

MSX N 19

DISK

£ 14.000

giochi
news
utilità

- Galious
- Stormer
- Piggy
- Apeman
- Notebook
- Master
- Finance Plus 2

L'UNICA DISK MAGAZINE
DEDICATA ALLO STANDARD MSX



MSX

DISK

SOMMARIO

- 2 Sommario – Sul disco
- Caricamento – Avvertenze
- 3 Editoriale – Abbonamenti
- 4 Galious – Stormer
- 5 Piggy – Apeman
- 6 Notebook
- 8 Master
- 9 Finance Plus 2
- 10 Basic (Parte IV^a)
- 16 Corso di Zin (Terza e ultima parte)
- 19 Computer & Programmazione (Parte II^a)
- 21 Il libro del mese
- 22 Pirateria
- 24 News
- 25 Dentro l'MSX

SUL DISCO

- 1 Galious
- 2 Stormer
- 3 Piggy
- 4 Apeman
- 5 Notebook
- 6 Master
- 7 Finance Plus

CARICAMENTO

A computer spento inserite il disco nel driver. Tenendo premuto il tasto CTRL accendete il computer e tenetelo inserito fino alla comparsa sul video del sommario. Per caricare un programma premete il numero corrispondente (dall'1 all'8). Il caricamento avverrà automaticamente.

AVVERTENZE

Questo disco è stato registrato con cura e con i più alti standard di qualità. Leggete con attenzione le istruzioni per il caricamento. Nel caso in cui, per una ragione qualsiasi, trovaste difficoltà nel caricare i programmi, telefonate alla nostra redazione al numero (02) 89502256 oppure spedite il disco al seguente indirizzo:

Gruppo Editoriale International Education srl - viale Famagosta, 75 - 20142 Milano.

Testeremo il prodotto e, nel caso, lo sostituiremo con uno nuovo senza aggiunta di costi supplementari.

GALIOUS



La vostra ragazza alta, bella e bionda è stata rapita da Galious, il cattivo di turno, che guardacaso l'ha segregata nel suo castello.

Per pura combinazione il castello è un vero e proprio labirinto popolato da mostri di ogni tipo, anche Biscardi.

Così non vi resta altro che sfoderare la spada ultimo modello e correre fra le fetide stanze del castello-labirinto alla ricerca del vostro amore eterno.

Speriamo che non sia la solita cotta altrimenti rinuncereste subito.

Vestiti i panni del supereroe, dovrete affrontare i crudelissimi mostri alla ricerca degli indizi più disparati con in dotazione arco e frecce e una limitata scorta di energia. Sul suo cammino il nostro eroe Popolon potrà trovare nuova energia ma anche monete d'oro e chiavi altrettanto preziose per aprire cancellate e celle.

Sul suo cammino troverà anche nuove armi che dovranno essere usate con i mostri più solidi.

Raggiunta la bella Aphrodite dovrete difenderla strenuamente perché nel gioco esiste un contatore anche per la sua energia e se questa energia finisce morite in due.

COMANDI

Joystick in porta 1

Tasti:

[CURSORI] = Movimenti & Salto

[SPAZIO] = Fuoco & Inizio gioco

STORMER



In un cielo sempre più affollato dove le rotte devono essere prenotate con giorni di anticipo, voi coraggiosamente affrontate il vostro destino a bordo di un vecchio biplano.

Evitando mongolfiere e concorrenti vari dovrete cercare di viaggiare di qua e di là alla ricerca di bonus e carburante senza essere abbattuti altrimenti, ahimé, perderete un biplano.

Nella fase di gioco sullo schermo compare l'area di gioco in cui è arrivato il vostro aereo grazie ad un efficace scroll orizzontale.

Per aiutarvi avete anche una buona dose di tecnologia al vostro servizio: nella parte alta a sinistra troverete un radar che poi non è così potente ed efficiente come potreste pensare, così diffidate da eventuali messaggi errati.

Gli altri indicatori vi dicono quanto carburante avete, quanti biplani ed il punteggio e il livello (ROUND) raggiunti.

Avete già preso il brevetto?

Sì, lo stesso di Tom Cruise...

COMANDI

Joystick in porta 1

Tasti:

[CURSORI] = Movimenti

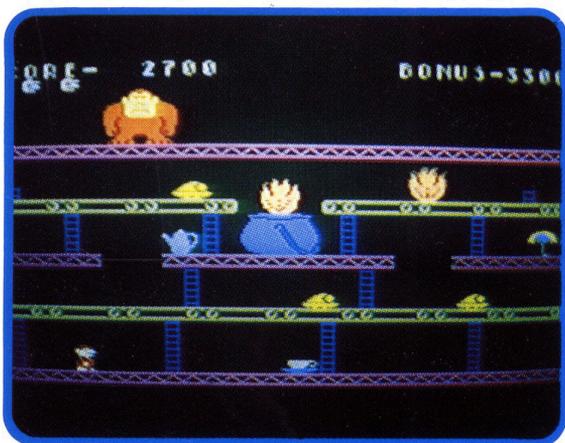
[SPAZIO] = Fuoco & Inizio gioco

PIGGY



Forse non tutti ricordano un gioco che, personalmente, mi ha fatto passare ore spensierate in un bar davanti ad un videogioco diversi anni fa. Questo gioco è uno di quelli in cui ho investito più monete da duecento lire. Come potete vedere nella foto, nella parte destra dello schermo c'è una cabina tipo ascensore in cui trova posto un simpatico maialino di cui vestite i panni. Il maialino, che si può spostare in su ed in giù grazie all'aiuto di altri suini, deve fare in modo che i lupi, dotati di innumerevoli palloncini, raggiungano la cima della montagna e

