



software

C-64

a cura di Tommaso Pantuso

Listing Plus per Commodore 64 con disk drive 1541 e stampante MPS 802 di Daniele Finocchiaro Linguaglossa (CT)

Listing Plus (di cui non pubblichiamo il listato a causa dell'eccessiva lunghezza) è un programma creato ad hoc per poter gestire la mia lista di programmi. Prima di scriverlo, ho provato ad utilizzare altri data base manager, come vengono chiamati, ma non ne ho trovato nessuno che rispondesse a queste due assolute esigenze:

- 1) gestire almeno 1500 record
- 2) stampare una lista molto compatta.

Sono rimasto stupito nel vedere che lo stesso SuperBase di mamma Commodore non risponde ai due requisiti (stampa infatti un solo record per riga, e si ferma a circa 1200 record). Listing Plus è nato dunque per soddisfare queste esigenze. Il programma che ne è derivato assolve molto bene al suo compito. Vediamone le caratteristiche principali:

- capacità massima 3600 record;
- ogni record ha 2 campi: nome (16 caratteri) e codice (8 caratteri);
- il file è suddiviso in 5 file relativi e 1 sequenziale;
- stampa in ordine alfabetico su 3 colonne;
- funzioni di insert, delete, variazione, ricerca presenti;
- velocità apprezzabile;
- scritto in Basic più breve routine in LM.

Prima di spiegare come funziona il programma, spieghiamo come è stato concepito.

È disponibile, presso la redazione, il disco con i programmi pubblicati in questa rubrica. Le istruzioni per l'acquisto e l'elenco degli altri programmi disponibili sono a pag. 265.

Sul Commodore 64 non si possono tenere 3000 record in memoria, quindi un file di dimensioni consistenti deve trovarsi necessariamente sul disco. Non tuttavia in un file sequenziale, che richiederebbe di essere caricato in memoria per essere gestito.

La soluzione più idonea è quella dei file «random» che mamma Commodore, chissà perché, ha deciso di chiamare «relatives» (e ha chiamato random quelli con gli U1 e U2). Tuttavia anche i file relativi hanno le loro limitazioni: oltre ad essere notoriamente lenti, non possono contenere più di 720 record.

Che fare, allora? Semplice, si divide il «megafile» in 5 «minifile», cosicché 5 file * 720 record = 3600 record. Come si divide il file grosso? Altrettanto semplice; si mantiene il file costantemente ordinato alfabeticamente, si prende la prima lettera di ogni record, e così:

record da «1» a «d» -> minifile n. 1
record da «e» a «h» -> minifile n. 2
record da «i» a «o» -> minifile n. 3
record da «p» a «r» -> minifile n. 4
record da «s» a «z» -> minifile n. 5

Vi do un minuto di tempo per trovare un inconveniente che comporta questa suddivisione alfabetica... 30 secondi... 15... vabbè, ve lo dico io: i 3600 record dichiarati difficilmente potranno essere raggiunti. Infatti, se inserite ad esempio tanti nomi iniziati con «s» il minifile n. 5 si riempirà, ed arrivato alla soglia dei 720 record si fermerà, anche se gli altri file sono ancora mezzi vuoti. È per questo che non ho diviso in megafile in 3 minifile, che avrebbero avuto un limite dichiarato di $720 \times 3 = 2160$ record, che poi si sarebbe ridotta a troppo poco...

Con 5 minifile, invece, il limite dichiarato sta bello alto, quota 3600, e noi ci dovremo fermare (secondo i miei calcoli, naturalmente approssimati) intorno a quota 2700. Chi possiede più di 2700 programmi non si metta le mani ai capelli: se si è un poco abili nel programmare si può modificare il programma, in modo che divida il file in 10, 20, 30 pezzi (io non l'ho fatto perché non mi serviva).

