

6. L'EDITOR ED

SOMMMARIO

Questo capitolo contiene la descrizione dell'editor ED.

INDICE

<u>INTRODUZIONE</u>	6-1
<u>RICHIAMO DI ED</u>	6-1
<u>INTRODUZIONE DEI COMANDI ED</u>	6-3
<u>COMBINAZIONE DI COMANDI ED</u>	6-5
POSIZIONAMENTO DEL CURSORE	6-6
VISUALIZZAZIONE DEL TESTO	6-6
EDITING DEL TESTO	6-7
<u>MESSAGGI DI ERRORE</u>	6-7
COMANDI ED	6-9
number (SKIP LINES)	6-9
number (GO TO LINE)	6-10
:number (THROUGH LINE)	6-10
a (APPEND)	6-11

b (BEGINNING/BOTTOM)	6-12
c (CHARACTER)	6-12
d (DELETE)	6-13
e (EXIT)	6-14
f (FIND)	6-14
h (HEAD OF FILE)	6-16
i (INSERT)	6-16
j (JUXTAPOSE)	6-17
k (KILL)	6-19
l (LINE)	6-20
m (MACRO)	6-21
n (NEXT)	6-22
o (ORIGINAL)	6-23
p (PAGE)	6-23
q (QUIT)	6-24
r (READ)	6-25
s (SUBSTITUTE)	6-26
t (TYPE)	6-27
v (VERIFY)	6-27

w (WRITE) 6-28

x (TRANSFER) 6-29

z (SLEEP) 6-30

INTRODUZIONE

Il programma di utilita' ED e' un editor "line-oriented" e "context. Cio' significa che permette di creare e modificare file linea per linea oppure fare riferimento a singoli caratteri all'interno di una stessa linea.

ED fornisce due modi di operare: modo "comando" e modo "insert". Quando ED viene richiamato, compare il prompt ":*" su video. Questo indica che ED e' in modo comando. L'introduzione del comando i provoca il passaggio di ED da modo comando a modo insert (se si vuole inserire ancora del testo); l'introduzione dei rimanenti comandi permette di manipolare a piacere il contenuto del file.

ED usa parte della memoria utente come buffer per memorizzare il testo al quale vengono aggiunti, cancellati o modificati caratteri. Un comando legge tutto o parte del file in questo buffer. Questo comando deve essere sempre richiamato per primo quando si vuole editare un file. I comandi w ed e riportano in parte o interamente il contenuto del buffer nel file.

Un immaginario "puntatore di carattere", detto CP, e' sempre posizionato all'inizio, oppure su un carattere intermedio, o alla fine del buffer. Sebbene CP non appaia sul video, esso puo' essere manovrato come un cursore dai vari comandi di ED.

Per agevolare la ricerca delle linee, ED associa ad esse un numero. Questi numeri compaiono su video alla sinistra della linea ma non fanno parte del testo del file. Il numero di linea puo' essere soppresso mediante il comando -v.

RICHIAMO DI ED

Per richiamare ED occorre usare la sintassi seguente:

```
ED filespec1 [filespec2]
```

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
filespec1	nome del file da editare o da creare
filespec2	nome del file che conterra' il testo editato

Tipo

Transiente

Caratteristiche

ED permette di creare o di editare il file identificato da filespec1.

Il parametro filespec2 deve essere specificato quando non si vuole alterare il contenuto del file identificato da filespec1 (questo file deve, pero', esistere). Il file filespec2 riceverà tutto il testo modificato mentre il file filespec1 rimarrà inalterato. Il file filespec2 non deve ancora esistere; in caso contrario ED emette il messaggio

Output File Exists, Erase It

Se filespec2 contiene solamente l'identificatore di drive, allora al file destinazione verrà assegnato lo stesso nome e la stessa estensione del file sorgente.

Se il file identificato da filespec1 non esiste ancora, allora ED provvede a crearlo ed emettere il messaggio:

```
NEW FILE
:*
```

Quando non viene specificato filespec2, ED preserva il file sorgente cambiandone l'estensione in BAK prima di rimpiazzarlo. Se viene editato un file con estensione BAK, allora ED rimuove il file sorgente non lasciandone così alcuna copia. Per ovviare a questo inconveniente, occorre usare il comando REN per modificare l'estensione BAK del file sorgente e quindi richiamare ED il quale associerà nuovamente BAK al file.

Se filespec2 coincide con filespec1, allora ED rimpiazza il file sorgente con quello contenente il testo editato.

INTRODUZIONE DEI COMANDI ED

La seguente tabella elenca tutti i comandi di ED. La loro descrizione, in ordine alfabetico, compare alla fine del capitolo.

