver fundamentalmente con la facilidad de manejo del lenguaje, pero hemos preferido comentarla aparte por ser un elemento importante.

Para Papert es algo que contribuye a familiarizar rápidamente al usuario con su "ayuda a pensar" -los gráficos de la tortuga-, que la hace agradable e inmediata. Más importante por lo tanto desde el

punto de vista de los factores emotivos que influyen en el proceso de aprendizaje que cualquier otro de los ya citados. Evita al usuario los retrasos, permite la comprobación inmediata de cómo funcionan los procedimientos que ha estado creando. Será innecesario intentar explicarle razones tecnológicas que imponen retrasos en otros lenguajes y, al prescindir de esas explicaciones, hace más transparente el uso de la computadora, la cual se utiliza como una herramienta más, sin tener que preocuparse de esos aspectos del software. Al facilitar la tarea le permitirá concentrarse en lo que está haciendo, en el tema en el que está trabajando, sin tener que estar constantemente pendiente de las limitaciones y condicionantes que imponen otros lenguajes de la informática.

"La naturaleza interactiva del diálogo, el hecho de que ellos se encuentran al mando de la situación,... todo contribuye a dar un vigoroso sentido de realización a sus experiencias." (4)

La interpretación es, en la práctica, no tener que esperar a que la computadora compile los procedimientos después de su creación y antes de poderlos probar como sucede, por ejemplo, con el Pascal. Tanto las órdenes primitivas como el último procedimiento que se ha creado reciben el mismo tratamiento de manera instantánea.

La facilidad en la comprobación de los procedimientos se completa con el uso de mensajes de error mucho más explicativos que los usuales en informática; es fácil reconocer del texto el mensaje si el error procede de la ortografía, ejem.:

NO SE COMO HACER FIGARO

(4).- Seymour Papert. Desafío a la mente. B. Aires 1982 Pág. 138

cuando se ha escrito FIGARO en lugar de FIGURA o del olvido de espacios separadores, u otras causas.

Hasta ahora sólo hemos hecho una descripción de las características implementadas en el Logo, pero conviene apuntar que su uso en la clase suele dar un mayor rendimiento que el que permite suponer la mera descripción de las mismas.

La observación de la utilización del Logo con niños ha permitido reconocer que aporta otra serie de cualidades positivas que se exponen a continuación:

En el área de las matemáticas:

- fácil
- transparente
- sólo tiene un tipo de variables
- dispone del área de la geometría de la tortuga como objeto y ayuda a pensar
- analítico
- atractivo para el usuario, capaz de cambiar su actitud hacia las matemáticas que le parecían difíciles.
- capaz de acelerar la creación de estructuras mentales en el niño.

En la clase de uso general puede aportar el ser:

- gráfico
- en lengua materna
- puede procesar listas fácilmente
- combinatorio
- vocabulario no cerrado
- interpretado y por lo tanto interactivo
- mensajes de error explicativos

En la clase de introducción a la programación:

- orientado al uso de procedimientos
- recursivo
- potente.

2.2.- El valor pedagógico de LOGO

En el intento de desarrollar una utilización auténticamente renovadora de esta nueva técnica, se plantea la clásica necesidad de cambiar también la escuela y no solamente para acomodarla a la evolución de las necesidades sociales, sino para adaptarla a las metodologías de trabajo que se irán a utilizar. De esta manera la técnica será otro elemento más que podrá contribuir a la renovación pedagógica.

Lo interesante de la utilización de la computadora con el Logo es que ayuda a aprender a pensar, que su mismo uso, con cualquier estrategia, adquiere un nivel superior de significación que no sustituye ni es un mero complemento de la clase normal, si se utiliza como herramienta de trabajo. Como instrumento, de la misma manera que el profesor no se limita a proporcionar respuestas, sirve para que el estudiante vaya desarrollando por sí mismo la capacidad de descubrirlas autónomamente en los problemas que se planteen.

Aprender a pensar:

El Logo no suministra respuestas, sino que sirve para ayudar a que los estudiantes investiguen personalmente las posibles soluciones a los problemas, rectifiquen las ideas que resulten equivocadas, exploren nuevos enfoques con facilidad, creen hipótesis, teorías y soluciones, que las comprueben inmediatamente y que, si no funcionan

o si no concuerdan con lo que se espera de ellas, sean fáciles de desechar, perfeccionar o sustituir.

Esto es realizable en la práctica, de una manera sencilla y en niveles educativos bajos, si se puede empezar a trabajar con algo muy próximo al alumno, con algo que sea tan accesible como su, hasta cierto punto, limitado conjunto de experiencias y conocimientos; si puede utilizar, por ejemplo, su propio cuerpo para probar una hipótesis que se le haya ocurrido e intentar la búsqueda de una solución; si puede hacer personalmente una representación in vivo de lo que supone trazar un círculo intentando andar una trayectoria circular y analizar introspectivamente cómo mueve su cuerpo para hacerlo caminar.

"La geometría cobró vida cuando la conectamos con sus precursores en la experiencia humana más fundamental: la experiencia del propio cuerpo en el espacio." (5)

La programación de las computadoras con los actuales lenguajes de alto nivel supone un considerable esfuerzo de reformalización de la manera de reflexionar, para eliminar las ambigüedades e imprecisiones de nuestras ideas.

Así, al enseñarle a "pensar" a la computadora, los alumnos se involucran en una exploración del modo en que ellos mismos piensan.

El hecho de que la computadora sea incapaz de aceptar ningún error e incansable no los permita, obliga a replantear constantemente los problemas a dos niveles: primero, tal como lo permiten nuestras estructuras mentales y segundo, reformalizado, clarificado y concretando según las exigencias de la computadora.

(5).- Seymour Papert. "Desafío a la mente" B. Aires 1982 Pág. 201

Probablemente, el Logo contribuya al desarrollo intelectual del niño, pero quizá no en todas sus direcciones por igual. Hasta cierto punto resulta que favorece más el razonamiento inductivo que el deductivo, el método ascendente de lo concreto a lo general, pero creemos que esto ya es una aportación que merece la pena aprovechar en el ámbito de la escuela.

Es la traducción del lenguaje natural al informático, en el que las explicaciones deben ser perfectas, lo que lo convierte en un buen instrumento para hacer reflexionar de nuevo sobre el propio pensamiento. Y en esta actividad el procedimiento tiene asignado un papel relevante. Su importancia radica en que al desglosar, ordenar y precisar las acciones necesarias para realizar algo, éstas se objetivan y convencionan de una forma explícita, fija y observable, por ejemplo, en la pantalla de la computadora.

Modificar el procedimiento se convierte en una actividad mucho más controlada y objetiva que la remodelación de las ideas en el pensamiento. Al convertirlas en procedimiento, son algo sobre lo que se puede comentar, discutir, hablar con facilidad y concreción. El interés de hacer procedimientos y trabajar con ellos es que ayuda a remodelar el propio pensamiento, del cual constituye un modelo claro, fácil de manipular y evidente.

El esfuerzo de reformatización de estructuras, de analizar y concretar los propios pensamientos, que provoca el uso de la computadora, facilita la reflexión sobre cómo mejorar la propia manera de pensar y acaba por plantear reflexiones sobre el mismo proceso de aprendizaje.

De pensar en cómo se piensa hay un paso hacia aprender cómo se aprende.

CAPITULO III

CAPITULO III: APRENDIZAJE CON EL LENGUAJE LOGO

3.1.- APRENDIENDO CON EL LENGUAJE DE LOGO

Cargamos el programa de LOGO, como se describe en el capítulo VII y una vez que aparezca el letrero de bienvenida tecleamos CS.

Observamos a la tortuga apuntando hacia arriba en el centro de la pantalla. Es cierto, no parece tortuga, pero en sus orígenes LOGO no era un lenguaje para controlar a un objeto de la pantalla sino a un robot que caminaba por el piso. El robot tampoco parecía una tortuga pero se movía tan lento que parecía como uno de esos graciosos reptiles y así se le quedó el nombre.

Ordenamos a la tortuga que camine hacia adelante. Tecleamos la siguiente línea:

```
? FORWARD 50
```

y presionamos la tecla (enter). La palabra FORWARD indica a la tortuga que debe moverse hacia adelante. La abreviatura de FORWARD, FD, produce el mismo efecto. El número 50 indica la distancia que se moverá la tortuga.

La sintaxis de LOGO obliga a que haya un espacio entre la palabra FORWARD (o FD) y el número 50. Si olvidamos dejar el espacio, el sistema responderá con el mensaje:

```
I DON'T KNOW HOW TO FD50
```

```
(NO SE COMO HACER CON FD50)
```

Cualquier otro error de sintaxis es señalado con un mensaje similar. Si detectamos el error antes de presionar ENTER, podemos usar la tecla de Retroceso para borrar los caracteres erróneos.

La tecla ENTER indica a la computadora el momento en que debe ejecutar una instrucción. Al final de cada línea se debe oprimir ENTER, aunque ya no esté marcado en forma tácita; así pues, cuando hagamos una línea o escribamos un comando siempre deberemos presionar ENTER.

Intentemos que la tortuga recorra una distancia más grande. Tecleamos la siguiente línea:

? FD 100

Observamos que con la primera instrucción la tortuga dejó una marca de su trayectoria. Ahora la tortuga dejó una huella a su paso pero sobrepasó la pantalla y entró por la parte inferior de la misma. A este proceso se le llama "ENVOLVIMIENTO". La tortuga siempre hará este fenómeno cuando salga de la pantalla.

Siempre que queramos borrar todos los trazos, teclearemos CS. Un efecto adicional de esta instrucción es que la tortuga regresa a su posición inicial en el centro de la pantalla.

Borramos la pantalla (CS y ENTER) e introducimos la siguiente secuencia:

? FD 30

? RIGHT 90

? FD 30

La instrucción RIGHT 90 indica a la tortuga que gire 90 grados a la derecha. (RT es la abreviatura de la palabra RIGHT). La tortuga "entiende" el concepto grado como medida angular.

Podemos dibujar un cuadrado con las instrucciones FD y RT.

Limpiamos la pantalla y tecleamos las siguientes líneas:

? RT 45

? FD 50

Miremos la pantalla; ahora teclearemos:

? RT 10

Aparentemente no hubo ningún efecto. Ahora intentamos:

? FD 30

Notamos que hay un recodo en el trazo a pesar de que la tortuga no volteó su cabeza cuando le indicamos RT 10. La tortuga sabe exactamente cuál es su dirección pero su imagen en la pantalla sólo puede girar en pasos mayores de 15 grados. Entonces, la imagen de la tortuga apunta con la mayor aproximación que puede, pero no con precisión en la dirección de la tortuga real.

La instrucción RIGHT (derecha) produce cambio relativos de dirección. Cuando se le dice a la tortuga que haga un giro, voltea su cabeza, a partir de su posición actual, al número de grados indicado en la instrucción. De la misma manera cuando le indicamos a la tortuga que vaya hacia adelante, le estamos diciendo cómo de su posición, no le estamos indicando que vaya a algún lugar de la pantalla. En resumen, la posición y la dirección de la tortuga, después de ejecutar estas instrucciones, dependerá de las condiciones de inicio del movimiento.

Estamos en condiciones de dibujar un triángulo equilátero y un pentágono regular utilizando las instrucciones FD y RT.

LOGO incluye las dos instrucciones contrarias a FORWARD y a RIGHT, que son BACK (o su abreviatura BK) y que mueve a la tortuga hacia atrás y LEFT (su abreviatura LT) que gira la tortuga hacia la izquierda.

Borrarnos la pantalla y experimentamos con las instrucciones BK y LT.

Intento, por ejemplo:

? LT 90

? BK 30

? LT 90

? BK 30

Ahora tecleamos: HIDETURTLE ó (HT)

Antes de presionar (ENTER) observemos la pantalla.

HIDETURTLE ó HT significa esconder la tortuga (sólo la imagen).

Ahora tecleamos: ? LT 90

? BK 30

La tortuga es invisible pero su rastro es visible. Para hacer visible la tortuga tecleamos: SHOWTURTLE (ST).

De esta forma trataremos de encontrar cuántos pasos se puede desplazar la tortuga, a partir de su posición inicial, sin que se produzca envolvimiento, en las cuatro direcciones: arriba, abajo, izquierda y derecha.

3.1.1.- MODO EDIT Y COMANDOS DE EDICION

El objetivo de este apartado es que aprendamos los comandos del modo EDIT del sistema LOGO. Posteriormente veremos cómo se utilizará esta sección del sistema para crear y modificar procedimientos.

El modo **EDIT** es en sí un programa editor; un editor simple, orientado a líneas, que sólo maneja letras mayúsculas. Además de servir para trabajar con procedimientos, el modo **EDIT** es útil para crear documentos y luego imprimirlos.

En términos llanos, Editar significaría "enseñarle a la computadora". Una orden que se enseña a la computadora se llama **PROCEDIMIENTO**. Esto es lo que en el lenguaje informático se llamaría **PROGRAMACION**.

Elaboremos primero una figura, la que más nos guste, y le daremos un nombre por ejemplo:

? FD 30
 ? RT 90
 ? FD 60
 ? RT 90
 ? FD 30
 ? RT 90
 ? FD 60

Esta figura es un rectángulo, pero le podemos llamar como queramos (CAJA, BOTE, CXI, ESCALON). Ahora tecleamos el nombre que le pusimos y presionamos enter: Por ejemplo CAJA

El sistema envía un mensaje de error I DON'T KNOW HOW TO CAJA

(No sé cómo hacer CAJA)

Esto ocurre porque aún no le hemos enseñado a la computadora a hacer CAJA. Para ello podemos enseñarle de dos maneras:

1.- El primer método es con la orden **TO**.

Ejem: **TO CAJA**

> FD 30
 > RT 90

```
> FD 60
> RT 90
> FD 30
> RT 50
> FD 60
> END
```

La primera línea TO CAJA, comunica a la computadora que le está enseñando una nueva orden llamada CAJA. Las ocho líneas siguientes son las instrucciones para hacer CAJA. El símbolo > es un indicador ("prompt") o petición de orden que la computadora imprime en el lugar del usual ?. Este recuerda que está enseñando a la computadora una nueva orden. La última línea, END, indica a la computadora que ha terminado de enseñarle cómo hacer CAJA.

Después que escribamos la orden END, la computadora conoce que ha terminado e imprime el mensaje, CAJA DEFINED.

CAJA es ahora una orden Logo, tal como las ya conocidas. Para utilizar la nueva orden sólo teclearemos CAJA y enter.

2.- El segundo método para enseñar a la computadora nuevas órdenes es utilizando el editor de pantalla. Los errores de escritura son fáciles de corregir con el editor. Haremos el mismo ejemplo pero ahora utilizando el modo de edición:

En primer lugar, borraremos el procedimiento CAJA tecleando: ERASE "CAJA ò ER "CAJA

Ahora para utilizar el editor, tecleamos:

EDIT "CAJA, hay que asegurarse de que antes de la palabra a editar debemos escribir " (comillas), de lo contrario el sistema enviará mensaje de error.

Cuando escribimos EDIT "CAJA la computadora borrará la pantalla e introducirá el modo de edición; aparece en la parte inferior de la pantalla LOGO EDITOR, y se puede escribir todo lo que se desee. La computadora almacena información y no hace nada hasta que deje el modo de edición.

TO CAJA aparece en la parte superior de la pantalla. Ahora escribimos el resto de las órdenes, una cada vez. Podemos escribir las órdenes en línea vertical o en línea horizontal; en caso de error utilizamos las flechas de dirección y de retroceso para corregir cualquier error. Ejem:

TO CAJA

FD 30 RT 90 FD 60 RT 90 FD 30 RT 30 FD 60

END

Para dejar el modo de edición pulsamos la tecla ESC. = (ESCAPE) Ahora vemos la pantalla limpia y el letrero CAJA DEFINED en la parte superior. Para correr el procedimiento sólo tecleamos CAJA.

Veremos algo así:



Nota:

El sumario de LOGO contiene un resumen de las órdenes del modo EDIT.

3.2.- PROCEDIMIENTOS

Definición de procedimientos:

Se ha visto que la tortuga de LOGO ejecuta acciones simples en respuesta a las palabras FORWARD, LEFT, etc,. Estas palabras se llaman PRIMITIVAS. Una de las características más importantes de LOGO es que permite enseñar a la computadora nuevas palabras, las cuales se convierten en parte del vocabulario del lenguaje.