Vediamo ora una cosa fondamentale per la comprensione del programma: in ogni

singolo file relativo i record vengono scritti in ordine di immissione; per poterli poi mostrare in ordine alfabetico si usa un array «indice» che punta alle varie posizioni dei record nel file. Per non confonderci, diciamo che logicamente il file è ordinato (perché ai nostri occhi appare così), fisicamente no. È come avere un libro in cui sono disposte alla rinfusa le pagine. Sarà ancora leggibile se noi mettiamo all'inizio un indice con scritto:

- la 1ª pagina che devi leggere è pagina 14 (cioè la 14ª in ordine);
- la 2ª invece si trova in 4ª posizione;
- la 3ª, per ironia della sorte, è proprio la terza;
- la 4ª è invece a pagina 17; e così via...

Quanto detto finora serve da introduzione. Dopo le istruzioni del programma scenderemo più nei dettagli (per chi è forte di nervi), sempreché gli amici di MC non riterranno opportuno tagliare qualche parte.

Istruzioni per l'uso

Per poter lavorare con Listing Plus bisogna innanzitutto creare l'archivio. Ciò si fa tramite il programma File Creator.

Questo chiede innanzitutto il nome del file da creare, dopodiché attende l'inserimento di un record per ogni minifile che verrà creato. Bisogna dunque inserire un nome compreso tra «l» e «d», poi uno tra «e» e «i» e così via.

Questo procedimento è necessario poiché, per ragioni di velocità, il programma Listing Plus non è preparato (psicologicamente!) a trovare un file vuoto: deve esserci almeno un record per ogni minifile. Questo si riflette anche sulla cancellazione di record: non se ne può cancellare uno se è l'unico del suo «gruppo».

Comunque, una volta inseriti i nomi e confermato il tutto, non vi resta che attendere qualche attimo ed il file è pronto. Per lavorarci su potete ora caricare e far partire Listing Plus.

Come al solito, la prima domanda posta è il nome del file. Dopo averlo immesso, si deve attendere la lettura del file indice (quello seq). Se il file è molto lungo tale lettura potrà occupare un po' di tempo...

Si giunge così al menu principale, che propone 5 opzioni:

- Stampa lista
- Inserimento record
- Ricerca & variazione
- Aggiornamento file indice
- Fine lavoro.

La prima opzione stampa la lista. Viene chiesto innanzitutto un messaggio da stampare nell'intestazione (non più di 40 caratteri); immettendo la freccetta a sinistra si ritorna al menu. Prima di premere RETURN assicuratevi che la stampante sia accesa e la carta pronta. A proposito di stampante: deve essere una MPS 802, altrimenti la spaziatura «stretta» non funziona.

Inizia dunque la stampa dell'intestazione, che va naturalmente personalizzata durante la battitura del listato (a meno che non vogliate stampare liste col mio nome sopra!). La stampa si ferma ogni tanto, ma non preoccupatevi: è il programma che legge dal disco quello che deve stampare.

Se c'è bisogno di cambiare pagina, il programma lo chiede ed attende la pressione di un tasto. Alla fine si ritorna al menu principale.

Vorrei far notare che, se si usano fogli singoli, non si deve lasciare molto spazio di sopra, altrimenti alla fine la stampante si blocca segnalando che la carta è finita. Casomai succedesse, non preoccupatevi: cambiate il foglio, premete il pulsante sulla stampante, e quando sarà il programma a chiedervi di cambiare pagina limitatevi a premere un tasto.

La stampa è molto compatta: avviene su 3 colonne, con 105 linee per ogni pagina... più stretto di così ci vuole la lente d'ingrandimento... tanto per avere un'idea la mia lista giochi (1208 programmi) prende 4 pagine.

L'opzione inserimento serve, appunto, per inserire i record. Compare un numero progressivo che indica il numero di record del file, ed una sfilza di due punti. Se si preme solo RETURN si ritorna al menu. Altrimenti si batte il nome, poi RETURN e quindi il codice, seguito anch'esso dal RETURN.

Se per nome si batte «\$» (il simbolo del dollaro) verrà visualizzata la directory. Per far ciò ho utilizzato la routine in LM di Bert Bunse (nel programma File Rescue, MC 64). Spero che non me ne voglia...