COMANDO	DESCRIZIONE
number (Skip Line)	muove il CP di un numero di linee specificato
number: (Go To Line)	muove il CP alla linea specificata
:number (Through Line)	esegue comandi sulla linea specificata
a (Append)	carica linee di testo nel buffer
b (Begin/Bottom)	muove il CP all'inizio o alla fine del buffer
c (Character)	muove il CP a destra o a sinistra di un carattere
d (Delete)	cancella caratteri alla destra od alla sinistra del CP
e (End)	termina la sessione di editing e provvede alla scrittura del contenuto del buffer
f (Find)	muove il CP all'inizio di una data sequenza di caratteri
h (Head)	scrive il contenuto del buffer e ritorna all'inizio del file
i (Insert)	cambia da modo comando a modo insert
j (Juxtapose)	scrive caratteri in modo contiguo
k (Kill Lines)	rimuove una linea di testo sopra o sotto al CP

l (Line)	muove il CP su linee precedenti o seguenti
m (Macro)	permette l'esecuzione ripetuta di un gruppo di comandi
n (Next)	ricerca la prossima sequenza
o (Original)	rimuove il testo editato, ri-inizializza la sessione
p (Page)	muove il CP indietro o avanti di 23 linee
q (Quit)	rimuove il testo editato, nessun cambiamento
r (Read)	legge LIB o testo trasferito
s (Substitute)	sostituisce sequenze di caratteri
t (Type)	visualizza linee di testo
v (Verify)	numera le linee oppure visualizza il buffer
w (Write)	scrive il contenuto del buffer
x (Transfer)	trasferisce linee a/da file temporaneo
z (Sleep)	ritarda l'esecuzione dei comandi

La maggior parte di questi comandi vengono richiamati introducendo il relativo identificatore seguito da un argomento numerico. Alcuni comandi richiedono invece un argomento stringa.

Quando si deve specificare un argomento stringa occorre impostare l'identificatore di comando in carattere minuscolo, questo per poter valutare la stringa esattamente nel modo in cui e' stata impostata. Se il comando viene introdotto in carattere maiuscolo, allora anche la stringa viene convertita in caratteri maiuscoli. Quando non vi sono argomenti stringa, il comando puo' essere introdotto indifferentemente

in caratteri minuscoli o maiuscoli.

I comandi introdotti possono essere modificati, prima della loro esecuzione, per mezzo dei tasti di editing di linea. Impostare /CR/ per iniziare l'esecuzione della linea di comando.

Le seguenti regole valgono per gli argomenti numerici dei comandi di ED:

- se viene omissso un argomento numerico, viene forzato il valore 1
- un numero negativo permette di eseguire comandi all'indietro (su linee precedenti). Il comando b non segue questa regola
- il carattere "#" al posto di un argomento numerico, forza il valore 65535. Il carattere "-#" fa eseguire il comando sulle precedenti 65535 linee
- in alcuni comandi il numero 0 fa eseguire il comando approssimativamente meta' del numero previsto di volte. In altre parole, esso ostacola il movimento di CP.

Vedere i singoli comandi per informazioni dettagliate.

COMBINAZIONE DI COMANDI ED

Per risparmiare tempo di digitazione e di editing, conviene combinare comandi di editing e di visualizzazione. Una linea di comando puo' contenere un numero qualsiasi di comandi ED. Essa viene eseguita solamente dopo aver impostato il carattere /CR/. Usare i tasti editing di linea di CP/M-86 per gestire l'input da tastiera.

Quando una linea contiene piu' comandi, ED li esegue nell'ordine in cui sono stati introdotti, cioe' da sinistra a destra. Vi sono quattro restrizioni sulle linee di comando ED:

- la linea di comando non deve eccedere 128 caratteri in lunghezza.
- se la linea di comando contiene una stringa di caratteri, allora non deve essere lunga piu' di 100 caratteri.
- i comandi di terminazione di sessione non devono comparire in una linea di comando.
- i comandi come i,s,j,x e r che richiedono una stringa di caratteri oppure un "filespec", devono essere specificati per ultimi nella linea di comando oppure devono essere terminati da /CTRL/ /Z/ o da Esc (tasto RESET).

POSIZIONAMENTO DEL CURSORE

Per muovere il CP alla fine della linea indipendentemente dal numero di caratteri contenuti, occorre combinare il comando l con il comando c: l-2c. Questa combinazione di comandi e' analoga alla sequenza <cr><lf> alla fine della linea.

Modificare il comando c in questa sequenza per muovere il CP di piu' caratteri verso sinistra. Questa sequenza e' utile per effettuare una modifica alla fine della linea senza calcolare il numero dei caratteri precedenti la modifica stessa; esempio:

```
1: *t
1: Emily Dickinson said,
1: *l-7ct
said,
1: *
```

VISUALIZZAZIONE DEL TESTO

Il comando t visualizza il contenuto della linea a partire da CP fino alla fine della linea stessa. Per visualizzare l'intera linea occorre combinare i comandi l e t. Impostare Ol t per muovere il CP all'inizio della linea e quindi visualizzarla. Nell'esempio seguente, CP e' all'interno della linea; Ol lo riporta all'inizio. t puo' cosi' visualizzare la linea intera.

```
3: *t
sense of living
3: *Olt
3: the mere sense of living
3: *
```

Il comando Ott visualizza l'intera linea senza muovere il CP.

