

testo in una finestra (ecco come fanno i word processor!!), manipolazione della data e dell'ora! Niente male, vero? Per arrivare a 150 ce ne vuole!

Conclusioni

Microsoft Basic rappresenta, ormai, una delle implementazioni più efficaci di questo linguaggio. Fino alla versione 2.1 non era ancora stato possibile accedere in tutto e per tutto al toolbox, la scatola delle meraviglie del Macintosh. E, a dire la verità, quando ho avuto tra le mani la nuova versione sono rimasto da principio deluso ed un po' irritato; la somiglianza quasi totale tra questa versione e la precedente per quanto attiene allo standard del linguaggio è tale che ho pensato, onestamente che fosse del tutto ingiustificato il passaggio dalla 2.1 alla 3.0. Niente di più errato! Le routine e le librerie della Clear Lake, che

rappresentano la chiave d'accesso di MS Basic al toolbox (anche se alcune caratteristiche più specialistiche restano ancora inaccessibili) giustificano, in maniera indiscutibile, il varo di questa nuova edizione. Permane, ancora, la notevole lentezza del linguaggio ma il fatto che Microsoft abbia messo in vendita, pressoché contemporaneamente, un compilatore, risolve di per sé il problema.

Qualcosa sui prezzi; essi sono abbastanza modesti e comunque allineati con la concorrenza, per l'interprete. Il compilatore, invece, è molto più caro (costa quasi il doppio dell'interprete) e pur essendo inferiore a quello del True Basic, è ancora troppo alto, se si considera che un linguaggio di altissimo pregio quale lo ZBasic della Zedcor offre un compilatore-interprete, peraltro di rara efficienza, al prezzo del solo secondo. A

questo punto viene da chiedersi se non sia il caso che la Borland cominci a fare un pensiero anche nel campo del BASIC-MAC; se ne vedrebbero delle belle...

Nota

La versione 3.0 di Ms Basic ha sanato un problema deontologico ed etico piuttosto pesante, almeno per le persone oneste; chi utilizzava, in precedenza, MS Basic per produrre programmi da destinare alla vendita doveva, per forza di cose, chiedere al suo cliente di comprare un originale del linguaggio (o, più disonestamente, gli forniva una copia della sua) per poter far girare il programma da lui creato. Con 3.0 ciò non è più, visto che è possibile creare, tramite una opzione, programmi che non è più necessario siano supportati dalla presenza fisica del linguaggio.

Qualche utility di programmazione in Basic

parte prima

La prova di MS. Basic ci porge l'occasione di aprire una rubrica nella rubrica, destinata a presentare routine, utility, piccoli trucchi destinati a rendere più facile la vita del programmatore. Era fin dalla nascita di questo spazio che Marinacci aveva consigliato, con la dolcezza che gli è congeniale, di interessarmi al problema. Avevo sempre rimandato in quanto ho per diverso tempo pensato, probabilmente non a torto, che Mac, per la sua stessa filosofia, è soprattutto una macchina "enduser", vale a dire destinata a "macinare" programmi, e, solo più raramente, da utilizzare per la programmazione autonoma.

Ma il tempo è passato, Mac, anche se con un certo ritardo, si è guadagnato un invidiabile posto al sole, ed una notevole messe di linguaggi, prodotti da case di prim'ordine, è a disposizione dei programmatori; Fortran, Logo, Lisp, Modula2, addirittura un APL, alcuni Forth, un Neon (linguaggio simile al precedente, ma con una sintassi più chiara ed intuitiva), per non parlare dei numerosissimi Pascal, Basic e C, che, ovviamente, fanno la parte del leone, affollano l'area della programmazione, pronti a contendere il mercato a forza di token ed utility.

Sono sicuro che tra i lettori di MacCorner, che forse non saranno oceanici, ma che conosco, dalla loro corrispondenza, attenti ed avanzati, ce ne sono molti che hanno sviluppato routine, utility, o scoperto locazioni di memoria particolari capaci di allietare ancora di più le lunghe sere d'inverno seduti davanti al "melone". Non parlo solo di programmi, sempre benvenuti, ma anche di semplici

sequenze operative, combinazioni di tasti, tecniche di utilizzo del mouse, anche in uno specifico programma. Questo spazio, la cui presenza sarà direttamente proporzionale anche all'apporto dei lettori, è aperto a questi interventi e sopravviverà anche grazie alle vostre contribuzioni. Perciò, al lavoro! Come diceva un mendicante: «Nessun contributo è troppo piccolo o grande».

Credo che la maggior parte delle routine che illustreremo su queste pagine faranno capo, più o meno direttamente, alle ROM di sistema; questo mondo immenso, tenebroso, dotato di un Piccolo Popolo pronto a rispondere alle chiamate (CALL) dell'utente, è troppo affascinante ed accattivante per non essere oggetto, anche da parte del programmatore più sempliciotto, di qualche tentativo di esplorazione. Oggi, con la possibilità di

quasi tutti i linguaggi di accedere al ToolBox per intero, è lecito a tutti poter usufruire della potenza nascosta del Mac (non si dimentichi che il modello SE, che pur non è ai vertici delle prestazioni, vanta ben 256 K di ROM, anche se non così pienotti come accadeva nelle ROM da 128).

Tanto per riscaldare il motore, e dato che ormai il nostro spazio si è ridotto alquanto, vediamo di risolvere qualche problema facile facile: andiamo a curiosare nelle font.

Non credo che esista utente di Mac che non si sia installato qualche font supplementare, giusto per personalizzare il suo lavoro, nel suo sistema. I programmi applicativi presenti sul mercato mettono generalmente a disposizione dell'utente tutte le font installate (ma ci sono alcune pesanti eccezioni, come Mac Draw, che consente di utilizzare solo le ultime 8 font installate in ordine di tempo). Bene; ma come fare ad utilizzare le nostre font preferite nei nostri programmi in Basic?

Molto semplice: ogni font è caratterizzata, univocamente, da un ID Number, un numero identificativo compreso in un byte (0-255) che, richiamato tramite l'istruzione CALL TEXTFONT (n) dove [n] è l'ID Number, mette a disposizione della tastiera il carattere prescelto.

Le font distribuite direttamente dall'Apple sono rappresentate dagli ID in tabella.

E per lo stile, ed il modo d'interferire con lo schermo? E per le font aggiunte?

Perdonatemi, Marinacci guarda con occhio torvo la lunghezza del mio scritto; ne parleremo la prossima volta!

MC

ID	Font	note
font ImageWriter		
0	system appl.	(generalmente Chicago) (font di default dell'applicazione utilizzata)
1	Geneva	
2	New York	
3	Monaco	
4	Venice	
5	London	
6	Athens	
7	S.Francisco	
8	Toronto	
9	Seattle	
10	Cairo	
11	Los Angeles	
12	Los Angeles	
font Laser		
20	Times	
21	Helvetica	
22	Courier	
23	Symbol	
24	Talliesin	