

UNIVERSIDAD AUTONOMA DE NUEVO LEON

FACULTAD DE CIENCIAS FISICO MATEMATICAS



EDITOR

T E S I S

QUE PARA OBTENER EL TITULO DE:  
LICENCIATURA EN CIENCIAS COMPUTACIONALES

PRESENTA  
JOSE CARLOS RODRIGUEZ REYNA

MONTERREY, N. L.

NOVIEMBRE DE 1987



TL  
QA 76  
.76  
.T49  
R64  
1987  
c.1

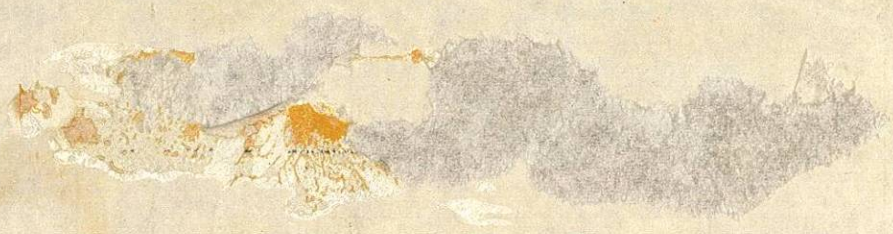
35

C-1





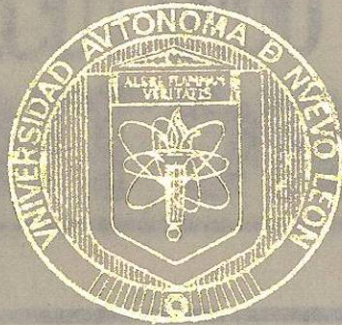
1080171541





UNIVERSIDAD AUTONOMA DE NUEVO LEON

FACULTAD DE CIENCIAS FISICO MATEMATICAS



EDITOR

TESIS

QUE PARA OBTENER EL TITULO DE:  
LICENCIATURA EN CIENCIAS COMPUTACIONALES

PRESENTA  
JOSE CARLOS RODRIGUEZ REYNA

MONTERREY, N. L.

NOVIEMBRE DE 1987



---

**EJECUTANDO  
EDITOR**

---

---

---

# CONTENIDO

---

**Reconocimiento** **I**

**Introduccion** **II**

Que es un editor,II  
Sobre este manual,II

---

**1 Creando y Editando Un Archivo De Texto** **1**

Preparandonos para el ejemplo,1  
Creando un nuevo archivo de texto,2  
Entrada de lineas,4  
Desplegando lineas,5  
Agregando lineas,6  
Terminando una sesion de edicion,8  
Imprimiendo tu archivo,8  
Sumario,9

**2 Usando Editor Para Revisar Un Archivo** **10**

Preparandonos para el ejemplo,10  
Revisando tu archivo,11  
Reemplazando un string por otro,12  
Borrando lineas,13  
Cancelando una sesion de edicion,14  
Archivo de respaldo,15  
Recuperando lineas,16  
Moviendo lineas,18  
Funcion ayuda,20  
Sumario,21

**3 Conclusiones** **22**

---

# RECONOCIMIENTO

---

A Dios quien me dió la suficiente fuerza, capacidad y paciencia en todos los pasos que he dado, los que doy y los que si me lo permite seguire dando.

A mis padres por el apoyo recibido en todo momento, además de ser la motivación para seguir siempre adelante.

A mis abuelos quienes me apoyaban en el momento preciso y a quienes quiero mucho.

A mis tios, en especial a mi tío Chema quien me apoyo en los momentos que se lo pedía.

A la familia Jiménez Cruz quien me dió un apoyo muy grande y que les estaré agradecido por siempre.

A mi novia Reyna quien me ha apoyado constantemente en todo momento lo cual estaré por siempre agradecido.

A mis amigos con quienes pasé los momentos mas agradables en toda la carrera.

A mis amigos Pedro, Chano y Sergio que me apoyaron cuando más lo necesitaba y a quienes considero mis hermanos.

A los padres de mis amigos quienes nos aguantaban cuando nos reuniamos a estudiar.

A mis maestros quienes además de ser mis amigos siempre ponían lo mejor de ellos para ayudarnos.