APEMAN



Anche questo è un grande classico in una nuova versione decisamente fedele all'originale. Il protagonista è il solito baffuto idraulico Mario a cui un panciuto scimmione sottrae la ragazza. Rifugiatosi in un grattacielo in costruzione, lo scimmione cerca di sfuggire a Mario che, implacabile, lo insegue ovunque. Ma riuscirà Mario a raggiungere la

spingano tutti insieme la grossa roccia proprio sulla vostra testa. Per impedire ai lupi di arrivare in alto avete a disposizione un arco e un'illimitato numero di frecce con cui dovete cercare di colpire i palloncini dei lupi. Quando il numero di lupi è sufficiente a spingere la roccia non avete più scampo. Quando cresce la difficoltà del gioco a seconda del livello raggiunto, i lupi potranno insediarsi anche sulla scala posta alla vostra destra tentando di azzannarvi il popò. Tra i diversi livelli del gioco troverete diversi livelli-bonus in cui dovrete colpire e affondare il maggior numero possibile di palloncini. Prima di poter giocare dovrete selezionare, tramite un minuscolo menù, il tipo di controllo: joystick (manette) oppure tastiera (clavier). Questa selezione può essere fatta tramite i cursori e la barra spaziatrice oppure con il joystick.

COMANDI

Joystick in porta 1

[CURSORI] = Movimenti cabina & Scelta opzione

[SPAZIO] = Fuoco & Selezione opzione

cima del grattacielo e a far cadere il gorillone salvando la bella Maria? Questo dipenderà da voi che dovrete vestire i panni del nostro beneamato idraulico. Gli schemi dei quattro livelli di gioco base sono gli stessi della versione arcade: primo livello composto da diversi piani in salita con barilozzi da saltare, fiamme e la possibilità di martellare tutti; secondo livello con fiamme che escono dalla caldaia, nastri trasportatori che portano bulloni roventi e scale mobili che con difficoltà consentono di raggiungere l'ultimo piano; terzo livello con molloni impazziti che spadroneggiano ostacolando la salita, ascensori che vanno e le solite fiamme maligne; quarto e ultimo livello con tantissime fiammelle, sempre più cattive e veloci, e pezzi di trave da rimuovere per togliere il pavimento e far cadere il gorilla. Ce la farete?

COMANDI

Joystick in porta 1

[CURSORI] = movimento

[SPAZIO] = Salto & Inizio gioco

NOTEBOOK



Oltre a Finance Plus, che ormai è un appuntamento fisso, anche Notebook è un programma originale che vi offriamo in esclusiva, proprio come la Gestione Libri dello scorso numero.

Come potete capire dal titolo, il programma è un utility di archiviazione che permette di creare un vero e proprio blocco notes, Notebook in inglese, che permette di archiviare su disco delle pagine di testo.

Le pagine hanno una dimensione di 40 caratteri per riga con un numero massimo di righe per pagina pari a 120, cioè circa 4800 caratteri per pagina.

Una volta scelta l'opzione di Notebook nel menù di Msx Disk dovreste attendere alcuni brevi istanti, dopodiché sullo schermo comparirà la scritta:

“Inserire il disco archivio e premere un tasto...”

Come è chiaro, questo messaggio invita ad inserire nel drive un disco diverso da quello di Msx Disk.

Questo è necessario in quanto non ci sarebbe sufficiente spazio sul disco per ospitare gli archivi di Notebook.

Per rendere più efficiente il sistema potete copiare il programma - che sul nostro disco è chiamato 5.BAS - direttamente in un disco archivio che userete solo per questo scopo.

Rinominando il file come

AUTOEXEC.BAS

potrete far partire automaticamente il programma di gestione ogni volta che inserite il disco archivio e accendete il computer.

Dopo la pressione di un tasto qualsiasi una volta inserito il disco archivio, sullo schermo comparirà il menù principale di Notebook che potete vedere nella foto.

Il programma offre 7 diverse opzioni selezionabili premendo il tasto numerico corrispondente e quindi il tasto di invio. Le opzioni disponibili sono:

1 - INSERIMENTO DATI

Selezionando questa opzione sullo schermo apparirà la richiesta di una chiave che identificherà i diversi fogli del blocco.

Se usate più volte la stessa chiave i nuovi fogli aggiunti diventeranno il secondo, il terzo, il quarto e così via di fogli appartenenti a quel soggetto.

Nuovi fogli con la stessa chiave vengono inseriti ogni volta che superate lo spazio destinato ad una singola pagina. In questo caso il computer archivia la pagina precedente e vi porta direttamente alla nuova pagina.

Dopo aver inserito la chiave potete passare alla fase di inserimento vera e propria.

Per terminare l'inserimento della pagina premete il tasto di Escape [ESC] che riporta l'esecuzione alla richiesta della chiave.

Inserendo una chiave nulla l'esecuzione ritorna al menù principale.

2 - MODIFICA DATI

Questa opzione, come la precedente, vi richiede la chiave della pagina da modificare.

Se esistono più pagine con la stessa chiave vi verrà chiesto il numero di pagi-

NOTEBOOK

na, da 1 a N dove N è il numero di pagine con la stessa chiave.

Dopo aver raggiunto la pagina la fase di editing sarà in tutto e per tutto uguale a quella dell'inserimento. Come nella prima opzione, dovrete usare il tasto

[ESC]

per terminare la modifica.

