



EDITORIAL

EL COMPUTADOR PERSONAL es un concepto y una actividad que nació con el uso de los microcomputadores, como una herramienta de sobremesa en actividades más relacionadas con la toma de decisiones, que con actividades características de los computadores.

Si bien es cierto, que los computadores con su capacidad de ordenamiento de datos dentro de la empresa, ha significado que la toma de decisiones sea sistematizada como nunca antes lo fue, la aparición del programa VISICALC, desarrollado en principio como algo propio de los microcomputadores, marcó un hito importante en el uso de los computadores personales en la gran empresa, específicamente en el área de Finanzas, pues permitió hacer un modelo financiero de la empresa a un costo mínimo. Lo anterior se tradujo en que los ejecutivos pudieran hacer todo tipo de sensibilizaciones en forma inmediata, sin tener que recurrir al Departamento de Informática y Computación de la empresa, que con otras prioridades nunca podía ser oportuno y efectivo con la información solicitada por el departamento en particular.

Las sensibilizaciones nos permiten entrar en el juego de auscultar el futuro financiero de la empresa, utilizando un modelo que responda a todas las preguntas de: *¿qué pasaría si?*, al modificar cualquiera de los parámetros del modelo que representa al negocio.

Las Bases de Datos, son también otra de las principales aplicaciones del Computador Personal en manos de un profesional o ejecutivo de empresa.

En ésta se pueden almacenar los antecedentes históricos, resumidos o clasificados, de una actividad particular. Una buena Base de Datos, nos permite luego jugar con estos antecedentes y empezar a conocerlos como nunca antes.

Las Bases de Datos actúan como veloces kárdexs electrónicos que entregan sus antecedentes en distintas clasificaciones y prioridades, como también corregirlas y actualizarlas con mucha rapidez. Sus listados de informes, nos permiten ordenar y clasificar esta importante información.

Los famosos diagramas de barras, como también los gráficos de tortas, tienen la virtud de entregar valiosa información en una fracción de segundos, la cual en forma de listados estadísticos necesita de varios minutos para ser analizada. Los programas graficadores de datos, han pasado a constituir parte importante del quehacer del Computador Personal.

Finalmente, una de las mayores aplicaciones que alcanzan los Computadores Personales hoy en día, es en el Procesamiento de Textos. Estos programas implican una gran economía de tiempo en la preparación de informes, cartas, artículos, publicaciones y todo tipo de documentos escritos.

El Computador ATARI 800 XL, asociado a una disquette ATARI 1050, representa hoy en día la mejor y más económica solución en materia de Computación Personal, ya que dispone de una serie de excelentes programas para este fin, como son: VISICALC como planilla electrónica, FILE MANAGER, SYNFILE y DATA PERFECT como Bases de Datos y ATARI WRITER como Procesador de Textos.

Y más aún, ya que hoy se encuentra disponible el SYNCALC, obra maestra en planillas electrónicas, similar al famoso LOTUS 1 2 3, con la gran diferencia, que este último cuesta más de cuatro veces el valor del SYNCALC y debe ser utilizado en un equipo con un costo que supera en más de 10 veces a una configuración con ATARI 800 XL.

CUESTA CREERLO ¿VERDAD?

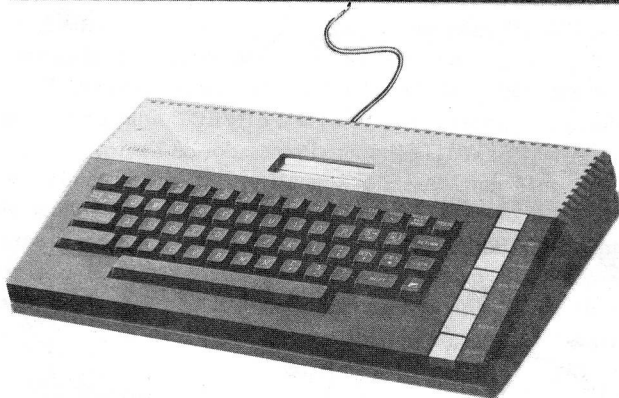
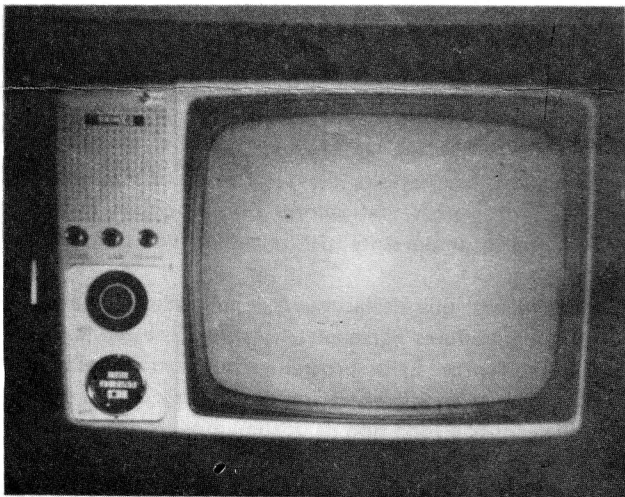
HARDWARE DEL MES

BT - 312 KM MONITOR 12" BLANCO Y NEGRO

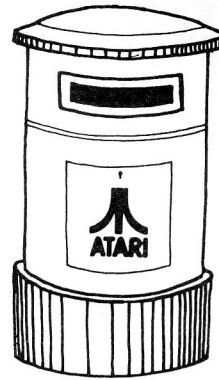
El computador ATARI 800 XL, dispone de una salida posterior para conectarle un monitor de video, a colores o blanco y negro, según necesidad.

El Monitor Blanco y Negro BT-312 KM de 12 pulgadas, es una excelente solución de bajo costo y alta nitidez de imagen, debido a la mayor resolución y estabilidad de la imagen presentada.