A mi asesor Lic. Fernando Torres C. quien me apoyó en forma desinteresada en este proyecto y del cual como maestro siempre agradeceré su apoyo.

A mis compañeros, jefes, amigos DeRCA con quienes he aprendido mucho y tenido un constante apoyo.

A todos ellos G R A C I A S por todo.



---

# INTRODUCCION

---

## QUE ES UN EDITOR

Un programa de computadora diseñado para ejecutar funciones de reordenamiento, modificaciones y eliminación de datos de acuerdo con las reglas establecidas.

## SOBRE ESTE MANUAL

Este manual describe y explica el uso de el **Editor**. Además de introducir al usuario debutante de una manera sencilla y clara a el manejo de edición de archivos y/o programas.

El capítulo 1 proporciona la información básica sobre como empezar a trabajar con el **Editor**. En el capítulo 2 se da la información necesaria para que puedas revisar tu archivo y hacer las modificaciones que quieras.

Una sección final, CONCLUSIONES, se expresan las conclusiones y se habla de como se fue formando este proyecto hasta llegar a el producto final el **Editor**.

Con el **Editor** tú podras editar archivos de datos y programas, ya sea BASIC, COBOL, PASCAL, etc. esto a nivel Microcomputadora y ejecutar estos programas con su respectivo compilador.



---

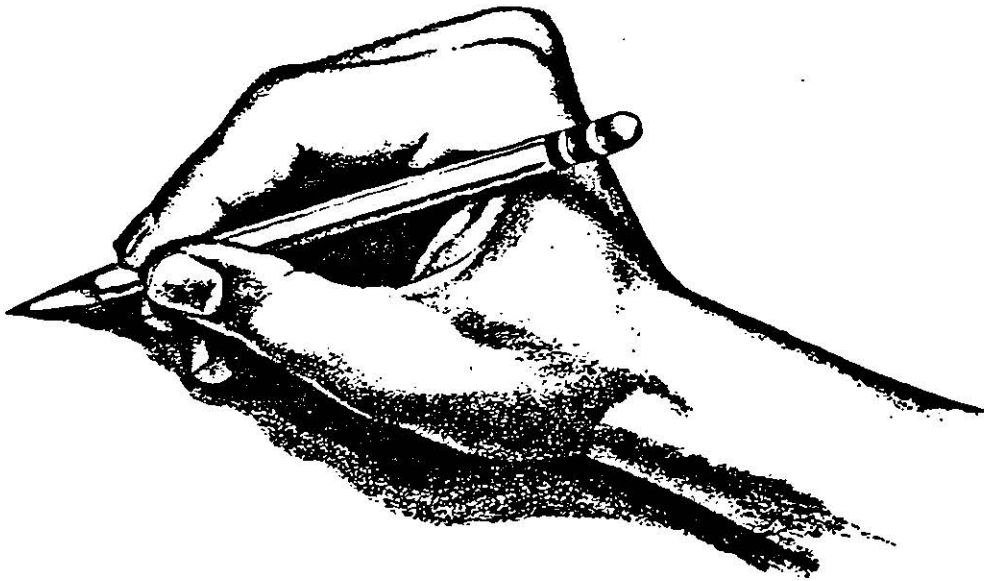
**CAPITULO**

**1**

---

**CREANDO Y EDITANDO  
UN ARCHIVO DE TEXTO**

---





Cuando es necesario escribir una nota o un documento corto y tenemos una microcomputadora es realmente facil utilizar el programa **Editor**

Ademas puedes capturar un programa en cualquier lenguaje (Basic, Cobol, Turbopascal, etc) y ejecutarlo con su respectivo compilador.

En el ejemplo que veremos en este manual presenta un programa sencillo que se asimila a los que son encargados en la escuela, etc.

Despuès de practicar con este ejemplo, tendràs la habilidad para manejar el **Editor**, para lo que tu lo requieras en lo que sea necesario.

## PREPARANDONOS PARA EL EJEMPLO

Para este ejemplo necesitas tener un diskette formateado, si tienes 2 drives coloca el programa **Editor** en el drive A y tu diskette formateado en el B.

