

CATALOG

di Fulvio Peruggi - Qualiano - (NA)

Questo programma sarà utile a tutti coloro che hanno l'abitudine o la necessità di mantenere numerosi programmi caricati in macchina. Siccome la memoria (nostra) non è perfetta, e può capitare di non avere gli appunti o i listati a portata di mano, spesso è necessario spulciare la memoria (dello SHARP) col LIST e coi tasti ↑, ↓ per ricordare cosa avevamo messo in macchina, dove sta, e magari cosa fa. CATALOG risolve tutto con la pressione di due soli tasti. Esso cerca la prima linea di ogni programma, ne mostra il numero, e visualizza l'etichetta, il nome, ed eventuali commenti da noi precedentemente associati al programma in questione. Il tutto avviene con un'occupazione di circa 150 byte di RAM, e con la necessità di una banale codifica da parte dell'utente. Quest'ultima è illustrata dalla riga 10 dello stesso CATALOG: la prima riga di ogni programma deve essere identificata con un'etichetta vuota "" (due virgolette adiacenti), seguita da un REM dopo il quale si può specificare tutto ciò che si vuole far visualizzare da CATALOG. La vera etichetta del programma (se c'è) va messa all'inizio della linea successiva. Nel nostro caso CATALOG parte con <DEF> C e quando si autoleggerà mostrerà sul display:

```
10 "C" *** CATALOG *** by
Fulvio PERUGGI
```

Il funzionamento del programma è chiarito dal diagramma di flusso in figura 2. Il valore assegnato ad S in riga 20 vale per le macchine con modulo di espansione 8K RAM; in generale esso si può calcolare mediante l'espressione:
256*PEEK 30821+PEEK 30822

La scrittura carattere per carattere, se riempie il display, lo cancella automaticamente e ricomincia da capo: regolate a vostro piacimento la velocità di presentazione modificando l'argomento del WAIT in riga 20. Il secondo IF in linea 40 sostituisce un AND ed è stato usato per evitare un continuo (ed inutile) doppio confronto, con un risparmio di circa il 10% sul tempo di elaborazione. Se un'etichetta vuota non è seguita da un REM, CATALOG funziona lo stesso, ma la riga potrebbe essere decodificata in modo poco chiaro (attenzione a non far seguire "" da un'altra etichetta, perché ciò va bene per CATALOG, ma all'atto del RUN di quel programma provocherà un ERROR 11 o 21). Il gran numero di etichette vuote non ha alcun effetto negativo sul comportamento del PC-1500: comandi o istruzioni del tipo LIST "" o GOTO "" vedranno sempre e solo la prima coppia "" in memoria.

Concludo saltando ad un argomento completamente diverso, e cioè una proprietà, forse non molto nota, dell'istruzione RND. Il manuale ci spiega che essa genera solo numeri casuali interi, per cui si può pensare che siano necessari sottoprogrammi del tipo:

```
500 B=1E10-1:A=RND B
510 PRINT A/B:RETURN
```

per generare numeri casuali con 10 cifre significative compresi fra 0 (escluso) e 1 (incluso). Tutto ciò non è necessario perché la macchina accetta l'istruzione RND 0 che genera numeri casuali con le stesse caratteristiche.

```
10 ""REM "C" *** CATALOG *** by Fulvio PERUGGI
20 "C" S=14533:WAIT 3:USING
30 P=PEEK S:T=S+3+PEEK (S+2):IF P=255 END
40 IF PEEK (S+3)=34 IF PEEK (S+4)=34 GOSUB 60
50 S=T:GOTO 30
60 CLS:PRINT P#256+PEEK (S+1);" ";
70 FOR I=S+7 TO T-2:PRINT CHR$ PEEK I;:NEXT I
80 RETURN
```

Figura 1 - Listato del programma CATALOG.