Para muchas aplicaciones profesionales, como procesamiento de textos, manejo de bases de datos, planillas electrónicas tipo VISICALC y otros, que no requieren el uso de las capacidades de colores del computador ATARI, la utilización del monitor de video BT-312 KM, representa la mejor solución al menor precio como pantalla.



Para una demostración del mismo, acérquese a su distribuidor ATARI más cercano.



BUZON ATARI

Señor Director:

Primero que nada, vayan mis felicitaciones para ustedes por su excelente Boletín Informativo ATARI.

Mi consulta particular, se refiere a cómo lograr construir la instrucción TAB(X) —que otros lenguajes BASIC disponen— para poder tabular, dado que el ATARI BASIC no la tiene, y muchas veces es necesario ubicar la información en algún lugar específico de la pantalla o impresora.

Esperando su ayuda al respecto, le saluda atentamente,

Gastón Rioseco V.
Antofagasta

R.D.

Muchas gracias por sus palabras de estímulo y nos sentimos satisfechos de poder atender las consultas de nuestros clientes.

Efectivamente, el lenguaje ATARI BASIC no dispone de la instrucción TAB(X) para tabular, como es el caso de ATARI MICROSOFT BASIC. Sin embargo, resulta muy simple implementar una solución al respecto, que cumple exactamente con la misma función.

```

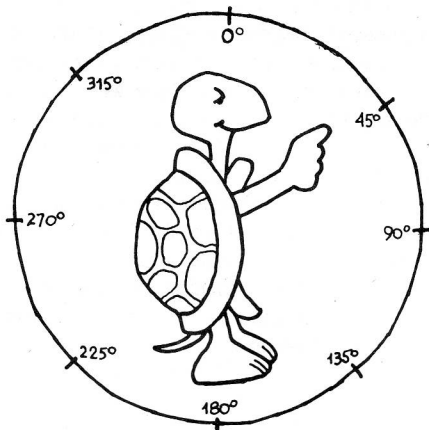
10 REM Implementación función TAB(X)
20 DIM TAB$(40)
22 REM llena con 40 blancos TAB$
30 TAB$(1) = " ": TAB$(40) = TAB$: TAB$(2) = TAB$
32 REM limpia pantalla
40 PRINT CHR$(125)
50 PRINT "INGRESE COORDENADA X, 2 > X < 30: ";
  INPUT X
60 PRINT "INGRESE NUMERO INICIAL, 1 > N < 20: ";
  INPUT N
62 REM el número 6 es la cantidad de la palabra NUMERO
70 PRINT : PRINT TAB$(1, X); "NUMERO"; TAB$(1, 20 - 6);
  "CUADRADO"
80 PRINT
90 FOR P = N TO 5 * N
94 REM STR$ = convierte números a letras
100 Q = LEN(STR$(P))
110 PRINT TAB$(1, X); P; TAB$(1, 20 - Q); P * P
120 NEXT P
130 END
  
```

SOFTWARE DEL MES

CONOCIENDO EL UNIVERSO desarrolla la comprensión acerca del medio que rodea al hombre, de la importancia de los avances de las ciencias para conocer el mundo. Cada unidad tiene test de evaluación de conocimientos.

La primera unidad sobre el Universo nos ubica en

CURSO LOGO EN ESPAÑOL



GRAMATICA DE LA TORTUGA

(continuación)

VARIABLES

La mejor manera de entender las variables de LOGO es imaginarlas como recipientes con nombres afuera y contenido dentro. Los dos puntos al comienzo de una palabra indican que su contenido debe estar disponible para el procedimiento. Si se tipea:

```
IM :JUAN
```

El LOGO busca un casillero denominado JUAN. Si encuentra uno, busca en su interior y realiza todo lo que se encuentra disponible para IM.

El procedimiento IM enseguida despliega el contenido (valor) de JUAN en la pantalla. Si no encuentra algo en su interior, imprime un mensaje de error:

```
JUAN SIN VALOR
```

Existen dos formas de poner cosas o valores dentro de estos casilleros. El primero ya está explicado y consiste en usar procedimientos con ENTRADAS.

El segundo es usando el comando HAZ.

```
HAZ "JUAN 25
```

```
IM :JUAN
```

```
25
```

El procedimiento HAZ necesita dos ENTRADAS: una palabra y un valor que puede ser una palabra, una lista o un número. Aquí se crea un casillero llamado JUAN y coloca 25 en su interior.

Tenga presente que un procedimiento no detiene su ejecución cuando llama a un subprocedimiento. Así, una variable local de un procedimiento puede ser usado por sus subprocedimientos.

EN QUE CONSISTE UNA LINEA LOGO

La definición de procedimientos consiste en líneas de instrucciones. Estas se llaman líneas LOGO debido a que pueden ser mucho más largas que una línea física de la pantalla, por ejemplo:

```
HAZ "VARIOSNOMBRES [ JOSE JUAN PEDRO  
MARIA → JUANA ROSA ]
```

La flecha (→) indica que la próxima línea de la pantalla es una continuación de la primera línea LOGO. Para conseguir líneas más largas, sólo es necesario seguir escribiendo sin presionar la tecla RETURN.

Resumiendo podemos decir que:

1. Cada vez que se use un procedimiento, asegúrese de saber:
 - a) Cuantas entradas tiene
 - b) Cuando es un comando u operación
2. La primera palabra de una línea LOGO debe ser siempre un comando.
3. Una operación siempre es una entrada para otro procedimiento.
4. Asegúrese de contabilizar cada entrada para procedimiento.
5. Cuando las entradas o un comando han sido contabilizados, el próximo procedimiento debe ser otro comando.

El siguiente ejemplo muestra líneas de LOGO más complejas:

```
CR COMENTARIO :ADJ  
IM SE [ TU ERES ] :ADJ  
IM SE [ YO SOY ] PLBR SU :ADJ "ISIMO  
FIN
```

```
COMENTARIO "DIVERTIDO  
TU ERES DIVERTIDO  
YO SOY DIVERTIDISIMO
```

LAS TORTUGAS MULTIPLES

El ATARI LOGO tiene cuatro tortugas para realizar tareas gráficas y se le pueden asignar tareas específicas a cada una mediante distintas instrucciones. Las tortugas se identifican por su nombre, que son 0, 1, 2 y 3. Cuando es la primera instrucción gráfica, es la tortuga 0 que aparece y realiza las tareas solicitadas.

