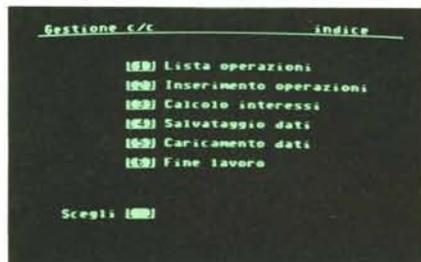


## Gestione CC

di Roberto Ceccarelli - Rimini

Ho già collaborato con la vostra rivista per il software SOA nel n. 20: grazie anche al vostro compenso, a luglio ho acquistato un Commodore 64, e visto che ve ne curate in una apposita rubrica vi invio del materiale. Il programma che propongo consente la gestione personale di un conto corrente bancario, ed utilizza la cassetta come memoria di massa. Questa versione può girare su tutti i Commodore 40 colonne — basta ignorare le due POKE per il colore in linea 50040 (leggermente più nascoste le modifiche per il VIC: si tratta dell'uscita per le 22 colonne, nonché di adattare la routine di hardcopy che parte in linea 60000, ponendo in 91 — linea 60020 —



l'indirizzo di partenza della memoria video nella propria configurazione, e alterando i cicli allo schermo 22\*23 anziché 25\*40, ndr). Le operazioni previste sono quelle di inserimento e visualizzazione, nonché salvataggio e rilettura su nastro, più il calcolo degli interessi maturati. Ho cercato di dividere il listato in blocchi per semplificare eventuali adattamenti, in particolar modo la gestione tramite l'unità a dischi.

Tra le peculiarità interessanti cito l'ordinamento per date, che consente inserimenti di conti i cui riscontri arrivano in data successiva al pagamento (bollette, etc), e il controllo sui dati, che ci informa del cambiamento del file attualmente in memoria prima che la chiusura del programma o il caricamento di un successivo file li distrugga.

Termino con alcune note sull'uso. Come richiesto dal programma, le date vanno inserite nella forma <giorno, mese>; il computo degli interessi è effettuato sulla base di più periodi (fino a 10) ognuno con il suo tasso percentuale, e la fase di input dei suddetti periodi termina solo quando si inserisce la data <31, 12>. Il calcolo non è accurato per gli anni bisestili (tra cui il 1984!), ma lo stesso listato, nelle linee 50170 e 50175, informa delle modifiche da farsi in questo caso.