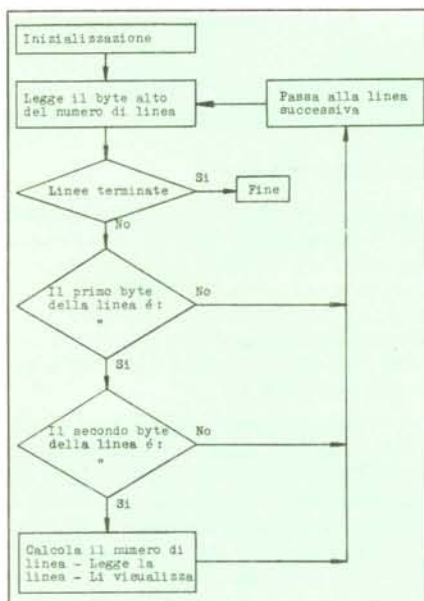


Figura 2 - Diagramma di flusso del programma CATALOG.

COMPACT

di Lorenzo Spoladore - Rovigo

Talvolta, nel corso di un programma, capita di dover rappresentare sul display del PC-1500 lunghi messaggi oppure menu particolarmente "nutriti". Quando, però, finalmente facciamo girare il programma, ci si accorge, con un po' di dispetto, che la scritta viene troncata inesorabilmente dal margine destro del visore.

Che fare? Ci sono due alternative: o si

ricorre ad abbreviazioni, spesso incomprensibili, soprattutto se il programma viene ripreso a distanza di tempo o viene utilizzato da altri operatori, oppure ci si serve della routine COMPACT. Questo programmino serve, infatti, a "compattare" tutte le scritte, che compaiono sul display tramite le istruzioni del tipo PRINT.

COMPACT non utilizza alcuna biblioteca di nuovi caratteri; si serve, infatti, di quelli standard, opportunamente modificati.

I caratteri del PC-1500 sono contenuti in una matrice di 7 x 6 punti. Le prime 5 sono effettivamente utilizzate, la sesta è bianca e serve a separare i caratteri contigui. Ora, 6 colonne per carattere si possono rivelare, nelle circostanze descritte sopra, un vero e proprio spreco di spazio.

Si può cominciare, allora, col "sopprimere" la colonna bianca, ma, poi, ci si domanda perché non continuare, togliendo ancora una o due colonne. Dopo qualche tentativo, ci si convince che si può fare a meno della terza colonna, oppure della seconda e della quarta, senza compromettere la leggibilità dei caratteri. In verità, se si taglia la terza colonna, per esempio, della lettera T maiuscola, cioè la sbarretta verticale, il carattere perde completamente di significato. È, allora, evidente che COMPACT deve porre rimedio a questo genere d'inconvenienti.

Esistono diversi formati:

FORMATO 2, toglie la 2a, la 4a e la 6a colonna di quasi tutti i caratteri. Rappresenta bene le lettere maiuscole e i numeri.

FORMATO 3, toglie la 3a e la 6a colonna. Va bene per maiuscole e minuscole.

FORMATO 6, toglie da solo la 6a colonna di ogni carattere. Va bene per ogni applicazione.

FORMATO 0, è quello normale.

Come già accennato, le colonne sono tolte solo se non pregiudicano la leggibilità dei caratteri. Così, il FORMATO 3 toglie solo la colonna bianca della T, mentre il FORMATO 2 lascia al suo posto la 2a colonna della D.

Il programma decide analizzando ogni colonna da togliere attraverso la funzione POINT. Quindi, se il valore d'uscita è uno di quelli che non verifica la condizione che compare alla riga 40085, la colonna viene "salvata". I valori corrispondono a schemi di punti, indispensabili per poter leggere un carattere. Ad esempio, 127 rappresenta la sbarretta verticale, senza la quale le lettere come la T o la I sono assolutamente irricognoscibili.

Ovviamente, per poter utilizzare l'istruzione POINT, il carattere esaminato deve essere presente sul display. Infatti, le colonne dalla 150 alla 155 sono riservate a questo scopo e non possono essere usate

dalla frase da stampare. Tuttavia, il numero di caratteri che si possono visualizzare è ancora molto elevato. Precisamente circa 28 col FORMATO 6, 35 col FORMATO 3, 49 col FORMATO 2.

Con il FORMATO 2 vengono utilizzate quasi sempre solo 3 colonne per ogni carattere, quindi la capacità del display risulta raddoppiata.