Para mandar a otra tortuga se activa con la instrucción DILE enseguida del número que representa a la tortuga de interés. Luego, todas las instrucciones que se ordenen, serán realizadas por dicha tortuga. Ejemplo:

```
CR CUADRADO :LADO  
REPITE 4 [ AV :LADO DE 90 ]  
FIN
```


DILE 3 CUADRADO 25
AV 50
DE 45
CUADRADO 10

La tortuga 3 hará un cuadrado de 25 por lado, luego avanzará 50 pasitos, girará 45 grados a la derecha y hará un cuadrado más pequeño de 10 pasitos por lado.

Se pueden activar varias tortugas juntas a la vez con el comando DILE seguido de una lista de números representando las tortugas que se deseen activar.

Ejemplo:

DILE [0 1 2]
CUADRADO 28

Las tres tortugas en forma simultánea harán un cuadrado de 28 pasitos, partiendo de la posición en que se encuentren, que puede ser cualquier lugar de la pantalla o campo de acción.

Para que las instrucciones sean ejecutadas secuencialmente, por cada tortuga se utiliza la instrucción CADA seguida de una lista de instrucciones a ejecutar. Las tortugas que realizarán esta acción serán aquellas que previamente se han activado con la instrucción DILE.

Ejemplo:

CR PONE
LM DILE [0 1 2 3]
CADA [DE 90 * QT?]
FIN

Primer caso:

PONE
DILE [0 1 2 3] REPITE 4 [AV 50 DE 90]

Segundo caso:

PONE
CADA [REPITE 4 [AV 50 DE 90]]

En el primer caso las tortugas en forma simultánea harán un cuadrado.

En el segundo caso cada tortuga hará un cuadrado en forma secuencial, es decir, primero la tortuga 0 hará el cuadrado y las otras esperarán su turno, luego la 2 y así sucesivamente.

Existe otra forma de dirigirse a una tortuga para que realice una actividad. Es la instrucción PI que significa pide. Esta instrucción no afecta la situación de las tortugas activas. La instrucción PI va seguida de un número o lista de números que representan a las tortugas, y de una lista de instrucciones. Una vez que las tortugas llamadas con PI terminan su tarea, vuelven a su estatus anterior, es decir, las que estaban activas (con la instrucción DILE) siguen activas. Las que habiendo actuado bajo el comando PI y no estaban previamente activadas, luego de la ejecución de la lista de instrucciones, vuelven a quedar inactivas.

A su vez existe una instrucción que permite conocer las tortugas activas en un determinado momento. La instrucción QT? nos entrega un número o lista de números que representan a las tortugas previamente activadas con la instrucción DILE.

Ejemplo:

DILE 1 TG
DILE 2 TG
IM QT?
2
SL PONXY [-30 0]
PONR PI 1 [RUMBO?]
IM QT?
2

En la segunda línea del ejemplo (DILE 2 TG) queda activa sólo la tortuga 2.

En la quinta línea la tortuga 2 es colocada en la coordenada $X = -30$ e $Y = 0$.

En la sexta línea su cabeza apunta hacia abajo o rumbo 180.

En la séptima línea se le ordena a la tortuga activa que ponga el mismo rumbo de la tortuga 1.

Finalmente IM QT? nos indica que la tortuga activa sigue siendo la 2.



Las instrucciones empleadas en este artículo, corresponden al cartidge LOGO EN ESPAÑOL de la fotografía.



* programe su Atari con el profesor von Byte

En múltiples ocasiones, y especialmente con el manejo de archivos, se requiere de algún método de búsqueda de un registro específico, dentro de algún gran string que contiene todos los registros del archivo. Por ejemplo, supongamos que tenemos un archivo de información que contiene registros de clientes, con los siguientes datos:

Código del cliente	(3 bytes)	
Nombre del cliente	(25 bytes)	Largo total
Dirección del cliente	(25 bytes)	del registro: 78 bytes
Ciudad del cliente	(15 bytes)	
RUT del cliente	(10 bytes)	

Cada uno de estos registros se encuentra almacenado dentro de un gran string; uno a continuación de otro, conteniendo la información de cada uno de los clientes.

Cuando se desea ubicar un cliente en particular, digamos por el código, es necesario recorrer el string comparando el registro buscado con cada uno de los registros del archivo. Naturalmente, cuando el número de registros totales es elevado y esta operación se realiza en lenguaje BASIC, el tiempo requerido para encontrar el registro puede ser considerable, transformándose en un método ineficiente.

Justamente para solucionar este problema, en la columna de este mes, se entrega una subrutina en lenguaje de máquina para realizar esta operación, la que prácticamente transforma la búsqueda en un proceso instantáneo.

El listado adjunto, entrega un pequeño programa de ejemplo que hace uso de la subrutina de máquina ubicada entre las líneas 1000 a 1100.