Si tienes disco duro y deseas instalar el programa en C entonces teclea:

```
A>COPY A:EDITOR.COM C:
```

Con esto dejamos lista la preparaci3n para el ejemplo





## CREANDO UN NUEVO ARCHIVO DE TEXTO

### **EDITOR<RETURN>**

Cuando tu teclees **Editor** , enseguida deberás teclear RETURN, como respuesta el sistema copia en memoria el programa **Editor** y empieza a ejecutarlo, el **Editor** te pedirá entonces el archivo que quieres editar de la siguiente manera:

Archivo:

En este momento deberas teclear el nombre de archivo que tu quieras.

Archivo:<drive:><nombre de archivo>[.extensión]<RETURN>

Ahora el **Editor** verifica si el archivo que tecleaste existe en tu diskette. Si el archivo existe copia el archivo en la memoria y espera a que teclees los comandos que requieras. Si el archivo no existe, el **Editor** asume que quieres crear uno nuevo y espera a que teclees un comando.

Para empezar este ejemplo crearemos un archivo llamado ARCH.PAS en el drive A. Tecleando lo siguiente.

A>EDITOR<RETURN>

El **Editor** contestará de la siguiente forma:

Archivo:ARCH.PAS<RETURN>



El **Editor** te dira que ARCH.PAS es un archivo nuevo y espera por un comando:

Para terminar teclea  
EXIT

Archivo nuevo  
\*

El (\*) de el **Editor** indica que esta listo para aceptar un comando.

Como el archivo es nuevo lo único que puedes hacer es insertar líneas, los comandos del **Editor** son letras sencillas, en este caso lo que necesitas es teclear I (para insertar):

\*I<RETURN>

El responde con:

\*I  
Listo

Después de esto el **Editor** acepta cada línea que teclees para agregar al archivo, tu le diras al **Editor** el fin de cada línea al teclear <RETURN>.Cada vez que teclees <RETURN>, **Editor** mueve al principio de la siguiente línea y le asigna el número de línea.Cuando quieras terminar de insertar líneas teclea EXIT.



## ENTRADA DE LINEAS

Si hiciste algún error en la línea puedes corregir con el backspace (<--> antes de teclear <RETURN>.

Pero no te preocupes si tecléas <RETURN> en una línea con errores; veras que es muy sencillo corregir esto.

Teclea las siguientes líneas, pero no el número de línea; recuerda que el número de línea es asignado por el **Editor**. Para terminar debes de teclear EXIT y <RETURN>.

```
Program A;  
VAR Z:integer;  
Begin  
read(z);  
write(z);  
end;  
( Este es un Programa  
  de Prueba )  
EXIT<RETURN>  
*
```

El **Editor** indica que no puedes insertar otra línea y mueve el cursor a la siguiente línea y despliega (\*), con el cual te indica que espera otro comando.



## DESPLEGANDO LINEAS

**< rango >**

Estas son algunas razones por las que tu quedras desplegar tu archivo:

- . Ver si tienes incluido todo lo que debes tener
- . Verificar que tienes
- . Desplegar tu archivo

El comando LISTAR (L) despliega una linea, un rango de lineas o todo tu archivo si asi lo deseas. Si tu tecleas T, el **Editor** despliega todo tu archivo empezando desde la linea 1. Si tecleas L seguido de un número de linea y una (,) el **Editor** despliega el número de linea si existe. Si tecleas L seguido de un rango de numeros de linea, el desplegará el rango que tu especificaste. Tecleas:

\$T<RETURN>

El **Editor** responde con:

```
1 Program A;  
2 VAR Z:integer;  
3 Begin  
4 read(z);  
5 write(z);  
6 end.  
7 ( Este es un Programa  
8 de Prueba )  
$
```





Tu puedes desplegar una linea especifica ya sea para modificarla o verificarla unicamente. Teclea:

```
*L 5<RETURN>
```

El **Editor** responde con:

```
5 write(z);  
*
```

Tambien puedes desplegar un rango de lineas separando el rango con una (,). Teclea:

```
*L 3,5<RETURN>
```

El **Editor** responde con:

```
3 Begin  
4 read(z);  
5 write(z);  
*
```

## AGREGANDO LINEAS AL ARCHIVO

### **(linea)**

Tu puedes insertar en cualquier lugar de tu archivo utilizando el comando Insertar con un número de linea.