Per verificare il corretto posizionamento di CP, combinare il comando c con il comando t per visualizzare la linea. L'esempio seguente illustra una combinazione dei comandi c e t:

```
2: *8ct
ecstasy in living -
2: *
```

L'esempio seguente e' invece una combinazione dei comandi b e t. Esso visualizza l'intero contenuto del buffer.

```
4: *b#t
1: Emily Dickinson said,
2: "I find ecstasy in living -
3: the mere sense of living
4: is joy enough.
1: *
```

EDITING DEL TESTO

Per editare il testo ed effettuare veloci modifiche, conviene combinare comandi di editing con comandi di movimento del CP e di visualizzazione. Linee di comando, come quella dell'esempio sottostante, muovono il CP, cancellano caratteri specifici e verificano velocemente le modifiche.

```
1: *15c5d0lt
1: Emily Dickinson,
1: *
```

Per cancellare linee intere ed effettuare la verifica occorre combinare il comando k con altri comandi ED. Esempio:

```
1: *212kb#t
1: Emily Dickinson said,
2: "I find ecstasy in living -
1: *
```

La forma abbreviata del comando i (insert) permette di effettuare semplici modifiche del testo. Per modificare e poi visualizzare i cambiamenti occorre combinare il comando i con c e la stringa 0lt come nell'esempio seguente. Ricordare che la stringa di inserzione deve terminare con /CTRL/ /Z/.

```
1: *20ci to a friend^Z0lt
1: Emily Dickinson said to a friend,
1: *
```

MESSAGGI DI ERRORE

ED restituisce due tipi di messaggi di errore: messaggi relativi all'impossibilita' di eseguire un comando ED e messaggi relativi all'impossibilita' di leggere o scrivere in un file.

Il formato generico dei messaggi di errore ED e':

```
BREAK "x" AT c
```

dove x e' uno dei simboli elencati nella tabella seguente e c e' l'identificatore del comando in cui si e' verificato l'errore.

SIMBOLO	DESCRIZIONE
#	ricerca senza successo. ED non riesce a trovare la stringa specificata in un comando f,s o n.

?	identificatore non riconosciuto. ED non riesce ad identificare tale comando oppure uno dei comandi e,h,q o o non compare da solo in una stringa comando
0	file .LIB non trovato. ED non riesce a localizzare il file .LIB specificato nel comando R
>	buffer pieno. ED non riesce a memorizzare ulteriori caratteri nel buffer, oppure la stringa specificata in uno dei comandi f,n o s e' troppo lunga
E	comando interrotto. L'impostazione di un tasto ha interrotto l'esecuzione del comando.
F	errore di file. Puo' essere seguito da uno dei messaggi: 'DISK FULL' o 'DIRECTORY FULL'

Esempi

SE l'utente incontra...	ALLORA
BREAK ">" AT A	ED ha riempito completamente il buffer in memoria prima di terminare l'esecuzione di un comando. Quando viene emesso questo errore, CP e' posizionato alla fine del buffer e non e' consentito ulteriore editing. Usare il comando 0w per rilasciare al file meta' del buffer oppure uno dei comandi o od h e ri-editare il file.
BREAK "#" AT F	ED ha raggiunto la fine del buffer senza aver incontrato la stringa specificata in un comando f. A questo punto CP e' posizionato alla fine del buffer. Occorre muovere CP mediante b oppure n: per riprendere l'editing.
BREAK "F" AT F DISK FULL	non vi e' piu' spazio su disco. Usare il comando 0x per rimuovere qualche file oppure il comando b#xd: buffer.sav per riportare il contenuto del buffer su un altro disco.

BREAK "F" AT n
DIRECTORY FULL

Non vi e' piu' spazio disponibile nella directory. Intraprendere le stesse azioni correttive illustrate nell'esempio precedente.

BDOS ERR ON d:R0

il drive d: e'in stato R0 (Read-Only). Questo puo' accadere quando e' stato inserito un nuovo disco nel drive.

FILE IS READ ONLY

il file specificato nella chiamata ad ED ha l'attributo R0. Il file puo' essere letto ma ED non puo' modificarne il contenuto.

COMANDI ED

number (SKIP LINES)

Muove il CP e visualizza la linea di destinazione.

[[-]n]

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di linee di cui CP deve avanzare o regredire

Caratteristiche

ED muove il CP avanti od indietro di n linee; la linea di destinazione viene quindi visualizzata.