Entonces las nuevas palabras pueden usarse como si fueran primitivas. Una nueva palabra se define como un procedimiento. La definición de un procedimiento consta de tres partes:

- La línea del título, formada por la palabra TO seguida del nombre elegido para el procedimiento.

- El cuerpo, que es la secuencia de órdenes que conforma la definición del procedimiento,

- El fin de la definición del procedimiento, marcada por la palabra END.

Hay que tomar en cuenta que los procedimientos deben definirse en el modo de edición y ejecutarse en el modo de trabajo.

En la realización de los procedimientos existe un comando de mucha utilidad y que nos puede ayudar a abreviar órdenes es REPEAT; que permite hacer cosas interesantes, por ejem:

Para hacer cuadrado teníamos que poner lo siguiente:

TO CUADRADO

FD 50 RT 90 FD 50 RT 90 FD 50 RT 90 FD 50 RT 90

END

Lo que se traduce en repetir 4 veces FD 50 y 4 veces RT 90; con la orden REPEAT quedaria así:

```
TO CUADRADO
REPEAT 4[FD 50 RT 90]
END
```



Esta nueva fórmula hace exactamente lo mismo que la anterior.

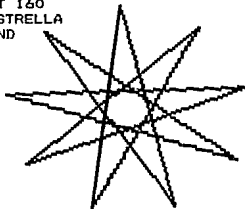
REPEAT necesita dos entradas. La primera es el número de repeticiones. La segunda es una lista de órdenes a repetir. Esta lista siempre debe de ir dentro de corchetes [].

Probaremos a elaborar las figuras geométricas que más nos agraden utilizando el comando REPEAT.

3.2.1.- RECURRENCIA:

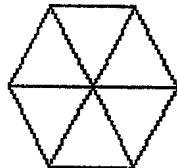
Existe un elemento poderoso que permite hacer que la computadora repita algo una y otra vez y es hacer un procedimiento que llame a una copia de sí mismo como un subprocedimiento. Esta forma de repetir algo se llama recurrencia. Probemos estos ejemplos:

```
TO ESTRELLA
FD 100
RT 160
ESTRELLA
END
```



```
TO TRIANGULO
REPEAT 3[FD 60 RT 120]
END
```

```
TO FLOR
TRIANGULO
RT 60
FLOR
END
```



Si se observa bien, la tortuga está dando vueltas y vueltas y no se parará hasta que nosotros la detengamos con las teclas CTRL-BREAK o bien si en la fórmula del procedimiento introducimos [STOP] (del que se hablará más adelante).

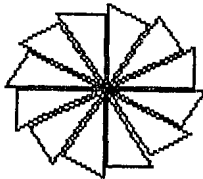
Podemos poner otro ejemplo más para acabar de comprender esta forma de procedimiento, con el que podemos obtener una serie de figuras que tal vez nunca las hubiéramos imaginado.

```

TD ASPA
FD 50 LT 90 FD 20 LT 120 FD 30
END
    
```

```

TD RUEDA
ASPA
RUEDA
END
    
```



3.2.2.- LOS COLORES DE LOGO

Una de las cosas más bonitas del Lenguaje Logo es que permite hacer procedimientos a colores y eso se debe que el sistema está diseñado para hacer más atractivo el aprendizaje. Para ello tenemos que tomar en cuenta lo siguiente:

(Debemos asegurarnos de tener un monitor a color por supuesto)

La tortuga puede dibujar en seis colores diferentes usando la orden SETPC (activar el color de la pluma) para establecer el color de la huella de la tortuga y la orden SETPAL (activa el grupo de colores de la pluma) para seleccionar un grupo de colores. Los colores reales

dependerán del tipo de monitor con que se cuente y de la resolución que tenga el mismo.

La orden SETPAL necesita como entrada un 0 o un 1.

La clasificación sería como sigue:

SETPAL 1

SETPC 1 dibuja en Cyan (azul claro)

SETPC 2 dibuja en Magenta (púrpura)

SETPC 3 dibuja en Blanco

SETPAL 0

SETPC 1 dibuja en Verde

SETPC 2 dibuja en Rojo

SETPC 3 dibuja en Marrón

SETPC 4 dibuja en el color del fondo, borrando por

donde cruce.

Con la orden SETBG (set Background = pon color de pantalla), podemos hacer que cambie el color del fondo de la pantalla; y podemos localizar dieciséis colores en la misma, que van del número 0 hasta el 15. Ejem:

? SETBG 0

? SETPAL 1

? SETPC 2

? FD 60 LT 45 FD 80

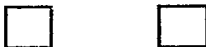
? SETBG 12

? BK 30 RT 90

3.2.3.- ORDENES CON PU Y PD

Hay dos comandos útiles para que la huella de la tortuga en el momento que se desea desaparezca y vuelva a aparecer más tarde en la pantalla; se trata de la orden PU (pen up = levanta pluma y PD (pen down = baja pluma). Esto permite que cuando intentemos hacer figuras separadas lo podamos hacer utilizando estos comandos. Ejem:

```
TO DOBLE
CUADRADO LT 90 PU FD 30 RT 90 PD CUADRADO
END
```



3.3.- VARIABLES

Se llama variable a una localidad donde pueden almacenarse números (uno a la vez) bajo un nombre. El nombre de una variable consiste en el símbolo "dos puntos" (:) seguido de cualquier número de letras y/o dígitos numéricos.

El siguiente procedimiento sirve para dibujar cuadros de diferentes tamaños:

```
TO CUADRO :LADO
REPEAT 4[FD :LADO RT 90]
END
```

Corremos el procedimiento con estas medidas:

```
CUADRO 20
CUADRO 50
```

Los números 20 y 50 son parámetros de entrada o simplemente "entradas" del procedimiento CUADRO.

En general, un procedimiento con una entrada se define poniendo en la línea de título el nombre de la entrada (o nombre de variable). Luego

se usa el nombre de la entrada en el lugar donde debe ir el valor de la entrada dentro del cuerpo del procedimiento.

También se pueden definir procedimientos con más de una entrada. Este es un ejemplo:

```
TO RECTANGULO :BASE :ALTURA
REPEAT 2[FD :BASE RT 90 FD :ALTURA RT 90]
END
```

Probemos los siguientes comandos:

```
RECTANGULO 50 20
RECTANGULO 100 10
```

El máximo número de entradas que puede tener un procedimiento de Logo es cinco.

Ciertamente que cuando hagamos procedimientos con variables, las deberemos editar primero y luego correrlas para comprobar si están bien elaboradas; sin embargo, algunas veces los procedimientos no resultan correctos y se hace necesario corregirlos, para ello podemos volver al editor y corregirlas. Para volver al editor tecleamos Edit "... y el nombre de nuestro procedimiento y procedemos a efectuar las correcciones necesarias utilizando las instrucciones dadas en el capítulo VII para este efecto.

Existen otros comandos que nos ayudan a recordar qué procedimientos hemos efectuado, estos son:

3.3.1.- POTS (Printout titles) = (salida impresa de títulos) visualiza en la pantalla todos los títulos (nombres) de procedimientos.

3.3.2.- PD (printout) = (salida impresa) Mostrará el desarrollo completo en la pantalla de un procedimiento. Si no pudieramos ver el procedimiento completo impreso en la pantalla, teclearemos F1, con lo que se mostrará la pantalla de texto en su integridad. Más adelante teclearemos F2 para volver a la pantalla dividida normal.

Ejem: PD "CAJA

3.3.3.- POALL (printout all) = (salida impresa de todo) Imprimirá todos los títulos y desarrollo de los procedimientos que estén en la memoria de la computadora en ese momento. En caso de que la impresora esté activada, todos los procedimientos que existan quedarán impresos en papel, además de la pantalla del monitor.

3.3.4.- LOS COMANDOS SAVE = (SALVAR) Y LOAD = (CARGAR)

Al apagar la computadora, todo lo que exista en la memoria de ella, se borrará automáticamente. Esto incluirá todos los procedimientos que se hicieron durante una jornada de trabajo. La manera de prevenir esto, es "salvando" el trabajo hecho con el comando SAVE. Esto es; antes de apagar el equipo podemos guardar los procedimientos editados con la siguiente orden: SAVE "NOMBRE <enter>", donde NOMBRE, corresponderá a la identificación que el usuario quiera darle a sus procedimientos. Ejem: SAVE "FIGURAS.

El trabajo que se hizo ahora queda guardado en el disco de trabajo que previamente se ha preparado para tal efecto. El nombre del procedimiento queda guardado en un archivo del disco con la terminación LF. (Logo File)

Conviene saber que el disco deberá estar previamente formateado, es decir un disco inicializado con un procedimiento que puede consultarse en los manuales del sistema operativo del equipo. (Ver Comando Format del Sistema Operativo del Sistema)

Para volver a utilizar los procedimientos que se guardaron en el disco, una vez que retornamos al trabajo de la computadora, podemos llamar esos procedimientos, cargando el archivo del disco a la memoria de la computadora por medio del comando LOAD. Bastará con introducir a la máquina el disco con el archivo salvado y cargarlo a la memoria. Ejem: LOAD "FIGURAS <enter>.

Ahora estamos listos para ver nuevamente nuestro trabajo anterior y poder corregirlo, aumentarlo o borrarlo. En este último caso usamos el comando ERASEFILE "NOMBRE <enter> y quedará borrado del disco y de la memoria el procedimiento que no deseamos.

3.4.- ELABORANDO FIGURAS

Con los comandos vistos hasta este apartado, podemos empezar a elaborar las figuras más sencillas que nos darán la pauta para de aquí en adelante ir a diseños más elaborados. Resultaría importante que el alumno practicara la forma de salvar procedimientos en su disco de trabajo al final de una sesión de trabajo y los cargara al inicio de la siguiente.

Iniciaremos con el procedimiento para hacer CUADRADO:

```
TO CUADRADO
REPEAT 4[ FD 80 LT 90]
END
```


Si observamos bien, la tortuga hace 4 giros de 90 grados cada uno, con lo que obtenemos una suma total de 360 grados, lo que representado en forma numérica sería $90 = 360/4$.

Entonces el procedimiento CUADRADO también lo podríamos escribir así:

```
TO CUADRADO
REPEAT 4[FD 80 LT 360/4]
END
```

Con este tipo de procedimiento nos damos cuenta que para hacer un cuadrado tenemos que dar una vuelta completa dividiendo los 360 grados entre el número de lados del cuadrado, o sea 4.

Si la figura a editar fuera un triángulo el giro de la tortuga sería de $360/3$; lo cual se traduce en que para elaborar cualquier figura, bastará dividir 360 entre el número de lados de la figura a realizar.

Ejemplo: Para hacer un hexágono, el procedimiento sería:

```
TO HEXAGONO
REPEAT 6[FD 70 RT 360/6]
END
```

Con el conocimiento que tenemos de las variables, podemos facilitar el confeccionamiento de los procedimientos, si utilizamos un elemento común en nuestra fórmula. Esto es:

```
TO POLIGONO :LADOS
REPEAT :LADOS [FD 70 RT 360/:LADOS]
END
```

Esta nueva forma permite que al correr este procedimiento tengamos que hacer solamente una entrada de lados para obtener la figura deseada, por ejemplo:

```
POLIGONO 4
POLIGONO 12
POLIGONO 20
```

Donde 4, 12 y 20 es el valor que se le asigna a la variable !LADOS y que se traducirá en un cuadrado (POLIGONO 4), un dodecágono (POLIGONO

12) y un icosaágono (POLIGONO 20), pero todos con una misma medida (FD 70).

Sin embargo, también podríamos modificar el tamaño de esas figuras si introducimos una segunda variable que nos permita agrandar o reducir el tamaño de nuestros diseños; en este caso la nueva fórmula quedaría replanteada en estos términos:

```
TO POLIGONO :LADOS :TAMANO
REPEAT :LADOS [FD :TAMANO RT 360/:LADOS]
END
```

Algunas veces este tipo de procedimiento clarifica al alumno la tarea que va a realizar la computadora en el momento de correr el trabajo; no obstante una vez comprendida la fórmula, podemos abreviar el procedimiento en cuanto a los nombres de las variables, ya que Logo permite que las palabras de éstas puedan estar representadas por una letra. Ejem:

```
TO POLIGONO :N :L
REPEAT :N [FD :L RT 360/:N]
END
```

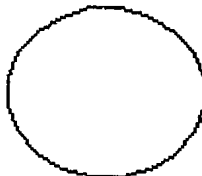
Donde (N) será el número de lados, y (L) corresponderá al largo de la figura. Estas dos formas de procedimiento ejecutarán el mismo tipo de trabajo, sólo que habrá que dar dos entradas para que se cumpla la fórmula. Ejem:



POLIGONO 3 90



POLIGONO 12 50



POLIGONO 20 30

Induiremos al alumno que observe que con esta última entrada el POLIGONO tiende a parecer una circunferencia, razón por la cual, estaremos en condiciones que él mismo busque la forma de elaborar un círculo, modificando las entradas hasta que logre descubrir, cuáles serían los números que le permitan hacer un círculo perfecto. Una de tantas respuestas podría ser esta:

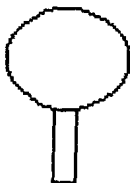
POLIGONO 300 1; POLIGONO 100 2; POLIGONO 75 5.

El alumno habrá descubierto cosas interesantes que puede hacer con una sencilla fórmula, y habremos despertado su curiosidad por elaborar otras muchas figuras. Sin embargo lo importante es que el alumno descubrió el principio de la confección de figuras y la manera de establecer principios geométricos.

Estamos en condiciones que el alumno elabore figuras conocidas por él, donde combine figuras rectas con curvas. Por ejemplo:



UNA CASA



UN ARBOL



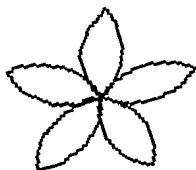
UNAS PEGAS

Ayudándonos con este principio, podemos hacer que el alumno descubra ahora la manera de hacer ARCOS o SEMICIRCULOS; le daremos la oportunidad que utilice su creatividad y con un poco de ayuda tendremos que por simple deducción de su parte encuentra la manera de hacerlos.

Una forma pudiera ser la siguiente:

```
TO POLIGONO :N :L :G
REPEAT :N [FD :L RT 360/:G]
END
```

Con este procedimiento podríamos elaborar lo siguiente:



UNA FLOR



UNAS NUBES

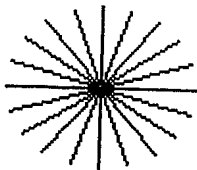
Una vez que los alumnos logran la adquisición de estos conocimientos, conviene enseñarles que también pueden obtener otro tipo de diseños que sirven como base para hacer figuras diferentes, como sería lo siguiente:

```
FD 60
BK 60
RT 10
```

Estas sencillas órdenes al verse repetidas obtendrán una serie de figuras que seguramente agradarán al alumno. Por ejemplo, si elaboramos el siguiente procedimiento:

```
TO SOL
REPEAT 36[FD 60 BK 60 RT 10]
END
```

Obtendríamos algo así:



SOL

Utilizando variables, podemos obtener los siguientes:

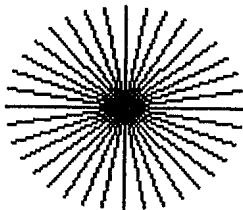
```
TO SOL :RAYOS :LARGO
```

```
  REPEAT :RAYOS [FD :LARGO BACK :LARGO RT 360/:RAYOS]
```

```
END
```



SOL 20 40



SOL 36 50

CAPITULO IV

CAPITULO IV: ALGUNAS APLICACIONES DEL LENGUAJE LOGO

4.1.- APLICACIONES DE LOGO EN LA ESCUELA

CALCULOS ARITMETICOS:

Con la computadora podemos efectuar operaciones aritméticas:

SUMA:	PRINT 23 + 46
RESTA:	PRINT 1045 - 749
MULTIPLICACION:	PRINT 67 * 18
DIVISION:	PRINT 1200 / 35
POTENCIA:	PRINT 96

La orden PRINT significa imprime y aunque no es indispensable, conviene utilizarla para ver el resultado.