\* \* \*

Il programma, che è grazioso (oltre a venire incontro ad un'esigenza quotidiana) si presta ad essere adoperato con tranquillità; visto il poco spazio occupato dai dati,

```

10 REM *****
20 REM *** GESTIONE CC ***
30 REM *** ROBERTO CECCARELLI ***
46 REM *****
48 :
50 GOSUB50000
99 :
100 REM MENU' PRINCIPALE
101 :
120 PRINT" [M] GESTIONE C/C", "INDICE":PRINT#
140 PRINT, " [M] [1] LISTA OPERAZIONI"
160 PRINT, " [M] [2] \INSERIMENTO OPERAZIONI"
180 PRINT, " [M] [3] \ALCOLO INTERESSI"
200 PRINT, " [M] [4] \SALVATAGGIO DATI"
220 PRINT, " [M] [5] \CARICAMENTO DATI"
240 PRINT, " [M] [6] \FINE LAVORO"
260 PRINT" [M] [0] \CEGLI [M] "
280 GETA$:IFA$(←"1"ORA$)←"6"←GOTO280
300 ONVAL(A$)GOSUB1000,2000,3000,4000,5000,6000
320 GOTO120
999 :
1000 REM LISTA OPERAZIONI
1001 :
1020 FORJ=0TONDSTEP16:PRINT" [M] ", "LISTA OPERAZIONI"
1040 PRINT"-----";
1060 PRINT" |DATA | CAUSALE | IMPORTO |";
1080 PRINT"-----";
1100 FORK=JTOV:IFK<NDTHENPRINT"-----";K=V:GOT
01100
1120 D$=RIGHT$(STR$(V%(K)),2)+ " +M$(INT(V%(K)/100))
1140 NN=I(K):GOSUB10000:XX=12-LEN(X$)
1160 PRINT" |D$| |LEFT$(C$(K)+SP$,18)|";TAB(POS(0)+X$)X$|";
1180 NEXT:GOSUB10100:NEXT
1200 PRINT" [M] ", "\IEPILOGO TOTALI":PRINT#
1220 NN=TA:GOSUB10000:PRINT" [M] [0] TOTALE AVERE ="TAB(30-LEN(X$))X$
1240 NN=TD:GOSUB10000:PRINT" [M] [0] TOTALE DARE ="TAB(30-LEN(X$))X$
1260 NN=TA-TD:GOSUB10000:PRINT" [M] [0] SALDO ATTUALE ="TAB(30-LEN(X$))X$
1280 GOTO10100
1999 :
2000 REM INPUT DATI
2001 :
2020 ND=ND+1:PRINT" [M] \NGRESSO OPERAZIONE N.",ND+1:PRINT#
2040 CH=1
2060 INPUT" [M] [0] \ATA (GG,MM)":G,M
2080 V%=G/M#100
2100 FORJ=0TOND:IFV%(V%(J))THENK=J:J=ND:NEXT:GOTO2500
2120 NEXT:K=ND
2140 V%(K)=V%
2160 INPUT" [M] \CAUSALE":C$(K)
2180 INPUT" [M] \MPORTO":I(K):IFI(K)<0THENTD=I(K):GOTO2220
2200 TA=TA+I(K)
2220 PRINT" [M] [0] [F1] = ALTRO DATO [F3] = MENU'"
2240 GETA$:IFA$=" "THENRETURN
2260 IFA$(←" ")THEN2240
2280 GOTO2000
2500 FORJ=ND-1TOKSTEP-1
2520 V%(J+1)=V%(J):C$(J+1)=C$(J):I(J+1)=I(J)
2540 NEXT:GOTO2140
2999 :
3000 REM VALUTAZIONE INTERESSI
3001 :
3020 PRINT" [M] ", "\ALCOLO INTERESSI":PRINT#
3040 J=1:PRINT" [M] "
3060 INPUT" [M] INTERESSE %":IN:IN(J)=IN/100
3080 PRINTTAB(20)" [M] [0] ";INPUT"FINO AL":G,M:D%(J)=G%(M)+G
3100 IFD%(J)<365THENJ=J+1:GOTO3060
3120 PRINT" [M] [0] TAB(16) "\IEPILOGO DI FINE ANNO":PRINT#
3140 NN=TA:GOSUB10000:PRINT" [M] [0] TOT. MOVIMENTO AVERE"TAB(39-LEN(X$))X$
3160 NN=TD:GOSUB10000:PRINT" [M] [0] TOT. MOVIMENTO DARE"TAB(39-LEN(X$))X$
3180 PRINTTAB(25)" [M] [0] "
3200 NN=TA-TD:GOSUB10000:PRINT" [M] [0] \ALDO AL 31-12"TAB(39-LEN(X$))X$
3220 NN=0:K=1:SA=0:FORJ=0TOND
3240 D=G%(V%(J)/100)+VAL(RIGHT$(STR$(V%(J)),2))
3260 IFD%(K)THENGOSUB3800
3280 SA=SA+I(J):NN=NN+I(J)*IN(K)/365*(366-D):NEXT
3290 GOSUB3840
3300 NN=INT(NN+.5):GOSUB10000:PRINT" [M] \INTERESSI MATURATI"TAB(39-LEN(X$))X$
3320 PRINTTAB(25)" [M] [0] "
3340 NN=TA-TD+NN:GOSUB10000:PRINT" [M] [0] \ALDO AL 1 - 1"TAB(39-LEN(X$))X$
3360 GOTO10100
3800 DI=(IN(K+1)-IN(K))*SA*(366-D%(K))/365
3820 NN=NN+DI:K=K+1:RETURN
3840 IFD%(K)=365THENRETURN
3860 GOSUB3800:GOTO3840
3999 :

```

In alcune linee appaiono dei simboli grafici, corrispondenti alle maiuscole poiché il programma si basa sul set minuscolo. Ad esempio, nelle linee 160/240 sono tali le iniziali delle voci con menu - Inserimento, Calcolo, Salvataggio etc; lo stesso in molte altre. Così per la linea 50060, che è l'intestazione del menu principale. Per aiutarvi, fate riferimento alle fotografie illustrate.