Una forma abbreviata di questo comando e' quella in cui non viene specificato alcun numero. In questo caso ED assume n=1 e muove quindi il CP sulla linea seguente visualizzandola. Analogamente, il segno "-" senza numero muove il CP indietro di una linea.

number: (GO TO LINE)

Quando vengono numerate le linee, ED accetta uno di questi numeri come indicatore di linea di destinazione.

n:

Dove

ELEMENTO SINTATTICO	DESCRIZIONE
n	numero della linea di destinazione.

Caratteristiche

Questo comando pone il CP all'inizio della linea specificata. Ad esempio, il comando 4: muove il CP all'inizio della quarta linea.

Ricordare che ED numera automaticamente le linee man mano che vengono inserite e/o cancellate. Il numero della linea di destinazione puo' quindi cambiare durante la fase di editing.

: number (THROUGH LINE)

Indica che un comando deve essere eseguito per un certo numero di linee. E esso puo' essere usato solamente in congiunzione con i tre comandi ED: T (type), L (Line) e K (Kill).

:ncommand

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di linee per cui deve essere eseguito il comando.
command	uno dei comandi t, b, l o k.

Caratteristiche

La parte :n del comando non muove il CP, puo' far cio' solo il comando che la segue. Si puo' combinare n: con :n per specificare un insieme di linee su cui eseguire il comando. Ad esempio, il comando 2::4t visualizza la seconda, la terza e la quarta linea:

```
1: *2::4T
2: "I find ecstacy in living -
3: the mere sense of living
4: is joy enough."
2: *
```

a (APPEND)

Trasferisce linee da un file sorgente nel buffer in memoria.

[n]a

Dove

ELEMENTO SINTATTICO	SIGNIFICATO
n	numero di linee di codice sorgente da trasferire nel buffer.

Caratteristiche

Il carattere "#" al posto di n viene interpretato come 65535. Dato che il buffer in memoria puo' contenere completamente la maggior parte dei file aventi dimensioni ragionevoli, e' spesso possibile usare il comando #a all'inizio della sessione ED per trasferire in memoria l'intero file di codice sorgente.

Se n=0, allora ED trasferisce tante linee di codice sorgente quanto ne bastano per riempire approssimativamente meta' buffer. Se n non viene specificato, allora ED trasferisce solamente una linea.

b (BEGINNING/BOTTOM)

Muove il CP all'inizio o alla fine del buffer.

[-]b

Caratteristiche

-b muove il cursore alla fine del buffer, b lo muove invece all'inizio.

c (CHARACTER)

Muove il CP avanti od indietro del numero di caratteri specificato.

[-]nc

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
<code>n</code>	numero di caratteri di cui CP deve avanzare o regredire.

Caratteristiche

Se `n` e' positivo, allora il CP avanza verso la fine della linea corrente. Se `n` e' negativo, allora CP si sposta nel verso opposto, cioe' verso l'inizio della linea e quindi anche del buffer. Il valore di `n` puo' essere sufficientemente grande da muovere il CP su un'altra linea. In questo caso occorre ricordare che ogni linea e' separata da quella che la precede o da quella che la segue, da due caratteri invisibili: un carriage return ed un line feed. Questi caratteri devono essere contati nello stabilire l'ampiezza dello spostamento del CP.

d (DELETE)

Cancella un dato numero di caratteri.

[[`-`]`n`]`d`**Dove**

ELEMENTO DI SINTASSI	SIGNIFICATO
<code>n</code>	numero di caratteri da cancellare.

Caratteristiche

Se `n` non viene specificato, allora ED cancella il carattere alla destra di CP. Un valore positivo di `n` cancella `n` caratteri alla destra di CP, cioe' verso la fine del file. Un valore negativo di `n` cancella `n` caratteri a sinistra di CP, cioe' verso l'inizio del file.

Il comando `d` permette di cancellare i caratteri `<cr>` e `<lf>` presenti tra due linee; in questo modo le due linee vengono riunite in una sola.

Ricordare che <cr> e <lf> sono due caratteri separati.

e (EXIT)

Provoca la terminazione della sessione ED.

e

Caratteristiche

Quando viene impostato questo comando, ED provvede a scrivere le linee del buffer e quelle del file sorgente nel nuovo file. Se esiste il file .BAK, allora ED lo cancella e rinomina il file originale associando l'estensione .BAK. ED cambia infine l'estensione temporanea del nuovo file da .\$\$\$ a quella specificata e ritorna il controllo a CP/M-86.

La sequenza delle operazioni intraprese dal comando e, sconsiglia l'editing di un file .BAK. In questo caso ED rimuove il file sorgente dato che ha estensione .BAK. Per evitare questo occorre sempre cambiare l'estensione BAK del file originale prima di incominciare una sessione ED.

Nota

Il comando di fine sessione deve sempre comparire da solo all'interno di una linea comando.

f (FIND)

Effettua la ricerca di una stringa.

