

ST Basic, nuova versione

Colgo l'occasione della presentazione della versione riveduta e corretta dell'ST Basic per parlare di questo Basic che in passato ha fatto tribolare non poco coloro i quali, armati di tanta buona volontà, si sono adoperati per costruire qualcosa di utile.

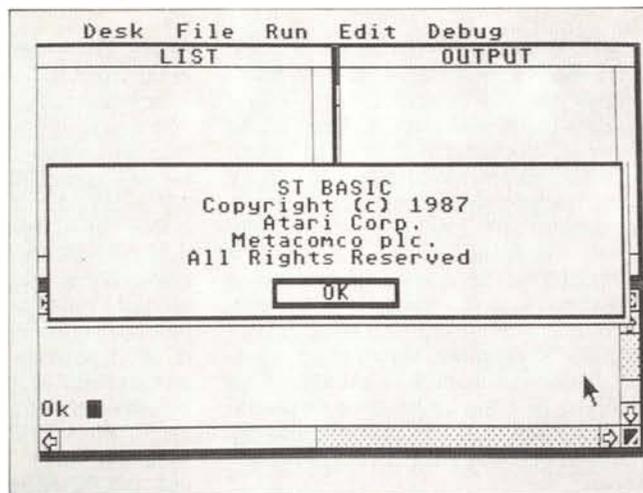
La prima versione aveva diversi bug ed, adoperando un ambiente a finestre piuttosto nuovo, risultava ai più piuttosto scomoda da adoperare, tanto che diverse software house hanno pensato bene di non lasciarsi impressionare dal fatto che tutti gli utenti ST disponessero già dell'ST Basic e si sono adoperate per scrivere altri interpreti Basic, anche di qualità superiore tra cui: il GfA Basic, ora alla versione 2.0 anche con compilatore

minato gli errori presenti nella precedente versione, la ha velocizzata di circa tre volte, ha aumentato le dimensioni dei numeri interi validi e ne ha migliorato la sintassi.

Oltre a questo, le stesse capacità operative di questo Basic sono aumentate dal momento che sono state aggiunte 33 nuove parole riservate ed il sistema di comunicazione di errori è ora più esplicito ed esteso.

Vediamo in particolare queste differenze, tenendo presente che ST Basic © 1987 è generalmente compatibile con i programmi scritti per ST Basic © 1985, tranne che per quello che andrò ad illustrare.

Per cominciare l'affermazione DEF



Il nuovo
ST Basic.

e versione per coprocessore matematico M68881, il Fast Basic, l'HiSoft Basic con compilatore (v 1.1), l'Omicron Basic con compilatore, fino al nuovissimo True Basic per ST, anche disponibile tanto su MS-DOS che su Macintosh e Amiga.

Tornando alla nuova versione dell'ST Basic, il file relativo, BASIC.PRG, è grande 123984 byte e si distingue dal precedente per la dialog box delle informazioni generiche che ci dirà che si tratta di una versione di ST Basic datata 1987 alla quale ha contribuito anche la Metacomco, la software house per la quale ha lavorato anche il versatile Tim King. A quanto pare la Metacomco avrebbe eli-

SEG che serviva a stabilire il modo d'operare dei PEEK e POKE e l'offset usato dai comandi non esiste più; al suo posto sono stati inseriti PEEK_W, PEEK_B, PEEK_L e POKE_W, POKE_B, POKE_L

che restituiscono o scrivono valori rispettivamente a 8, 16 e 32 bit dall'indirizzo di memoria indicato con numero intero. Quindi i programmi che utilizzavano DEF SEG vanno riscritti adoperando le nuove affermazioni. A proposito di interi, il nuovo ST Basic passa dai -32768 e +32767 come intero più piccolo e più grande della precedente versione a -2147483648 e +2147483647.

La sintassi relativa a GEMSYS e VDISYS è stata modificata: in passato era GEMSYS(< AES Op Code >) e VDISYS (< Dummy Argument >) mentre ora è GEMSYS < AES Op Code > e VDISYS [(Dummy Argument)] dove le parole tra < > descrivono il genere di dati da inserire, le parentesi quadre [] indicano valori opzionali che non possono essere ripetuti e le parentesi tonde () indicano valori opzionali che possono essere ripetuti.

Come ho detto sopra, 33 nuove parole ora sono riservate; vediamole insieme alla sintassi ed ai relativi significati:

AREA < point list >: Disegna un poligono pieno.

ASK MOUSE < x >, < y >, < b >: Restituisce le attuali coordinate del mouse e lo stato dei pulsanti nelle variabili specificate.

ASK RGB < reg >, < r >, < g >, < b >: Mette i reali valori di rosso, verde e blu della palette specificata nelle variabili listate.

BIOS < numeric expression >, < arglist >: Fornisce una chiamata di sistema operativo al BIOS.

BOX [FILL] < x1, y1 >, < x2, y2 >: Disegna una box.

CLEAR setta tutte le variabili numeriche a 0 e le variabili di stringa a nulla. Nega le definizioni di tutti gli array.

DRAW < point list >: disegna una linea attraverso i vertici indicati.

DRAWMODE < integer variable >: Setta l'attuale modo di disegno.

ED < line number >: Edita l'attuale programma.

ERRS (< n >): Restituisce un messaggio d'errore per il dato codice.

GEM_ADDRIN (< offset >) = < expression >: Una variabile di sistema legata al Gem che può funzionare come un array incorporato.

GEM_ADDROUT (< offset >) = < expression >: Una variabile di sistema legata al GEM che può funzionare come un array incorporato.

GEM_CONTRL (< offset >) = < expression >: Una variabile di sistema legata al GEM che può funzionare come un array incorporato.

GEM_GLOBAL (< offset >) = < expression >: Una variabile di sistema legata al GEM che può funzionare come un array incorporato.

GEM_INTIN (< offset >) = < expression >: Una variabile di sistema legata al GEM che può funzionare come un array incorporato.

GEM_INTOUT (< offset >) = < expression >: Una variabile di sistema legata al GEM che può funzionare come un array incorporato.

GEMDOS < numeric expression >, < arglist >: Fornisce una chiamata di sistema operativo al GEMDOS.

GSHAPE < x 1 >, < y 1 >, < array >: Scrive un raster memorizzato nel dato array sullo schermo.

LINEPAT < style > [< pattern >]: Setta il pattern di linea.