

IST. EL. INF.  
BIBLIOTECA  
Posiz. ARCHIVIO

Consiglio Nazionale delle Ricerche

# ISTITUTO DI ELABORAZIONE DELLA INFORMAZIONE

PISA

LAF - EDITOR

MANUALE D'USO

C. LAMI - R. FAVILLA

Nota interna B 77 - 1

Gennaio 1977.

## LAF - EDITOR: MINI EDITORE DI FILES

Il LAF-EDITOR è un programma di tipo conversazionale che permette di creare o modificare files di tipo S.S. (Source Sorgent), a formato variabile e compatibile con il sistema DOS-III-HP, che abbiano una occupazione massima di disco nella misura di 58-59 settori (2 tracce, 15.000 car.)

Il LAF-EDITOR è scritto interamente in linguaggio FORTRAN-HP ed a "RUN-TIME" occupa per intero l'area di memoria (32k) che il sistema DOS mette a disposizione dell'utente. Dispone di "Features" (primitive), che sono proprie di "EDITORS" avanzati, come quelle di: Cambiare una stringa di caratteri con un'altra, cancellare records, inserire records, stampare records, ricercare una stringa di caratteri, ecc...

### Organizzazione del file in memoria.

Sia che il file venga letto da disco che creato dall'utente tramite il terminale, il LAF-EDITOR, venendo incontro alle esigenze dell'utente, fissa alcuni "punti di riferimento" caratterizzanti il file in memoria. Questi "punti" consentono all'utente una facile gestione del file ed un buon uso delle primitive.

### Punti di riferimento.

Top  
Pointer o Record corrente  
Bottom  
\*EOF\*

### Top.

Il punto "Top" del file rappresenta la testa del file; cioè la posizione immaginaria che stà tra l'inizio del file e il primo record di quest'ultimo.

### Pointer.

Questo punto indica il "record corrente" del file. Il "record corrente", come si vedrà nella successiva descrizione dei comandi, costituisce per il LAF-EDITOR un punto di riferimento per l'esecuzione di molti comandi.

### Bottom.

Il punto "Bottom", definito dal LAF-EDITOR, indica l'ultimo record utile del file corrente.

### \*EOF\*

Questo punto costituisce la fine fisica del file corrente; cioè la posizione immaginaria che stà tra l'ultimo record utile (Bottom) e la fine del file.

Fig. N. 1: Struttura file in memoria.