A continuación se entrega una descripción del programa:

- Línea 10:** Dimensionamiento de las variables de string a utilizar. En STRING\$ se almacenarán los registros, en B9\$ se mantiene la rutina de máquina y REG\$ corresponde al registro que se desea buscar dentro de STRING\$.
- Línea 110:** Se va a la rutina de la línea 1000 donde se carga en B9\$ la subrutina de máquina para la búsqueda.
- Línea 120:** Se rellena de corazones el string total STRING\$.
- Línea 130:** Se asignan los valores del largo y número total de registros.
- Líneas 140-180:** Se ingresa información en el string REG\$ y se ubica ésta al azar dentro del string STRING\$.
- Líneas 190-320:** Se muestra en pantalla la ubicación donde queda almacenado REG\$ dentro del gran string STRING\$
- Línea 330:** Se pide ahora la información de REG\$ para proceder a la búsqueda dentro de STRING\$.
- Línea 340:** Se ejecuta la subrutina en lenguaje de máquina para realizar la búsqueda, entregando los parámetros indicados a la instrucción USR.
- Líneas 350-360:** Si como resultado la variable C=0, significa que el string REG\$ no se encuentra dentro de STRING\$.
- Líneas 370-450:** Si C=0, significará que lo encontró y lo despliega en pantalla. Luego consulta si desea continuar la búsqueda con otro string.
- Líneas 1000-1030:** Lee la subrutina de máquina y la almacena como caracteres en la variable B9\$.
- Líneas 1040-1100:** Datos correspondientes a la subrutina en lenguaje de máquina.

```

2 REM * Búsqueda rápida en STRING *
4 REM * BOLETIN CENTRO ATARI *
6 REM * MAYO 1985 *
8 REM *****
10 DIM STRING$(20000),B9$(200),REG$(50)
),RESP$(1)
20 REM STRING#=Largo total del string
30 REM B9#=Lenguaje Maquina /busqueda
40 REM REG#=Registro
50 REM RESP#=Usada para respuestas
58 REM *****
100 GRAPHICS 1+16
110 POSITION 3,10: ? #6;" INICIALIZANDO"
:GOSUB 1000
118 REM Llena de corazones STRING#
119 REM *****
120 STRING#="♥":STRING$(20000)=STRING#
:STRING$(2)=STRING#
126 REM LR=Largo Registro fijo <50>
127 REM TREG=Total Reg. en STRING#
128 REM LREG=Largo Registro Variable
129 REM *****
130 LR=50:TREG=INT(LEN(STRING#)/LR)
140 GRAPHICS 0: ? "Dígame información a
entrar en STRING":INPUT REG#
150 LREG=LEN(REG#):IF LREG=0 THEN GOTO
140
155 REM NREG=No Registro
156 REM PI=Posicion Inicial
157 REM PF=Posicion Final
158 REM Dentro de STRING#
159 REM *****
160 NREG=INT(RND(0)*TREG):PI=INT(NREG*
LREG+1):PF=PI+LREG-1
170 STRING$(PI,PF)=REG#
180 TRAP 2000
190 GRAPHICS 0

```

```

200 ? "Numero registro Maximo =";TREG
210 ? CHR$(34);REG#;CHR$(34)
220 ? "Posicion inicial en STRING# es
"
230 ? "STRING#(";PI;",";P9;")"
240 POKE 752,1
250 ? "DESEA VER REGISTRO EN STRING# S
/N ";;INPUT RESP#
260 IF RESP#<>"S" THEN GOTO 280
270 ? STRING#;FOR PAUSA=1 TO 500:NEXT
PAUSA
280 GRAPHICS 0
290 ? CHR$(34);REG#;CHR$(34)
300 ? "ENCONTRADO EN LAS SIGTES. POSICI
ONES"
310 ? "POS.INICIAL =";PI;" POS.FINAL =
";PF
320 ? "STRING#(";PI;",";PF;")"
330 ? "DEME INFORMACION A BUSCAR":POKE
752,0;INPUT REG#;LREG=LEN(REG#);C=0
340 C=USR(ADR(B9#),TREG,ADR(STRING#),A
DR(REG#),LR,LREG)
350 IF C=0 THEN POSITION 6,10: ? "INFOR
MACION NO ENCONTRADA":POSITION 2,12: ?
"PRESIONE RETURN PARA PROBAR OTRA VEZ"
360 IF C=0 THEN INPUT R#;REG#="" :GOTO
280
370 GRAPHICS 0
380 ? "LARGO DE STRING#=";LEN(STRING#)
390 ? "Cant.Reg. Maximo =";TREG
400 ? CHR$(34);REG#;CHR$(34);"Localiza
do como reg.# =";TREG-C
410 SP=(TREG-C)*LR+1
420 ? STRING$(SP,SP+LR-1)
430 ? "BUSCA MAS INFORMACION EN STRING
# S/N ";;INPUT RESP#;REG#=""
440 IF RESP#="S" THEN GOTO 330
450 GOTO 140
1000 I=1;SRCH=0
1010 READ SRCH:IF SRCH=-1 THEN RETURN
1020 POKE 708,RND(0)*250;POKE 709,RND(
0)*250
1030 B9$(I,I)=CHR$(SRCH):I=I+1:GOTO 10
10
1040 DATA 216,104,104,133,204,104,133,
203,104,133,209,104,133,208,104,133,21
5,104,133,214
1050 DATA 104,104,133,205,104,104,133,
206,169,0,133,212,169,0,133,213,162,0,
160,0
1060 DATA 177,214,224,0,208,2,132,216,
209,208,208,43,232,228,206,240,22,200,
196,205
1070 DATA 240,50,72,152,72,138,168,177
,214,133,207,104,168,104,165,207,24,14
4,219,72
1080 DATA 165,204,133,213,165,209,133,
212,104,162,0,224,0,240,17,224,0,240,6
,160
1090 DATA 0,177,214,162,0,164,216,200,
196,205,208,186,165,208,24,101,205,133
,208,144
1100 DATA 2,230,209,165,203,208,6,165,
204,240,7,198,204,198,203,24,144,156,9
6,-1
2000 GRAPHICS 0
2010 ? "REGISTRO NO ENCONTRADO"
2020 FOR PAUSA=1 TO 500:NEXT PAUSA
2030 TRAP 40000:GOTO 430

```

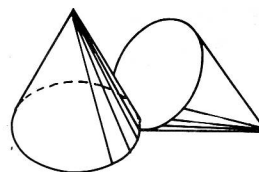
A partir del ejemplo dado, se puede utilizar la subrutina de máquina para realizar búsqueda de registros dentro de un gran string que conforma el archivo, de acuerdo a la aplicación particular que cada uno tenga.

Estudie cuidadosamente el ejemplo y transfórmelo de acuerdo a sus propias necesidades.