Attenzione: si deve battere «\$» mentre c'è ancora il dischetto col file; dopo la richiesta da parte del computer si inserisce il disco di cui si vuol vedere la directory e poi, se non avete altre directory da guardare, si deve reinserire il dischetto con il file (che viene chiamato «master») prima di inserire nuovi record.

Non ho messo la visualizzazione della directory nel menu principale perché serve, generalmente, ai fini dell'immissione. Comunque è possibile (se avete seguito bene le istruzioni) guardarsi la directory senza immettere nessun record (Soluzione: si batte «2» per inserire record, poi «\$» e RETURN, quando ricompare il cursore RETURN di nuovo).

La terza opzione, ricerca & variazione, è

quella forse più usata. Viene mostrata una maschera video contenente 11 record. Quello centrale è in bianco perché si lavora su quello. Premendo i tasti funzione si accede a varie... funzioni:

- f1 - avanza di un record
- f2 - va all'ultimo record
- f3 - retrocede di un record
- f4 - va al primo record
- f5 - cerca un record dato il nome
- f6 - va al numero di record dato
- f7 - per variare il record corrente
- f8 - per cancellare il record corrente
- ret - torna al menu
- P - stampa gli 11 record presenti sullo schermo
- ? - mostra l'help.

Non credo ci siano difficoltà, soprattutto per chi ha una certa familiarità col SuperBase. Va notato solo che f7 permette di variare anche il NOME del record, cosa che il SuperBase non permette. L'ordinamento viene effettuato automaticamente. A proposito, nome e codice vanno variati insieme e si preme RETURN una sola volta (non come nell'immissione, dove si deve immettere prima il nome e poi il codice).

Come ho detto prima, se si tenta di cancellare l'unico record di un minifile, apparirà un messaggio di diniego («Impossibile cancellare»).

La funzione di «aggiornamento indice» serve per evitare che uno scherzo dell'elettricità vi distrugga l'archivio. Infatti, quando cancellate un record o ne inserite uno nuovo l'indice subisce delle modifiche. Se manca l'elettricità prima che abbiate memorizzato l'indice vi ritroverete con un indice vecchio (quello che c'era sul disco) e tutto il resto nuovo (perché la cancellazione è avvenuta fisicamente sul disco). L'archivio non è comunque completamente perduto: ci saranno solo alcuni record... scomparsi o fuori posto. Per questo vi consiglio:

- 1) di fare una copia dell'intero dischetto prima di lavorare sul file
- 2) aggiornare l'indice appena avete finito il lavoro.

Naturalmente ciò vale solo per le zone a rischio...

«Fine lavoro» serve per tornare al Basic. Se c'è bisogno, il programma aggiorna automaticamente l'indice. Se non c'è bisogno, non lo fa. Comunque sia, fidatevi.

Andiamo ora a vedere cosa fa il programma «Copy file». Come abbiamo detto, il file è ordinato logicamente, mentre fisicamente due record adiacenti (logicamente) possono distare mille miglia. Se ci sono molti record, la testina può essere costretta a fare su e giù per l'intero dischetto. Ora, certo non ci importa che la testina si affatichi..., ma il programma ne viene un po' rallentato. Ragion per cui, ogni tanto, è bene fare una copia del file con «Copy file». Questo programma serve a copiare il file ma, mentre lo fa, ordina anche fisicamente i record. Riprendendo l'esempio del libro, questo programma copia il libro mettendo però le pagine al posto giusto.

Prima di fare la copia di un file lungo (diciamo da 1000 record in su) vi racco-

mando di premunirvi di qualche tazza di caffè... In verità il programma è piuttosto lento, ma non scoraggiatevi: la copia va fatta una volta ogni tanto.

Vi rammento che una copia «normale» (cioè senza riordino dei record) va fatta con un qualsiasi copiatore che copi tutto il disco. I copiatori di singoli file non sono infatti capaci di raccapezzarsi coi file relativi (almeno tutti quelli che ho visto io).