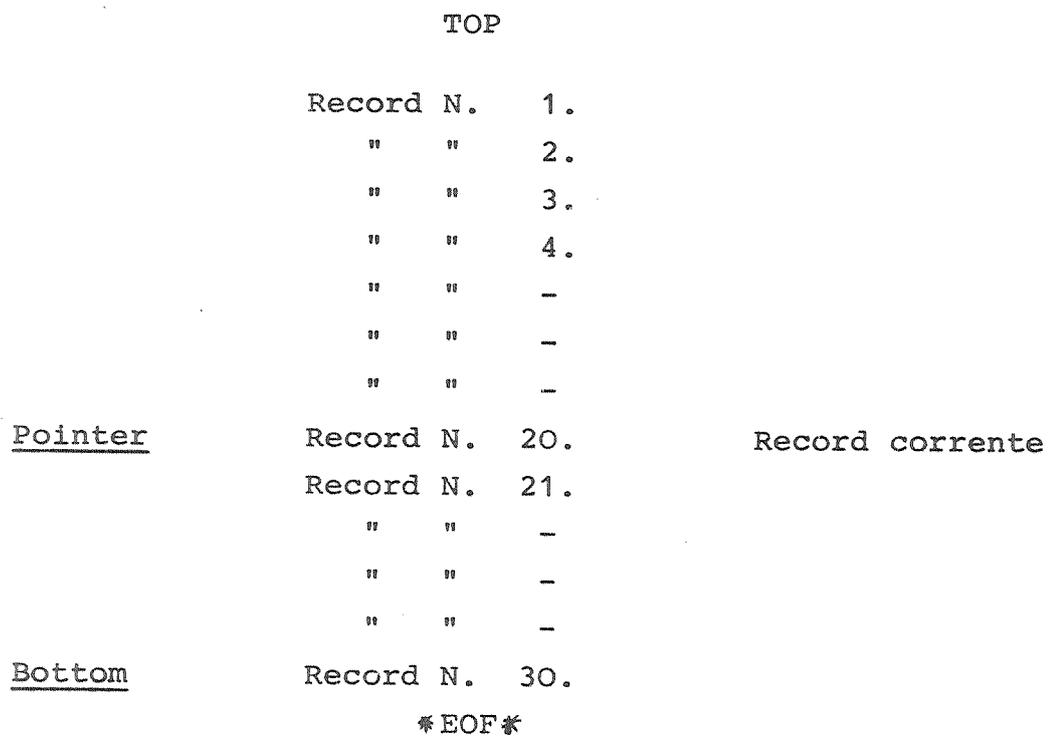
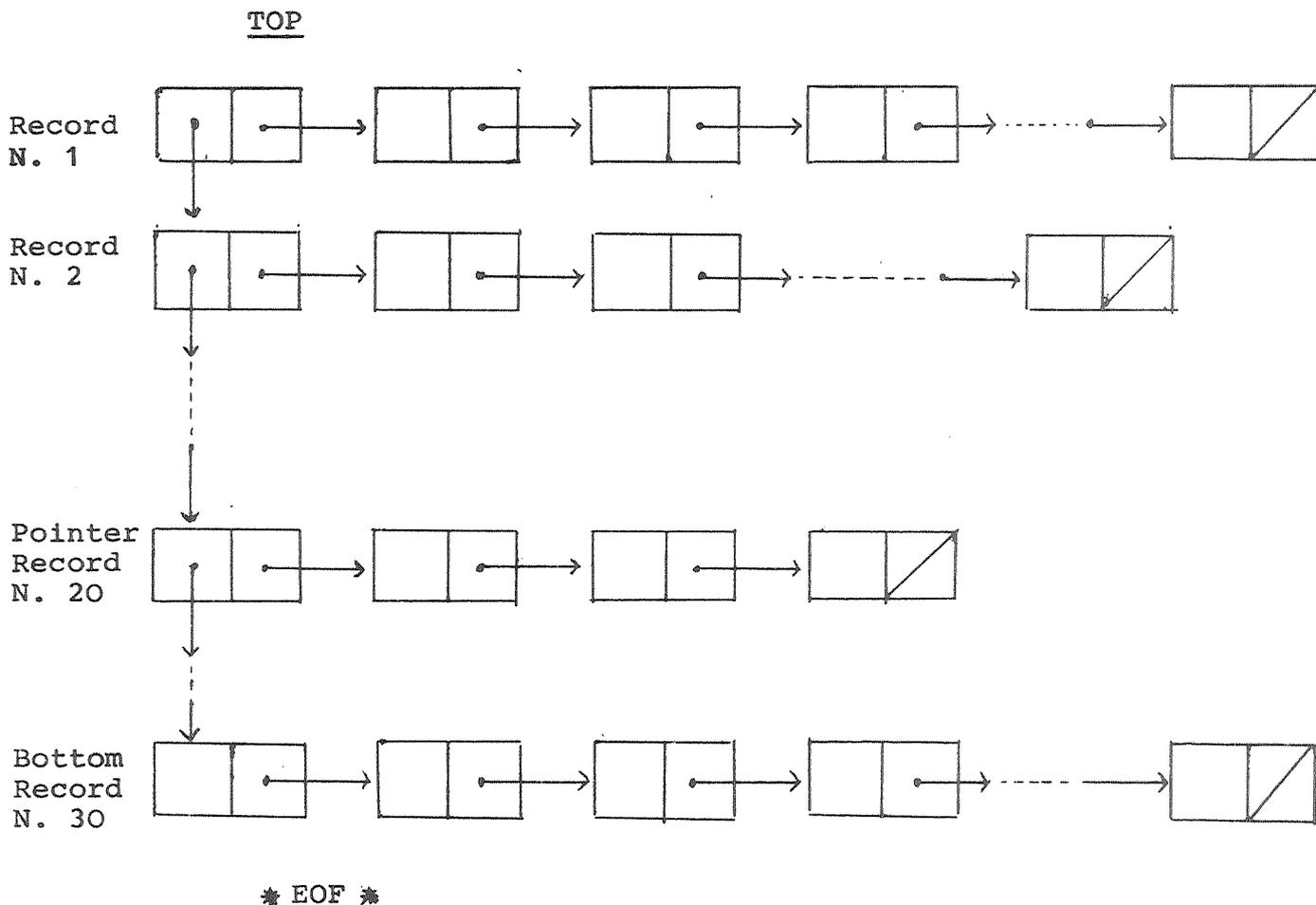


Fig. N. 1.b: Struttura file in memoria.