## APPEND 64

da Commodore Computing International  
dicembre 1983

Per la prima volta, (anzi la seconda, è successo una volta per lo Spectrum), sulle pagine di MC trovate una routine presa da altri giornali. L'episodio si ripeterà probabilmente in futuro, poiché a volte si trovano cose che val davvero la pena di riportare: in questo caso abbiamo una routine in LM (di cui riportiamo il list in mnemonici) con programmino caricatore in BASIC che realizza la funzione di APPEND, che consiste nell'allacciare due programmi di cui uno residente in memoria e l'altro su cassetta o su disco. Il caricatore contiene un utilissimo controllo di correttezza: in fase di inserimento dei dati li somma, e poi verifica che il risultato sia pari a quello esatto, eventualmente stampando un messaggio di errore. A differenza da quanto avviene con il MERGE, usando l'APPEND i due programmi vengono caricati in memoria uno dopo l'altro, indipendentemente dai numeri di linea che possono anche in tutto o in parte, essere gli stessi.

L'idea di base per implementare un comando di questo tipo non è complicata. Le linee di programma sono messe in memoria a partire dalla locazione indicata da PEEK(43) + PEEK(44)\*256, che senza manipolazioni è 2049. Il primo e secondo byte di ogni linea contengono il puntatore alla locazione di partenza della prossima linea; l'ultimo byte di ogni linea è sempre 0, così come quella iniziale del listato BASIC (usualmente in 2048) più i due finali del programma; dopo il programma è presente una serie di informazioni che dicono all'interprete BASIC cosa fare dopo il RUN o il LIST.

```

1 REM *****
2 REM *** APPEND 64 ***
3 REM *** COMM COMP ***
4 REM *** DIC 1983 ***
5 REM *****
10 I=49152:T=0
20 READ A: IF A=-1 THEN 50
40 POKE I,A: I=I+1: T=T+A: GOTO20
50 IF T<>9910 THEN PRINT "DATA ERROR":GOTO 90
60 PRINT "DATA OK"
70 PRINT "USE SYS 49152"CHR$(34)"FILENAME"CHR$(34)".DEV."
80 PRINT "APPEND FILENAME."
90 END
100 DATA 169, 0,133, 10, 32,212,225
110 DATA 165, 45, 56,233, 2,170,165
120 DATA 46,233, 0,169,165, 10, 32
130 DATA 213,255, 32, 51,165,165, 45
140 DATA 164, 46, 56,233, 2,133, 87
150 DATA 152,233, 0,133, 89,160, 0
160 DATA 177, 87, 208,27,200,177, 87
170 DATA 208, 22,165, 87, 24,105, 2
180 DATA 133, 45,133, 47,133, 49,165
190 DATA 88,105, 0,133, 46,133, 48
200 DATA 133, 50, 96,160, 0,177, 87
210 DATA 133, 89,200,177, 87,133, 88
220 DATA 165, 89,133, 87, 76, 40,192
230 DATA -1

```

Il caricatore in BASIC dell'APPEND 64. È molto interessante la verifica di correttezza in linea 50, basata sulla somma dei valori numerici dei codici in LM che realizzano la funzione (checksum): se il totale non è corretto viene stampato un messaggio, "DATA ERROR".

Va precisato che il programma, essendo caricato a partire dalla locazione 49152, è compatibile con il Simons' BASIC.

A questo punto per aggiungere un'altra parte a quanto sta in memoria bisogna (a) coprire la RAM, agendo sulle locazioni 43 e 44; (b) caricare il programma; (c) ricostruire in fondo al vecchio programma i

puntatori così che l'interprete veda i due pezzi come uno solo.

Tramite il caricatore, la routine viene ad essere caricata in RAM, a partire dalla locazione 49152. Le istruzioni per il carica-

```

PC SR AC XR YR SP
803E 32 00 83 00 F6

C000 A9 00 LDA #00
C002 85 0A STA #0A
C004 20 D4 E1 JSR #E1D4
C007 A5 2D LDA #2D
C009 38 SEC
C00A E9 02 SBC #02
C00C AA TAX
C00D A5 2E LDA #2E
C00F E9 00 SBC #00
C011 A8 TAY
C012 A5 0A LDA #0A
C014 20 D5 FF JSR #FFD5
C017 20 33 A5 JSR #A533
C01A A5 2D LDA #2D
C01C A4 2E LDA #2E
C01E 38 SEC
C01F E9 02 SBC #02
C021 85 57 STA #57
C023 98 TYA
C024 E9 00 SBC #00
C026 85 58 STA #58
C028 A0 00 LDY #00