Dopo aver lanciato il programma, si seleziona il formato desiderato. Quindi, alla successiva richiesta di input, si scrive la frase da elaborare. Se la frase è troppo lunga, il programma la tronca e vi aggiunge un punto per segnalare la operazione svolta. Immediatamente dopo, il primo carattere della frase appare sulla destra del visore, mentre alla sinistra compaiono le

prime colonne. Ad elaborazione conclusa, le ultime 6 colonne del display vengono ripulite.

Naturalmente, quando COMPACT è usato come subroutine, la frase da elaborare viene caricata nella variabile WS(0) in una linea di programma e il formato è già predisposto. Inoltre, a seconda del formato usato, si può modificare la condizione di cui alla riga 40085. Ciò è utile perché, ad esempio, nel FORMATO 2 non sono molto significative le colonne il cui valore, misurato da POINT, è 120 oppure 96 o 63.

Presentiamo, ora, un'applicazione in cui l'uso di COMPACT è indispensabile. Il programma, chiamato DOMUS, serve a stabilire che peso hanno le spese per il gas, energia elettrica, telefono, acquedotto e

fitto sull'economia domestica. DOMUS calcola la percentuale di ogni voce sulla spesa totale e riporta i risultati su un istogramma orizzontale. Quindi, di seguito sul display, stampa i valori numerici delle percentuali.

Il formato usato in questa applicazione di COMPACT è il 2.1, che differisce dal 2 perché salta la prima colonna bianca (se c'è) di ogni carattere. Questa nuova possibilità è contenuta nelle linee 60130 e 60195. Con il 2.1 si possono rappresentare comodamente più di 50 caratteri.

Il flow-chart in figura 5 si riferisce al solo programma COMPACT. I blocchi contengono anche le variabili più importanti, che intervengono nel corso delle operazioni descritte.

```

40000: REM ** COMPACT **
40001: " " CLEAR:CLS:CURSOR:INPUT "FORMATO=";F:D=24
40004: IF F=2 LET D=49
40005: IF F=3 LET D=35
40006: IF F=6 LET D=28
40010: DIM WS(0)*D+1
40020: INPUT "FRASE=";WS(0):L=LEN WS(0)
40030: IF L>D LET WS(0)=LEFT$(WS(0),D)+".":L=D+1
40040: CLS:WAIT(1/N)*D:FOR Z=1 TO L
40050: BS=MID$(WS(0),Z,1)
40060: CURSOR 25:PRINT BS
40070: H=0:FOR J=1 TO 6:H=H+1:IF H>F LET H=1
40080: I=POINT(J+149)
40085: IF H=F AND I<=65 AND I<=125 AND I<=96 AND I<=120
      THEN 40100
40097: H=H+1:BEEP 1,100,1
40090: CURSOR H:GPRINT I;
40100: NEXT J
40110: NEXT Z
40110: WAIT:CURSOR 25:PRINT " "
40150: BEEP 1:END
  