N.B.- Il file in memoria è costituito da due vettori di nome ICAR e ICDR.

ICDR è un vettore di puntatori; ICAR è un vettore che contiene sia puntatori che informazioni relative al record. Nella prima cella di ogni record, in ICAR, è contenuto il puntatore del record successivo; nelle altre celle che compongono il record una coppia di caratteri.

## STRUTTURA DEI COMANDI.

- 1) AGAIN
- 2) BOTTOM
- 3) CHANGE<\*)␣ / STRING 1 / STRING 2/
- 4) DELETE ␣ <N>
- 5) FILE ␣ <NAME>
- 6) GO ␣ <N >
- 7) INSERT
- 8) LOCATE ␣ / STRING /
- 9) MEMORY
- 10) NEXT ␣ <N >
- 11) OREAD ␣ NAME 1
- 12) PRINT <N>
- 13) QUIT
- 14) ROW
- 15) SAVE <NAME >
- 16) TOP
- 17) UP <N>
- 18) WRITE <N>

AGAIN.

Ripete l'esecuzione dell'ultimo comando inviato dall'utente.

BOTTOM.

Sposta il "pointer" sull'ultimo record del file corrente.

CHANGE<\*>L/STRING 1/STRING 2/

Cambia la stringa di caratteri STRING 1 con la stringa STRING 2 del record corrente. L'opzione " \* " permette di cambiare la stringa di caratteri STRING 1 con la STRING 2 alla prima occorrenza di STRING 1 in ogni record per tutti i records compresi tra quello corrente e l'END OF FILE (\* EOF \*).

"/" Carattere delimitatore. Può essere un carattere qualsiasi, ma non deve essere contenuto in nessuna delle due stringhe di carattere STRING 1 e STRING 2. Una volta eseguito il comando con l'opzione " \* " il "pointer" si troverà posizionato sulla \* EOF \* .

DELETE L <N>

Cancella "N" records del file a partire da quello corrente. Dopo che il comando è stato eseguito, il "pointer" sarà posizionato sul record immediatamente successivo all'ultimo cancellato. Per default N = 1.

FILE <NAME> .

Memorizza su disco in forma permanente il file corrente. L'opzione "NAME" permette di copiare su disco il file corrente con un nome uguale o diverso da quello che l'utente ha assegnato al file corrente nel momento in cui il file è stato richiamato (da disco) o creato (da terminale) in memoria. (Vedere uso del LAF-EDITOR).

Una volta eseguita la primitiva FILE, l'ambiente di Edit viene abbandonato ed a terminale verrà stampata l'informazione seguente:

```
EDIT 1 : STOP 0 0 0 0
```

Per default il nome assegnato al file è quello corrente.

#### GO <N>

Sposta il "pointer" sul  $N^{\text{mo}}$  record del file corrente. Se "N" è maggiore del numero dei records che compongono il file corrente, a terminale viene stampato: \*EOF\* e il "pointer" si troverà su questa posizione.

Per default  $N = 1$ .

#### INSERT.

Permette di inserire nuovi records nel file corrente. Dopo che il comando è stato riconosciuto il LAF-EDITOR risponde con la dicitura: INPUT. Da quel momento, passando dall'ambiente conversazionale di EDIT a quello di INPUT, si possono inviare tramite il terminale nuovi records. I records vengono inseriti immediatamente dopo la posizione indicata dal "pointer" e terminati gli inserimenti il "pointer" si troverà posizionato sull'ultimo record inserito. Nel caso in cui il "pointer" sia sul "Top" prima dell'inserimento, i records inviati saranno collocati in testa al file corrente. Analogamente, se il "pointer" si trovasse sul Bottom o sull'\*EOF\*, i nuovi records saranno aggiunti in coda al file corrente.