Se pueden efectuar varias operaciones dentro de una misma instrucción, por ejemplo:

456 / 34 * 67 + 197 - 19

La computadora hará la operación rápidamente, pero sin embargo al efectuarla sigue los siguientes pasos:

1.- Primeramente efectúa las potencias sin hacer caso de las otras operaciones, haciendo una recorrido de lectura de izquierda a derecha.

2.- Recorriendo en el mismo sentido la operación, ejecuta las multiplicaciones y divisiones que encuentre.

3.- Finalmente efectúa las sumas y restas que se encuentren.

Veamos un ejemplo:

430 / 5 + 60 * 3 - 70

PASO 1.- POTENCIAS 430 / 5 + 60 * 3 - 70 (NO HAY)

PASO 2.- MULTIPLICACIONES Y DIVISIONES:

$$(430 / 5) + (60 * 3) = 70$$

$$86 + 180 = 70$$

PASO 3.- SUMAS Y RESTAS:

$$266 - 70 = 196$$

Observe las siguientes operaciones aritméticas, sus equivalencias en Logo y la forma en que éste las resuelve:

OPERACION ARITMETICA

EXPRESION EN LOGO

$$40 + \frac{60}{20} - 10 =$$

$$40 + 60 / 20 - 10 =$$

$$40 + 30 - 10 =$$

$$= 60$$

$$2 * 5 - 3 + \frac{7}{2} =$$

$$2 * 5 - 3 + 7 / 2 =$$

$$10 - 3 + 3.5 =$$

$$= 10.5$$

Si en una instrucción aritmética en Logo encierra varias operaciones dentro de un paréntesis, éstas se resolverán primero, siguiendo la lectura normal del proceso. Ejem:

$$9 + \frac{12}{5 + 1} - 2 * 3 =$$

$$9 + 12 / (5 + 1) - 2 * 3 =$$

$$9 + 12 / 6 - 2 * 3 =$$

$$9 + 2 - 6 =$$

$$= 5$$

Cuando el alumno haya comprendido el proceso de solución de operaciones aritméticas, lo introduciremos en el proceso de obtener algunas fórmulas aritméticas en Logo.

En esta ocasión trataremos con un trapecio:

FIGURA

FORMULA

TRAPECIO

$$\frac{B + b \times h}{2}$$

Le damos los datos:

Base mayor = 40

Base menor = 25

Altura = 12

Representación aritmética

Representación en Logo

$$\frac{40 + 25 \times 12}{2} =$$

$$(40 + 25) \uparrow 12 / 2 =$$

$$65 \uparrow 12 / 2 =$$

SOLUCION:

SUPERFICIE = 390

Con representaciones como éstas lograríamos que el alumno encontrara el perímetro de las figuras, su superficie e incluso el volumen de ellas , como por ejemplo:

CUERPO

FORMULA PARA VOLUMEN

PRISMA TRIANGULAR

BASE X ALTURA

Nuevamente damos los datos:

Base del triángulo = 71

Altura del triángulo = 34

Altura del Prisma = 68

Representación aritmética

Representación en Logo

$$\frac{71 \times 34}{2} \times 68 =$$

$$(71 \uparrow 34) / 2 \uparrow 68 =$$

$$2424 / 2 \uparrow 68 =$$

$$1212 \uparrow 68 =$$

$$= 82,416$$

Hasta este momento las operaciones que se tratan de enseñar al niño siguen un proceso lógico y de fácil comprensión, especialmente porque la computadora mediante el programa de Logo nos ayudan a la satisfacción de los problemas, y porque podemos cambiar las cantidades cuantas veces queremos y revisar los procedimientos una y otra vez hasta quedar satisfechos de los resultados del mismo.

Otra aplicación que podemos hacer con las operaciones matemáticas, es la utilización de la probabilidad con la computadora, al emplear el comando RANDOM = (Números al azar).

Este comando elige al azar la respuesta de las órdenes que le demos, como en este ejemplo:

```
RANDOM 9
RANDOM 9
RANDOM 9
.....
(15 VECES)
```

¿Qué sucede? Enlista número de manera aleatoria, o sea, si jugarámos a la suerte de ver cuántas veces salía el número 5 podríamos decir que N veces, según los datos que nos diera la computadora.

Podemos utilizar la siguiente orden:

```
RANDOM 20          Y genera números entre 0 y 19
```

En cambio:

```
1 + RANDOM 20      Genera números entre el 1 y el 20
```

Esta instrucción puede sernos útil cuando elaboramos figuras y queremos jugar a la probabilidad con ellas. Por ejemplo:

Ya sabemos cómo hacer un cuadrado y cómo hacerlo de diferentes tamaños con un solo procedimiento; a recordar:

```
TO CUADRADO :LADO
REPEAT 4[FD :LADO RT 360/4]
END
```

Ahora utilizaremos el comando RANDOM para ver que figuras nos hace:

```
CUADRADO RANDOM 20  
CUADRADO RANDOM 50  
CUADRADO RANDOM 80
```

Notaremos que los cuadrados que hace no son de la medida que le damos sino de los que aleatoriamente saca el comando RANDOM.

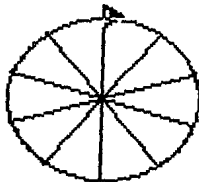
Podemos hacer que nuestros alumnos practiquen este comando e intercalen algunas otras órdenes de figuras para que entiendan mejor el funcionamiento de la probabilidad.

Otros ejemplos podrian ser:

```
POLIGONO 5 70  
POLIGONO 5 RANDOM 70  
POLIGONO 3 + RANDOM 8 20  
POLIGONO 3 + RANDOM 5 RANDOM 40
```

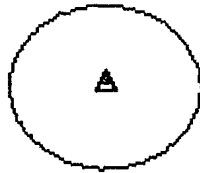
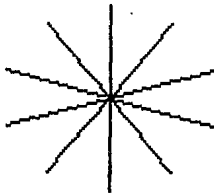
OTRAS APLICACIONES:

Vamos a hacer el procedimiento RUEDA . Tendrà un radio de 50.



RUEDA

RUEDA puede hacerse combinando SDL y POLIGONO, usándolas con variables, para poder hacerla de distintos tamaños.



SDL :RAYOS :LARGO
 :LARGO en este procedimiento
 es el radio de la rueda.
 :RAYOS es el número de
 rayos.

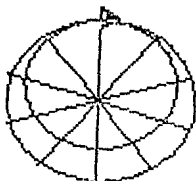
POLIGONO :LADOS :TAMANO
 :TAMANO en este procedimien-
 to es la longitud de cada lado.
 Si :TAMANO = 1, :LADOS = peri-
 metro.

Por ejemplo: POLIGONO 100 1
 dibuja una circunferencia
 con perímetro = $100 \times 1 = 100$.

De acuerdo con ésto, hagamos primero la rueda paso a paso y después
 construiremos un procedimiento.

Necesitaremos los procedimientos SDL y POLIGONO. De estar grabados
 cárgalos a la memoria de la computadora, en caso contrario hay que
 crearlos nuevamente. Ahora haremos primeramente:

```
SOL 8 50
FD 50
RT 90
```



Ahora, dibujaremos la circunferencia encima del SOL con nuestro procedimiento POLIGONO. Tal vez no quede bien en un principio, así que lo intentaremos varias veces hasta que salga perfecto.

Podemos hacer la observación que el diámetro cabe un poco más de 3 veces en el perímetro; de donde surge el valor de 3.1416. Con esto, podemos entonces cambiar el valor del primer número de la variable que pertenece a LADOS. Ejem:

```
SOL 8 50
FD 50 RT 90
POLIGONO 50*2*3.1416 1
```

Ahora tendríamos el funcionamiento correcto para hacer ruedas; de aquí entonces que podemos generalizar el procedimiento para hacer RUEDA:

```
TO RUEDA :RAYOS :RADIO
SOL :RAYOS :RADIO
FD :RADIO RT 90
POLIGONO :RADIO*2*3.1416 1
END
```

Con este procedimiento podemos hacer ruedas de cualquier tamaño.

EL COMANDO "MAKE" = (HAZ)

Probemos la orden: MAKE "A B

Escribimos: PRINT !A

Como se puede observar, MAKE creó la variable !A y le asignó el valor de B.

Tomaremos en cuenta que al crearla se le antepusieron comillas (")

```
Probemos ahora:      MAKE "D 4 * 5 + 2
Desplegamos :D      PRINT !D
Ahora:               MAKE "C (:A + !D) * 2
                   PRINT !C
```

Los resultados son evidentes por sí mismos.

Usando MAKE, el procedimiento RUEDA puede quedar de mejor forma:

```
TO RUEDA :RAYOS :RADIO
  SOL :RAYOS :RADIO
  FD :RADIO RT 90
  MAKE "DIAMETRO :RADIO * 2
  MAKE "PERIMETRO 3.1416 * :DIAMETRO
  POLIGONO :PERIMETRO 1
  LT 90 BK :RADIO
  END
```

Se usaron dos instrucciones MAKE para mayor claridad de los cálculos, pero daría el mismo resultado una sola:

```
MAKE "PERIMETRO :RADIO * 2 * 3.1416
```

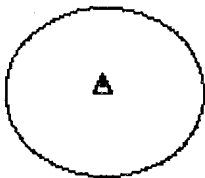
El procedimiento POLIGONO nos permite dibujar una circunferencia partiendo de un punto de ella, cuando conocemos el perímetro.

```
POLIGONO 300 1
```

RUEDA nos sugiere la necesidad de tener otro que dibuje la circunferencia a partir del centro cuando conocemos el radio.

```
TO CIRC :RADIO
  FU FD :RADIO RT 90 PD
  MAKE "PERIMETRO :RADIO*2*3.1416
  POLIGONO :PERIMETRO 1
  FU LT 90 BK :RADIO PD
  END
```

Probemos:



CIRC 100



CIRC 40

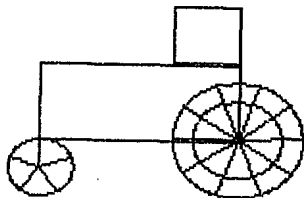


CIRC 10

Usando los procedimientos RUEDA, CIRC, y RECTANGULO podríamos hacer el procedimiento TRACTOR. EJEM:

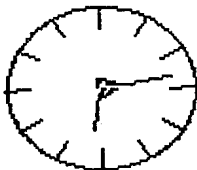
```

TO TRACTOR
REPEAT 2(FD 40 RT 90 FD 50 RT 90)
PU BK 15 PD
RUEDA 5 15 LT 90
PU RT 90 FD 90 PD
RUEDA 10 30 LT 90
BK 30 CIRC 20
SETH 0 FD 40
REPEAT 4(FD 30 LT 90)
END
    
```



TRACTOR

4.2.- JUGANDO AL RELOJ:



Usando los conocimientos adquiridos pretendemos ahora elaborar un proyecto cuyo fin serà hacer el diseño de un RELOJ y que nos marque la hora que nosotros indiquemos. Haremos este trabajo en dos partes:

RELOJ
 (Nos dibuja el reloj
 sin manecillas).

HORA
 Coloca las manecillas de
 acuerdo al tiempo :H (horas)
 y :M (minutos)

Vamos a ocupar el procedimiento CIRC, que ya conocemos.

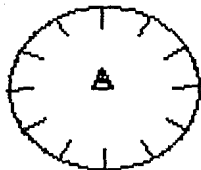
Ahora el procedimiento:

```
TO RELOJ
CIRC 50
REPEAT 12 [PU FD 40 PD FD 10 PU BK 50 PD RT 30]
END
```

Ahora, necesitamos elaborar la HORA, para esto:

```
TO HORA :H :M
MAKE "ANGHORA :H * 30 + :M * 0.5
MAKE "ANGMIN :M * 6
SETH :ANGHORA FD 25 BK 25
SETH :ANGMIN FD 35 BK 35
END
```


LISTO! Probemos resultados:



RELOJ



HORA

Las órdenes MAKE crean las variables :ANGHORA que guardan el ángulo que girará el horario a partir de las 12 hs.

:ANGMIN que guarda el ángulo que formará el minuterero con el 12.

El comando SETH, hace que la tortuga gire en el ángulo que nosotros deseemos, entendiendo que en su posición inicial estará en grado 0. Así, al escribir SETH 45, la tortuga girará hasta quedar en el grado 45 sin importar en que posición estaba anteriormente. Tampoco dejará huella de su giramiento.

Ejem:

```
SETH 90  FD 50 BK 50
SETH 45  FD 50 BK 50
SETH 180 FD 50 BK 50
SETH 340 FD 50 BK 50
```

Con esta ejemplificación, podemos ahora volver al procedimiento anterior y ver la función de SETH.

APRENDIZAJE POR COMPUTADORA

Podemos facilitar la operación si incluimos un procedimiento que borre la imagen de las manecillas sin perder la figura del reloj y estar listos para dar una nueva hora. Para esto, necesitamos hacerlo mediante dos pasos:

1.- Vamos a hacer el procedimiento BORRA:

```

TO BORRA.HORA
PE
SETH :ANGHORA FD 25 BK 25
SETH :ANGMIN FD 35 BK 35
PD
END
    
```

La orden PE nos ayuda a borrar, como si fuera una goma. Una vez este nuevo procedimiento podemos poner varias veces la hora, así:

```

CS
DIBUJA RELOJ
HORA 2.47
BORRA.HORA
HORA 16.53
BORRA.HORA
    
```

El procedimiento BORRA.HORA podemos incluirlo en el procedimiento principal, pero nos hace falta uno más para que la hora se borre a un determinado tiempo y podamos poner otra diferente. Elaboraremos el procedimiento ESPERA:

```

TO ESPERA :T
REPEAT :T []
END
    
```

Ahora creo que ya podemos escribir todo nuestro procedimiento para consultar la hora del reloj:

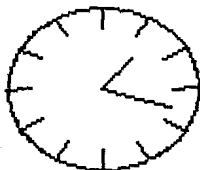
```

TO HORA :H :M
MAKE "ANGHORA :H * 30 + :M * 0.5
MAKE "ANGMIN :M * 6
SETH :ANGHORA FD 25 BK 25
SETH :ANGMIN FD 35 BK 35
ESPERA 5000
BORRA.HORA
END
    
```

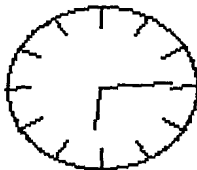
Con este nuevo procedimiento bastará lo siguiente:

CS
RELOJ
HORA 1 18
HORA 18 14
HORA 22 56

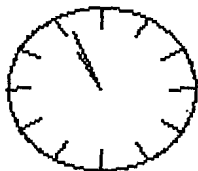
Una vez comprobado este mecanismo, guardaremos nuestro procedimiento en disco para ser utilizado en cualquier momento de la clase.



HORA 1 18



HORA 18 14



HORA 22 56

4.3.- DIBUJOS A ESCALA

Algo que sin duda será una delicia para los aficionados al dibujo, es poder hacer escala de sus diseños. Los procedimientos para poder efectuar este apartado es muy simple y nos ayudará más adelante cuando veamos el plano cartesiano.

Para poder efectuar la escala simplemente tenemos que editar nuestro procedimiento base y hacerle unas pequeñas modificaciones; así por ejemplo:

Vamos a hacer el dibujo de la fachada de una casa.

```
TO CASA
RECT 50 100
RT 90 FD 20 LT 90
RECT 40 20
RT 90 FD 30 LT 90
PU FD 20 PD
RECT 20 40
END
```



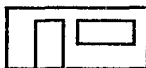
CASA

Nótese que empleamos un procedimiento ya conocido que es RECT =RECTANGULO y que tiene dos variables :L = :Lado y :B = :Base.

De manera que RECT 50 100 nos hará un rectángulo de 50 x 100 ; luego, más adelante hay otro de 40 x 20 y finalmente otro de 20 x 40, sólo que están acomodados en diferente posición, tal como lo vemos en el dibujo.

Ahora llegamos uno a escala. Usamos el mismo procedimiento, pero vamos a efectuarle unas variantes; ahora quedará así:

```
TO CASAESC :N
RECT 50% :N 100% :N
RT 90 FD 20% :N LT 90
RECT 40% :N 20% :N
RT 90 FD 30% :N LT 90
PU FD 20% :N PD
RECT 20% :N 40% :N
END
```

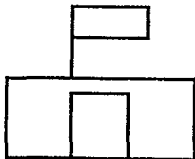


CASAESC 1/2

El resultado es asombroso, pues podemos hacer nuestra fachada de casa del tamaño que nosotros queramos hacerla.