```

Figura 3 - Listing programma COMPACT.

```

60000: REM ** DOMUS **
60005: " " CLEAR:CLS:CURSOR:WAIT 0:DIM WS(0)*42,S(4),P(4),M(4)
60010: INPUT "GAS=";S(0),"ENEL=";S(1),"TEL=";S(2),"ACQUA=";S(3),
      "FITTO=";S(4)
60015: CLS
60020: FOR Z=0 TO 4:S=S(Z)+S: NEXT Z
60025: REM CALCOLA LA PERCENTUALE DI OGNI VOCE
60030: FOR Z=0 TO 4:P(Z)=100*S(Z)/S:P(Z)=INT(P(Z)*10)/10:
      P(Z)=STR$(P(Z)):NEXT Z
60035: REM STAMPA L'ISTOGRAMMA
60040: FOR Z=0 TO 4:Q=0:S=0:H=0:N=0:IF U>10 LET H=1
60045: IF H=5 LET I=64
60047: IF H=10 OR Z=0 LET I=96
60050: FOR Y=0 TO 4:Z=Z+1
60060: IF P(Y)/2>Z+1 LET I=I+3
60070: NEXT Y
60075: CURSOR Z:GPRINT I;
60080: NEXT Z
60090: REM * COMPACT *
60095: REM CARICA LA FRASE
60100: WS(0)=" "+P(0)+" "+P(1)+" "+P(2)+" "+P(3)+" "+P(4)+
      ".":L=LEN WS(0)
60110: L=LEN WS(0)
60120: F=2:H=50:FOR Z=1 TO L
60130: Q=0
60150: BS=MID$(WS(0),Z,1)
60160: CURSOR 25:PRINT BS
60170: H=0:FOR J=1 TO 6:H=H+1:IF H>F LET H=1
60180: I=POINT(J+149)
60185: IF I=16 THEN 60210
60190: IF H=F AND I<=127 THEN 60220
60195: IF I=0 LET Q=Q+1:IF Q=1 THEN 60210
60200: H=H+1:BEEP 1,100,1
60210: CURSOR H:GPRINT I;
60220: NEXT J
60230: NEXT Z
60240: WAIT:CURSOR 25:PRINT " "
60250: BEEP 1:END
  
```

Figura 4 - Listing programma DOMUS.

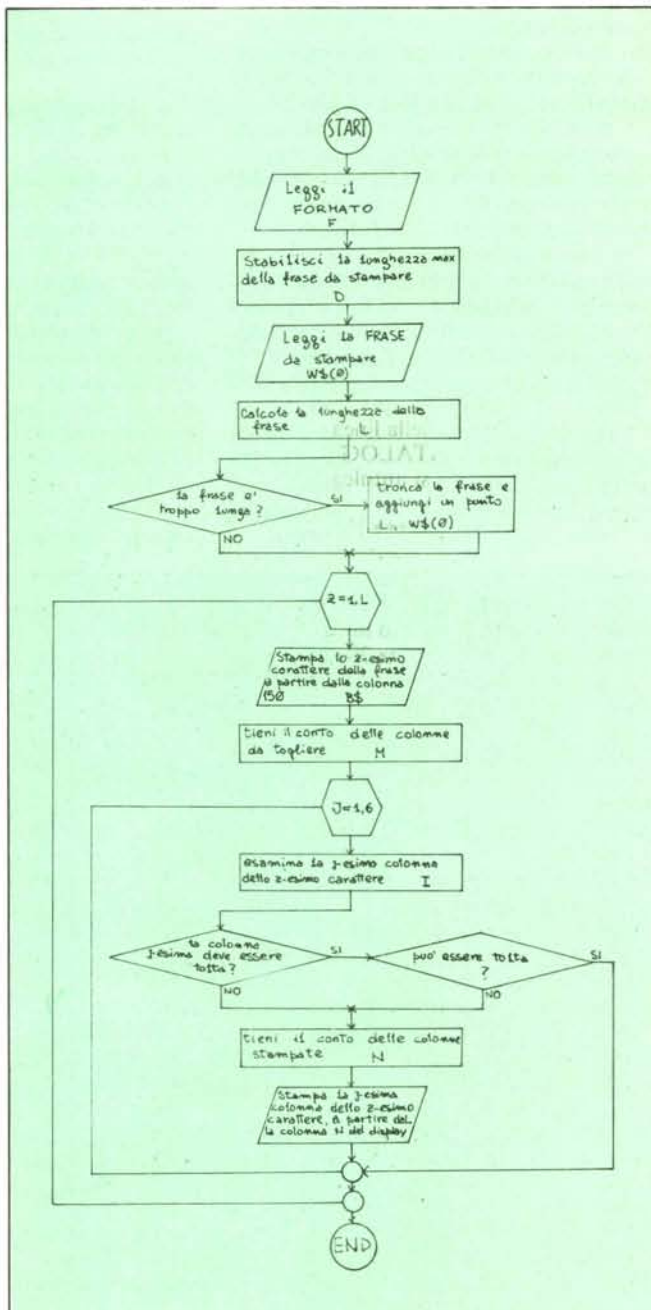


Figura 5 - Flow-chart programma COMPACT.