Credevo di aver finito, quando mi sono ricordato che devo spiegarvi come funziona il codice. Be', il codice è una stringa qualsiasi di 8 caratteri. Il programma considera i primi tre caratteri come la lunghezza del programma, e gli ultimi 3 come il numero di dischetto dove si trova il programma. Ad esempio:

COLOSSUS 4.0 148*c197

indica che questo programma occupa 148 blocchi del disco numero 197. I due caratteri nel mezzo sono arbitrari, io li interpreto: se c'è l'asterisco il programma è registrabile su cassetta, altrimenti no; la lettera seguente indica che si tratta di un programma buono (c) discreto (b) o scarso (a). A proposito della lunghezza, se il programma occupa una facciata o più io uso l'espressione:

MUSIC CONSTR.SET 1 f b000

A che serve tutto ciò? Il programma Listing Plus calcola (in base ai codici) la lunghezza media dei programmi; indica anche il numero dell'ultimo dischetto immesso.

Voi potete comunque usare il codice che più vi aggrada: basterà che ignoriate i messaggi «lunghezza media» e «ultimo dischetto».

Lo show finisce qua (per ora). Prima di leggere il commento al listato, vi consiglio di consultare il vostro medico di fiducia!

Commento al funzionamento

Quando si deve commentare un programma di 42 blocchi non si sa proprio da dove partire... Io direi di riprendere il discorso che facevo all'inizio, con più dovizia di particolari.

Per non confonderci, prendiamo un singolo file relativo, con record che vanno da «1» a «d» compresi, e contenente 5 record. Fisicamente, cioè sul dischetto, tali record sono scritti, ad esempio, così:

1 chess 7.0	1 f b000
2 bmx simulator	191*b221
3 ant attack	248 b089
4 champ wrestling	2 f c000
5 ballblazer	183*b112

In questa condizione noi avremo un indice:

3 - 5 - 2 - 4 - 1.

Questo significa che il primo record logico (cioè in ordine alfabetico) è il numero 3, il secondo il numero 5, il terzo il numero 2, e così via. Inseriamo ora un sesto record, che viene scritto di seguito agli altri:

6 back to future 151*b171

Questo record va ad inserirsi (logicamente) tra il primo e il secondo. L'indice diverrà quindi:

3 - 6 - 5 - 2 - 4 - 1.

Quando si inserisce un record bisogna quindi:

- 1) cercare la posizione alfabetica (= logica) giusta nel file;
- 2) far slittare avanti l'indice di tutti i record seguenti logicamente;
- 3) inserire il numero della posizione fisica nell'indice;
- 4) scrivere il record nell'ultima posizione fisica.

Passiamo alla cancellazione. Ad un primo sguardo, per far scomparire un record basta far slittare indietro l'indice di tutti i record seguenti logicamente. Mettiamo di voler cancellare il quarto record logico: l'indice diverrà

3 - 6 - 5 - 4 - 1.

Il quarto record logico (secondo fisico) è scomparso.

A lungo andare, questo metodo causa però un accumulo di spazio inutilizzato sul disco. La posizione fisica numero 2, infatti, resterà vuota, nel nostro esempio. Per aggirare il problema, ho pensato di copiare l'ultimo record fisico nella posizione del record cancellato; così si libera l'ultima posizione, che può essere riutilizzata. Fisicamente, il file diverrà quindi così:

1 chess 7.0	1 f b000
2 back to future	151*b171
3 ant attack	248 b089
4 champ wrestling	2 f c000
5 ballblazer	183*b112

Naturalmente l'indice cambierà, e sarà:

3 - 2 - 5 - 4 - 1.

In questa maniera il prossimo record immesso andrà ad occupare la sesta posizione fisica, senza spreco di spazio.

Vediamo ora la variazione di un record. La variazione, dal nostro punto di vista, può essere di due tipi: può lasciare immutata la posizione del record nell'ordine alfabetico, oppure può mutarla.

Nel primo caso, basta scrivere il record modificato sul vecchio: non c'è bisogno di modificare l'indice, dato che il record occupa la stessa posizione logica.