... TIPEANDO EN ATARI BASIC

Unos interesantes y atractivos efectos gráficos se pueden lograr con un computador ATARI, utilizando los modos gráficos especiales GRAPHICS 9, 10, 11, denominados modos GTIA.

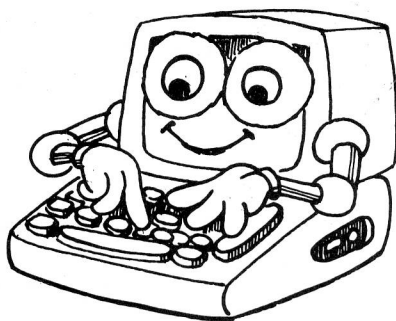
A continuación encontrará algunos pequeños ejemplos de estos modos:



```

2 REM CONOS EN MOVIMIENTOS
5 DEG
10 GRAPHICS 10
15 FOR I=0 TO 7:POKE 705+I,128+2:NEXT
I
17 POKE 705,136
20 FOR ANG=180 TO 360+180 STEP 6
30 X=8+8*COS(ANG)
40 Y=16+8*SIN(ANG)
50 COLOR (ANG-180)/45+1:PLOT X,Y
60 DRAWTO X,50+Y
70 COLOR 0:PLOT X,Y
90 NEXT ANG
120 FOR ANG=180 TO 360+180 STEP 6
130 X=26+8*COS(ANG)
140 Y=16+8*SIN(ANG)
150 COLOR 9-(ANG-180)/45:PLOT X,Y
160 DRAWTO X,50+Y
170 COLOR 0:PLOT X,Y
190 NEXT ANG
220 FOR ANG=180 TO 360+180 STEP 6
230 X=44+8*COS(ANG)
240 Y=16+8*SIN(ANG)
250 COLOR (ANG-180)/45+1:PLOT X,Y
260 DRAWTO X,50+Y
270 COLOR 0:PLOT X,Y
290 NEXT ANG
320 FOR ANG=180 TO 360+180 STEP 6
330 X=62+8*COS(ANG)
340 Y=16+8*SIN(ANG)
350 COLOR 9-(ANG-180)/45:PLOT X,Y
360 DRAWTO X,50+Y
370 COLOR 0:PLOT X,Y
390 NEXT ANG
410 GO TO 500
420 FOR ANG=180 TO 360+180 STEP 6
430 X=50+8*COS(ANG)
440 Y=16+8*SIN(ANG)
450 COLOR (ANG-180)/45+1:PLOT X,Y
460 DRAWTO X,50+Y
470 COLOR 0:PLOT X,Y
490 NEXT ANG
500 A=PEEK(705)
510 FOR I=705 TO 711
520 POKE I,PEEK(I+1)
530 NEXT I
540 POKE 712,A
550 GO TO 500

```



*aquí opinan los lectores de Atari

El aporte escogido para el presente mes, corresponde al señor Guillermo Rioseco Higuera, calle Los Angeles 958 de la ciudad de Los Angeles, quien nos hizo llegar el programa adjunto, para dibujar en modo gráfico 15.

El programa requiere sobre 16 KB de memoria libre para poder correr, sin embargo, si se reemplaza la línea 200 por modo gráfico 14 (GRAPHICS 14), puede ser utilizado con menor cantidad de memoria.

El programa de dibujo, permite mover el cursor que aparece en el centro de la pantalla, presionando las teclas de flechas: +, -, =, *. Para asignar el color al trazo del dibujo, se presionan las teclas A, B, C o D; en que A = COLOR 0 y D = COLOR 3.

Además, presionando simultáneamente las teclas CONTROL y Q, se pasa a las opciones para dibujar líneas rectas, círculos o rellenar áreas con color. Una vez finalizado el dibujo, éste puede ser grabado en cassette para uso posterior, presionando la tecla escape ESC.

Todos los pasos y acciones necesarias para utilizar el programa, son explicadas directamente por pantalla mediante mensajes.

Agradecemos y felicitamos al señor Rioseco por su programa de aporte y lo invitamos a escribirnos, para hacernos saber del premio en mercadería que desea obtener, equivalente a \$ 6.000 (seis mil pesos), a fin de poder despacharlo a su ciudad.

```

100 ? "M":POSITION 5,5:? " INSTRUCCION
ES DE USO+"
105 ? "El cursor se mueve presionando
las teclas con flechas -, +, =, *+"
110 ? "Se asigna color presionando las
teclas A-B-C-D; A=Color 0,B=1,C=2,D=3
+"
115 ? "Presionando CONTROL Q se puede;
trazar líneas, círculos y rellenar+"
120 ? "Presionando ESC Se Graba el
DIBUJO+"
125 ? "ESPERE UN MOMENTO":GOSUB 3000