Con esta forma de hacer los proyectos a escala, elaboraremos una escuela, unos portales, un árbol, una iglesia y los usaremos como gráficos para nuestro trabajo un poco más adelante.

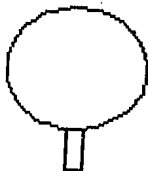
Así nos quedarían algunos de nuestros dibujos:



ESCUELA



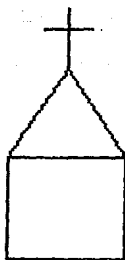
ESCUELAESC 1/4



ARBOL



ARBOLESC 0.5



IGLESIA



IGLESIAESC 0.4



PORTAL

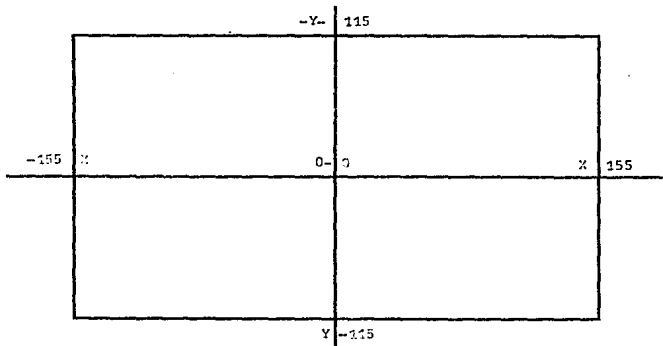


PORTALESC 1/3

4.4.- PLANO CARTESIANO

Uno de los trabajos que realizamos en la escuela primaria es que los alumnos aprendan a ubicar sus lugares donde viven en un plano. Para este efecto podemos utilizar el PLANO CARTESIANO y hacer simulaciones de las calles y los lugares más representativos de nuestros alumnos. Primeramente necesitamos elaborar el plano en nuestra computadora; tendremos que auxiliarnos de algunas instrucciones que aún no hemos revisado y que son el de las coordenadas.

La pantalla de Logo simula un plano con 4 cuadrantes. Cada uno está ubicado en coordenadas de X y de Y. Esto es la pantalla de la computadora podemos dividirla imaginariamente con dos líneas que la cruzan. Una de arriba hacia abajo por la parte media y otra de izquierda a derecha también por la parte media. Nos quedaría una pantalla dividida en 4 partes. Algo así:



Tenemos ahora que la tortuga está ubicada en el centro de la pantalla, o sea en el punto 0-0

Podemos hablar de que la línea horizontal es el eje de la "X" y la línea vertical el eje de la "Y". Las dimensiones que tiene la línea horizontal "X", son de 155 "pasos" positivos a partir del 0 hacia la derecha y -155 pasos negativos hacia la izquierda, también a partir del 0.

En el caso de la línea vertical "Y", los pasos hacia arriba son positivos y van del 0 al 115; y los pasos hacia abajo son negativos y van del 0 al -115.

Para ubicar la tortuga en algún punto de la pantalla debemos hacer lo siguiente:

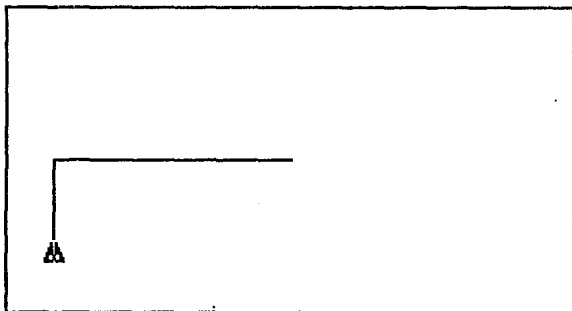
Cerciorarnos que la pluma de la tortuga este levantada antes de ir a la posición que deseamos, pues de lo contrario dejará marca en su trayectoria. Ejem:

Si deseamos ir al extremo izquierdo inferior, debemos hacer este mecanismo:

```

PU
SETX -140
SETY -110
PD
    
```


Si solamente diéramos las coordenadas SETX -140, SETY -110, tendríamos lo siguiente:



Obviamente este trazo ya está limitando nuestro diseño.

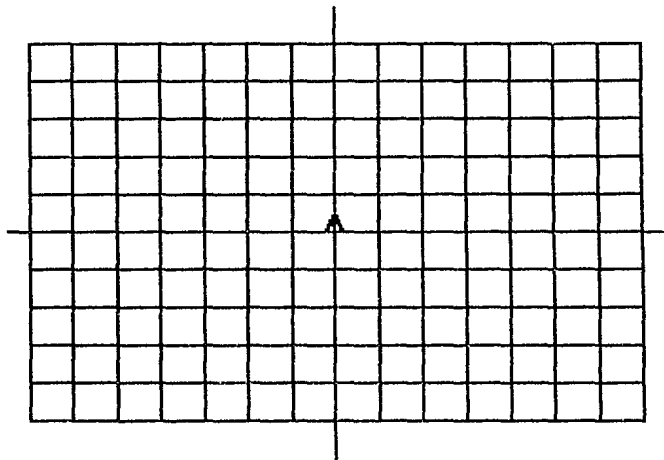
Con estos antecedentes nos ocuparemos en primer lugar de hacer un plano cartesiano y luego de tenerlo lo ocuparemos para ubicar nuestros diseños a escala. Para ello, diseñaremos el plano con el siguiente procedimiento:

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

APRENDIZAJE POR COMPUTADORA

```
TO CUADRICULA
SETBG 8 SETPC 1 PU
SETX -140 SETY -100 PD
REPEAT 14IFD 200 BK 200 SETX XCOR +201
FD 200 BK 200
PU SETX -140 SETY -100 SETH 90 PD
REPEAT 10IFD 280 BK 280 SETY XCOR + 201
FD 280 BK 280
PU SETX 0 SETY 0 PD SETH 0 SETPC 2
FD 120 BK 240 FD 120 RT 90 FD 150 BK 300
FD 150 LT 90 SETPC 3
END
```

Nuestro plano quedaria de esta manera:



APRENDIZAJE POR COMPUTADORA

Ahora los procedimientos que permitan ubicar nuestros diseños en el plano:

```
TO ORIENTE :P
SETX 20# :P
END
```

```
TO PONIENTE :P
SETX -20# :P
END
```

```
TO NORTE :P
SETY 20# :P
END
```

```
TO SUR :P
SETY -20# :P
END
```

Una vez que tenemos todos nuestros procedimientos listos, procedemos a ubicarlos dentro de nuestro plano con las figuras que hicimos a escala y que nos ayudarán a comprender mejor el uso del plano cartesiano.

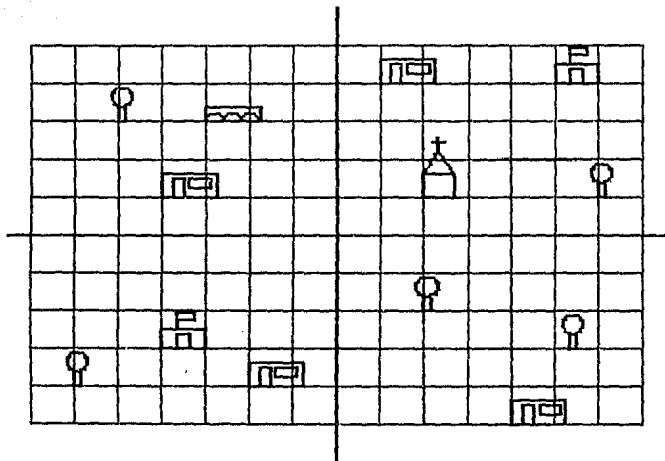
Con este tipo de actividad, estamos seguros que los alumnos encontrarán una forma divertida de aprender ayudándose con la computadora.

Para ubicar todos nuestros diseños dentro del plano cartesiano, elaboraremos un procedimiento que los contenga a todos.

```
TO CIUDAD
CS FS CUADRICULA HT
ORIENTE 2 NORTE 2 IGLESIAESC 0.4
ORIENTE 3 NORTE 4 ARBOLESC 0.5
ORIENTE 4 SUR 3 ESCUELAESC 1/4
ORIENTE 2 SUR 4 ARBOLESC 0.5
PONIENTE 3 NORTE 2 PORTALESC 1/3
PONIENTE 5 NORTE 5 CASAESC 1/2
PONIENTE 6 NORTE 1 ARBOLESC 0.5
PONIENTE 3 SUR 3 CASAESC 1/2
PONIENTE 4 SUR 5 ARBOLESC 0.5
FIN
```

Ahora el resultado que tendríamos sería como sigue:

OBJETOS UBICADOS EN EL PLANO CARTERIANO



Con este ejemplo, creo que podemos hacer más diseños a escala e insertarlos dentro de nuestro plano, o bien pedir a los alumnos que ellos mismos elaboren algunos otros y les den cabida dentro del plano.

4.5.- EJERCICIO DE ESTADISTICA

Para concluir este capítulo de aplicaciones en la escuela, elaboraremos una sencilla gráfica que podrá ser utilizada en muchas otras aplicaciones que se quieran hacer.

Se trata de elaborar un conjunto de figuras geométricas creadas al azar y después representadas mediante una gráfica.

Primeramente generaremos el conjunto de figuras que en este caso serán de pequeños polígonos. Para esto generaremos el siguiente procedimiento:

```

TO POLIGNITO
MAKE "N 3 + RANDOM 6
POLIGNO N: 50/:N
END

```

Este procedimiento genera un polígono aleatorio que va desde 3 hasta 8 lados. La orden RANDOM elegirá al azar el número de lados que va a tener cada polígono.



Ahora con otro procedimiento generaremos un grupo de 14 figuras con diferentes números de lados; para esto:

```

TO POLIGONITOS
PU SETX -120 SETY 85 PD
SETH 270
REPEAT 14 [POLIGONITO PU BK 18 PD]
SETH 0
END
    
```

Esta nueva orden nos dará como resultado lo siguiente:



Necesitamos ahora elaborar nuestros ejes donde estén contenidos los datos que nos arrojan el número de poligonos generados:

Primero el eje de Y:

```

TO EJEY
REPEAT B[FD 10 LT 90 FD 2 BK 2 RT 90]
BK 90
END
    
```

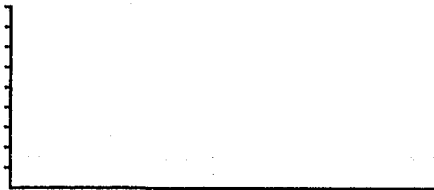
Esto nos dará el primer EJE VERTICAL:



Ahora necesitamos el eje de X, donde se colocarán las barras que representen el número de figuras. Utilizamos el siguiente procedimiento:

```
TO EJEX
PU SETX -90 SETY 0 PD
EJEX
RT 90 FD 180
END
```

Nos da como resultado el tener listos el EJE VERTICAL y el EJE HORIZONTAL:

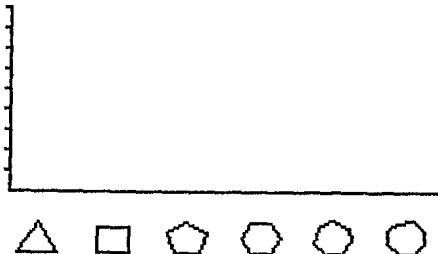


Luego necesitamos ubicar las figuras debajo de este plano para que podamos definir las veces que se repite cada una de ellas y conocer de que figura estamos hablando. Procedemos!

```

TO FIGURAS
EJEX EJEX
PU SETX -70 SETY -30 PD SETH 270
MAKE "N 3
REPEAT 6[ POLIGOND IN 50/IN PU BK 30 PD MAKE "N IN+1]
END
    
```

Ya tenemos listo nuestro plano de X y de Y, y además las figuras que se representarán en cada barra. Nuestro diseño se vería ahora de esta forma:



Finalmente, procedemos a elaborar el procedimiento que nos permita introducir las barras que representen el número de figuras del mismo número de lados en nuestro plano; para esto haremos como sigue!

```

TO BARRAS
PU SETX -90 SETY 0 PD
SETH 0
REPEAT 6[MAKE "FIRST RW RECT :A#10 30 SETX XCOR +30]
END
    
```

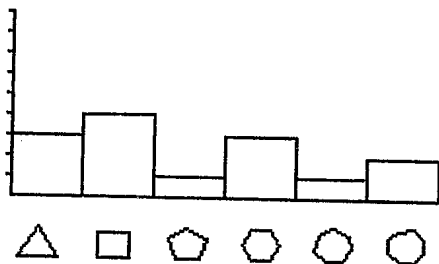

Y esto nos permite hacer todo el procedimiento maestro para que el alumno o nosotros mismos con una sola palabra tengamos listo nuestro trabajo:

```

TD GRAFICA
POLIGONITOS
EJEX EJEY
FIGURAS
BARRAS
END
    
```

Podemos ver nuestro trabajo completo y bastará que el alumno cuente las figuras y anote el número de ellas, para que nuestra gráfica se complete por sí sola.

GRAFICA



El alumno tecleó las veces que aparecieron las figuras generadas al azar y tendrá cada vez una gráfica distinta, con la que podrá entender para qué nos sirven y cuáles son sus aplicaciones.

Todo lo que presentamos en este apartado, puede servir para que aquellos usuarios que ocupen la computadora con el programa de Logo, hagan un sinnúmero de pruebas o ejercicios que lleve al alumno a concebir de una manera más racional sus inquietudes o logren resolver problemas con mejor entendimiento.

CAPITULO V

palabras, entonces debemos ponérselas, así vayan combinadas con números.

A esto se le podría llamar trabajar con PALABRAS.

Si nos ayudamos con la orden MAKE, tendremos una secuencia muy interesante:

```
MAKE "N 123
PRINT :N
123
MAKE "P GATO
PRINT :P
GATO
```

Como podemos observar la orden MAKE, tanto en la letra N como en la P, guardaron un número y una palabra; ambas serán conservadas en MAKE hasta que no le demos otra entrada.

Podemos hacer que MAKE guarde una LISTA de palabras si las encerramos dentro de corchetes, ejem:

```
MAKE "L [TELEFONO ALAMBRE BOCINA CABLE]
PRINT :L
TELEFONO ALAMBRE BOCINA CABLE
```

NO debemos olvidar que para las entradas de MAKE debemos usar comillas y para el despliegue de la variable se usan : (dos puntos)

Ahora introduzcamos nuevos elementos para el uso de la PALABRAS y de las LISTAS.

Usaremos las órdenes:

FIRST, LAST, BUTFIRST, BUTLAST, ITEM, COUNT, IFTRUE, IFFALSE

Primero las más sencillas; un ejemplo puede darnos la idea exacta para que se utilicen:

APRENDIZAJE POR COMPUTADORA

```

MAKE "X AUTOBUS

PRINT :X
AUTOBUS           =   IMPRIME TODA LA PALABRA

PRINT FIRST :X
A                 =   IMPRIME LA PRIMERA LETRA

PRINT LAST :X
S                 =   IMPRIME LA ULTIMA LETRA

PRINT BUTFIRST :X
UTOBUS           =   IMPRIME SIN LA PRIMERA LETRA

PRINT BUTLAST :X
AUTOBU           =   IMPRIME SIN LA ULTIMA LETRA

PRINT ITEM 4 :X
0                 =   IMPRIME EL ELEMENTO No. 4

PRINT COUNT :X
7                 =   IMPRIME EL TOTAL DE ELEMENTOS
    
```

Esto nos dará una buena idea para poder hacerlo ahora con LISTAS.

```

MAKE "A (LECHE CARNE FRUTA DULCE MIEL VERDURAS)

PRINT :A
LECHE CARNE FRUTA DULCE MIEL VERDURAS

PRINT FIRST :A
LECHE

PRINT LAST :A
VERDURAS

PRINT BUTFIRST :A
CARNE FRUTA DULCE MIEL VERDURAS

PRINT BUTLAST :A
LECHE CARNE FRUTA DULCE MIEL

PRINT ITEM 5 :A
MIEL

PRINT COUNT :A
6
    
```

Podemos recurrir al editor de Logo para hacer muchos ejemplos de Listas; lo haremos primero con un procedimiento sencillo:

```

TO PALABRA
  PRINT [ANALICEMOS ESTA PALABRA:]
  PRINT :P
  PRINT [ELEMENTOS QUE TIENE =]
  PRINT COUNT :P
  PRINT [PRIMER ELEMENTO =]
  PRINT FIRST :P
  PRINT [ULTIMO ELEMENTO =]
  PRINT LAST :P
  PRINT [CUARTO ELEMENTO =]
  PRINT ITEM 4 :P
END
    
```

Con este procedimiento, podemos hacer análisis de palabras.