Nel secondo caso, invece, la situazione si fa più complicata. Potrei addirittura variare la prima lettera del record, e questo magari deve andarsene in un altro minifile. La soluzione che ho trovato io è questa: si cancella il vecchio record e poi si immette il record modificato come se fosse nuovo.

Se non vi è ancora evaporato il cervello, io andrei a precisare come tutto quello spiegato sopra avviene nel programma. Sottolineo che quanto detto può essere utilizzato in un qualsiasi programma di data base (infatti io ho scritto altri due programmi con lo stesso metodo).

Nel programma Listing Plus l'indice è costituito dall'array LO%(5,720). LO sta per logic. 5 è il numero di minifile, per

ognuno dei quali ci dev'essere un indice. 720 è il numero massimo di record che entrano in un minifile. Per sapere qual è l'ultimo record fisico senza scandire tutto l'indice, nel caso si dovesse ricopiarlo (come detto prima), si consulta l'array LA%(6), che indica il numero logico dell'ultimo record fisico (... spazio per riflettere ...). LA sta per last. Se riprendiamo gli esempi di poco fa, LA%(1) indicherebbe prima 2, e dopo la cancellazione 3 (perché prima era il secondo record logico «ballblazer» ad occupare l'ultima posizione del file, dopo l'inserimento era sempre il secondo «back to future» e dopo la cancellazione il terzo «di nuovo ballblazer» perché il secondo si era trasferito nella posizione lasciata vuota dal record cancellato). Notate che nella cancellazione abbiamo avuto bisogno di conoscere qual era l'ultimo record fisico, che è indicato appunto da LA.

I record sono numerati progressivamente. Se prendiamo un record qualsiasi, diciamo il numero n, in quale minifile si troverà? Non possiamo esaminare la prima lettera, perché sappiamo solo il numero! Bene, l'array PR%(6) indica qual è il numero del primo record di ogni file. PR%(1) indicherà sempre 1; PR%(6) corrisponderà invece al numero totale di programmi più 1. Se PR%(2) contiene 100, significa che i record di numero sino a 99 compreso si trovano nel primo minifile, da 100 in poi nel secondo o in uno seguente.

Per sapere poi il numero logico del record, all'interno del suo minifile, basta sottrarre dal suo numero «generale» quello del primo del suo file. Il record 125, ad esempio, sarà il numero 25 del secondo file (sempreché PR%(3) sia maggiore di 125). Il record 100 sarà invece lo... zeresimo del secondo file (dato che si comincia a contare da 0).

Per ogni minifile, si deve anche sapere il numero di record che lo compongono (quando si immette un nuovo record, lo si va a mettere nella prima posizione fisica libera che, non essendoci spazi vuoti dentro il file, corrisponde al numero di record presenti). A questo ci pensa NP%. Notate che

$$NP\%(i) = PR\%(i+1) - PR\%(i)$$

per i che va da 1 a 5.

Questa opposizione logico-fisica è un po' contorta per essere gestita dal programma principale. Questo funziona quindi guardando solo l'ordine logico dei record, e alla scrittura o lettura di questi (operazioni che implicano un «approccio» fisico) ci pensano delle subroutine (linee 2830-3480).

Una volta chiari (!?!?) i pilastri portanti di Listing Plus risulterà più semplice (!?!?!?) capire il resto del funzionamento.

Elenco delle variabili principali

- NR:** numero totale dei programmi;
- UD:** numero dell'ultimo dischetto immesso;
- SL:** somma delle lunghezze di tutti i programmi;
- LO%(5,720):** indice;
- PR%(6):** numero del primo record per ogni minifile;
- [PR%(1) = 1; PR%(6) = NR + 1]**
- LA%(6):** numero logico dell'ultimo record