[n]fstring

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	occorrenza della stringa da ricercare
string	stringa da ricercare

Caratteristiche

Il numero n deve essere positivo dato che la ricerca viene effettuata solamente in avanti. Se non viene specificato alcun numero, ED ricerca la prima occorrenza della stringa a partire da CP. Nel seguente esempio viene ricercata la seconda occorrenza della stringa 'living'.

```
1: *2fliving
3: *
```

CP viene posizionato all'inizio della terza linea ove e' stata localizzata la seconda occorrenza della stringa 'living'. Per visualizzare la linea, combinare il comando find con un comando type. Nel caso che il comando f sia seguito da un altro comando all'interno della stessa linea di comando, la stringa deve essere terminata dal carattere /CTRL/ /Z/, ad esempio:

```
1: *2fliving^Z01t
3: *the mere sense of living
```

Esiste una certa differenza nello specificare il comando f in carattere maiuscolo o minuscolo. Se viene specificato F, allora la stringa di ricerca viene automaticamente tradotta in caratteri maiuscoli. Il comando f invece lascia inalterata la stringa di ricerca. Ad esempio, il comando fCp/m-86 effettua la ricerca della stringa 'CP/M-86' mentre fCp/m-86 ricerca 'CP/M-86'.

Se la stringa non viene trovata, allora ED emette il messaggio

```
BREAK "#" AT
```

ove il simbolo # indica l'insuccesso di una ricerca effettuata dal comando f.

h (HEAD OF FILE)

Memorizza il contenuto del buffer senza terminare la sessione ED. Viene riportato nel buffer il testo iniziale del file. In questo modo il file può essere rieditato senza uscire da ED.

h

Caratteristiche

Per eseguire il comando h, ED prepara dapprima il nuovo file memorizzando il contenuto del buffer e del file sorgente. ED chiude il nuovo file, cancella l'eventuale file con nome uguale ed estensione .BAK e rinomina il file sorgente associandogli l'estensione .BAK. ED provvede quindi a cambiare l'estensione temporanea .\$\$\$ del nuovo file con quella specificata. Il nuovo file e quello originale vengono quindi riaperti per permettere un nuovo editing. Quando compare il prompt '*', CP è posizionato all'inizio del buffer di memoria.

i (INSERT)

Permette di inserire caratteri da video nel buffer in memoria.

i[string]

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
string	stringa da inserire al posto di CP

Caratteristiche

Quando viene impostato questo comando, ED si pone in modo 'insert'. Cio' significa che tutte le impostazioni fatte da tastiera vengono direttamente trasferite in un buffer di memoria. ED usa i caratteri introdotti per formare linee di testo; una nuova linea viene creata premendo /CR/. Ad esempio:

```
A>ED B:QUOTE.TEX
```

```
NEW FILE
: *i
1: Emily Dickinson said,
2: "I find ecstasy in living -
3: the mere sense of living
4: is joy enough."
5: ^Z
: *
```

Per uscire dal modo insert, impostare /CTRL/ /Z/.

Il comando i non pone in modo 'insert' quando gli viene associato il parametro stringa. Esso inserisce la stringa al posto di CP e ritorna immediatamente al prompt di ED. Si possono usare i caratteri di controllo della linea per editare la linea di comando.

Per inserire una stringa usare uno dei comandi di posizionamento CP. Esso deve essere posizionato dove si vuole inserire la stringa. Ad esempio, per inserire una stringa all'inizio della prima linea occorre usare il comando b seguito dal comando di inserzione; esempio:

```
iIn 1870, ^Z
```

Questo comando inserisce la stringa "In 1870 ," all'inizio della prima linea e ritorna quindi al prompt di ED. Nel buffer di memoria, CP viene posizionato alla fine della stringa inserita, cioè:

```
In 1870, ^Emily Dickinson said<cr><lf>
```

j (JUXTAPOSE)

Inserisce una stringa dopo una stringa di ricerca, quindi cancella tutti i caratteri presenti tra la stringa di ricerca e la stringa di cancellazione.

```
[n]searchstring^Zinsertstring^Zdeletetostring
```

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di occorrenza della stringa di ricerca
searchstring	stringa dopo la quale si vuole inserire il testo (terminata da /CTRL/ /Z/)
insertstring	stringa contenente il testo da inserire (terminata da /CTRL/ /Z/)
deletetostring	stringa fino alla quale si vogliono cancellare caratteri)

Caratteristiche

Se non viene specificato alcun numero, ED ricerca la prima occorrenza di search string nel buffer. Nell'esempio che segue, ED ricerca la stringa "Dickinson", inserisce "told a friend" dopo di essa e cancella quindi tutto il testo fino alla virgola.

```
1: *1t
1: Emily Dickinson said,
2: "I find ecstasy in living --
3: the mere sense of living
4: is joy enough.
1: *jDickinson^Z told a friend^Z
1: *01t
1: Emily Dickinson told a friend,
1: *
```

Quando questo comando viene combinato con altri, occorre terminare deletetostring con /CTRL/ /Z/ oppure con Esc (tasto RESET). Se il comando j viene impostato in carattere maiuscolo, ED trasforma in caratteri maiuscoli le stringhe specificate.