Antes, debemos decirle a la computadora, qué palabra queremos que analice. Creamos una con la orden MAKE. Ejem:

```
MAKE "P "FERROCARRIL (ENTER)
```

Ahora escribimos el título del procedimiento y la palabra que queremos analizar:

```

?PALABRA "FERROCARRIL (enter) y tenemos lo siguiente:
?ANALICEMOS ESTA PALABRA =
?FERROCARRIL
?ELEMENTOS QUE TIENE =
?11
?PRIMER ELEMENTO =
?F
?ULTIMO ELEMENTO =
?L
?CUARTO ELEMENTO =
?R
    
```

Podemos observar que la palabra "FERROCARRIL fue descompuesta y analizada en su totalidad.

Problemos ahora con una variante en el edición:

```

TO PALABRA :P
  (PRINT [ANALICEMOS ESTA PALABRA :] :P)
  (PRINT [ELEMENTOS QUE TIENE =] COUNT :P)
  (PRINT [PRIMER ELEMENTO =] FIRST :P)
  (PRINT [ULTIMO ELEMENTO =] LAST :P)
  (PRINT [CUARTO ELEMENTO =] ITEM 4 :P)
    
```

END

Observando cuidadosamente hemos reducido la cantidad de instrucciones en la edición y estaremos seguros que el resultado será el mismo. Lo que ocurre es que podemos envolver dos definiciones si las incluimos dentro de un paréntesis, pero sobre todo no necesitamos estar creando cada palabra con la orden MAKE en cada caso de analizar éstas. Ahora el resultado será como sigue:

```
?PALABRA "VERDURA
?ANALICEMOS ESTA PALABRA : VERDURA
?ELEMENTOS QUE TIENE = 7
?PRIMER ELEMENTO = V
?ULTIMO ELEMENTO = A
?CUARTO ELEMENTO = D
```

Otro ejemplo:

```
?PALABRA "ALUMNO
?ANALICEMOS ESTA PALABRA : ALUMNO
?ELEMENTOS QUE TIENE = 6
?PRIMER ELEMENTO = A
?ULTIMO ELEMENTO = O
?CUARTO ELEMENTO =M
```

También podemos hacerlo con Listas. Vamos a crear otro procedimiento similar que nos permita guardar en una lista varias palabras y hacer análisis de ellas:

```
TO FRASE
PRINT [ESCRIBE UNA LISTA DE PALABRAS DENTRO DE [ ] ]
MAKE "F RW
(PRINT [ANALICEMOS ESTA LISTA :] :F)
(PRINT [ELEMENTO QUE TIENE =] COUNT :F)
(PRINT [PRIMER ELEMENTO DE LA LISTA =] FIRST :F)
(PRINT [ULTIMO ELEMENTO DE LA LISTA =] LAST :F)
(PRINT [TODOS LOS ELEMENTOS MENOS EL ULTIMO =] BUTLAST :F)
(PRINT [TODOS LOS ELEMENTOS MENOS EL PRIMERO =] BUTFIRST :F)
(PRINT [CUARTO ELEMENTO DE LA LISTA =] ITEM 4 :F)
END
```

Con este procedimiento, estamos incluyendo en él la orden MAKE, y el comando RW (READWORD) = (LEER PALABRA), que nos será muy útil al hacer nuestras listas. Para ver los resultados obtenidos, basta con teclear la palabra FRASE y tendremos esto:

```
?FRASE
?ESCRIBE UNA LISTA DE PALABRAS DENTRO DE [ ]
? [PERA MANZANA TEJOCOTE CIRUELA MANGO]
?ANALICEMOS ESTA LISTA: PERA MANZANA TEJOCOTE CIRUELA MANGO
?ELEMENTOS QUE TIENE = 5
?PRIMER ELEMENTO DE LA LISTA = PERA
?ULTIMO ELEMENTO DE LA LISTA = MANGO
?TODOS LOS ELEMENTOS MENOS EL ULTIMO = PERA MANZANA TEJOCOTE
CIRUELA
?TODOS LOS ELEMENTOS MENOS EL PRIMERO = MANZANA TEJOCOTE
CIRUELA MANGO
?CUARTO ELEMENTO DE LA LISTA = CIRUELA
```

Ahora veamos dos nuevas órdenes : WORD Y SENTENCE

La orden Word (palabra) es una orden de Logo que toma dos palabras como entrada y da como salida la combinación de una palabra más grande. Ejem:

```
?PRINT WORD "FERRO "CARRIL
?FERROCARRIL
?PRINT WORD "FERROCARRIL "GIGANTE
?FERROCARRILGIGANTE
```

WORD puede tener más de dos entradas si la orden y sus entradas se encierran dentro de paréntesis. Ejem:

```
?PRINT (WORD "ARCHI "RECONTRA "GRANDE
?ARCHIRECONTRAGRANDE
```

Pero si la palabra WORD está fuera del paréntesis, el sistema nos enviará un mensaje de error:

```
TOD MUCH INSIDE ( ) 'S
```


SENTENCE se utiliza para combinar palabras y listas en una lista más grande. Por lo general tiene dos entradas. Cada una de ellas puede ser una palabra o una lista. Ejem:

```
? PRINT SENTENCE "DOS [GATOS ENCERRADOS]
? DOS GATOS ENCERRADOS
?PRINT SENTENCE [DOS GATOS ENCERRADOS] [EN UNA CAJA]
?DOS GATOS ENCERRADOS EN UNA CAJA
```

La orden SENTENCE tiene como abreviatura SE.

Otra orden importante es READLIST , cuya abreviatura es RL, es una orden que espera que el usuario teclee una línea desde el teclado y dé como salida esa línea como una lista. Ejem:

```
TO CONVERSAR
PRINT [ESCRIBA ALGUNA COSA QUE ME QUIERA DECIR]
PRINT SE [YO LO VOY A REPETIR] READLIST
END
```

Una vez que editamos este procedimiento podemos "conversar" con la computadora, pues obtendremos resultados como este:

```
?CONVERSAR
?ESCRIBA ALGUNA COSA QUE ME QUIERA DECIR
(La computadora espera que se teclee algo)
?ME GUSTARIA IR DE PASEO
?YO LO VOY A REPETIR ME GUSTARIA IR DE PASEO
```

La computadora contestó la frase que se tecleó, pero además concluyó con la frase que tenía guardada en la línea de SENTENCE. Ahora elaboraremos un procedimiento que nos permitirá añadir frases y más frases y no parará hasta que nosotros mismos lo

interrumpamos con las teclas CTRL-ESC., o sea, seguiremos la misma secuencia de la recurrencia de nuestros anteriores ejercicios.

```

TO ORACIONES :FRASES
PRINT [ESCRIBE UNA FRASE]
MAKE "AMADIR READLIST
MAKE "FRASES SENTENCE :FRASES :AMADIR
PRINT [LA FRASE YA ESTA ESCRITA]
PRINT :FRASES
ORACIONES :FRASES
END
    
```

El resultado de este procedimiento seria así:

```

ORACIONES HOLA
ESCRIBA UNA FRASE
DONDE ESTOY
LA FRASE YA ESTA ESCRITA
HOLA DONDE ESTOY,
ESCRIBA UNA FRASE
CREO QUE EN LA ESCUELA
LA FRASE YA ESTA ESCRITA
HOLA DONDE ESTOY, CREO QUE EN LA ESCUELA
ESCRIBA UNA FRASE
AHORA TENGO QUE IRME
LA FRASE YA ESTA ESCRITA
HOLA DONDE ESTOY, CREO QUE EN LA ESCUELA AHORA TENGO QUE IRME
ESCRIBA UNA FRASE....
    
```

El procedimiento continuaría hasta que nosotros quisiéramos y sólo se detendrá en cuanto nosotros oprimamos las teclas descritas.

Hay procedimientos que si ponemos la condición STOP (alto), si lograrán detenerse en su realización. Ejem.

```

TO PRUEBA1
PRINT [¿ COMO SE LLAMA UNO DE LOS HUESOS DE LA PIERNA?]
MAKE "CONTESTACION READLIST
IF :CONTESTACION = [FENUR] [PRINT [CORRECTO MUY BIEN!] STOP]
PRINT [TU RESPUESTA NO ES CORRECTA. TRATA DE RECORDAR OTRA VEZ...]
PRUEBA1
END
    
```

Estamos usando una condición que permitirá al alumno probar su conocimiento, si acaso no sabe la respuesta, la computadora le pide que vuelva a intentarlo, o bien, si la respuesta fue correcta, lo estimula y para el procedimiento.

En este caso se vería como sigue:

```

PRUEBA1
? COMO SE LLAMA UNO DE LOS HUESOS DE LA PIERNA?
TIBIA
TU RESPUESTA NO ES CORRECTA. TRATA DE RECORDAR OTRA VEZ...
CUBITO
TU RESPUESTA NO ES CORRECTA. TRATA DE RECORDAR OTRA VEZ...
CRANEO
TU RESPUESTA NO ES CORRECTA. TRATA DE RECORDAR OTRA VEZ..
FEMUR
CORRECTO MUY BIEN!
    
```

Veamos otro tipo de procedimiento que podemos utilizar con Listas de palabras, que en caso de no haber respuesta correcta el sistema puede abandonar inmediatamente su proceso:

```

TO PRUEBA2
PRINT [CUAL ES LA ABREVIATURA DE PROFESOR?]
MAKE "RESPUESTA READLIST
IF :RESPUESTA = [PROFR] [PRINT [PERFECTO, ASI SE ABREVI] STOP]
PRINT [NO, NO ESTA BIEN]
PRINT [QUIERE VOLVER A INTENTAR?]
MAKE "INTENTAR READLIST
IF :INTENTAR = [NO] [PRINT [LA RESPUESTA ES :PROFR] STOP]
IF :INTENTAR = [SI] [PRUEBA2 STOP]
PRINT [ABANDONO ! PORQUE NO RESPONDE NADA]
END
    
```

Ahora al correrlo en la máquina funcionará de esta manera:

```

PRUEBA2
CUAL ES LA ABREVIATURA DE PROFESOR?
PFR
NO, NO ESTA BIEN
QUIERE VOLVER A INTENTAR?
SI
CUAL ES LA ABREVIATURA DE PROFESOR?
PROF
    
```

NO, NO ESTA BIEN
QUIERE VOLVER A INTENTAR?
NO
LA RESPUESTA ES PROFR

En caso afirmativo la secuencia seria asi:

PRUEBA2
CUAL ES LA ABREVIATURA DE PROFESOR?
PROFR
PERFECTO, ASI SE ABREVI A

Y finalmente si no responde ni positiva ni afirmativamente el caso
quedaría de esta última manera:

PRUEBA2
CUAL ES LA ABREVIATURA DE PROFESOR?
?
ABANDONO! PORQUE NO RESPONDE NADA.

Podemos elaborar un tercer procedimiento para incluir dos
condiciones; una en letra y otra en número:

TO PRUEBA3
PRINT [¿CUANTO ES CINCO MAS CINCO?]
MAKE "CONCLUSION READLIST
IF [CONCLUSION = [10]] [PRINT [CORRECTO] STOP]
IF [CONCLUSION = [DIEZ]] [PRINT [MUY BIEN] STOP]
PRINT [NO ES EL RESULTADO CORRECTO]
PRUEBA3
END

El resultado que tendremos sera:

PRUEBA3
¿CUANTO ES CINCO MAS CINCO?
3
NO ES EL RESULTADO CORRECTO
CUANTO ES CINCO MAS CINCO?
NUEVE
NO ES EL RESULTADO CORRECTO
¿CUANTO ES CINCO MAS CINCO?
DIE?

MUY BIEN

O bien:

?CUANTO ES CINCO MAS CINCO?

10

CORRECTO

Indudablemente, estos sencillos ejemplos nos dan idea de cómo podríamos utilizar las listas de palabras para hacer una cantidad enorme de cuestionarios y que la máquina misma de contestación a las preguntas; o bien, ponerle condicionamientos al alumno para que pueda intervenir en su proceso creativo o mejor aún, investigar algunas cuestiones sencillas que le dieran oportunidad de contestar con mayor precisión y certeza.

El campo del lenguaje Logo en el tratamiento de palabras y listas es aún casi virgen, pues las posibilidades que nos brinda pueden ser enormes. Por mi parte creo que aún se tiene la perspectiva de hacer grandes investigaciones en esta área.

Confío plenamente que con la ayuda de nuevos elementos y las experiencias que logre obtener tanto personales como de mis alumnos, crearán un ambiente positivo para impulsar este región en la informática educativa.

CAPITULO VI

CAPITULO VI.- LOGO EN LA ESCUELA

6.1.- EXPERIENCIAS EN EL CAMPO DE LA DOCENCIA

Desde el comienzo de la utilización de la computadora fuera de los laboratorios de investigación, se vislumbraron las posibilidades que ofrecía en el mundo de la enseñanza. Su potencia, su rapidez y su capacidad de retener y utilizar un gran volumen de datos, le permitió proporcionar un acceso más diferenciado al proceso educativo, mejor adaptado a la situación concreta de cada estudiante, respetando niveles de conocimiento y otras variables individuales.

Puedo afirmar que esta reflexión que comentaban mis maestros de computación cuando inicié mi preparación en el Politécnico, guarda un enorme secreto que a medida que pasa el tiempo puedo corroborar más y más. Y es que, hasta hace poco tiempo, las computadoras únicamente estaban al servicio de grandes consorcios, administraciones de gobierno y particulares, de la milicia, etc.; pero no tenían un enfoque directo para la educación.

Los objetivos que persigo en este capítulo, responden perfectamente a los mismos que se buscan en la Licenciatura en Educación Primaria:

- 1.- Valorar críticamente las aportaciones de la Tecnología Educativa en el proceso enseñanza-aprendizaje de la Escuela Primaria.
- 2.- Fundamentar teórica y metodológicamente el empleo de la Informática en la práctica docente.

3.- Desarrollar el espíritu crítico, una actitud de apertura y el compromiso profesional para aprovechar racionalmente las innovaciones tecnológicas en el campo de la escuela.

Ahora bien, antes de iniciar el relato de mis experiencias en este campo concreto, me propongo resaltar algunas ideas que siento son de mucha importancia.

La implantación del Programa de Modernización Educativa pide que se hagan modificaciones a los actuales programas y planes de estudio, pone de manifiesto la necesidad de ejercer una educación más relevante y se enfatiza en elevar la calidad de la enseñanza; en este sentido es necesario hacer notar que este enfoque requiere de una innovación sustantiva en relación con el enfoque que tradicionalmente se ha venido manejando en la educación básica. Ahora bien, la adopción de este nuevo enfoque implica que alumnos y maestros generen conflictos y resistencias al cambio, puesto que deben tomar decisiones en relación a la necesidad de adoptar estrategias participativas para lograr una visión más amplia en su aplicación, especialmente cuando se pretenden nuevos perfiles de resultado o aprendizajes más significativos en los alumnos.

El Programa de Modernización Educativa señala también que el sistema educativo debe ser capaz de proporcionar al educando los conocimientos y habilidades para aprender de manera autónoma, descubrir y asumir valores, analizar y resolver problemas, vivir en sociedad y aportar todo ello para mejorar sus condiciones de vida y contribuir eficazmente al desarrollo del país.

Estos propósitos sólo se alcanzan si la educación se transforma; en el campo de la informática educativa, se vislumbran estas metas.

Creo que es necesario comentar que cualquier proceso de cambio educacional es complejo y requiere de tiempo para su nueva adaptación y que este cambio requiere necesariamente de conocimientos que le permitan acercarse a los objetivos que se pretenden lograr.