- fisico per ogni minifile;
- [LO%(i,LA%(i)) = NP%(i)];**
- NP%(6):** numero dei record per ogni minifile;
- [NP%(i) = PR%(i+1) - PR%(i)];**
- VR\$(11):** gli 11 record mostrati in ricerca & variazione;
- [VR\$(6) = record corrente];**
- X\$(360):** buffer per la stampa;
- PS:** contiene «p» + chr\$(2), per abbreviare la scrittura;
- SS\$:** contiene uno spazio shiftato per non scrivere sempre chr\$(160);
- FIS:** nome del file corrente;
- CU:** numero del record corrente (in bianco in ric.&var.);
- DT\$:** messaggio da stampare;
- LD:** linee disponibili per pagina;
- CN:** primo record della pagina corrente;
- FL:** 1 se è l'ultima pagina;
- F2:** 1 se vanno riletti i VR\$;
- AI:** 1 se va aggiornato l'indice;
- LL:** ultimo minifile utilizzato, lasciato aperto.

Elenco delle subroutine

- 370:** stampa lista;
- 920:** inserimento;
- 1080:** mostra directory;
- 1200:** ricerca & variazione;
- 1370:** stampa record attuali;
- 1460:** mostra l'help;
- 1640:** record seguente;
- 1730:** primo record;
- 1800:** record precedente;
- 1900:** ultimo record;
- 1970:** ricerca record;
- 2040:** numero record;
- 2090:** legge VR\$ da CU-5 a CU+5;
- 2160:** variazione;
- 2280:** cancellazione;
- 2340:** lettura indice;
- 2490:** legge canale errori;
- 2520:** scrittura indice;
- 2680:** maschera video 1;
- 2720:** maschera video 2;
- 2780:** pulizia schermo;
- 2830:** legge record (in:rc, out:z1\$);
- 2940:** variazione record (in:rc,z1\$);
- 3040:** scrive nuovo record (in:z1\$, out:rc);
- 3250:** cancellazione record (in:rc);
- 3490:** cerca record (in:z1\$, out:z0);
- 3660:** scrive routine LM per directory.

Il «cuore» del programma si trova alle linee 10-360. Qui c'è infatti l'inizializzazione delle variabili, il menu principale e le varie chiamate a subroutine.

Si inizializzano dapprima le variabili principali (90-100). POKE 650, 128 abilita l'autorepeat su tutti i tasti. Viene chiesto quindi il nome del file, memorizzato in FIS. Si legge l'indice (sub 2340) e si riempie la variabile VR\$(11) con i primi record del file (sub 2090). Dalla linea 170 si mostra il menu principale. POKE 204,0 serve a far lampeggiare il cursore, e POKE 204,1 a farlo star fermo. In base al carattere premuto vengono chiamati i vari moduli che compongono il programma.

La variabile AI viene settata ad 1 quando viene modificato l'indice. Se si sceglie l'opzione Fine, AI viene quindi consultato per sapere se bisogna aggiornare l'indice (sub 2520). Il main program si ferma alla linea

310. Le linee 320-340 servono per evitare un ?return without gosub error in 2500 (cioè dove va a finire la sub 2520).

Proseguiamo la nostra esplorazione. Il blocco 370-910 serve alla stampa della lista. Alla linea 410 si chiede il messaggio da stampare (si usa l'input dal device 0 per evitare il punto interrogativo). Vengono quindi aperti i canali di comunicazione con la stampante. Notate che, qui come altrove, il programma non è «proof error», e se lasciate la stampante spenta il programma andrà probabilmente in «crash».

I canali aperti sono due: 14 per la stampa e 6 per settare l'interlinea. Il canale 6 è una peculiarità della MPS 802; se avete un'altra stampante dovreste quindi eliminare tutte le istruzioni riferite a questo canale, e sostituirle se potete con quelle adatte alla vostra stampante (altrimenti vi vien fuori una cosa non tanto estetica).

Le linee 480-490 scrivono due volte la stessa scritta, in modo che anche col nastro usato questa venga scura. Naturalmente durante la battitura dovreste inserire il vostro nome e cognome in queste righe. Nelle linee 550-570 va invece l'indirizzo.

Nelle linee 710-810 avviene la vera e propria stampa. Prima si leggono tutti i record che entrano in una pagina del dischetto, poi si stampano a tre per riga. Il flag FL viene settato a 1 se si sta stampando l'ultima pagina. Alle linee 830-890 si stampano infine le ultime righe e si chiudono i canali.