"/\*" Una volta terminati gli inserimenti, per abbandonare l'ambiente Input e tornare in Edit, è necessario inviare da terminale a partire da colonna 1 la cop

pia di caratteri "/\*".

" > " In ambiente di Input questo carattere, solo se inviato nei primi 6 caratteri che fanno parte del record, viene interpretato come tabulatore. La sua funzione è quella di fare iniziare o continuare il record a partire dalla 7<sup>a</sup> colonna.

#### LOCATE <sup>L</sup> /STRING/

Ricerca la stringa di caratteri STRING a partire dal record successivo a quello corrente fino allo \* EOF\*. Se la stringa viene incontrata prima dell' \* EOF\*, a terminale viene stampato il record in cui è contenuta e il "pointer" si troverà posizionato sul record che contiene la stringa cercata. Se la stringa di caratteri STRING non viene incontrata prima dell' \* EOF\* il "pointer" si sposterà sul record immediatamente successivo a quello corrente ed a terminale verrà stampato \* EOF\* e il record corrente attuale.

"/" Per i delimitatori vale quanto detto per il CHANGE.

#### MEMORY.

Consente di ottenere la stampa di alcune informazioni relativa allo stato del file corrente. (records occupati, settori di disco eventualmente occupati, nome del file, tracce consecutive libere su disco, memoria disponibile.)

N.B. E' buona norma, prima di memorizzare un file su disco con SAVE o FILE, verificare la disponibilità di tracce consecutive libere. Nel caso che non vi sia spazio disco a sufficienza, è possibile recuperare il file corrente utilizzando il comando WRITE.

NEXT <N>

Sposta il "pointer" verso il Bottom di "N" records a partire da quello corrente e stampa a terminale il record raggiunto. Qualora il "pointer" si dovesse spostare di un numero di records maggiore di quello che stà tra il record corrente e il Bottom, a terminale verrà stampata la dicitura \*EOF\* punto sul quale il "pointer" si sarà posizionato.  
Per default N = 1.

ROW.

Stampa a terminale il numero di sequenza del record corrente.

CURRENT LINE: X X X X

SAVE <NAME>

Scrive su disco il file corrente e ritorna in ambiente di Edit.

Per l'opzione "NAME" vale quanto detto per il comando FILE.

TOP.

Sposta il "pointer" sulla posizione TOP del file corrente e stampa a terminale la dicitura TOF.

UP <N>

Sposta il "pointer" verso il TOP di "N" records a partire da quello corrente e stampa a terminale il record raggiunto. Qualora il "pointer" si dovesse spostare di un numero di record maggiore di quello che stà tra il record corrente e il TOP, a terminale verrà stampata la dicitura TOF. e il "pointer" verrà spostato su questa posizione.  
Per default N = 1.

WRITE <N>

Consente di ottenere una copia del file corrente sull'apparecchiatura di I/O identificata da N.

Per default N = 1.

N può assumere i seguenti valori:

1 = Teletype

4 = Perforatore

6 = Stampante

7 or 8 = Nastro Magnetico

L'output ottenuto è ancora compatibile con il sistema DOS e con il LAF-EDITOR.

OREAD L NAME 1

Permette di leggere un file da disco di tipo SS e di inserirlo immediatamente dopo il record corrente del file corrente. Quanto detto per il comando Insert, e nel caso specifico per il "pointer" e per l'inserimento di nuovi records, vale anche per questo comando. NAME1 non è un argomento opzionale ma fa parte del comando; rappresenta il nome del file memorizzato su disco da inserire nel file corrente.

PRINT L <N>

Stampa a terminale "N" records a partire da quello corrente. Una volta completato il comando il "pointer" si troverà posizionato sull'ultimo record stampato.

QUIT.

Permette di uscire dall'ambiente di Edit e ritornare sotto DOS senza alterare la copia del file su disco o quantomeno senza memorizzare su disco il file corrente.