```

```

130 ? "PRESIONE START PARA INICIAR"
135 IF PEEK(53279) <> 6 THEN 135
140 ? :? "INGRESE VALOR DEDE DIBUJADO=T
"
"
145 ? "Presione eleccion":OPEN #1,4,0,
"K:"
150 GET #1,A:IF A<67 AND A<84 THEN 1
50
155 IF A=67 THEN 2600
200 GRAPHICS 15:COLOR 1:PLOT 0,0:DRAWT
0 0,159:DRAWTO 159,159:DRAWTO 159,0:DR
AWTO 0,0
205 GOSUB 3100:PLOT 80,80
210 K=0:X1=159:X2=0:Y1=160:Y2=0
220 OP=2000:MC=2100:KOLOR=2200:LIMITES
=300
230 LINEA=400:CIRCULO=500:RELLENO=700
240 REM
250 GET #1,A
260 IF A=61 OR A=42 OR A=43 OR A=45 TH
EN GOSUB MC:GOTO 250
270 IF A=65, AND A=68 THEN GOSUB KOLO
R:GOTO 250
280 IF CHR$(A)="r" THEN GOSUB OP
285 IF A=27 THEN 2500
290 GOTO 250
300 REM LIMITES Y PLOT
310 C=PEEK(85):F=PEEK(84):COLOR 1:PLOT
C,F:FOR P=1 TO 5:NEXT P
320 COLOR K:PLOT PEEK(91),PEEK(90)
330 REM LIMITES
335 IF K=0 THEN 380
340 IF X1>C THEN X1=C
350 IF X2<C THEN X2=C
360 IF Y1>F THEN Y1=F
370 IF Y2<F THEN Y2=F
380 RETURN
400 REM LINEA
405 ? "Mueva el cursor hasta el extre
mo de la linea y presione CONTROL L"
410 L=K:K=0
420 COLOR L:L1=PEEK(91):L2=PEEK(90)
430 GET #1,A:IF CHR$(A)="r" THEN 450
440 C=PEEK(85):F=PEEK(84):GOSUB MC:GOT
O 430
450 COLOR L:PLOT L1,L2:DRAWTO C,F
460 K=L:GOSUB 3100:RETURN
500 REM CIRCULO
510 L=K:K=0
520 TRAP 520:? "Mueve el cursor hasta el centro
":INPUT R:TRAP 40000
530 ? :? "Mueva cursor hasta el centro
si esta correcto presione CONTROL C"
540 GET #1,A:IF A=61 OR A=42 OR A=43 O
R A=45 THEN GOSUB MC
550 IF CHR$(A)="r" THEN K=L:GOTO 570
560 GOTO 540
570 X=PEEK(91):Y=PEEK(90):COLOR K
575 C=X+C(0)*RX*.6:F=Y-S(0)*R:PLOT C,F
580 FOR I=0 TO 360 STEP 2
590 C=X+C(I)*RX*.6:F=Y-S(I)*R
600 DRAWTO C,F:GOSUB 330
610 NEXT I
620 GOSUB 3100:RETURN
700 REM RELLENA
702 IF K=0 THEN K=1
705 L=K:K=0:POKE 82,0:POKE 83,39
710 ? "Mueva el cursor hasta el punto
inferior derecho de el area a rellena
r y presione CONTROL R"
715 GOSUB 795:R1=PEEK(91):R2=PEEK(90)
720 ? "Mueva el cursor hasta el punto
superior derecho y presione CONTROL R
"
725 GOSUB 795:R3=PEEK(91):R4=PEEK(90)
730 ? "Mueva el cursor hasta el punto
superior izquierdo y presione CONTROL
R"
735 GOSUB 795:R5=PEEK(91):R6=PEEK(90)

```



```

740 ? "Mueva el cursor hasta el punto
inferior izquierdo y presione CONTROL
R"
745 GOSUB 795:R7=PEEK(91):R8=PEEK(90)
750 ? "K":K=L:COLOR K:PLOT R1,R2:DRAWTO
R3,R4:DRANTO R5,R6:POSITION R7,R8
755 POKE 785,K
760 XIO 18,#6,0,0,"S:"
765 POKE 82,2:POKE 83,39:GOSUB 3100:RE
TURN
795 GET #1,A:IF A=61 OR A=42 OR A=43 O
R A=45 THEN GOSUB MC
796 IF CHR$(A)="-" THEN RETURN
797 GOTO 795
2000 REM OPERACIONES
2005 ? "Mueve el cursor: CONTROL L = Linea:CONTROL C =
Circulo:CONTROL R = Relleno":? "Presi
one aleccion"
2010 GET #1,A
2020 IF CHR$(A)="|" THEN GOSUB LINEA:R
ETURN
2030 IF CHR$(A)="|" THEN GOSUB CIRCULO
:RETURN
2040 IF CHR$(A)="-" THEN GOSUB RELLENO
:RETURN
2050 GOTO 2000
*2100 REM MUEVE CURSOR*
2110 C=PEEK(85):F=PEEK(84)
2120 IF A=42 THEN POKE 85,PEEK(91)+1:P
OKE 84,F:GOSUB LIMITES:RETURN
2130 IF A=43 THEN POKE 85,PEEK(91)-1:P
OKE 84,F:GOSUB LIMITES:RETURN
2140 IF A=45 THEN POKE 84,F-1:POKE 85,
PEEK(91):GOSUB LIMITES:RETURN
2150 IF A=61 THEN POKE 84,F+1:POKE 85,
PEEK(91):GOSUB LIMITES:RETURN
2200 REM COLOR
2210 K=A-65:RETURN
2500 REM GRABA
2505 ? "Mueve la cinta y presione
CONTROL RECORD-PLAY-RETURN"
2510 OPEN #2,8,0,"C":GRAPHICS 15+32
2515 FOR FIL=Y1 TO Y2 STEP 2
2520 FOR COL=X1 TO X2
2530 LOCATE COL,FIL,W:IF W=0 THEN 2540
2535 PUT #2,COL:PUT #2,FIL:PUT #2,W
2540 LOCATE COL,FIL+1,W:IF W=0 THEN 25
50
2545 PUT #2,COL:PUT #2,FIL+1:PUT #2,W
2550 NEXT COL
2560 NEXT FIL
2565 CLOSE #2
2570 ? "Grabacion terminada sigue dib
ujando (S/N)":GET #1,A:IF A<>78 AND
A<>83 THEN 2570
2575 GRAPHICS 0:IF A=83 THEN RUN
2580 END
2600 REM INGRESA DATOS
2605 GRAPHICS 15
2610 ? "Mueve la cinta y presione
CONTROL PLAY RETURN"
2615 OPEN #2,4,0,"C:"
2620 TRAP 2640:GET #2,C:GET #2,F:GET #
2,W
2625 COLOR W:PLOT C,F
2630 GOTO 2620
2640 CLOSE #2
2645 END
3000 DIM C(360),S(360):DEG
3010 FOR N=0 TO 90:SOUND 0,N,10,12
3020 X=COS(N):Y=SIN(N)
3030 C(N)=X:S(N+90)=X:C(N+180)=-X:S(N+
270)=-X
3040 S(N)=Y:C(N+90)=-Y:S(N+180)=-Y:C(N
+270)=Y
3050 NEXT N:SOUND 0,0,0,0:RETURN
3100 ? "Mueve el cursor: Pres. -, =
, +, *, mueva el cursor: ESC=Graba di
bujo"
3110 ? "Mueve el cursor: Pres. -, =
, +, *, mueva el cursor: ESC=Graba di
bujo"
CONTROL Q=Operac. especiales":
RETURN