Il blocco 920-1190 serve per l'inserimento. Il funzionamento è abbastanza semplice. Dopo l'inserimento AI viene settato ad 1 e così anche F2. La sub 3040 si incarica di scrivere il record al posto giusto, sapendone il nome.

Il pezzetto 1080-1170 mostra la directory. Viene prima chiuso il canale rimasto aperto (minifile n. LL), poi si richiede l'inserimento nel drive del dischetto da leggere e poi di quello col file.

Dalla linea 1200 in poi si articola il modulo di «ricerca & variazione» o, più brevemente, di edit. Come detto la prima cosa che si fa è leggere gli 11 record attuali, se F2 è impostato ad 1. Vengono quindi mostrati questi record e si attende un comando (1230-1350). Dalla linea 1370 in poi ci sono le subroutine che svolgono le varie funzioni.

Da 1370 a 1450 si stampano gli 11 record attuali. Ricordate cosa ho detto a proposito del canale 6. La sub 2040 salta al numero di record specificato e legge tutti i record attuali. Attenzione: alla linea 2090 si arriva da vari posti (160, 1220, 2080).

Attenzione anche alle linee 2210-2220: se AS (il nome del record modificato) mantiene immutata la sua posizione nell'ordine, si fa una scrittura «semplice» (linea 2230) altrimenti si cancella il record e si scrive come nuovo quello modificato (linee 2250-2270). Il confronto, per ovvi motivi di velocità, viene fatto tra i VRS, e lascio a voi il piacere di scoprire che legame c'è tra le linee 2100-2110 e 2210-2220.

Nelle linee 2340-2510 avviene la lettura dell'indice memorizzato sul dischetto col suffisso «.0». La struttura del file indice è la seguente:

NR, SL, UD, PR%(1), LA%(1), PR%(2), LA%(2), PR%(3), LA%(3), PR%(4), LA%(4), PR%(5), LA%(5), LO%(1,0..NP%(1)-1), -1,

LO%(2,0..NP%(2)-1), -1, LO%(3,0..NP%(3)-1), -1, LO%(4,0..NP%(4)-1), -1, LO%(5,0..NP%(5)-1), -1, EOF.

Vi lascio il piacere di decifrare anche questo pseudocrittogramma. Sappiate anche che viene scritto su disco nelle linee 2520-2670.

La routine 3660 scrive il linguaggio macchina alle locazioni 49152-49265. Come ho già detto la routine è copiata pari pari da «File rescue» di Bert Bunse, che ringrazio e saluto.

Ed andiamo a commentare le subroutine «fisiche», cioè quelle che vanno da 2830 a 3650 e sono incaricate dell'interfacciamento con il disco. Queste routine ricevono in input certi valori, e ne danno altri in output, come ho indicato tra parentesi nell'elenco. Ricordo i comandi DOS che vengono utilizzati per trattare i file relativi:

PRINT # 15, «p» chr\$(2) chr\$(n-int(n/256)*256) chr\$(n/256) chr\$(0):

posiziona la testina sul primo carattere del record fisico numero n;

INPUT # 2,a\$: lo legge;
PRINT # 2,a\$: lo scrive.

Tutte le routine «fisiche» lasciano aperto il canale 2 sull'ultimo minifile su cui si è operato; se ad esempio si è letto un record del minifile numero 1, il canale 2 rimarrà aperto su questo file. Così eventuali successivi accessi allo stesso file (molto probabili) sono velocizzati. Il numero di file aperto è memorizzato nella variabile LL. Se in seguito si richiede l'accesso ad un altro file, il canale 2 viene chiuso, e riaperto settando LL al nuovo valore. Questo è quanto avviene alle linee 2900 e 3000. In alcune parti del programma si setta LL a 0, quando non c'è alcun file rimasto aperto (come dopo la lettura di una directory).