3 - ELIMINAZIONE

Anche qui dovrete indicare la chiave di ricerca. Trovata la pagina desiderata il computer vi chiederà la conferma dell'operazione di cancellazione della pagina. Dovrete necessariamente rispondere inserendo [S] altrimenti la pagina non verrà cancellata.

4 - ANALISI DATI

Questa opzione vi porta ad un sottomenù con tre opzioni:

LISTA PAGINE

che mostra una lista delle pagine presenti chiave per chiave

ANALISI PAGINA

che permette di visualizzare, riga per riga e velocemente, una o più pagine di una chiave data

MENU' PRINCIPALE

per tornare al menù precedente.

5 - STAMPA DATI

Come nel caso precedente, tramite questa opzione si accede ad un menù secondario con tre opzioni:

STAMPA PAGINA

che stampa una o più pagine con la stessa chiave

INSTALLAZIONE STAMPANTE

che permette di formattare la stampa

MENU' PRINCIPALE

che riporta l'esecuzione al menù principale.

6 - CAMBIO DISCO

Questa opzione serve per cambiare disco archivio durante l'elaborazione con anche la possibilità di formattare il nuovo disco archivio.

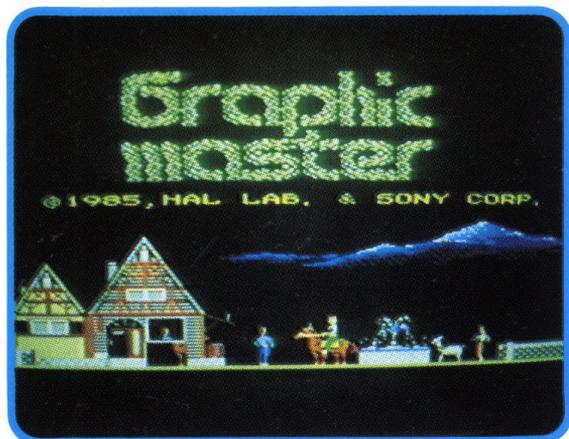
7 - FINE LAVORO

Questa opzione permette di abbandonare l'uso di Notebook.

Dopo averla selezionata dovrete confermare se abbandonare il programma inserendo [S] o [N].

Poichè le pagine di Notebook sono composte da 4800 caratteri in un disco formattato a 360 kilobyte potrete inserire un massimo di 75 diverse pagine, valore che raddoppia usando dischetti formattati a 720 kilobyte.

MASTER



Questo mese vi proponiamo una nuova utility grafica che esce decisamente dai soliti binari.

Il programma, che ricorda molto il Creative Graphic della Sony, offre tutte le potenzialità di altri programmi ma ha in più degli sprite grafici predefiniti.

Già nella presentazione - foto 1 - potete vedere alcuni sprite disponibili per creare le vostre schermate.

Il funzionamento è semplificato da una serie di menù ad icone decisamente facili da capire.

Per muovere il cursore sull'icona desiderata dovrete utilizzare i tasti del cursore, cioè quelli con le frecce, ed effettuare le selezioni con i tasti di funzione [F1] o [F3] mentre [F2] annulla l'ultima selezione.

Questi tre tasti corrispondono ai tre tasti del trackball Sony, di cui è possibile fare uso mentre [F2] e [F3] corrispondono ai tasti del mouse.

Purtroppo, non è stato possibile verificare la piena funzionalità del mouse con questo programma, così potreste anche avere dei problemi e quindi non poter usare il mouse come controllore del pro-

gramma.

In ogni fase di disegno o di selezione i tasti funzione [F1] e [F3], che sono corrispondenti, selezionano e confermano la scelta dell'utente, mentre con [F2] è possibile annullare l'ultima scelta.

Grazie alle chiare icone con matite e disegni sarà possibile usare velocemente il programma senza bisogno di aiuto.

Nella seconda foto potete vedere il menù secondario a cui si accede tramite l'icona che contiene diverse righe di diverso diametro.

Con questo menù è possibile scegliere il tipo di riga che si può disegnare o il tipo di retino per la funzione di riempimento, chiamata Fill o Paint.

Come già detto, oltre ad un sofisticato uso dei colori è possibile sfruttare degli sprite già presenti in memoria oppure crearsene dei propri tramite l'ultima opzione in basso a destra del menù principale.

Nella terza foto potete vedere il menù secondario destinato agli sprite e un esempio degli sprite esistenti.

Tramite l'icona con il dischetto potrete memorizzare sia su disco che cassetta i vostri file.

Per richiamare i file da voi creati su schermo sarà sufficiente settare lo screen 2 e poi leggere i dati tramite un'istruzione BLOAD con l'opzione S.

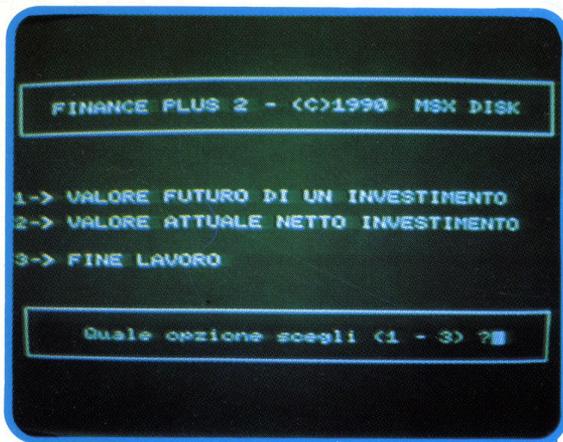
Per esempio,

BLOAD "ciao",S

carica sullo schermo il file CIAO.