E' possibile inoltre utilizzando in alcuni comandi l'opzione <.> (punto), inibire la stampa a terminale. Questa opzione deve immediatamente seguire il nome del comando.

es. LOCATE. ␣ /STRING/  
CHANGE. \*␣ /STRING 1/STRING 2/  
UP. ␣ 10  
NEXT.  
BOTTOM.  
GO. ␣ 12

Informazioni ottenute a terminale utilizzando il comando FILE (o SAVE)

OUTPUT LINES:	Numero records del file in memoria.
SECTORS:	Numero dei settori occupati su disco con la scrittura del file corrente.
FILE:	Nome del file corrente.
FREE TRACKS:	Tracce consecutive libere su disco.
FREE CELLS:	Celle libere utilizzabili per aggiungere nuove informazioni al file corrente.

Le stesse informazioni vengono ottenute utilizzando il comando MEMORY.

#### Restrizioni.

- Il formato dei comandi è libero ed è limitato a 80 caratteri. Il carattere blank (␣) nel comando o nell'opzione è considerato come separatore.
- Non è necessario che ogni comando sia esplicitato per intero; il comando è accettato un una qualsiasi di queste sequenze:  
B            C or C \*␣ /X/Y/



- i) Se il comando CHANGE modifica un record aggiungendo caratteri e superando i 72, a terminale verrà stampata la dicitura CHANGE-TRUNCATED e il record "troncato".
- l) Il LAF-EDITOR provvede alla gestione della memoria libera (quello non occupata dal file) alla quale aggancia i record cancellati (Delete) oppure cambiati (CHANGE) ove vi sia stata una riduzione in lunghezza del record. Per l'INSERT, OREAD e CHANGE (qualora aumenti il numero dei caratteri che compongono il record) il LAF-EDITOR acquisisce la memoria necessaria per inserire le nuove informazioni nel file corrente. Se, nel caso di acquisizione di memoria, si raggiunge il limite di memoria disponibile a terminale verrà stampato "MEMORY OVERFLOW" e EDIT1 : STOP  $\emptyset\emptyset\emptyset$  caso INSERT e change;  
 EDIT1 : STOP  $\emptyset\emptyset\emptyset$   
 FILE TOO LARGE nel caso di inserimento nel file corrente di un file di disco letto con il comando OREAD.
- m) Poichè la struttura di memoria dei calcolatori HP è data da parole di lunghezza fissa (2 bytes - 16 bits) ed anche perchè i records sono organizzati a formato variabile, la cui lunghezza è sempre calcolata in parole e non in caratteri, ai records inviati in ambiente di Input oppure modificati con il CHANGE, viene aggiunto un blank ( $\emptyset$ ) come ultimo carattere del record qualora il numero complessivo dei caratteri che lo compongono sia dispari. (Tutto ciò è compatibile con il sistema DOS-HP).
- n) Durante l'esecuzione del LAF-EDITOR si possono avere degli STOP di programma seguiti da un codice
- STOP 5252  
 STOP 7777 errori del sistema LAF (comunicarlo agli autori.)
- STOP 7171 errore di PURGE - FILE  
 STOP 7272 errore di CREATE - FILE  
 STOP 7373 errore di RENAME - FILE

o) Per ogni file che deve essere scritto su disco vengono sempre effettuati questi controlli:

1) Esistenza del file (vecchio)

a) se esiste viene cancellato (PURGE)

b) successivamente creato (CREATE e WRITE) file di tipo AD (ASCII DATA).

c) quindi rinominato (RENAME) per portare il tipo di ASCII DATA a S.S. (SOURCE SORT).

2) Se il file non esiste si procede come al punto 1b.

Il sistema DOS-HP non permette ad un utente che utilizza un linguaggio ad alto livello (FORTRAN) di creare file di tipo S.S. compatibili con i compilatori di cui dispone. E' necessario quindi per ogni file aggiungere l'ostacolo memorizzando le informazioni su disco come dati e quindi tramite il RENAME modificare il tipo (da AD ad S.S.).

p) La stampa dei caratteri ??? sta ad indicare che l'utente ha compiuto una operazione non permessa. Tutto questo vale anche per le diciture:

DELIMITER        Errato uso dei delimitatori, delimitatore mancante.

NO-AGAIN         Errato uso dell'AGAIN.

NO DEL -EOF POS.    L'utente tenta di cancellare record ma il pointer si trova sull'EOF.