Diseñar y producir libros de materias específicas, como lo es la computación, puede ser un trabajo realmente arduo, si no existen elementos para hacerlo; hoy en día existen muchos libros de cómputo, pero no de cómputo educativo, lo que se traduce en que hay necesidad de empezar a crearlos.

Mi trabajo que hoy presento no pretende ser un libro ni mucho menos, pero sí una contribución a la inquietud de muchas personas que como yo estamos interesados en aportar nuestro grano de arena a la vida educacional de nuestras escuelas primarias; y que mejor que empezar por los alumnos de Licenciatura, que son los que tendrán la oportunidad de preparar a las nuevas generaciones. Afortunadamente hoy en día, el interés de preparar alumnos para enfrentar los retos de la vida, ha hecho que las autoridades educativas piensen en proveerlos de una mejor calidad en la enseñanza y en sus actitudes personales, una conciencia de productividad.

La aplicación del Lenguaje Logo con niños, fue mi primera experiencia directa con computadoras. Apenas estaba conociendo el lenguaje cuando hubo necesidad de enseñar a estos pequeños alumnos, cuyas edades fluctuaban entre los 8 y 10 años. Realmente no estaba seguro si la impartición del curso fuera a dar resultados visibles en poco tiempo o a largo plazo; los niños que se presentaron a recibir el curso, no habían tenido oportunidad de manipular una computadora, esto es, desconocían por completo su manejo. Los trabajos se iniciaron con

computadoras de las llamadas "MICROSEP" que afortunadamente se manejan con instrucciones en español, lo cual facilitaba el aprendizaje de los comandos con cierta rapidez.

Las sesiones se desarrollaron diariamente con un horario de dos horas. Nos valimos de algunos recursos didácticos para dar la introducción al lenguaje Logo y procedimos a la familiarización con los equipos de cómputo.

No fue difícil para los niños entender el funcionamiento de las máquinas, realmente tienen una facilidad sorprendente; unos cuantos ejemplos de ejercicios y había soltura para ejecutar las órdenes iniciales.

Aplicando el método inductivo, hicimos que los alumnos empezaran por crear algunas líneas que se les ocurrieran y dando oportunidad a que su inventiva fuera el primer contacto con la computadora. Una vez que encontraron ciertas rutinas, procedimos a organizar las líneas hechas al azar por líneas con características más definidas. Vinieron las primeras figuras: cuadrado, rectángulo, rombo; los alumnos entendieron rápidamente los conceptos de giros a la derecha y a la izquierda. Creí que iba a ser necesario que les explicara los ángulos, pero para sorpresa mía uno de ellos intuyó la aplicación de éstos. Esto dio pie para que los otros niños tuvieran la curiosidad por saber más sobre ángulos en el momento de crear un triángulo, pues únicamente manejaron el giro a la derecha y a la izquierda de 90 grados. Para dos horas que trabajé el primer día con ellos, no podía creer que hubiera avanzado tan buen trecho en mi programación de enseñanza, pues llegamos más lejos de lo planeado.

Las sesiones siguientes fueron más intensas y fecundas. Los alumnos me sorprendían con la facilidad que elaboraban sus figuras, y no faltó quien lograra hacer una figura más audaz que los otros.

Mientras trabajamos en el modo directo, no hubo contratiempo; sin embargo, cuando hubimos de entrar al modo de edición, las cosas empezaron a tomar otro cariz. Empezaron las confusiones y los enredos en la mente de los alumnos, no entendían cómo había que editar y luego dejar que la computadora hiciera todo por ellos.

Pero el problema verdadero era que yo no sabía exactamente cómo explicarles este procedimiento, pues me faltaban algunos detalles de tipo técnico que no sabía esclarecerlos. Mi asesor de cómputo nos ayudó a resolver esas dudas y entonces el trabajo de los niños se disparó de manera insospechada. Tenían frente a ellos la oportunidad de hacer verdaderas creaciones con la ayuda de la máquina y esto les resultaba además de interesante muy entretenido. Mi conducción con el grupo realmente resultaba muy gratificante y sobre todo, yo aprendía al ritmo con que ellos aprendían.

Transcurridos los quince días del curso, hicimos una evaluación de los resultados del grupo de pequeños alumnos. Con plena satisfacción comprobamos que la creación de figuras geométricas, dejó de ser problema para ellos y el manejo de los ángulos fue aprendido de manera razonable y con mucha aplicabilidad para otras figuras.

En este sentido puedo afirmar que la actitud de los niños al iniciar el curso era de temor y expectación, pero al final era de seguridad y satisfacción porque aprendieron algo verdaderamente útil, pero sobre todo de una manera muy divertida. Sostienen que ese tipo de aprendizajes es mucho más atractivo que el de ir a la escuela y

observar las figuras en el pizarrón, en lugar de crearlas ellos mismos en la computadora.

Esto me hace pensar que si pudiéramos lograr que en las escuelas primarias hubiera computadoras para facilitar el aprendizaje de algunos conceptos matemáticos o de español, realmente sería útil y provechoso para los alumnos, ya que lograríamos que los alumnos incrementaran su pensamiento científico, despertaran su criticidad y acrecentaran su creatividad.

Sin duda otra de las experiencias que puedo narrar es la de mis alumnos de la Licenciatura, donde realmente es mi campo de acción.

La currícula de la carrera de Licenciatura en Educación Primaria, contempla en el quinto semestre, el espacio de Tecnología y Computación Educativa, mismo que imparto desde hace tres años. Conviene aclarar que el espacio mencionado sólo se da durante un semestre y se interrumpe hasta el año siguiente; esto desde luego ha propiciado que cuando se lleva un ritmo de trabajo, se ve interrumpido por el inicio de otro semestre que no contempla la continuidad de la materia de computación.

Sin embargo, las experiencias que tengo de cada curso puedo evaluarlas de manera diferente, ya que cada año escolar, se presenta con condiciones materiales y físicas de manera diferente.

Inicialmente la aplicación del lenguaje Logo estuvo condicionada a la cantidad considerable de alumnos y a la escasez de equipos de computación. Tenía 130 alumnos repartidos en cuatro grupos y tan sólo 4 equipos microsep, lo que representaba una computadora por cada 8 alumnos y eso definitivamente no propiciaba un buen aprendizaje.

Al año siguiente la situación empeora; al estropearse una computadora y aumenta la cantidad de alumnos, lo que se tradujo en menos tiempo de computación efectiva para cada alumno. Finalmente en el último año escolar, las autoridades de la S.E.P. tomaron conciencia de las necesidades de la institución y nos enviaron 6 computadoras personales de la I.B.M., más las tres microsep de las que ya disponíamos, dándonos un total de 9 computadoras que divididas entre los alumnos, el número de 10 alumnos por máquina se redujo a 3 alumnos por computadora, lo cual se traduce en otra forma de atender los intereses de los alumnos.

Después de esta pequeña reseña de mi escuela, creo que lo que importa comentar son las estrategias empleadas para solventar, hasta donde fue posible, el manejo del lenguaje y de las máquinas por supuesto.

Para poder trabajar con todos mis alumnos, inicié por reproducir un manual de trabajo del lenguaje Logo, a fin de que todos los alumnos pudieran entender rápidamente lo que pretendíamos aprender.

Mediante lluvia de ideas y las experiencias personales de los alumnos dimos una breve información de los alcances de la computación en la actualidad. Luego le hablé sobre la importancia de la Informática Educativa y las repercusiones que ésta tiene en la formación de los alumnos; posteriormente entramos en materia del conocimiento de lo que es el Lenguaje Logo.

Algunos de los conceptos que se manejan en el capítulo I y II de este trabajo, fueron dados a los jóvenes normalistas, a fin de que ajusten sus criterios y entiendan con mucho más amplitud lo que se persigue en el curso.

Trabajar directamente con los alumnos en las computadoras y con el lenguaje Logo, es muchas veces, más difícil que cuando recordaba la experiencia con los niños de 8 y 10 años. Los futuros licenciados, a pesar de tener entre los 20 y 22 años, presentan inseguridad en su manera de abordar la máquina, la ven como algo ajeno a su vida cotidiana, temen que su actuación sea no satisfactoria para ellos y para sus compañeros. Los conceptos de figuras, grados, repeticiones de órdenes, parece estar en un plano superior al de su intelecto y a pesar de que tienen el conocimiento de ellos no logran encontrar las soluciones con tanta rapidez como lo harían los alumnos de menor edad.

El hecho de que la computadora sea incapaz de aceptar ningún error e incansable no los permita, obliga a replantear constantemente los problemas a dos niveles: primero, tal como nos lo permiten nuestras estructuras mentales y segundo, reformalizado, clarificado y concretado según la exigencias de la computadora, con procedimientos, palabras, parámetros, etc. Sin embargo, una vez que superan esta etapa, los avances son notoriamente muy rápidos, pues ya existe la información necesaria para que ellos mismos dejen escapar su creatividad y conocimiento. En este punto, me doy a la tarea de solicitarles que piensen en las aplicaciones que se pudieran hacer del lenguaje Logo en la escuela primaria y para qué servirían estos conceptos y sus ejemplificaciones durante el proceso del desarrollo del lenguaje informático.

Puedo aseverar que los alumnos se entusiasman con lo alcanzado, pero al mismo tiempo se decepcionan al pensar que en el medio donde les tocará desempeñarse, difícilmente encontrarán lo más indispensables

de recursos didácticos, y menos aún, una computadora; sin embargo creo que lo importante de estos casos, no es el que los alumnos encuentren o no una máquina moderna, sino cómo se les modifica su manera de ver las cosas en el plano intelectual, pues al tratar de seguir los principios de la máquina, saben que lo que debe imperar es la lógica, el razonamiento y la explicación detallada para poder conseguir un objetivo.

Todo esto me lleva a pensar que con el tiempo, en la escuela aparecerán otras modalidades de enseñanza, administrada y gestionada por la computadora o simplemente en torno a ésta. Podría pensarse al final que se acabarían las colas de alumnos esperando que el profesor los atienda personalmente; el alumno tendría un tutor personal a su disposición en todo momento y quizás hasta en su casa tendría uno para seguir trabajando. Esto sería posible con la difusión y popularización de las computadoras personales, que se convertirían de esta manera en los elementos que proporcionarían un entorno educacional individualizado. Ciertamente que esto es quimérico, mas no imposible; en otros países ya está funcionando a pasos acelerados, nuestro país no debe quedarse a la zaga, los esfuerzos deben ser continuos e ininterrumpidos, nosotros podemos pugnar porque realmente suceda en la medida de nuestras posibilidades, pero sobre todo en la medida que concienticemos a las personas que tienen en sus manos la educación del país.

La educación que yo propongo y que persigue el Lenguaje Logo, no es otra cosa que el implantar nuevas formas de aprendizaje, donde el alumno participa de un manera más directa, creativa y divertida; nadie ha dicho que el aprendizaje tenga que estar revestido de

formalidad y adustez, los niños tienen tendencias lúdicas innatas y debemos aprovecharlas para que su aprendizaje sea más significativo. Probablemente, el lenguaje Logo contribuya al desarrollo intelectual del niño, pero quizá no en todas sus direcciones por igual. Hasta cierto punto resulta que favorece más el razonamiento inductivo que el deductivo, el método ascendente de lo concreto a lo general, pero creo que esto ya es una aportación que merece la pena aprovechar en el ámbito de la escuela.

Se puede decir que a través de la observación participativa y de la manipulación directa con los nuevos instrumentos tecnológicos, se pueden lograr aprendizajes que respondan a las necesidades actuales de las generaciones de alumnos que viven en un mundo lleno de tecnología avanzada pero de aplicaciones rudimentarias en el campo educativo.

Para concluir este capítulo quiero repetir un pensamiento que escuché en alguna ocasión:

"Dado que la computadora es capaz de asumir un millar de formas y cumplir un millar de funciones, puede ser atractiva para un millar de personas".

CONCLUSIONES

CONCLUSIONES:

De esta manera llegamos a la concepción más optimista, pero también controvertida, del papel del Lenguaje Logo en la escuela, asumiéndolo como modelo psicopedagógico, es decir, considerándolo como una herramienta con la que podemos superar el concepto tradicional de la educación, entendida como mera transmisión de saberes -que en muchos casos se ha mostrado incapaz de preparar a la mayoría de los alumnos para que elaboren autónomamente nuevas ideas- y darles el uso a las computadoras en la enseñanza el nuevo papel de facilitador del descubrimiento.

Quizá más que un modelo de renovación pedagógica, el Logo puede utilizarse en el aula como la herramienta tecnológica con la que podemos intentar que se conviertan en realidad algunos de los planteamientos que en el nivel teórico de la Modernización Educativa que se ha venido desarrollando en los últimos años, pero que la falta de medios lo suficientemente capaces, flexibles y potentes han limitado su puesta en práctica a centros pilotos, experimentales, de élite o simplemente no han sido extendidos con la profundidad y amplitud suficientes.

Por eso decimos que no es un programa ni un lenguaje más, sino una experiencia de nuevos conocimientos. Una aplicación apropiada de Logo puede funcionar como catalizador de la renovación pedagógica de la escuela. Los trabajos realizados hasta aquí, no demuestran que se

trate de una herramienta definitiva, ni la llamada a solucionar todos los problemas de la escuela, ni en el plano de la teoría ni en el plano de la práctica, pero es probablemente una herramienta que la innovación tecnológica ha puesto en nuestras manos para poder llevar a cabo ciertos objetivos.

El objetivo del lenguaje Logo es el poner en manos de los educadores una herramienta que hace posible realizar una enseñanza activa, individualizada, en la que predominan los aspectos de la creatividad, espíritu de investigación y centrada en los intereses del alumno, y que, además, tiene la decisiva ventaja de que puede utilizarse a partir de los 7 años o incluso antes, ya que los creadores del Logo se dejaron llevar por el objetivo ideal de crear una herramienta sin umbral ni techo. Esto hace posible que los niños más pequeños puedan manipular la computadora a través del teclado numérico de la misma, ya que el programa permite programar estas teclas para que quienes no saben aún ni leer ni escribir, puedan realizar sencillos dibujos a través de un código especial que habilita estas teclas.

Logo puede convertirse en un instrumento que introduzca a los niños en una actividad fundamental en el proceso educativo, según Piaget, el aprender sobre el aprendizaje.

Es de sobra conocida la figura del alumno que se muestra miedoso, o más bien perezoso, a la hora de aprender nuevos conocimientos,

proponer, establecer hipótesis o lanzarse a intentar un nuevo descubrimiento. Para estos objetivos, ciertamente muy ambiciosos, Logo presta una ayuda fundamental, en la medida en que no se le considere como una asignatura más ni como una isla en el conjunto de actividades de la escuela.

Creemos que Logo puede ayudar al estudiante a que adquiera una metodología que le permita resolver los problemas que se le van planteando constantemente en su experimentación con el entorno. Logo constituye algo así como el "método del problema" de Dewey, es decir, una herramienta con la que puede experimentarse con fines didácticos en una gran variedad de campos o micromundos, y en la que el entorno educativo se articula alrededor del principio de dar al alumno un control personal de los poderosos recursos de la informática y le permite establecer un contacto profundo con ideas de tipo científico, matemático y con el arte de construir modelos intelectuales.

Probablemente el niño llegue a conocer las reglas del cálculo de una manera más operativa y profunda si resuelve los problemas que se le van planteando, cuando se le presenta la necesidad de enseñarle a la tortuga a que dibuje en la pantalla algún diseño gráfico.

Ni recibe del maestro conocimientos que pueden no encajar en sus estructuras, ni se está amueblando ni atiborrando la cabeza con leyes

abstractas ni fórmulas copiadas, sino que está interiorizando, asimilando y resolviendo problemas o, por lo menos, aprendiendo a resolverlos; aprendiendo progresivamente las ventajas e inconvenientes de usar aquella regla que en algún caso no le ha funcionado y que tiene que modificar.

En todo caso, el Logo será tanto más útil como herramienta renovadora de la enseñanza cuanto más profundamente se implante en la atmósfera general de la escuela y cuanto más se acentúe el ambiente de cooperación mutua y de trabajo colectivo y no sólo entre el grupo de alumnos sino también entre el equipo de profesores.