Per maggiori informazioni fate riferimento al manuale del vostro computer nella parte dedicata ai file binari.

FINANCE PLUS 2



Ribadiamo ciò che abbiamo detto nello scorso numero inaugurando questa serie di programmi: questi programmi hanno il duplice scopo di fornire un aiuto nella gestione delle proprie finanze e offrire uno spunto didattico ai principianti che entrano nel mondo della programmazione.

Per utilizzare il programma sarà sufficiente premere il tasto [7] quando compare il menù principale di Msx Disk.

Il file 7.BAS è il file sorgente pronto per voi, per essere esaminato.

Questa volta il programma è dedicato al calcolo del valore futuro e del valore attuale netto di un investimento e, come la volta scorsa, il programma è diviso in tre parti.

Il programma offre un menù principale con le seguenti tre opzioni:

- 1 - VALORE FUTURO DI UN INVESTIMENTO;
- 2 - VALORE ATTUALE NETTO DI UN INVESTIMENTO;
- 3 - FINE LAVORO.

Nel file sorgente il calcolo del valore futuro viene espletato nell'algoritmo dalla linea 1000 alla linea 1300, mentre il valore attuale netto viene calcolato tramite la parte di programma che va dalla linea 2000 alla linea 2400.

Dalla linea 3000 in poi trovano posto le routine di servizio come, ad esempio, quella per la generazione della maschera del menù.

VALORE FUTURO DI UN INVESTIMENTO

Spesso è utile fare una previsione dei guadagni futuri - o conclusivi - che ci si aspetta di ricavare da un investimento.

Il criterio accetta/rifiuta stabilisce che si debba rifiu-

tare qualunque investimento il cui valore futuro di tutti i flussi di pagamento, incluso anche l'investimento iniziale, sia minore di zero. Questo algoritmo fornisce tale valore basandosi sulle scadenze (in anni), il tasso di interesse e l'ammontare dei flussi di pagamento per ogni anno. Il tasso di interesse dovrebbe essere quello al quale è possibile avere opportunità alternative di investire.

ESEMPIO:

Si vuole aprire un conto stanziando 200.000 lire il primo anno, 350.000 lire l'anno successivo e 250.000 lire l'anno seguente. Il tasso di interesse è del 6%. Quale sarà il valore del conto alla fine del terzo anno? Risposta : 845.720 lire.

VALORE ATTUALE NETTO DI UN INVESTIMENTO

Il valore attuale netto (VAN) è definito come il valore attuale di tutti i processi di pagamento associati ad un investimento, includendo la spesa iniziale.

Il criterio accetta/rifiuta del VAN per un investimento stabilisce che si debba accettare qualunque investimento il cui VAN sia maggiore di zero.

Per usare questo programma, prima di tutto dovrete inserire l'ammontare della spesa iniziale, la scadenza dell'investimento (in anni), la percentuale di guadagno richiesta e l'ammontare dei flussi di pagamento per ogni anno.

Per ottenere il valore attuale di un investimento, inseritene uno iniziale pari a zero.

ESEMPIO:

Si ha l'opportunità di un investimento che richiede una spesa iniziale di 10.000.000 di lire e offre una rendita di 3.000.000, 5.000.000, 4.000.000 di lire per i successivi tre anni. Si vuole almeno una rendita del 15% sull'investimento. Qual'è il valore attuale netto ? Si fa bene ad accettare ?

Risposta : il VAN di questo investimento è -980.520 lire. Non si dovrebbe accettare.



Le variabili, in informatica come in matematica, sono scatole vuote, ciascuna delle quali è contrassegnata da un proprio nome e serve per contenere un valore numerico od una stringa di caratteri. Così facendo il programma risulta pronto per la risoluzione di problemi analoghi. Dobbiamo, però per riutilizzarlo successivamente salvarlo su floppy.

ASSEGNAZIONE

Istruzione LET

L'istruzione sicuramente più importante di un linguaggio di programmazione è quella che permette di attribuire un valore ad una variabile, come anche di modificarlo o di calcolare un'intera espressione per esempio di tipo aritmetico. Abbiamo già incontrato le variabili e ne abbiamo dato una prima definizione. Avrete capito che una variabile è il nome simbolico che diamo a qualcosa in attesa di attribuirgli uno specifico valore. Il concetto di variabile usato in informatica è praticamente lo stesso di quello usato in matematica: un nome letterale per una grandezza numerica. Di variabili nei linguaggi di programmazione ve ne sono: variabili numeriche, variabili stringa e variabili logiche. Come abbiamo già detto all'inizio, preferiamo dapprima fare degli esempi e poi dare delle definizioni rigorose. Cominciamo quindi con il mostrare degli esempi di istruzioni di assegnazione. Nei programmi precedenti, i valori delle variabili A, B, C e D venivano chiesti come dati di ingresso. Se essi fossero invece noti al momento di scrivere il programma, potremmo allora assegnarli direttamente alle rispettive variabili. Per esempio così:

Programma 1

```
10 LET A = 25
20 LET B = 120
30 LET C = 5.6
40 LET D = 0
50 PRINT A,B,C,D
60 END
```

Se ora diamo il comando RUN, otteniamo:

```
25 120 5.60
```

(Per praticità non abbiamo più riportato il prompt READY in quanto riteniamo che il lettore abbia già capito quando esso appare).