```

Disassemblato del programma Append.

```

C02A B1 57 LDA (#57),Y
C02C D0 1B BNE #C049
C02E C8 INY
C02F B1 57 LDA (#57),Y
C031 D0 16 BNE #C049
C033 A5 57 LDA #57
C035 18 CLC
C036 69 02 ADC #02
C038 85 2D STA #2D
C03A 85 2F STA #2F
C03C 85 31 STA #31
C03E A5 58 LDA #58
C040 69 00 ADC #00
C042 85 2E STA #2E
C044 85 30 STA #30
C046 85 32 STA #32
C048 60 RTS
C049 A0 00 LDY #00
C04B B1 57 LDA (#57),Y
C04D 85 59 STA #59
C04F C8 INY
C050 B1 57 LDA (#57),Y
C052 85 58 STA #58
C054 A5 59 LDA #59
C056 85 57 STA #57
C058 4C 28 C0 JMP #C028

```

CARATTERI SPECIALI DEL 64  
NON DISPONIBILI SUL VIC

```

"0" CBM + 1 ARANCIO
"1" CBM + 2 MARRONE
"2" CBM + 3 ROSSO CHIARO
"3" CBM + 4 VIOLA
"4" CBM + 5 GRIGIO 2
"5" CBM + 6 GIALLO CHIARO
"6" CBM + 7 BLU CHIARO

```

mento sono le seguenti: (1) digitare il programma; (2) registrarlo su nastro o disco; (3) dare il RUN; (4) dare il NEW. A questo punto l'APPEND verrà chiamato con l'istruzione

SYS 49152 "nome pgm", unità  
ove unità vale 1 se si usa il registratore, e 8 se si usa il disco.

Il programma risiederà in memoria finché non si spegnerà la macchina (quindi niente problemi di RESTORE).

Un utile consiglio è di specificare sempre sia il "nome pgm" che l'unità perché in caso contrario può capitare che in questo modo venga caricato un programma che altrimenti sembra non esserci. **MC**

oggi, nei PC, l'alternativa è  
**Advance 86b**  
by FERRANTI INSTR.

**PERCHE' È L'UNICO  
COMPATIBILE 100% IBM PC  
HARD E SOFTWARE  
CON**



- SOFTWARE E SISTEMA OPERATIVO MS/DOS 2.X E CP/M 86
- 4 SLOT COMPATIBILI IBM PC E 2 SLOT A 16 BIT
- GARANZIA FERRANTI INSTRUMENTS SULL'INTERO SISTEMA
- DRIVES SHUGART ASSOCIATES
- GRAFICA E COLORI COMPRESI NEL PREZZO BASE
- PREZZO BASE FINO AL 40% INFERIORE AI PIU' NOTI PC DI PARI CARATTERISTICHE
- MEMORIA ESPANDIBILE FINO A 768KB CON CONTROLLO DI PARITA'
- DISPONIBILITA' DI COPROCESSORE ARITMETICO 8087
- CPU 8086 A 16 BIT A 4,77 MHZ
- COMPLETA GESTIONE DEL VIDEO
- QUATTRO PAGINE DI VIDEO
- RISOLUZIONE GRAFICA 300x200 O 640x200
- 16 COLORI DISPONIBILI
- DISK DRIVES DA 2x320K INCLUSI
- DISCO OPZIONALE DI TIPO WINCHESTER 5 1/4" DA 10 MBYTE (FORMATTATI)
- PORTA PER LA CASSETTA, PER PENNA OTTICA, PER PENNA GIOCHI (JOYSTICK)
- INTERFACCIA CENTRONICS E RS 232 C



CONDOR È PRESENTE A  
SALONE INF., 14-18/4 MILANO  
ROMAUFFICIO, 4-6/5 ROMA

CONDOR INFORMATICS ITALIA, VIA GRANCINI 8, 20145 MILANO  
TEL. 02/43.45-62-49.87.549-49.87.713, TELEX 315113

L'Advance 86b rappresenta un'implementazione del modello 86a, ancora oggi l'unico vero PC a 16 bit con un costo inferiore al milione e mezzo (IVA esclusa).

**FORTUNATI QUEI RIVENDITORI DI PC CHE  
HANNO LETTO QUESTO ANNUNCIO,  
VOGLIONO SAPERNE DI PIU' E SPEDISCONO  
QUESTO TAGLIANDO.**

NOME \_\_\_\_\_  
SOCIETA' \_\_\_\_\_  
INDIRIZZO \_\_\_\_\_  
CITTA' \_\_\_\_\_ TEL. \_\_\_\_\_

MC