Con él nos estamos sirviendo de los avances tecnológicos que han proporcionado los materiales y útiles necesarios para crear entornos propicios al mundo del aprendizaje. Gracias a estos lenguajes que pueden ser dominados incluso por los niños, sin dejar de ser potentes, podemos intentar que los procesos de aprendizaje mejoren, para buscar modelos de exploración más originales y profundos, o para construir alrededor de la computadora un ambiente en el que el niño se podrá enfrentar frecuentemente a situaciones ricas en el plano conceptual e ir adquiriendo una experiencia real de manipulación de conceptos intelectuales importantes.

La puesta en escena debe conducir a un efecto completo, observable y que no se preste a interpretaciones ambiguas y en donde las actividades sean motivantes, variadas, ricas y próximas, por lo menos al principio, a las actividades cotidianas del niño.

El Logo puede ser la herramienta que haga posible realizar lo que se ha estado pensando durante mucho tiempo y que en muchas ocasiones ha resultado utópico.

BIBLIOGRAFIA

BIBLIOGRAFIA

CISNEROS, PATRICIA. "LOGO PARA NINOS"
DIANA, MEXICO 1988. 175 p.

CHATEAU, JEAN. "LOS GRANDES PEDAGOGOS"
F.C.E., MEXICO 1978. 340 p.

CRATTY, BRIANT. "JUEGOS DIDACTICOS ACTIVOS"
ED. PAX, MEXICO 1982. 184 p.

CURTIS, RICHARD. "WORD STAR EN IBM PC"
MC GRAW HILL, MEXICO 1990. 208 p.

DENEY, JOHN "HOW WE THINK"
SELECTOR PUB, USA. 1987. 240 p.

DIENES, Z.P. "LOS PRIMEROS PASOS EN MATEMATICAS"
ED. TEIDE, BARCELONA 1986. 143 p.

JOYANES, AGUILAR. "PROGRAMACION BASIC PARA PC"
MC GRAW HILL, MEXICO 1989. 539 p.

MAYER, ROBERTO. "ACTITUDES POSITIVAS EN LA ENSEMANZA"
ED. PAX, MEXICO 1982. 184 p.

MICROSOFT. "MANUAL DE REFERENCIA DEL MS-DOS"
MICROSOFT CORP., USA. 1989. 480 p.

MICROSOFT. "MANUAL DE WORKS"
MICROSOFT CORP., USA. 1989. 592 p.

NERVI, RICARDO. "DIDACTICA NORMATIVA"
KAPELUZS, MEXICO 1985. 262 p.

APRENDIZAJE POR COMPUTADORA

PAPERT, SEYMOUR. "DESAFIO A LA MENTE"
ED. GALAPAGOS, MEXICO 1982. 255 p.

PALACIOS, JESUS. "LA CUESTION ESCOLAR"
EDIT. LATA/BARCELONA 1984, 647 p.

PAUSEWANG, ELFRIEDE. "JUEGOS DIDACTICOS"
KAPELUZS, ARGENTINA 1979. 106 p.

PIAGET, JEAN. "PSICOLOGIA DEL NINO"
ED. MORATA, MADRID 1984, 172 p.

PIAGET, JEAN. "SEIS ESTUDIOS DE PSICOLOGIA"
SEIX-BARRAL, BARCELONA 1976, 227 p.

PEREZ, CESAR. "CONJUNTO DIDACTICO DE LOGO"
MC GRAW HILL, MEXICO 1991. 180 p.

RODRIGUEZ, MAURO. "PSICOLOGIA EN EJEMPLOS"
TRILLAS, MEXICO 1980. 351 p.

STELLA, S. GILB. "JUEGOS PARA ESCOLARES"
ED. PAX, MEXICO 1980. 143 p.

WATT, DANIEL. "APRENDIENDO CON IBM LOGO"
MC GRAW HILL, MEXICO 1987. 332 p.

WATT, SOFIA. "BASIC AVANZADO PARA NINOS"
F.E.S.A., MEXICO 1988. 158 p.

WATT, SOFIA. "BASIC PARA NINOS"
F.E.S.A., MEXICO 1988. 128 p.

ANEXO

ANEXO : COMANDOS DE ORDENES RAPIDAS

A.- COMANDOS DE LOGO EN LA IBM PC O COMPATIBLES

Para cargar LOGO:

1.- Introduzca el disco del DOS (Sistema Operativo en Disco) en la unidad A y encienda la computadora.

2.- Luego de algunos mensajes aparecerá lo siguiente:
La fecha actual es Mar 1/01/1980
Introduzca nueva fecha (dd/mm/aa):

Teclée la nueva fecha y enter. Ejem: 24/10/91 (enter)

3.- Después aparecerá lo siguiente:
La hora actual es 01:03:28,00
Introduzca nueva hora:

Teclée la hora y enter. Ejem: 15.30 (enter)

4.- Cuando aparezca la letra A> teclée B! y enter

5.- Introduzca el disco de LOGO y teclée LOGO y enter.

6.- Al término de unos segundos aparece este mensaje:
WELCOME TO LOGO

Teclée CS (clear screen) para iniciar el trabajo.

COMANDOS (ORDENES) PARA INICIAR

ORDEN	ABREVIATURA	SIGNIFICA	EJEMPLO
PRINT	PR	IMPRIME	PRINT [HOLA]
CLEARSCREEN	CS	BORRA	CS
FORWARD	FD	ADELANTE	FD 120
BACK	BK	ATRAS	BK 75
RIBTH	RT	DERECHA	RT 45
LEFT	LT	IZQUIERDA	LT 90
ERALL		LIMPIA MEMORIA	

Si escribiera lo siguiente:

?ESCUELA NORMAL

verá este mensaje:

I DON'T KNOW HOW TO ESCUELA (no sé cómo hacer para escuela)

El sistema envía un mensaje de error para que el usuario corrija la última entrada.

Ahora escriba:

?PRINT [ESCUELA NORMAL] - observe resultados

ORDEN	ABREVIATURA	SIGNIFICA
PENUP	PU	LEVANTA PLUMA
PENDOWN	PD	BAJA PLUMA
SETPAL		ACTIVA GRUPO DE COLORES
SETPC		ACTIVA COLOR DE PLUMA
SETBG		ACTIVA COLOR DE PANTALLA
LOAD		CARGA ARCHIVO DE DISCO
HIDETURTLE	HT	ESCONDE TORTUGA
SHOWTURTLE	ST	MUESTRA TORTUGA
WRAP		
FENCE		
WINDOW		VENTANA
CLEAN		LIMPIA
HOME		CENTRO DE PANTALLA
FULLSCREEN	FS ó F4	PANTALLA COMPLETA
MIXEDSCREEN	MS ó F2	PANTALLA DIVIDIDA
TEXTSCREEN	TS ó F1	PANTALLA DE TEXTO

La orden PU hace que la pluma esté levantada y por lo tanto no efectúa ningún trazo.

La orden PD hace que la pluma baje y todas las órdenes que se le den a la tortuga quedan marcadas en la pantalla.

SETPAL selecciona los colores de la pluma y tiene dos entradas:

SETPAL 0 : dando los siguientes colores de la pluma:

SETPC 1	-	VERDE
SETPC 2	-	ROJO
SETPC 3	-	MARRON
SETPC 4	-	COLOR DEL FONDO DE LA PANTALLA

SETPAL 1 :

SETPC 1	-	AZUL CLARO (CYAN)
SETPC 2	-	PURPURA (MAGENTA)
SETPC 3	-	BLANCO

La instrucción SETBG selecciona el color del fondo de la pantalla y puede tener entradas desde el 0 al 15, o sea 16 colores de fondo.

FENCE - fuerza a la tortuga a permanecer adentro de la pantalla, si la medida es mayor que la pantalla, LOGD se quejará anotando: **TURTLE OUT OF BOUND (LA TORTUGA ESTA FUERA DE PANTALLA)**

Ejem: CS
FENCE
FD 200

WRAP.- Vuelve las cosas a la normalidad.

Ejem: FENCE
FD 200
WRAP
FD 200

WINDOW.- Permite que la tortuga se vaya del borde de la pantalla, pero al salir no se vuelve a ver!

Ejem: CB
WINDOW
RT 90
FD 180
LT 150
FD 180

CLEAN.- Borra la pantalla pero deja la tortuga en su lugar!

Ejem: CS
RT 30
FD 20
CLEAN
FD 20

HOME.- Envía la tortuga al centro de la pantalla mirando hacia arriba. Pero no borra nada.

Ejem: CB
FD 80
LT 45
FD 30
HOME

La combinación **HOME-CLEAN** hacen lo mismo que CS.

FULLSCREEN ó FS ó F4.- Le permite ver la pantalla completa de la tortuga pero oculta cualquier cosa escrita o impresa en la misma.

MIXEDSCREEN ó MS ó F2.- Restaura a pantalla dividida. Puede ver cuatro líneas de impresión pero puede ocultarse parte del dibujo de la tortuga.

TEXTSCREEN ó TS ó F1. - Le muestra la pantalla de texto completa.
 No puede ver ninguno de los dibujos de la tortuga aunque sigas
 estando allí. Para volver a ver la tortuga teclee F4 ó F2

Ejem: CS = CLEARSCREEN
 PU = PENUP
 LT 90 = LEFT 90
 FD 100 = FORWARD 100
 RT 90 = RIGHT 90
 PD = PENDOWN
 FD 50 = FORWARD 50
 F4 = FULLSCREEN
 F2 = MIXEDSCREEN
 F1 = TEXTSCREEN
 F2 = MIXEDSCREEN

B.- EDICION

Para editar un procedimiento haga lo siguiente:

Teclee **EDIT**
 (observe que desaparece la pantalla de la tortuga y aparece la
 pantalla del editor).

-Enseguida inicie el procedimiento como está acostumbrado pero con la
 palabra **TO** (en lugar de **ES** de la microsep).

-Luego las instrucciones del cuerpo del procedimiento con los
 comandos en inglés (FD, RT, BK, etc.)

-Termine el procedimiento con la palabra **END** (FIN de la microsep)

-Para dejar el modo de edición pulse la tecla **ESC**.

Ejem: EDIT
 TO BOX
 REPEAT 4[FD 40 RT 90]
 END
 ESC (TECLA)

La computadora imprimirá el nombre del procedimiento definido así:
BOX DEFINED

-Escriba ahora el nombre del procedimiento que editó -(BOX)- y
 aparecerá la figura.

-Para hacer correcciones al procedimiento definido deberá teclear:
EDIT "BOX
 y aparecerán las instrucciones del procedimiento que efectuó antes.

PD = PRINTOUT (SALIDA IMPRESA)

Con esta instrucción puede obtener un listado de todos los pasos de un procedimiento.

Ejem: PD "BOX

Si no puede ver todo el procedimiento completo impreso en la pantalla, teclee F1 más tarde F2 para ver la pantalla dividida.

- POTS = PRINTOUT TITLES (SALIDA IMPRESA DE TITULOS)

Imprimirá en la pantalla los títulos (nombres) de todos los procedimientos.

- PDALL = PRINTOUT ALL (SALIDA IMPRESA DE TODO)

Esta instrucción hace que aparezca impresa en la pantalla todos los procedimientos con sus nombres, así como las instrucciones de cada uno de ellos.

-ERASE = BORRAR

Borra un procedimiento de la memoria de la computadora.

Ejem: ERASE "BOX ó ER "BOX

UTILIZANDO ALGUNAS TECLAS CON EL EDITOR:

--> Desplaza el cursor a la derecha un solo espacio

<-- Desplaza el cursor a la izquierda un solo espacio

↑ Desplaza el cursor hacia arriba una línea

↓ Desplaza el cursor hacia abajo una línea

<<-- Borra el carácter a la izquierda del cursor y hace retroceder el cursor

(enter) Crea una nueva línea y desplaza el cursor hacia abajo al principio de esa nueva línea.

<--> Desplaza el cursor al final de la línea

Shift <--> Desplaza el cursor al inicio de la línea

Pg Dn Desplaza el cursor hacia adelante una pantalla completa cuando la pantalla está llena de texto.

Pg Up Desplaza el cursor hacia atrás una pantalla completa cuando la pantalla está llena de texto.

DEL.- Borra el carácter donde está el cursor y no retrocede un espacio. Del es la abreviatura de "Delete" (borrar).

Ctrl--> Borra la línea completa a la derecha del cursor.

INS.- Crea una línea nueva en el cursor y desplaza hacia abajo todo el resto del texto en una línea. Ins es la abreviatura de "insert" (insertar) .

-SAVE = SALVAR (GUARDAR)

Conserva todos los procedimientos que están en la memoria de trabajo de la computadora en el disco de trabajo de LOGO en un archivo que tendrá un nombre que identifique los procedimientos.

Ejem: SAVE "FIGURAS

-LOAD = CARGA

Lee todos los procedimientos del archivo -x- y los pone en la memoria de trabajo de la computadora.

Ejem: LOAD "FIGURAS

ERASEFILE = BORRA ARCHIVO

Borra un archivo del disco de trabajo de Logo.

-Nunca borre un archivo sin estar seguro de ello, pues la información que tiene el archivo, una vez borrada, es imposible recuperarla.

Ejem: ERASEFILE "FIGURAS

DIR = DIRECTORIO

Este comando presenta una lista de todos los archivos del disco. Todos los archivos tienen la terminación (.LF) después del nombre.

DRIBBLE "LPT1

Imprime en la impresora todo lo que aparece normalmente en la pantalla.

NODRIBBLE

Desactiva la impresora y hace que la computadora imprima sólo en la pantalla del monitor.

REPEAT = REPITA

Hace que la computadora repita una lista de órdenes tantas veces como quiera. Es muy útil cuando se sabe cuántas veces hay que repetir algo.

Ejem:

```
TO CUADRO
REPEAT 4[BK 50 LT 90]
END
```

VARIABLES

La variable es un elemento de información con un nombre. La computadora almacena el nombre y la información en su memoria de trabajo. Puede cambiar fácilmente o variar la información siempre que lo quiera. La variable puede ser un número, una letra, una palabra o una lista.

```
Ejem: TO TRIANGULO #LADO
REPEAT 3[FD #LADO LT 120]
END
```

COMANDO	EJEMPLO
+	FD #LADO + 10
IF	IF #LADO < 10 [STOP]
<	IF #LADO < 5 [STOP]
>	IF #ANGULO > 20 [STOP]
STOP	IF #LADO = 40 [STOP]
=	IF #LADO = 10 [STOP]
	PRINT 5 = 3 + 2
HEADING	IF HEADING = 0 [STOP]
	PRINT HEADING
MAKE	MAKE "COMIENZO HEADING

SIGNO	SIGNIFICA
+	MAS
-	MENOS
/	ENTRE
*	POR
[]	CORCHETES

APRENDIZAJE POR COMPUTADORA

COMANDO	ABREVIATURA	EJEMPLO
WORD		PRINT WORD "HO "LA
SENTENCE	SE	PRINT SENTENCE [HOLA] [OYE]
FIRST		PRINT FIRST "HOLA
BUTFIRST	BF	PRINT BF [HOLA OYE AMIGO]
LAST		PRINT LAST "ADIOS
BUTLAST	BL	PRINT BL [VAMOS TODOS]
READLIST		MAKE "RESPUESTA READLIST
TYPE		TYPE [ADIVINA UN NUMERO]
RANDOM		PRINT RANDOM 20
TEST		TEST !RESPUESTA = 7
IFTRUE	IFT	IFT PRINT [HURRA]
IFFALSE	IFF	IFF PRINT [LO SIENTO]
AND		IF AND !NUM1 = 1 !NUM2 = 0 [PRINT [DE ACUERDO]]
OR		IF OR !AND = [DOS] !ANS= [] [PRINT [CORRECTO!]]
READNUMBER		MAKE "RESPUESTA READNUMBER

Estos son los comandos (ordenes) que se usarán a través del desarrollo de la metodología para aprender con Logo y que servirán como guía rápida en caso de tener alguna duda acerca del empleo de ellos.