Le istruzioni 10, 20, 30 e 40 sono quattro istruzioni di assegnazione. La parola LET si può tradurre come "poni" e vuol dire proprio "poni A uguale a 25", "poni B uguale a 120", ecc. Con l'istruzione 50 facciamo invece visualizzare il contenuto delle

quattro variabili e vediamo proprio i valori che avevamo loro assegnato. E' importante fare subito una precisazione. La parola LET fu coniata per le primissime versioni di BASIC, ma nella maggior parte dei BASIC attuali non è più obbligatorio usarla. Il Programma 1 si può scrivere più semplicemente così:

```
10 A = 25
20 B = 120
30 C = 5.6
40 D = 0
50 PRINT A,B,C,D
60 END
```

Ancora un esempio:

```
10 A = 25 + 10
20 B = 3.14 - 3
30 PRINT A,B
40 END
```

```
RUN
35 0.14
```

Che cosa c'è di diverso nelle ultime istruzioni 10 e 20? A destra del segno di eguale (il segno di assegnazione) invece di un valore costante abbiamo posto una espressione algebrica che il calcolatore ha calcolato ed assegnato alle rispettive variabili. Un'istruzione di assegnazione permette anche di fare dei calcoli, per esempio algebrici, e poi attribuire il risultato ottenuto alla variabile posta a sinistra del segno di eguale. Facciamo qualche altro esempio:

```
10 M = 2*(3 + 6)           M=18
10 M = (180 + 20)/(10 - 6) M=50
```

Con la freccetta indichiamo, in modo simbolico, il risultato di qualche operazione o il valore assunto da un variabile. L'asterisco * e la barra / (in inglese "slash") che ci sono nelle espressioni degli esempi sono rispettivamente i simboli per la moltiplicazione e la divisione usati in quasi tutti i linguaggi di programmazione. Si potrebbero fare altri esempi con espressioni molto più complicate, ma lasciamo al lettore il divertimento di pensarle e, se possibile, anche di provarle su un calcolatore. In questo momento è importante dire due cose. Per primo va precisato, ma lo abbiamo già detto, che il segno di eguale = usato nelle istruzioni di assegnazione non rappresenta una eguaglianza matematica! Non è quindi possibile scrivere cose del genere:

```
10 45 + 5 = A
```

Le assegnazioni in informatica portano un valore da

destra nella variabile di sinistra. Viceversa non significa nulla. E' anche un errore scrivere eguaglianze del tipo:

```
20 Y + 30 = Z
```

Ripetiamo che questa non è un'assegnazione. Chiarito quindi che le assegnazioni permettono di avere a sinistra del segno = solo il nome di una variabile, diciamo qualcosa di più su cosa si può scrivere in un'espressione posta a destra. Un'espressione può contenere delle variabili che, in tal caso, vengono trattate come fossero dei valori già noti. L'unica limitazione all'uso di variabili nella parte destra di un'istruzione di assegnazione è che devono avere un valore al momento in cui tale istruzione viene eseguita. Facciamo degli esempi:

```
50 A = 50
60 B = A + 100      B = 150
30 C = 20 + 15
40 F = 15 / 3
50 Z = C / F      Z = 7
10 K = 1
20 K = K + 1      K = 2
```

Quest'ultimo esempio è molto significativo poiché presenta un caso molto usato: la creazione di un contatore. Se cerchiamo di leggere passo l'istruzione 20 possiamo dire: somma al contenuto attuale di K il valore 1; siccome K vale 1 si ottiene 2. Ma che cosa succede se l'istruzione 20 dovesse essere nuovamente eseguita? K ora vale 2 per cui il nuovo valore sarebbe 3, e così via. Per ripetere varie volte l'istruzione 20 vi anticipiamo quella che sarà una delle istruzioni di salto che vedremo in seguito GOTO ("go to" in italiano "vai a"). Ecco come modificare l'ultimo programma:

```
10 K = 1
20 K = K + 1
30 GOTO 20
```

L'istruzione 30 dice di "saltare" alla 20 (invece che all'istruzione col numero successivo) per cui queste due istruzioni vengono indefinitamente eseguite e il valore di K cresce continuamente: 1, 2, 3, 4, 5, 6, ecc.. Negli esempi di programmi successivi avremo occasione di chiarire molti altri aspetti delle istruzioni di assegnazione.

VARIABILI

Abbiamo visto qualche semplice esempio di programma BASIC. Facciamo ora un passo avanti e proviamo a studiare programmi che siano in grado di svolgere un determinato compito. Forse qualche

punto di questi programmi non risulterà molto chiaro essendo solo all'inizio. Non preoccupatevi troppo, ma cerca di comprendere gli aspetti fondamentali di quanto spiegheremo. Vedere e possibilmente provare molti esempi pratici è di grande aiuto nell'apprendimento, di un linguaggio di programmazione, anche se all'inizio non se ne comprendono tutti i dettagli. Il Programma 2, che possiamo chiamare "programma per la stampa di biglietti da visita", è un'esempio d'impiego delle istruzioni INPUT e PRINT. Ricordiamo che con "stampa" si intende spesso dire "visualizzazione sullo schermo video", come appunto in questo caso. Solo se disponete di una periferica stampante potete effettivamente stampare i risultati di un programma. A parte questo chiarimento sui termini, vediamo cosa fa il Programma 2.