El **Editor** inserta la nueva linea o lineas después de el número de linea que tu especificaste, y automáticamente renumera las lineas.

Por ejemplo, insertar unos comentarios despues de la  
linea 1. Teclea:

```
*I 1<RETURN>
```

El **Editor** responde con:

```
*I 1  
Listo  
< Nombre: Jose Carlos Rdz. R. >  
Quieres insertar otra linea (S/N) ==>S  
< Grupo: 7-B >  
Quieres insertar otra linea (S/N) ==>N  
*
```

Para insertar lineas al final de el documento, usando  
el comando Insertar, es necesario especificar un número  
grande de linea (si no sabes cuál es el último número de  
linea) y el **Editor** asume que es la ultima linea de tu  
archivo, colocando la linea o lineas que insertaste después  
de la última linea.

Ahora despliega el archivo para ver las inserciones  
teclea:

```
*T<RETURN>
```

El **Editor** te muestra:

```
1 Program A;  
2 < Nombre: Jose Carlos Rdz. R. >  
3 < Grupo: 7-B >  
4 VAR Z:integer;  
5 Begin  
6 read(z);  
7 write(z);  
8 end;  
9 < Este es un Programa  
10 de Prueba >  
*
```

Nota que el **Editor** renumeró todas las lineas.



## TERMINANDO UNA SESION DE EDICIÓN

Cuando quieres terminar una sesión de edición teclea S (comando del **Editor** para terminar una sesión de edición), el **Editor** almacena tu archivo en tu diskette y regresa a tu DOS. Después de que tu archivo ha sido almacenado, DOS despliega (A>). Teclea el comando de fin de sesión:

```
#S<RETURN>
```

El **Editor** responde con:

```
#S  
Quieres salvar ? ARCH.PAS (S/N) ==>S  
Salvando ARCH.PAS
```

```
8 líneas salvadas  
A>
```

## IMPRIMIENDO TU ARCHIVO

Tu archivo almacenado en tu diskette no te es útil si no lo puedes imprimir, el DOS hace fácil imprimir tu archivo. Primero, verifica que la impresora este lista. En seguida utiliza el comando del DOS (Type) para mandar tu archivo a la impresora. Teclea lo siguiente:

```
A>Type ARCH.PAS <Ctrl P> <RETURN>
```

Utilizando el **Editor** tu puedes crear e imprimir un archivo en pocos minutos.





## SUMARIO

Así termina la primera sesión con el **Editor**. El siguiente capítulo muestra cómo puedes revisar un archivo y describe algunos otros comandos adicionales.

Si tú no vas a continuar con el siguiente capítulo en este momento no olvides salvar ARCH.PAS en tu diskette.