FIGURAS

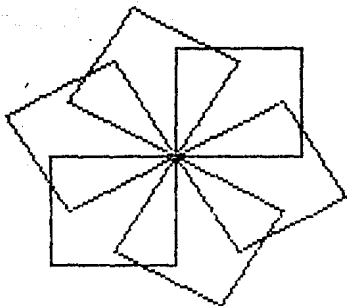
CON

LOGO

```
TO OCHO  
CS REPEAT 6 [UNO LT 60]  
END
```

```
TO UNO  
REPEAT 4 [FD 50 RT 90]  
END
```

LOGO EDITOR

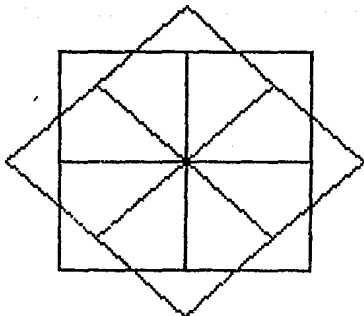


```
TO FIGURA  
CS SETPC 3 VENTANA RT 45  
VENTANA  
END
```

```
TO VENTANA  
REPEAT 4 [VIDRIO LT 90]  
END
```

```
TO VIDRIO  
REPEAT 4 [FD 50 RT 90]  
END
```

LOGO EDITOR



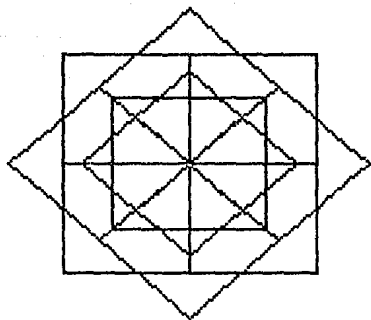
```
TO FIGURA  
REPEAT 8 [CUADROS RT 45]  
END
```

```
TO CUADROS  
CUADRO  
CUADRITO  
END
```

```
TO CUADRO  
REPEAT 4 [FD 50 RT 90]  
END
```

```
TO CUADRITO  
REPEAT 4 [FD 30 RT 90]  
END
```

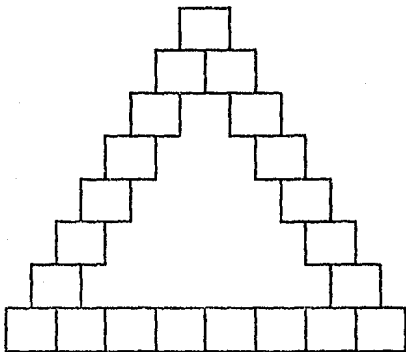
LOGO EDITOR



```
TO PYRAMIDE
CS PU
SETX -80 SEY -30 FT HT
REPEAT 7 ESCALON FD 20 RT 90
FD 10 LT 90
RT 90 FD 10 RT 90
REPEAT 7 ESCALON FD 20 LT 90
FD 10 RT 90
REPEAT 6 ESCALON RT 90 FD 20
LT 90
END
```

```
TO ESCALON
SETPC 2
REPEAT 4 [FD 20 RT 90]
END
```

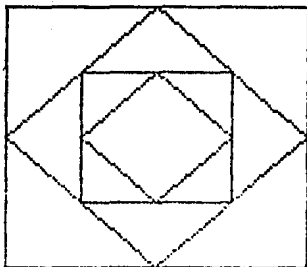
LOGO EDITOR



```
TO REHILETE :N
HT CS CT SETH 0 PU
SETX -:N / 2 SETY -:N / 2 PD
FOR :N
  FOR :N / SORT 2
  FOR (:N / SORT 2) / SORT 2
  FOR (:N / SORT 2) / SORT 2
  / SORT 2
  END
END
```

```
TO ROM :N
REPEAT 4 [PD :N RT 90]
RT 90 PD :N / 2
RT 270 MAKE "K :K + 1
END
```

LOAD EDITOR



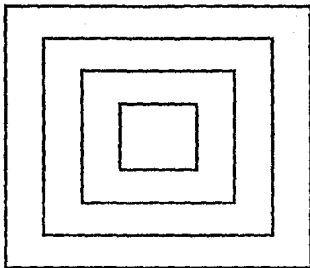

```

TO CONCENTRICOS :N
  IF :N < 0 [ PU SETX 0 SETY -1N / 2 SETY -1
  N / 2 PD
  CUADRI :N
  PU SETX -1N / 2 + 1N / 8 SETY -1N / 2 +
  1N / 8 PD
  CUADRI 3 / 4 1N
  PU SETX -1N / 2 + 1N / 4 SETY -1N / 2 +
  1N / 4 PD
  CUADRI 1 / 2 1N
  PU SETX -1N / 2 + 3 / 4 1N / 8 SETY -1N /
  2 + 3 / 4 1N / 8 PD
  CUADRI 1 / 4 1N
  END

TO CUADRI :N
  REPEAT 4 [FD :N RT 90]
  END

```

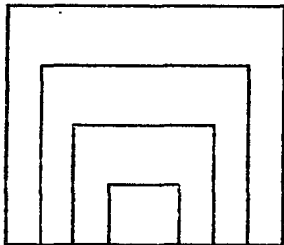
LOGO EDITOR



```
TO UNILATERALES :N  
HT CB RT PU SETX 0 SETX :N / 2 SETX :N  
N / 2 PD  
CUADRO :N  
PU SETX :N / 8 :N PD  
CUADRO 3 / 4 :N  
PU SETX :N / 4 :N  
CUADRO :N / 2  
CUADRO :N / 4 PD  
CUADRO :N / 4  
END
```

```
TO CUADRO :N  
REPEAT 4 [FD :N RT 90]  
END
```

LOGO EDITOR



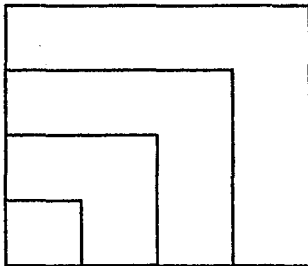
```
TO CUADROSINTERIOS :N
  RT 90 CT BETH 0 FU
  SETX -:N / 2 SETY -:N / 2 PD
  CUADRO :N
  CUADRO :N / 4 1 :N
  CUADRO :N / 2 1 :N
  CUADRO :N / 4 1 :N
  END

```

```
TO CUADRO :N
  REPEAT 4 [FD :N RT 90]
  END

```

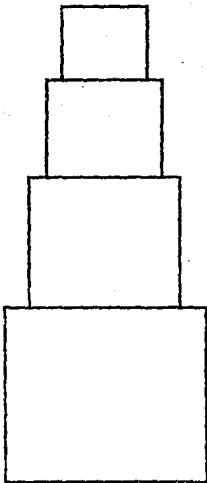
LOGO EDITOR



```
TO T088E IN
HT 08 07 SETH 0 PU
SETX 1IN / 2
SETX 1E / 4 2 IN PD
CUG IN
CUG 6 / 3 1 IN
CUG 6 / 3 1 6 / 8 1 IN
CUG 6 / 3 1 6 /
CUG 6 / 3 1 IN
```

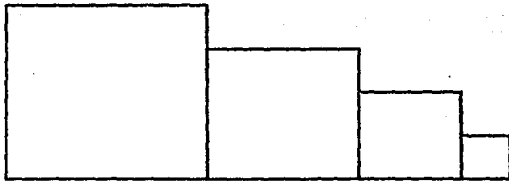
```
TO CUG IN
REPEAT 08FD IN RT 700
FD 1 / 8 1 IN SETH 0
END
```

LAGO EDITOR



```
TO HORIZONTALS :N  
HT CS CT DETH 0 SETX 5 / 4 * :N  
REPEAT 4 CFD :N RT 90  
FU SETX XCOR + 1 :N PD  
REPEAT 1 CFD 3 / 4 * :N RT 90  
FU SETX XCOR + 3 / 4 * :N PD  
REPEAT 4 CFD 1 / 2 * :N RT 90  
FU SETX XCOR + 1 / 2 * :N PD  
REPEAT 4 CFD 1 / 4 * :N RT 90  
FU
```

LOGO EDITOR

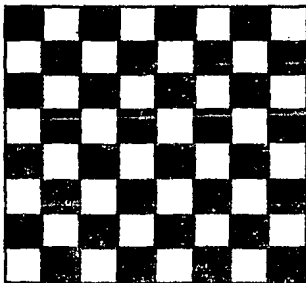


TO AJEDREZ
HT
POSICION
REPEAT 4 (LINEAR
PD REPEAT 4 (REPEAT 4 (PD 150 LT 90 FD
100 LT 90))
END

TO POSICION
CS CT
FU
SETX -64
SETY 64
END

TO LINEA
REPEAT 8 (REPEAT 4 (PD FD 150 PU FD 150)
RT 90 FD 1 RT 90 REPEAT 4 (FU FD 15 PD
FD 150 LT 90 FD 1 LT 90)
REPEAT 8 (REPEAT 4 (FU FD 15 FD FD 150)
RT 90 FD 1 RT 90 REPEAT 4 (PD FD 15 FU
FD 150 LT 90 FD 1 LT 90)
END

LOGO EDITOR

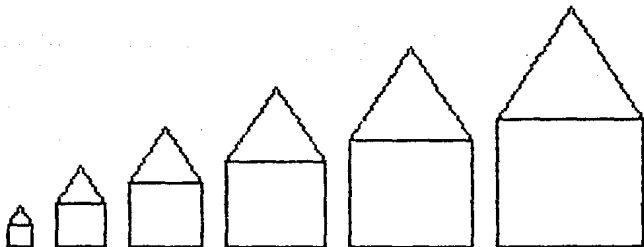


```
TO CASAS
CS FU
SETX -120 SETY -40 PD
CASA 10 AVANZAR 10
CASA 20 AVANZAR 20
CASA 30 AVANZAR 30
CASA 40 AVANZAR 40
CASA 50 AVANZAR 50
CASA 60
END
```

```
TO AVANZAR :LADO
FU RT 90 FD :LADO + 10
LT 90 PD
END
```

```
TO CASA :LADO
CUADRO :LADO FD :LADO
RT 30 TRIANGULO :LADO
LT 30 BK :LADO
END
```

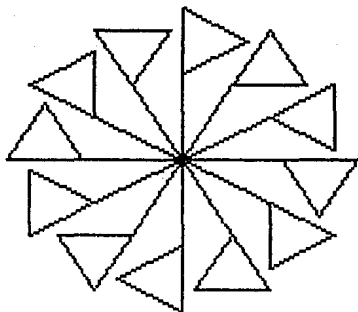
LOGO EDITOR



```
TO BANDERAS
CS SETPC 2
REPEAT 12[BANDERIN RT 30]
END
```

```
TO BANDERIN
FD 40
REPEAT 3[FD 30 RT 120] BK 40
END
```

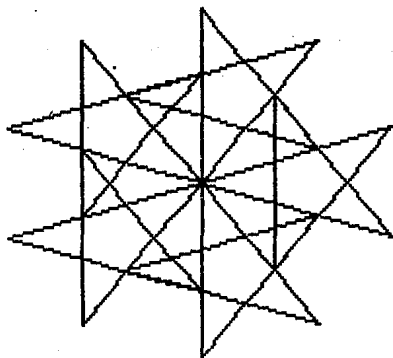
LOGO EDITOR




```
TO ESTRELLAS  
CS  
REPEAT 5[PENTA RT 360/5]  
END
```

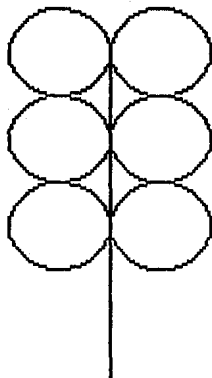
```
TO PENTA  
SETPC 1  
REPEAT 5[FD 80 RT 144]  
END
```

LOGO EDITOR



```
TO ARDS
CS SETPC 2 HT BK 70
FD 150 RCIRCLE 20
LCIRCLE 20 BK 40 SETPC 1
RCIRCLE 20 LCIRCLE 20
BK 40 SETPC 3 RCIRCLE 20
LCIRCLE 20
END
```

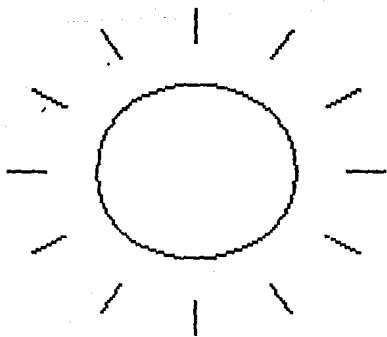
LOGO EDITOR



```
TO SOL
CS SETPC 2 HT PU
FD 40 RT 90 FD
RCIRCLE 40 LT 90
PU BK 40
REPEAT 12 [RAYOS LT 30]
END
```

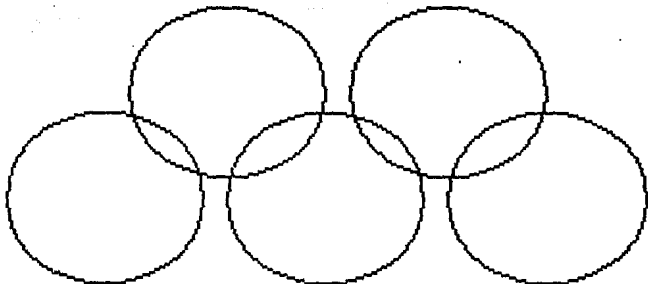
```
TO RAYOS
SETPC 1 PU
FD 60 FD FD 15
BK 15 PU BK 60
END
```

LOGO EDITOR



```
TO OLYMPIC
CS SETPC 1 PU
SETX -110 SETY -30 PD
RCIRCLE 40 SETPC 2 PU
SETX -20 SETY -30 PD
RCIRCLE 40 SETPC 3 PU
SETX 70 SETY -30 PD
RCIRCLE 40 PU
SETX -60 SETY 30 PD
RCIRCLE 40 SETPC 1 PU
SETX 30 SETY 20 PD
RCIRCLE 40
END
```

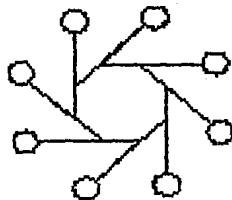
LOGO EDITOR



```
TO PALETA
FD 40 LT 90 SETPC 1
RCIRCLE 5 SETPC 2 RT 90
BK 25 RT 45
END
```

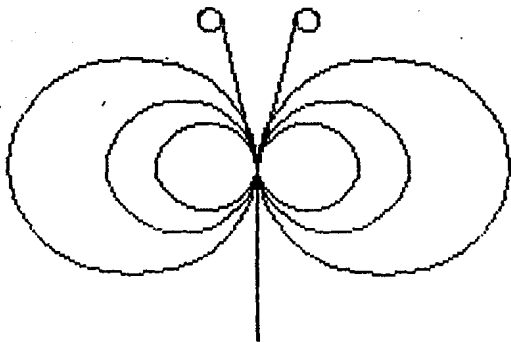
```
TO PALETON
CS HT REPEAT [PALETA]
END
```

LOGO EDITOR



```
TO MARIPOSA
  SETPC 1 RCIRCLE 50 LCIRCLE 50
  SETPC 2 RCIRCLE 30 LCIRCLE 30
  SETPC 3 RCIRCLE 20 LCIRCLE 20
  SETPC 1 BK 80 FD 80 SETPC 2
  RT 12 FD 70 SETPC 3 RCIRCLE 5
  SETPC 2 BK 70 LT 24 FD 70
  SETPC 3 LCIRCLE 5 SETPC 1
  HOME HT
END
```

LOGO EDITOR

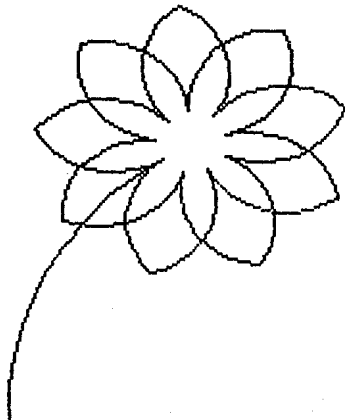


```
TO FLOR
CS HT TALLO
REPEAT 9 [PETALO]
END
```

```
TO PETALO
REPEAT 10 [FD 6 RT 10]
RT 60
REPEAT 10 [FD 6 RT 10]
LT 180
END
```

```
TO TALLO
PU LT 135
FD 100 RT 135 PD
SETPC 1
REPEAT 10 [FD 14 RT 6]
LT 180
END
```

LOGO EDITOR



```
TO ESPERAS  
CS HT  
REPEAT 12[CIRCULO LT 30]  
END
```

```
TO CIRCULO  
SETPC 2  
REPEAT 90[FD 4 RT 4]  
END
```

LOGO EDITOR