Il comando j e' utile specialmente quando si vuole modificare i comandi presenti in un file sorgente assembler. Esempio:

```
236: SORT   LXI   H, SW   ;ADDRESS TOGGLE SWITCH
236: *j;^ZADDRESS SWITCH TOGGLE^ZL^ZOLT
236: SORT   LXI   H, SW   ;ADDRESS SWITCH TOGGLE
236: *
```

In questo esempio ED ricerca il primo carattere ';', inserisce TOGGLE' dopo di esso e cancella quindi tutto il testo fino alla sequenza <cr> <lf>, rappresentata da /CTRL/ /L/ (l'uso di /CTRL/ /L/ per rappresentare <cr> <lf> permette di estendere la ricerca su piu' linee. Il carattere /CTRL/ /L/ viene usato per rappresentare i tab).

Nota

Se il testo di una linea e' piu' lungo di una linea di video, si puo' usare il carattere /CTRL/ /E/ per forzare un carriage return. Questo carattere provoca il posizionamento del cursore sulla parte sinistra del video ma non termina la linea di ED. Il numero massimo di caratteri che una linea di ED puo' contenere e' 100.

k (KILL)

Cancella linee intere dal buffer di memoria.

[[-]n]k

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di linee da cancellare

Caratteristiche

Se n e' positivo, allora viene cancellato un corrispondente numero di linee dopo l'attuale posizione di CP. Se n e' negativo, allora la cancellazione procede in senso inverso. Se non viene specificato, ED cancella la linea corrente. Supponiamo che CP sia posizionato all'inizio della seconda linea come nell'esempio seguente:

```
Emily Dickinson said,<cr><lf>
  "I find ecstasy in living -<cr><lf>
```

```
the mere sense of living<cr><lf>
is joy enough."<cr><lf>
```

Il comando -k cancella la linea precedente ed il buffer diventa:

```
"I find ecstasy in living -<cr><lf>
the mere sense of living<cr><lf>
is joy enough."<cr><lf>
```

Se CP e' all'interno di una linea, allora k cancella tutti i caratteri a partire da CP fino alla fine della linea stessa; i caratteri che precedono CP vengono concatenati con la linea seguente. Il comando -k cancella tutti i caratteri dall'inizio della linea fino a CP compreso. Il comando k cancella i caratteri della linea fino a CP escluso. Il simbolo speciale # puo' essere usato per cancellare tutto il testo compreso tra CP e l'inizio (o la fine) del buffer. Esso deve essere usato con cautela dato che non vi e' possibilita' di ripristinare le linee una volta cancellate dal buffer di memoria.

1 (LINE)

Muove il CP di un determinato numero di linee. Dopo questo comando CP si trova sempre all'inizio di una linea.

[-][n]1

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di linee di cui CP deve avanzare o regredire.

Caratteristiche

Un numero positivo fa avanzare CP verso la fine del buffer, un numero negativo lo fa regredire verso l'inizio. Il comando 21 muove CP in avanti di due linee e lo posiziona all'inizio della linea destinazione.

Il comando -1 muove CP all'inizio della linea precedente, anche se CP si trova all'interno della linea corrente.

Il comando m permette il raggruppamento di comandi ED per esecuzioni ripetute.

[n]mcommandstring

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di volte che la commandstring deve essere eseguita.
commandstring	sequenza di comandi da eseguire

Caratteristiche

Il numero n deve essere positivo. Se n non viene specificato, allora ED assume il valore # ed esegue la commandstring fino al raggiungimento della fine del buffer o del file di codice sorgente, a seconda dei comandi specificati. Nell'esempio seguente, ED esegue in maniera ripetitiva i quattro comandi specificati fino a che non raggiunge la fine del buffer.

```
1: *mfliving^Z-6diliving^Z01t
2: "I find ecstasy in Living -
3: the mere sense of Living
```

```
BREAK "#" AT ^Z
3: *
```

Il terminatore del comando m e' un carriage return; il comando m deve quindi comparire come l'ultimo comando della linea. Tutte le stringhe di caratteri specificate nel comando m devono terminare con /CTRL/ /Z/ oppure Esc (tasto RESET). Se una stringa termina la sequenza combinata di comandi, si deve usare il carattere /CTRL/ /Z/ seguito da /CR/.

L'esecuzione di un comando m termina sempre con un messaggio BREAK "#" anche se il numero di ripetizioni e' limitato e ED non ha raggiunto la fine del buffer o del file. L'identificatore di comando che compare nel

messaggio si riferisce, generalmente, ad uno dei comandi della sequenza di m e non a m stesso.