```

... TIPEANDO EN ATARI BASIC

(CONTINUACION)

Otros ejemplos interesantes de ver son los siguientes:

```

2 REM RAYOS LAZER
10 GRAPHICS 11
20 OLDX=0
30 OLDY=0
50 X=INT(RND(0)*80)
60 Y=INT(RND(0)*192)
70 C=INT(RND(0)*15+1)
75 COLOR C
77 PLOT OLDX,OLDY
80 DRAWTO X,Y
90 COLOR 0
100 PLOT OLDX,OLDY
110 DRAWTO X,Y
120 COLOR C
130 PLOT OLDX,OLDY
140 OLDX=X:OLDY=Y
150 FOR V9=1 TO 15:SOUND 1,200,8,V9:NE
XT V9:SOUND 1,0,0,0:GOTO 50

```

*

```

1 REM ABANICO
2 GRAPHICS 9:DEG
5 PLOT 40,0
20 FOR I=-90 TO 240
21 X=39+39*SIN(I)
30 COLOR INT((X-40)/2.5+16)
33 IF X<76 THEN 35
34 COLOR 0:GOTO 40
35 IF X>39 THEN COLOR 16-(X-37)/2.5
39 Y=120+60*COS(I)
40 DRAWTO X,Y
50 PLOT 40,0
60 NEXT I
70 FOR X=0 TO 15:SETCOLOR 4,X,0
80 FOR Y=1 TO 300:NEXT Y
90 NEXT X
100 GOTO 70
110 GOTO 110

```

*

```

1 REM HIPNOSYS
2 DIM A$(30)
3 A$="h u v t a e s o a i n o u l e i a a a a a a"
10 GRAPHICS 10
15 FOR I=1 TO 8:POKE 704+I,(I-1)*32+22
:NEXT I
20 Q=1
30 FOR Y=0 TO 191
40 COLOR Q
50 PLOT 0,Y
60 DRAWTO 79,191-Y
70 Q=Q+0.4166666666:IF Q>8 THEN Q=1
75 REM FOR T=1 TO 100:NEXT T
80 NEXT Y
120 Q=1
130 FOR X=79 TO 0 STEP -1
140 COLOR Q
150 PLOT X,0
160 DRAWTO 79-X,191
170 Q=Q+1:IF Q>8 THEN Q=1
180 NEXT X
190 REM COLOR 0:PLOT 0,0:DRAWTO 79,191
:PLOT 79,0:DRAWTO 0,191
200 X=USR(ADR(A$))
210 FOR J=1 TO 4:NEXT J
220 GO TO 200

```


ATARI[®]

EN LA EDUCACION

SAINT ROSE SCHOOL



Como comentábamos en el Boletín del mes pasado en esta misma columna, se darán a conocer las experiencias logradas a la fecha por los diferentes colegios poseedores de talleres computacionales ATARI.

Este mes expondremos el particular enfoque que le ha dado el colegio SAINT ROSE SCHOOL, como lo describe uno de sus directivos don José Miguel Oportus.

“A mediados de 1983, el Saint Rose School decide en conjunto con la Dirección del Colegio y el Centro de Padres, introducir la computación como un ramo obligatorio del Plan Anual para los cursos desde 5o. Básico a 3o. Medio. Luego de un acabado estudio de las diferentes alternativas del mercado de microcomputadores, se decidió, sobre sólidas bases, instalar un taller de computación con equipos ATARI, el que fue inaugurado en marzo de 1984. Desde esa fecha han pasado centenares de alumnos por este taller, los que han vivido la espectacular interacción que se produce entre el estudiante y el computador.

Al frente de estos cursos están los profesores Osmán Jofré y Marta Ulloa. La idea básica de esta metodología es familiarizar al estudiante con la máquina, de tal modo que utilizando su natural creatividad, puede llegar después de un tiempo, que hemos estimado en 3 años como mínimo, a un importante grado de capacitación para programar sus propias inquietudes.

El Colegio tiene un proyecto que a la fecha no se ha materializado que consiste en que todos los estamentos participen de las ventajas de contar con un taller en su propio local,

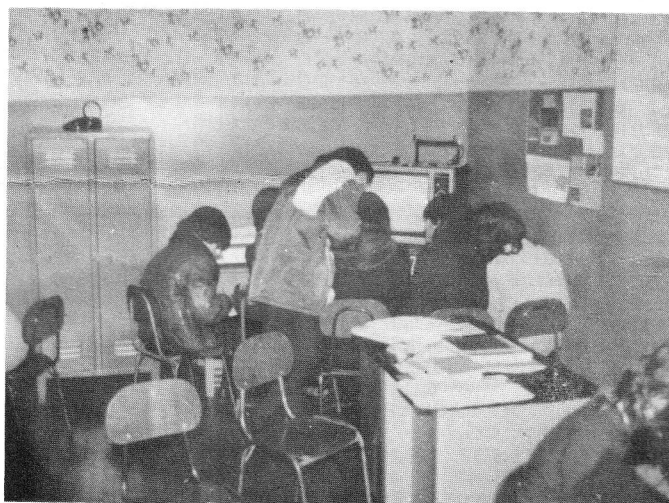
esto es, intentar llegar a todos los alumnos, todos los profesores y en lo posible, a un importante número de apoderados que se interesen por la práctica de esta tecnología.

Como proyecto más ambicioso, el Saint Rose School pretende instalar un nuevo taller enfocado exclusivamente a los alumnos del Primer Ciclo Básico, orientado al lenguaje LOGO, que es un sistema altamente motivador como para introducir al estudiante en este fascinante campo.