---

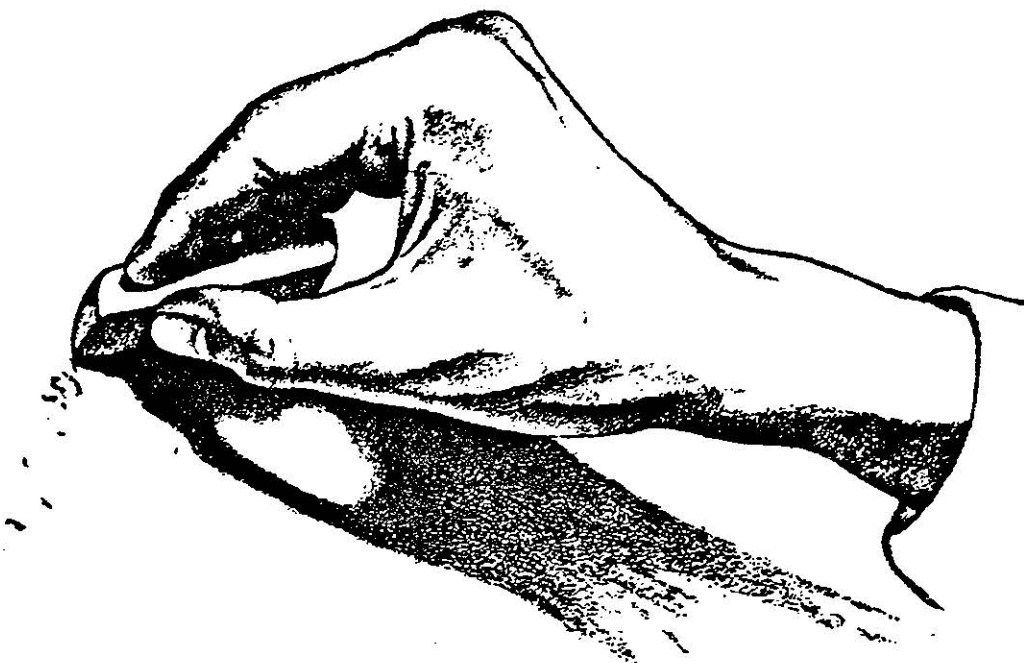
**CAPTULO**

**2**

---

**USANDO EDITOR PARA  
REVISAR UN ARCHIVO**

---



En el capítulo anterior vimos como tú puedes crear un archivo de texto.

El **Editor** incluye comandos adicionales para que tu edites un archivo rápidamente .

Este capítulo muestra como puedes revisar un archivo de una manera muy sencilla y el manejo de otros comandos de el **Editor**

## PREPARANDONOS PARA EL EJEMPLO

Si estas continuando directamente del capítulo 1, no es necesaria la preparación. Así que nos vamos hasta el encabezado que dice "REVISANDO TU ARCHIVO".

Si vas a empezar una nueva sesion, necesitas el archivo que creaste en el capítulo 1.

Coloca el diskette que contiene el archivo ARCH.PAS en el drive B y teclea lo siguiente para editar ARCH.PAS:



A>EDITOR<RETURN>

El **Editor** responde con:

Archivo:

teclea entonces

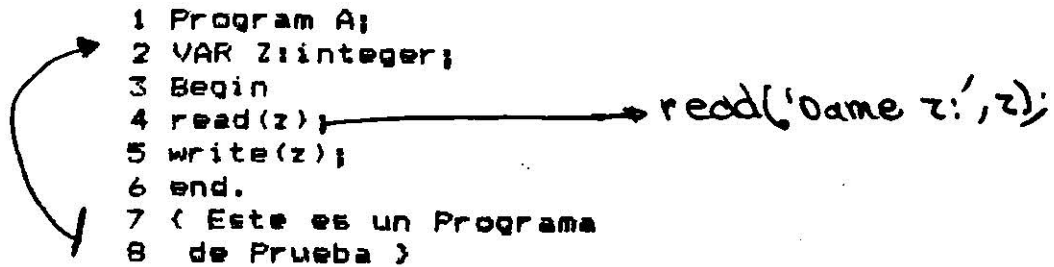
Archivo:ARCH.PAS<RETURN>

y el **Editor** responde con:

```
1 Program A;  
2 VAR Z:integer;  
3 Begin  
4 read(z);  
5 write(z);  
6 end.  
7 ( Este es un Programa  
8 de Prueba )  
*
```

## REVISANDO TU ARCHIVO

Asumiendo que ya terminaste tu archivo no significa que lo tengas que mandar salvar en tu diskette, ya que puedes revisarlo en la pantalla. Y decidir si hay que agregar líneas o efectuar cambios que creas conveniente. La Figura 2.1 muestra una copia de el archivo ARCH.PAS marcado con los cambios que debes de hacer.



```
1 Program A;  
2 VAR Z:integer;  
3 Begin  
4 read(z);  
5 write(z);  
6 end.  
7 ( Este es un Programa  
8 de Prueba )
```

Figura 2.1 ARCH.PAS marcado con cambios

## REEMPLAZANDO UN STRING POR OTRO

**A n/m**

En el ejemplo ARCH.PAS y en base a la Figuar 2.1 queremos modificar una linea (4 read(z);) agregandole una leyenda antes de la variable. Tu puedes cambiar todas las lineas que quieras con el comando MODIFICA (M), para usar este comando es necesario posecionarse en la linea que quieres efectuar el cambio con el comando LISTA (L),  
Teclea:

\*L 4,

E) **Editor** responde con:

```
4 read(z);  
*M z/'Dame z: 'z
```

E) **Editor** responde con:

```
4 read('Dame z: 'z);  
*
```

## BORRANDO LINEAS

### B Borrando

Tu puedes utilizar el **Editor** para borrar una línea tecleando el comando BORRAR (B) seguido de un número de línea.