Per interrompere l'esecuzione del comando m, impostare /CTRL/ /C/.

n (NEXT)

Estende la ricerca di una stringa dal buffer di memoria al file sorgente. Se la ricerca ha esito positivo, allora CP viene posizionato sul primo carattere che segue la stringa cercata.

[n]nstring

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	occorrenza della stringa da ricercare
string	stringa da ricercare

Caratteristiche

Se non viene specificato il numero n, ED ricerca la prima occorrenza della stringa. Se questo comando viene specificato in maiuscolo la stringa di ricerca viene convertita in maiuscolo come per il comando f. Se n e' seguito da un altro comando, la stringa di ricerca deve terminare con /CTRL/ /Z/.

ED esegue, nel buffer di memoria, la ricerca della stringa specificata; se questa stringa non viene trovata, non viene emesso alcun messaggio di errore e la ricerca viene estesa alla parte del file sorgente non ancora in memoria. ED esegue il comando Oa per trasferire nel buffer la porzione del file non ancora usata per la ricerca. La ricerca continua in questo modo sino alla localizzazione della stringa o sino alla fine del file. In quest'ultimo caso appare il messaggio

BREAK "#" AT

Dato che ED riporta il testo presente nel buffer nel nuovo file prima di considerare gli ulteriori dati presenti nel file sorgente, il buffer viene sempre trasferito prima di trovare la fine del file con relativa segnalazione. Il comando h deve essere usato per continuare la sessione di editing dopo che si e' arrivati alla fine del file sorgente ed il buffer in memoria e' vuoto.

o (ORIGINAL)

Tutte le modifiche, le aggiunte e/o le cancellazioni effettuate vengono annullate ed il file sorgente e' riportato alla situazione iniziale.

o

Caratteristiche

Quando viene impostato il comando o, ED richiede conferma mediante il messaggio

O (Y/N)?

Qualsiasi impostazione diversa da "Y" o "N" provoca la riemissione del messaggio. Se viene impostato 'Y' allora ED rimuove il file temporaneo e cancella il contenuto del buffer di memoria. Quando ritorna il prompt '*', CP e' posizionato all'inizio del buffer in memoria, esattamente come quando e' stato lanciato ED.

p (PAGE)

Visualizza un determinato numero di pagine di testo.

[[-]n]p

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di pagine da visualizzare

Nonostante si possa visualizzare una qualsiasi porzione del file mediante il comando t, a volte e' piu' conveniente avanzare o retrocedere sul testo di un certo numero di 'pagine', ponendo CP ogni volta all'inizio di una nuova pagina. Il comando p permette di effettuare questa operazione.

Se n non viene specificato, allora ED visualizza le 23 linee che seguono CP posizionandolo in avanti della quantita' visualizzata; in questo modo CP coincide con il primo carattere che appare su video.

Per visualizzare la pagina corrente senza muovere il CP, impostare Op. Il carattere 0 previene il movimento di CP. Un valore negativo di n provoca la visualizzazione delle pagine che precedono CP.

q (QUIT)

Termina la sessione di ED senza effettuare modifiche al file.

q

Caratteristiche

Quando viene impostato il comando q, ED esegue la verifica voluta e invia il messaggio

Q (Y/N)?

Occorre rispondere 'Y' o 'N'; qualsiasi altro carattere causa la riemissione della domanda. Se viene impostato 'Y' allora ED rimuove il file temporaneo, chiude il file sorgente e ritorna il controllo a CP/M-86.

Nota

L'impostazione di /CTRL/ /C/ provoca un ritorno immediato a CP/M-86. Questo non permette però a ED di chiudere il file sorgente e quelli creati; previene però la cancellazione di quelli temporanei.

r (READ)

Il comando x trasferisce n linee di testo in un file di libreria. Il comando r permette di leggerle.

r[filespec]

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
filespec	identificatore del file temporaneo

Caratteristiche

Se non viene specificato alcun file, allora ED assume il file X\$\$\$\$\$. Se non viene specificata alcuna estensione, viene assunta .LIB. Il comando R inserisce il contenuto del file di libreria immediatamente dopo CP; in questo modo CP non cambia posizione pur trovandosi all'inizio di una nuova linea. Se r viene combinato con altri comandi, filespec deve essere separato dal seguito mediante /CTRL/ /Z/. Occorre verificare l'esecuzione di R.

```
1: *41
  : *R`ZB#T
1: "I find ecstasy in living -
2: the mere sense of living
3: is joy enough."
4: Emily Dickinson said,
1: *
```

S (SUBSTITUTE)

Ricerca una data stringa all'interno del buffer di memoria. Quando viene localizzata, viene sostituita da un'altra.

```
[n]ssearchstring^Znewstring
```

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero delle sostituzioni da effettuare
searchstring	la stringa da sostituire (terminata da /CTRL/ /Z/)
newstring	la nuova stringa

Caratteristiche

Se non viene specificato alcun numero, ED ricerca la prima occorrenza di searchstring nel buffer. Ad esempio, il comando

```
sEmily Dickinson^ZThe poet
```

ricerca la prima occorrenza della stringa 'Emily Dickinson' che sostituisce con 'The poet'; CP viene quindi posizionato dopo la nuova stringa.