Detto questo, esaminiamo le singole routine. La 2830 legge un record. Innanzitutto controlla che il numero di record (RC) sia nel range lecito. Poi individua (tramite il confronto con PR%) a quale minifile appartiene. Viene quindi posizionata la testina e letto il dato (Z1\$).

La sub 2940 scrive Z1\$ sul record numero RC. È quasi identica alla 2830, salvo che alla fine si scrive, anziché leggere, il dato. Questa subroutine è usata per la variazione «semplice», cioè che mantiene inalterata la posizione del record nell'ordine alfabetico. Viene anche usata dalla 3250 per ricopiare l'ultimo record fisico nella posizione liberata.

La sub 3040 (scrittura record) è un po' più complessa. Si usa innanzitutto la sub 3490 (cerca) per individuare la posizione «assoluta» che Z1\$ andrà ad occupare (Z0), e il minifile a cui apparterrà (Z1); viene quindi incrementato il numero di record presenti in quel file (NP%(Z1)), il numero totale (NR) e si slittano avanti tutti i PR% seguenti (3040-3100). Dopo ciò si calcola la lunghezza (Z8) e se il numero del disco è maggiore di UD. Arrivati alla linea 3160 si calcola la posizione «relativa» del record all'interno del suo minifile (Z5) e si slitta avanti l'indice da quella posizione in poi. Non rimane quindi che scrivere il record fi-

sicamente nell'ultima posizione del proprio file (che è la prima disponibile), non prima di aver fatto puntare l'indice LO% a quella posizione (3200-3240).

La sub 3250 presiede alla cancellazione di un record. Dapprima legge il record condannato, per sottrarre la lunghezza da SL. La sub 2830 (lettura record) dà anche il numero del minifile del record (Z7, copiato in Z1 alla linea 3300). Se il record è l'unico del suo file la cancellazione non può essere eseguita (3310-3320). Alla linea 3340 si legge l'ultimo record fisico del minifile, e alla 3360 lo si fa scrivere sopra il record cancellato (tramite la sub 2940, che non tocca l'indice). Si fa quindi slittare indietro l'indice (3370-3390) e PR% (3410-3430); si scalano i puntatori (3400) e si scandisce l'indice per sapere qual è ORA l'ultimo record fisico del minifile (3440-3470).

La sub 3490 ricerca un record dato il nome e, se non lo trova, dà il numero di quello successivo in ordine alfabetico. In base alla prima lettera si stabilisce in quale minifile si deve cercare (3510-3580). Si effettua poi la ricerca dicotomica su questo file (3580-3640). Ricordo che la ricerca dicotomica è la più usata su file già ordinati (non può essere utilizzata su file non ordinati) ed è molto veloce.

File Creator

Il programma File Creator non fa altro che leggere cinque record, registrarli su disco (uno per ogni minifile) e scrivere l'indice, che naturalmente è brevissimo.

Attenzione alla linea 440: per creare un file relativo ex novo la sintassi è:

OPEN 2, 8, 2, «0:nome,1,»+chr\$(lunghezza record)

Nel nostro caso il record è lungo 26 caratteri:

11111111	1	11222222	2
123456789-	7	89012345	6
0123456			
nome	return	codice	return

Alle linee 340-410 viene scritto l'indice.

Copy File

Il programma, schematicamente, svolge le seguenti operazioni:

- legge l'indice (linee 260-330);
- scrive l'indice (350-430);
- ripete per ogni minifile (5 volte);
- legge file in ordine logico (460-530);
- lo scrive (560-630);
- fine.

L'unica cosa particolare da notare è nella scrittura dei file relativi: viene dapprima scritto l'ultimo record, poi tutti quelli dal primo al penultimo. Perché? Semplice: dando il comando di posizionarsi sul record numero NP%(i) (l'ultimo), il DOS crea lo spazio per tutti i record intermedi; così quando si scrivono questi ultimi lo spazio è già pronto. Se invece si parte dal primo e così via fino all'ultimo, il DOS è costretto a creare ogni volta lo spazio per un record, poi per un altro e un altro ancora, e ci mette quasi il doppio di tempo (provato cronometro alla mano).