Por ejemplo supongamos que decides que la línea 3 no la quieres incluir en tu archivo. Para borrar la línea, Teclea:

```
*B 3,
```

El **Editor** responde con:

```
3 Begin  
Quieres borrar la línea 3 (S/N) ==>S  
*L 2,4  
2 VAR Z:integer;  
3  
4 read(z);  
*
```

Para borrar un rango de líneas, se utiliza el mismo comando BORRA (B) con el número de la primera y última línea a ser borrada, separadas por una (,).

Por ejemplo para borrar de la línea 4(4 read(z);) hasta la 5(5 write(z);) en tu archivo.





Tecllea:

\*B 4,5

El **Editor** responde con:

4 read(z);

5 write(z);

Quieres borrar desde la linea 4 hasta la 5 (S/N) ==>S

\*L 4,6

4

5

6 end.

\*

## CANCELANDO UNA SESION DE EDICIÓN

Supongamos que tu hiciste cambios al archivo pero te das cuenta que realmente no quieres estos cabios en este momento por lo que quieres dejar tu archivo tal como estaba antes de esta sesión. Tu puedes cancelar la sesión de edición con el comando QUIT (Q) y regresar al DOS. Así los cambios que que hiciste no serán almacenados en tu diskette si tu no lo deseas.

Esta cancelación no se hace inmediatamente ya que el **Editor** te pregunta si quieres salvar tu archivo, si tecleas N, el **Editor** regresa al DOS sin tomar en cuenta las modificaciones hechas a tu archivo. Pero si tecleas S el **Editor** salva tu archivo en el diskette con las modificaciones que hiciste.

Por ejemplo cancelando la sesión de edición.

Tecllea:

```
*0<RETURN>  
Quieres salvar ? ARCH.PAS ==>N  
Nos vemos  
A>
```

Ahora edita ARCH.PAS nuevamente:

```
A>EDITOR<RETURN>
```

```
Archivo:B:ARCH.PAS
```

El **Editor** responde con:

```
1 Program A;  
2 VAR Z:integer;  
3 Begin  
4 read(z);  
5 write(z);  
6 end.  
7 ( Ete es un Programa  
8 de Prueba )  
*
```

Como puedes ver en tu archivo no se hicieron los cambios que hiciste.

## ARCHIVO DE RESPALDO

Quando tu editas un archivo que ya existe, el **Editor**, trabaja con una copia que es colocada en la memoria de tu computadora, de esta forma la version original queda almacenada en tu diskette.



Cuando tu terminas una sesión de edición, el **Editor** cambia la extensión de tu versión original, a BAK (para respaldo) y almacena la nueva versión modificada en tu diskette además de tu versión BAK.

Si existe en tu diskette un archivo con extensión BAK, el **Editor** borra la versión anterior al archivo BAK. Si tu necesitas la versión original puedes usar el archivo BAK.

Tu necesitas cambiar la extensión de tu archivo de cualquier manera, porque el **Editor** no te editará si el archivo tiene extensión BAK.

Quizá, por ejemplo, deberas usar el comando RENAME del DOS para cambiar ARCH.BAK por ARCH.OLD.

## RECUPERANDO LINEAS

### R (Recup)

Cuando tu borras una línea y después te das cuenta que realmente esa línea era necesaria, por lo regular tu tendrias que teclear esa línea nuevamente. El **Editor** te permite recuperar esta línea con el comando RECUPERA (R), siempre y cuando esta línea haya sido la última que borraste.

Teclea:

\*B 2,

El **Editor** responde con:

```
2 VAR Z:integer;
Quieres borrar la linea 2 (S/N) ==>S
*L 1,3
1 Program A;
2
3 Begin
*R<RETURN>
```

El **Editor** responde con:

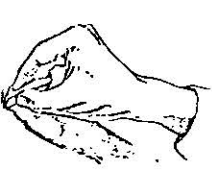
```
*
*L 1,3
1 Program A;
2 VAR Z:integer;
3 Begin
*
```

Si te fijas la linea a sido recuperada.