El Saint Rose School está seguro que la inclusión de la computación en todos los ciclos de la educación, ayudará y motivará la creatividad del alumnado, así como también servirá para ayudar en el proceso educativo propiamente tal, como es la nivelación del alumno, ensayos de pruebas, etc.

A partir de este año 1985, el Saint Rose School, utiliza la computación en su Departamento Administrativo llevando el control de matrículas y todo el sistema de evaluación del alumno.

En el aspecto programático de la enseñanza de la computación, los alumnos tienen, semanalmente, una hora de clases sistemáticas y 2 horas de taller controlado y la posibilidad, si el alumno así lo desea, taller libre por las tardes.”



Las instalaciones del Colegio Saint Rose School se encuentran ubicadas en Av. Manuel Antonio Matta 393, Santiago.

OFERTA ESPECIAL

35 % DESCUENTO

ANTES \$ 25.286

AHORA \$ 16.500



ATP - 1064

EXPANSION DE MEMORIA DE 64 KB

Una muy buena noticia para todos los poseedores de computadores ATARI 600XL.

Como una oferta especial del CENTRO ATARI de Santiago, se está ofreciendo al público el módulo de expansión de memoria ATO-1064, que incrementa la memoria del computador 600 XL de 16 KB al máximo de 64 KB de RAM, a un precio de oferta de promoción de \$ 16.500 con IVA incluido. Esta oferta se mantendrá vigente sólo hasta agotar el stock.

A fin de satisfacer tanto a los clientes de Santiago como de provincias, se ha implementado un sistema de atención por correo para la compra y despacho de la mercadería. Para este efecto, se suministrará el cupón inferior, el cual deberá ser remitido a **CENTRO ATARI SANTIAGO, Andrés de Fuenzalida No. 79, Providencia, Santiago**, adjuntando: vale vista o cheque cruzado a nombre de COELSA S.A.

Una vez recibida esta documentación se procederá al despacho de la mercadería. Agradeceremos considerar un tiempo de espera entre 5 y 10 días para la recepción de la misma.

ATARI®

CUPON
ADQUISICION MODULO
EXPANSION MEMORIA
ATP - 1064

nombre cliente _____

fono _____

factura a nombre de _____

RUT _____

domicilio para despacho _____

ciudad _____

Adjunto:

Cheque

Vale vista No. _____ del Banco _____

Por la suma de \$ 16.500 correspondiente a la compra de un módulo de expansión ATARI 1064, a nombre de COELSA S.A.

NOTA:
 Los datos del cupón deben ser llenados con letra de imprenta o a máquina.

día	mes	año

_____ firma

GUIA PARA TIPEAR PROGRAMAS

Antes de digitar cualquier programa, usted se debe familiarizar con su computador. Aprenda a usar el teclado para digitar y corregir programas BASIC. Lea el manual del equipo para entender como grabar y cargar sus programas BASIC hacia y desde disquettes o cassette. Para ayudarle con la tarea de digitación de los programas aparecidos en el Boletín Centro ATARI, éstos vienen impresos tal como aparecen en la pantalla de su televisor, incluyendo los caracteres gráficos y de control que el programa tenga incorporados.

A continuación se entrega un listado de los diferentes caracteres o símbolos que pueden aparecer en los listados, junto a la secuencia que debe presionarse para lograrlos.

símbolo teclas a presionar

▼	CTRL .
↑	CTRL A
	CTRL B
	CTRL C
	CTRL D
	CTRL E
↘	CTRL F
↙	CTRL G
▲	CTRL H
■	CTRL I
■	CTRL J
■	CTRL K
■	CTRL L
■	CTRL M
■	CTRL N
■	CTRL O
■	CTRL P
■	CTRL Q
■	CTRL R
■	CTRL S
■	CTRL T
■	CTRL U
■	CTRL V
■	CTRL W
■	CTRL X
■	CTRL Y
■	CTRL Z
ESC	ESC ESC
↑	ESC CTRL ↑
↓	ESC CTRL ↓
←	ESC CTRL ←
◆	CTRL .
◆	CTRL :
↵	ESC SHIFT CLEAR
␣	ESC BACK SPACE
␣	ESC TAB
◻	INVERSO CTRL

símbolo teclas a presionar

◻	INVERSO CTRL A
◻	INVERSO CTRL B
◻	INVERSO CTRL C
◻	INVERSO CTRL D
◻	INVERSO CTRL E
◻	INVERSO CTRL F
◻	INVERSO CTRL G
◻	INVERSO CTRL H
◻	INVERSO CTRL I
◻	INVERSO CTRL J
◻	INVERSO CTRL K
◻	INVERSO CTRL L
◻	INVERSO CTRL M
◻	INVERSO CTRL N
◻	INVERSO CTRL O
◻	INVERSO CTRL P
◻	INVERSO CTRL Q
◻	INVERSO CTRL R
◻	INVERSO CTRL S
◻	INVERSO CTRL T
◻	INVERSO CTRL U
◻	INVERSO CTRL V
◻	INVERSO CTRL W
◻	INVERSO CTRL X
◻	INVERSO CTRL Y
◻	INVERSO CTRL Z
◻	ESC SHIFT DELETE
◻	ESC SHIFT INSERT
◻	ESC SHIFT TAB (SET)
◻	ESC SHIFT TAB (SET)
◻	INVERSO BARRA ESPACIO
◻	INVERSO SHIFT -
◻	INVERSO CTRL .
◻	INVERSO SHIFT =
◻	ESC CTRL 2
◻	ESC CTRL BACK SPACE
◻	ESC CTRL INSERT

SR. (A) :

№ 90467

FERNANDEZ TORRES ENRIQUE

CATEDRAL 1029 OF 703

SANTIAGO

291

**BOLETIN
INFORMATIVO
CENTRO ATARI®**

ANDRES DE FUENZALIDA 79
PROVIDENCIA, SANTIAGO

CONDUCCION GRATUITA