FILE EMPTY        Questa dicitura la si ottiene quando un utente tenta di inviare dei comandi, quando il file è vuoto oppure quando con il comando DELETE si cancellano tutti i records che compongono il file in memoria.

#### USO DEL LAF-EDITOR

Per poter usare il programma editore LAF-EDITOR è necessario averlo compilato e posto sotto il controllo del sistema DOS-HP e inoltre con la dicitura DOS:

: RUN, EDIT 1

è possibile entrare in ambiente di EDIT.

A questo punto il sistema stamperà a terminale la dicitura:

LAF-EDITOR ed ancora

FILENAME : ?

Alla richiesta FILENAME l'utente dovrà rispondere, inviando da terminale, il nome del file che intende creare o modificare.

IFUNZ        nome del file dell'utente.

Nel caso che il file esista il LAF-EDITOR si preoccupa di verificarlo e di leggerlo portandolo in memoria. A terminale verrà stampato il numero dei records che compongono il file.

INPUT LINES : X X X

EDIT :

da questo momento in poi si possono utilizzare tutti i comandi che il LAF-EDITOR mette a disposizione dell'utente. L'unica eccezione è costituita dal comando AGAIN il quale per essere eseguito deve essere preceduto da un altro comando.

Nel caso in cui il file non esista su disco a terminale avremo:

NEW FILE :

EDIT :

il file in memoria a questo punto è vuoto (EMPTY FILE), occorre crearlo utilizzando i comandi INSERT or OREAD.

Una volta giunti a termine della creazione o della modificazione del file in memoria basta inviare da terminale il comando FILE con o senza l'opzione "NAME".

Il file corrente verrà scritto su disco e qualora su questo esistesse la vecchia copia si provvede a cancellarla.

## ESTENSIONE

CHANGE <+> /STRING 1 / STRING 2/  
CHANGE <^> / STRING1 / STRING 2/

Queste opzioni (+, ^) permettono rispettivamente di cambiare la stringa di caratteri STRING 1 con la STRING2 in ogni occorrenza del record attuale (+); e di cambiare la STRING 1 con la STRING 2 (^) in tutte le occorrenze del record corrente per tutti i records che stanno tra quello corrente e l'\*EOF.\*

## BREVE SEDUTA A TERMINALE.

Viene riportato qui di seguito un breve esempio di seduta a terminale per la scrittura di un programma FORTRAN.

Le diciture in caratteri maiuscoli vengono stampate dal LAF-EDITOR e quelle in minuscolo dall'utente. Vengono riportati sia l'esempio di creazioni di file che di modifica di un file esistente su disco.

Caso di creazione file.

In ambiente D.O.S.

```
:run, edit 1
LAF-EDITOR
FILENAME :?
ifunz
NEW FILE :
EDIT
TOP
???
insert
INPUT
ftn4
c          programma di prova.
c          questa funzione calcola
```

```

c          la somma di (a+b)/c
c
a colonna 7 function ifunz (a,b,c,)
          ifunc = (a+b/c
          ritorna
          end
/* (FINE INSERIMENTI)
top
TOF :
print 100

FTN 4
c          Programma di prova
c          questa funzione calcola
c          la somma di (A+B)/C
c
          Function Ifunz (A,B,C)
          Ifunz = (A+B/C
          Ritorna
          End
          * EOF *

locate   /A+B/
          Ifunz = (A+B/C

change   /+b/+b)/
          Ifunz = (A+B)/C

next
          ritorna

change  /ITORNA/RETURN/
          RRETURN

File    (vengono stampate alcune informazioni rela
          tive allo stato del file)

EDIT 1 : STOP
" dopo aver compilato si scopra che vi è un errore:
RRETURN ha una R di troppo.

```

E' necessario richiamare il file IFUNZ in memoria e modificarlo, quindi memorizzarlo nuovamente su disco e finalmente compilarlo.

Caso di modifica di un file.

:run; edit 1

LAF-EDITOR

FILENAME : ?

ifunz

INPUT LINES : 9

EDIT :

bottom

END

up

RRETURN

Change

/RR/R/

RETURN

file

(vengono stampate alcune informazioni del file.)

EDIT : STOP 0000.