Lo mismo pasa cuando borramos un rango de lineas y luego queremos recuperar este rango, se utiliza el mismo comando RECUPERA (R). Por ejemplo teclea:

\*B 2,3

El **Editor** responde con:



```

2 VAR Z:integer;
3 Begin
Quieres borrar desde la linea 2 hasta la 3 (S/N) ==>S
*L 1,3
1 Program A;
2
3
*R<RETURN>

```

El **Editor** responde con:

```

*
*L 1,3
1 Program A;
2 VAR Z:integer;
3 Begin
*

```

Si te fijas el rango a sido recuperado.

## MOVIENDO LINEAS

Tu puedes mover una linea y colocarla en otro lugar de tu archivo, tambien con el comando RECUPERA (R n), lo primero que tienes que hacer es borrar el rango que deseas mover, ya que este borrado lo unico que tienes que hacer es utilizar el comando RECUPERA. Por ejemplo si queremos mover la linea 3. Tecllea:

```
*B 3,
```

El **Editor** responde con:

```

3 Begin
Quieres borrar la linea 3 (S/N) ==>S
*R 4<RETURN>

```



El **Editor** responde con:

```
*  
*L 1,6  
1 Program A;  
2 VAR Z:integer;  
3 read(z);  
4 write(z);  
5 Begin  
6 end.  
*
```

De esta forma hemos movido una linea, pero que pasa si quieres mover un rango, pues muy sencillo ya que es de la misma forma. Por ejemplo si queremos mover la linea 7 (Este es un Programa) a la B ( de Prueba ).

Tecllea:

```
*B 7,8
```

El **Editor** responde con:

```
7 ( Este es un Programa  
8 de Prueba )  
Quieres borrar desde la linea 7 hasta la B (S/N) ==>S  
*R 1<RETURN>
```

El **Editor** responde con:

```
*  
*L 1,8  
1 Program A;  
2 ( Este es un Programa  
3 de Prueba )  
4 VAR Z:integer;  
5 Begin  
6 read(z);  
7 write(z);  
8 end.
```

En ambos casos el **Editor** renumera las lineas.



## FUNCION AYUDA

Cuando sea la primera vez que utilizas el **Editor** en un principio tendrás que acostumbrarte a la estructura de los comandos que el **Editor** tiene por lo cual incluye en estos el comando AYUDA (Help).

Cuando lo necesites solamente necesitas teclear:

A<RETURN>

El **Editor** responde con:

### COMANDOS\_DEL\_EDITOR

L n,	Lista la linea n
L n,m	Lista de la linea n a la m
B n,	Borra la linea n
B n,m	Borra de la linea n a la m
M n/m	Modifica el string n por el string m
S	Salva el archivo
Q	Sale del EDITOR sin hacer modificaciones al archivo
R	Recupera o Mueve la ultimas lineas borradas
T	Lista todo el archivo
A	Ayuda ( Help )
I n	Inserta en la linea n + 1

Presione\_cualquier\_tecla\_para\_continuar

## SUMARIO

Como hemos visto el **Editor** es un sencillo editor de textos, el cual cumple con muchos de los requerimientos del futuro de un procesador de palabra, es rápido y hace que de una manera sencilla podamos crear y revisar un archivo de texto corto. El ejemplo que seguimos en los capítulos 1 y 2 mostramos que tu puedes usar el **Editor** para escribir un archivo de texto corto.

---

# CONCLUSIONES

---

- . Software fácil de usar
- . Trato amable con el usuario
- . Cumple con la definición de un EDITOR
- . Crea y Edita archivos de texto
- . Aplicable a Microcomputadoras compatibles con IBM

Este proyecto fué concebido en base a la experiencia obtenida en los primeros semestres de la carrera, basicamente en el manejo de EDITORES tanto de Microcomputadoras como de la VAX, además en mi trabajo (CDC, DG, NCR, VAX y Micros), siempre se encontraba con comandos diferentes para editar un archivo, algunos mas complicados que otros, de aquí extraje lo que fuera mas claro, sencillo y práctico de manejar a nivel Micros únicamente, además de incluir algunos comandos adicionales que en la actualidad no todos los EDITORES los tienen.