Se viene impostato il comando maiuscolo S, allora ED si comporta come se entrambe le stringhe fossero state specificate in caratteri maiuscoli; la ricerca e la sostituzione avvengono quindi in accordo a questa regola: Quando questo comando e' combinato con altri, newstring deve terminare con /CTRL/ /Z/.

t (TYPE)

Visualizza un dato numero di linee su video.

[[-]n]t

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di linee da visualizzare

Caratteristiche

Se viene specificato un numero negativo, ED visualizza le n linee che precedono CP; se viene specificato un numero positivo, vengono visualizzate le n linee che seguono CP. Se non viene specificato alcun numero, ED visualizza il contenuto della linea corrente a partire da CP fino alla fine della linea stessa. La posizione di CP rimane inalterata indipendentemente dal numero di linee visualizzate. Ad esempio, se CP e' all'inizio del buffer e viene specificato il comando 4t, allora vengono visualizzate quattro linee; CP rimane pero' all'inizio della linea 1.

Se CP e' all'interno di una linea, allora il comando t, senza n specificato, visualizza solamente i caratteri tra CP e la fine della linea. CP rimane inalterato.

Durante la fase di visualizzazione e' possibile impostare il carattere /CTRL/ /S/ per interromperla ed il carattere /CTRL/ /Q/ per riprenderla. /CTRL/ /C/ interrompe definitivamente lunghe visualizzazioni.

v (VERIFY)

Abilita e disabilita l'inserimento di numeri di linea. Visualizza pure il numero di byte liberi e la dimensione totale del buffer di memoria.

[-][0]v

Caratteristiche

All'inizio della sessione ED, viene abilitata la numerazione automatica delle linee. Impostare -v per disabilitarla. e v per riabilitarla.

Il comando 0v visualizza il numero di byte liberi e la dimensione totale del buffer di memoria. Il formato usato e' xxxxx/yyyy ove xxxxx sono i byte liberi e yyyy la dimensione del buffer.

w (WRITE)

Riporta nel file le linee contenute nel buffer di memoria.

[n]w

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
n	numero di linee, a partire dall'inizio del buffer, da riportare nel file (Il valore di default e' 1)

Caratteristiche

Se n e' maggiore di 0, ED riporta nel file le prime n linee contenute nel buffer. Se n e' 0, ED riporta un numero di linee tale da svuotare circa meta' buffer. Il comando 0w e' un modo conveniente per lasciare nel buffer di memoria spazio per il testo ancora memorizzato nel file. Se il buffer e' pieno, allora 0w riporta meta' buffer nel file. Il comando #w svuota completamente il buffer; si puo' quindi usare il comando 0a per riempire il buffer con ulteriore testo contenuto nel file.

Nota

Dopo l'esecuzione di un comando `w`, occorre usare il comando `h` per poter rieditare le linee riportate nel file durante la sessione corrente.

x (TRANSFER)

Trasferisce linee in un file temporaneo.

```
[n]x[filespec]
```

Dove:

ELEMENTO DI SINTASSI	SIGNIFICATO
<code>n</code>	numero di linee, che a partire da CP verso la fine del buffer devono essere trasferite in un file temporaneo. Il numero <code>n</code> deve essere sempre positivo; il valore di default e' 1.
<code>filespec</code>	identificatore del file temporaneo

Caratteristiche

Se non viene specificato `filespec`, allora viene assunto `X$$$$$$`. Se non vengono specificate estensioni, allora viene assunta `.LIB`. Se il comando `x` non e' l'ultimo nella linea di comando, allora deve essere terminato da `/CTRL/ /Z/` oppure da Esc (tasto RESET).

Alla fine della sessione corrente, il file temporaneo viene rimosso. Per poter salvare il testo trasferito occorre prima usare il comando `r`.

Se non viene specificato il nome del file, allora ED ricerca il file `X$$$$$.LIB`. Se non esiste, viene creato. Se esiste, le linee specificate vengono trasferite alla fine del file.

Il carattere `0`, come argomento di `x`, permette di cancellare, da ED, il file specificato.

z. (SLEEP)

Inserisce una pausa tra l'esecuzione dei comandi ED.

`n[z]`

Dove

ELEMENTO DI SINTASSI	SIGNIFICATO
<code>n</code>	numero di secondi di pausa (default=1).

Caratteristiche

Questo comando e' utile all'interno di macro per permettere, in ogni momento, l'interruzione di operazioni ripetute.