Programma 2

```
10 PRINT "PROGRAMMA PER LA STAMPA DEL"
20 PRINT "      VOSTRO"
30 PRINT "BIGLIETTO DA VISITA."
40 PRINT
50 INPUT "VOSTRO NOME: ";NOME$
60 INPUT "VOSTRO COGNOME: ";COGNOME$
70 INPUT "VIA: ";V$
80 INPUT "NUMERO: ";N
90 INPUT "CITTA': ";C$
100 PRINT
110 PRINT "ECCO IL VOSTRO BIGLIETTO DA VI-
SITA: "
120 PRINT
130 PRINT "-----"
140 PRINT "SIG. ";NOME$," ";COGNOME$
150 PRINT
160 PRINT "VIA ";V$," ";N
170 PRINT C$
180 PRINT "-----"
190 GOTO 40
200 END
```

Prova di esecuzione

PROGRAMMA PER LA STAMPA DEL VOSTRO BIGLIETTO DA VISITA

```
VOSTRO NOME: ? SERGIO
VOSTRO COGNOME: ? ROSSI
VIA: ? CARPACCI
NUMERO: ? 12
CITTA': ? MILANO
```

Alle linee 10, 20 e 30 vi sono tre istruzioni PRINT per la visualizzazione del titolo del programma. La PRINT della linea 40 serve solo per andare a capo. Alle linee da 50 a 90 vi sono le istruzioni d'ingresso

ga, cioè variabili che invece di numeri contengono stringhe di caratteri. Notate che il nome di queste variabili termina con il carattere \$ (dollaro). Notate anche che lo spazio è un carattere come tutti gli altri. Pensiamo che il Programma 3 sia molto facile da capire e basti osservarlo. Provate a modificarlo o a ideare altri programmi che disegnino, per esempio, una cornice o una quadrettatura sullo schermo. I programmi visti finora sono molto semplici e permettono di effettuare solo delle operazioni di ingresso dei dati (con INPUT) nella memoria centrale, di stampa degli stessi (con PRINT) sullo schermo. Abbiamo accennato a come sia possibile inviare i risultati di un programma ad una stampante. I dati possono anche essere memorizzati (scritti) su memorie esterne (dischetti o cassette magnetiche), da dove poi possono essere nuovamente prelevati e riportati nel computer. La registrazione di dati su una memoria esterna, cioè la scrittura di file, è uno degli aspetti più interessanti ed utili della programmazione, ma prima di affrontarlo è meglio avere un po' di dimestichezza col BASIC in generale. E' utile invece imparare subito a registrare il programma su una memoria esterna, in modo che possa essere conservato. La memoria centrale di un calcolatore perde infatti ogni contenuto nel momento in cui la macchina viene spenta (si dice che è una "memoria volatile"). E' necessario quindi registrare un programma su una memoria esterna non volatile, ad esempio un dischetto magnetico, per poterlo conservare. Per effettuare questa operazione si usa l'istruzione SAVE descritta in dettaglio nel relativo riquadro. Poiché l'uso di SAVE non è uguale in tutte le macchine, vi consigliamo di consultare il manuale del computer che state usando. Il trasferimento inverso del programma, dalla memoria esterna a quella centrale, si effettua con l'istruzione LOAD (in alcune versioni BASIC si usa l'istruzione OLD che è equivalente a LOAD).

Sul supporto su cui si vuole salvare il programma (cassetta, dischetto o altro) non deve esistere un altro programma con lo stesso nome. In caso contrario, è probabile che il vecchio programma vada perso e venga sostituito con quello nuovo (alcuni sistemi operativi chiedono prima una conferma). Questa operazione viene spesso eseguita per aggiornare la registrazione di un programma, registrando al suo posto una versione più recente dello stesso. E' una pessima pratica, perché in caso di errore fisico di scrittura, o di caduta di rete (mancanza di corrente), si perdono entrambi i programmi. E' molto meglio registrare i programmi con i numeri progressivi, conservando le ultime due o tre versioni i cancellando quelle vecchie. Per poter essere eseguito da un computer, un programma BASIC deve risiedere completamente nella memoria centrale. Questo significa che non può essere più lungo, in numero di

byte, dell'area di memoria disponibile per i programmi. Poiché alcuni computer sono in vendita con diverse dimensioni (configurazioni) di memoria, non è detto che un programma registrato da una macchina sia sempre usabile su un'altra: può non esserci memoria a sufficienza (ricordiamo che le memorie centrali si misurano in Kbyte: 16, 32, 48, 64, 128 Kbyte...). Per esempio, un programma molto lungo scritto su un Sinclair Spectrum da 48K (48 Kbyte) può non funzionare su uno Spectrum da 16K. In mancanza di informazioni, si può provare a caricare, in ogni caso, il programma e vedere che non venga segnalato alcun errore del tipo "Out of memory" (memoria insufficiente) o simili, e che il programma poi funzioni regolarmente. Può essere utile, dopo un'operazione di LOAD, fare il listato del programma (LIST) per controllare che sia proprio quello voluto, prima di dare il comando RUN, se si desidera eseguirlo. Il comando LOAD, prima di cominciare a caricare il nuovo programma, cancella completamente ogni altro programma e/o dato presente in memoria (centrale). Prima di dare un comando LOAD, assicurarsi quindi che l'eventuale programma presente in memoria sia stato salvato con SAVE, o possa comunque essere cancellato. In alcune versioni BASIC esistono speciali comandi, come CHAIN o MERGE, che permettono di sovrapporre o unire due programmi senza cancellare completamente il precedente ed i suoi dati (variabili). Questa possibilità si utilizza quando un programma è molto lungo e non è caricabile interamente in memoria. Lo si fraziona allora in due o più parti, tra cui esista una connessione logica, e queste si caricano e si eseguono una alla volta. La prima viene caricata con LOAD, le successive con CHAIN o MERGE. Un'altra istruzione che abbiamo già visto è quella di assegnazione, che in BASIC si chiama LET. L'istruzione di assegnazione, che esiste in tutti i linguaggi di programmazione, permette di assegnare (attribuire) un certo valore ad una variabile. Non abbiamo ancora detto cosa siano le variabili in un linguaggio di programmazione, anche se le abbiamo già usate negli esempi. Se ricordate le variabili imparate a scuola durante le lezioni di matematica, possiamo allora dire che in informatica sono pressappoco la stessa cosa. Una variabile è come una scatola che può contenere qualcosa (in BASIC, un numero od una stringa). Più propriamente, è un nome simbolico per indicare una grandezza a cui può essere assegnato un valore effettivo. Forse un semplice esempio può chiarire alcuni dubbi. Supponiamo di dover riempire un modulo per una domanda di assunzione in un'azienda. Quasi tutti questi moduli iniziano richiedendo i dati anagrafici: cognome, nome, anno di nascita, ecc. Vicino ad ognuno si trovano dei puntini per indicare dove dobbiamo scrivere i nostri dati personali:

Cognome..... Rossi
Nome..... Sergio
Anno di nascita..... 1958

“Cognome”, “Nome”, e “Anno di nascita” sono variabili a cui chi scrive attribuisce dei valori effettivi (nel nostro caso “Rossi”, “Sergio” e “1958”). Tutte le domande di assunzione portano indicate le stesse variabili, che ognuno “riempie” con i propri dati. Nel Programma 2 le variabili erano NOME\$, COGNOME\$, V\$, N e C\$. Si potrebbe notare che le variabili imparate a scuola trattavano “numeri”, mentre in questo caso “Rossi” e “Sergio” non sono numeri. E' vero: le variabili usate in informatica sono un concetto più ampio di quello algebrico. In BASIC infatti esistono due tipi di variabili: quelle numeriche e quelle di stringa. I dati che si possono attribuire alle variabili numeriche sono dei numeri, mentre per quelle dette di stringa sono, appunto, delle stringhe. Che cosa siano con precisione le stringhe, o per meglio dire le stringhe di caratteri, lo vedremo più avanti. Per ora limitiamoci a dire che i dati “Rossi” e “Sergio” sono due esempi di stringhe, come stringhe sono gli insiemi di spazi ed asterischi delle istruzioni 10 e 20 del Programma 3. Una nota importante riguardo alle variabili: in molti dialetti BASIC, solo le prime due lettere del nome di una variabile sono significative, cioè contano per distinguerla dalle altre. Le lettere successive possono essere impiegate per comodità, ma non hanno importanza. Occorre perciò fare attenzione a non usare nomi con le prime due lettere uguali, per esempio, MINIMO e MINUTI o COGNOME\$ e CODICE\$.

ESPRESSIONI NUMERICHE

Le espressioni matematiche, in BASIC, sono praticamente identiche a quelle che si studiano a scuola, con alcune piccole differenze nella scrittura degli operatori (i simboli delle operazioni). Prendiamo un'espressione algebrica, scritta come a scuola, e proviamo a riscriverla in BASIC. Per fare questo, dobbiamo usare i simboli delle operazioni algebriche (operatori algebrici) accettati dal BASIC. Come si vede le operazioni di “somma” e di “sottrazione” si indicano come al solito con “+” e “-”. Il prodotto viene invece indicato con l'asterisco “*” e non con “x” o il punto come si usa in matematica. Spesso in matematica il prodotto di due valore si indica semplicemente scrivendoli uno dopo l'altro; così “a b” vuol dire “a per b”. In informatica, invece, nulla è mai sottinteso, per cui il prodotto deve essere espressamente indicato con l'asterisco: A * B (notare che in BASIC tutto va in lettere maiuscole). La divisione è indicata con il carattere “/” (barra, in inglese slash). I due punti “:” hanno in BASIC un significato completamente diverso e non costante in tutti i

dialetti. L'elevamento a potenza è un'altra operazione con un simbolo particolare. Non si scrive l'esponente come un numero più piccolo posto in alto a destra della base come “4-”. Si scrivono invece base ed esponente sulla stessa riga, separati da uno dei simboli “**” o “^” o “^”; cioè, per il nostro esempio, 42 o 4**2 o ancora, 4^2. Per quanto riguarda le parentesi, le uniche che si usano nelle espressioni sono quelle rotonde, che funzionano esattamente come in matematica. Le parentesi quadre e graffe sono usate in questo testo e in alcuni manuali, ma solo per descrivere le istruzioni del linguaggio di programmazione. Non fanno parte del linguaggio BASIC (sono usate invece in altri linguaggi, come PASCAL e C). Altre funzioni matematiche che si trovano normalmente pronte in BASIC sono l'esponenziale EXP(X) e la sua funzione inversa LOG(X) che è il logaritmo in base “e”, le funzioni trigonometriche seno, coseno e tangente, rispettivamente SIN(X), COS(X) e TAN(X). Di queste ultime bisogna ricordare che si riferiscono ad angoli in radianti (20 radianti = 360 gradi). Dato un angolo espresso in gradi, per convertirne il valore in radianti basta moltiplicarlo per:

$$X_{\text{rad}} = x * (3.1416/180)$$

ALCUNI PROGRAMMI

Vediamo ora un programma in cui vengono calcolate alcune espressioni algebriche. La prima serve per il calcolo del volume di un parallelepipedo e la seconda per quello di una sfera. Il Programma 4 chiede le dimensioni del parallelepipedo, l'altezza, lunghezza e larghezza, poi alla linea 80 le moltiplica tra di loro per calcolare il volume e assegna il risultato alla variabile V. Successivamente chiede il raggio di una sfera e poi ne calcola il volume alla linea 140. Notate che viene usato lo stesso nome di variabile (V) sia per il volume del parallelepipedo che per quello della sfera. Questo è possibile purché si usi la variabile V correttamente. Vediamo di chiarire questo punto. Inizialmente V non contiene niente (contiene zero), poi alla linea 80 le viene assegnato il valore del volume del parallelepipedo. Subito dopo, alla linea 100, viene visualizzato con PRINT V il suo contenuto (volume parallelepipedo). In un secondo tempo, alla linea 140 viene calcolato il volume della sfera che viene assegnato alla stessa variabile V: da questo momento V contiene solo il volume della sfera e quello del parallelepipedo è stato cancellato. In effetti a noi interessa ora il volume della sfera che viene infatti visualizzato con PRINT V (linea 160). Ovviamente, se avessimo avuto bisogno di conservare sia il valore del primo che del secondo volume avremmo dovuto usare due variabili diverse, per esempio VP e VS. Questo

esempio conferma che le variabili in informatica si comportano come scatole in cui si può mettere, di volta in volta, ciò che si vuole. L'ultimo oggetto (numero) introdotto "caccia via" quello che c'era prima.

Programma 4

```
10 PRINT "PROGRAMMA PER IL CALCOLO DI
VOLUMI"
20 REM -----
30 PRINT
40 PRINT "CALCOLO DEL VOLUME DI UN PA-
RALLELEPIPEDO"
50 INPUT "ALTEZZA ";H
60 INPUT "LUNGHEZZA ";LG
70 INPUT "LARGHEZZA ";LA
80 V=LA*LG*H
90 PRINT "IL VOLUME E': ";
100 PRINT V
110 PRINT
120 PRINT "CALCOLO DEL VOLUME DI UNA
SFERA "
130 INPUT "RAGGIO DELLA SFERA ";R
140 V=3. 1416*(4/2)*R^2
150 PRINT "IL VOLUME DELLA SFERA E': ";
160 PRINT V
170 END
```

Prova di esecuzione

```
PROGRAMMA PER IL CALCOLO DI VOLUMI

CALCOLO DEL VOLUME DI UN PARALLELEPI-
PEDO
ALTEZZA ? 5
LUNGHEZZA ? 6
LARGHEZZA ? 8
IL VOLUME E': 240

CALCOLO DEL VOLUME DI UNA SFERA
RAGGIO DELLA SFERA ? 5
IL VOLUME DELLA SFERA E': 523. 6
```

Il Programma 5 è un esempio di come si possa tenere una piccola contabilità e calcolare dei valori percentuali.

Supponiamo di aver fatto un viaggio e voler sapere come si sono ripartite le spese, in percentuale, tra automobile, hotel e ristoranti. Dopo avere ricevuto i dati in ingresso, alla linea 90 viene calcolato il totale delle spese, che viene conservato nella variabile T. Alle linee 100, 110 e 120 vengono calcolate le tre percentuali e assegnate rispettivamente alle variabili APERC, HPERC e RPERC.

Programma 5

```
10 REM PROGRAMMA PER IL CALCOLO
20 REM DEI VALORI PERCENTUALI DI
30 REM SPESA RISPETTO AL TOTALE
40 REM IN OCCASIONE DI UN VIAGGIO
50 REM -----
60 INPUT "SPESE AUTO ";A
70 INPUT "SPESE HOTEL ";H
80 INPUT "SPESE RISTOR. ";R
90 T=A+H+R
100 APERC=A*(100/T)
110 HPERC=H*(100/T)
120 RPERC=R*(100/T)
130 REM
140 PRINT "SPESE AUTO IN PERC. ",APERC
150 PRINT "SPESE HOTEL IN PERC. ",HPERC
160 PRINT "SPESE RISTOR. IN PERC. ",RPERC
170 END
```

Prova di esecuzione

```
SPESE AUTO ? 20000
SPESE HOTEL ? 120000
SPESE RISTOR. ? 35000
SPESE AUTO IN PERC.          11. 4286
SPESE HOTEL IN PERC.        68. 5714
SPESE RISTOR. IN PERC.      20
```

Quest'ultimo esempio sembra banale e ci si può chiedere perché usare un calcolatore, o scrivere un programma, per calcoli più facilmente eseguibili con una calcolatrice tascabile.

Ma naturalmente questi sono soltanto esempi, che devono essere il più possibile semplici.

Quello che conta, per ora, è il principio. Un programma effettivo potrebbe essere impostato più o meno nello stesso modo e lavorare non su tre voci, ma su centinaia, e ricevere i dati non a mano dalla tastiera, ma da un archivio su disco.

E da ultimo potrebbe anche tracciare un "grafico" della ripartizione delle spese o estrapolare (predire sulla base dei dati precedenti) previsioni per gli anni successivi.

Programmi di questo genere, fatti per immagazzinare e successivamente rivedere, elaborare, presentare grandi quantità di dati, si chiamano "programmi di data base", o semplicemente "data base".

Fine Quarta Parte

Con quest'ultima parte terminiamo il nostro mini corso all'uso di ZIN. Vi daremo ulteriori informazioni sull'uso dell'Editor/Assembler, parleremo delle etichette (Label), della tabella dei simboli, degli operatori e degli operandi e degli errori.

USCITA DALLA LISTA

I comandi Assemble, Sort e Disassemble possono tutti generare grandi quantità di outputs al video, alla stampante e ai dispositivi esterni. Con questi comandi l'output sarà generato una pagina alla volta con una breve pausa fra una pagina e l'altra. Premendo un tasto qualsiasi si fermerà l'emissione alla fine della pagina, per ripartire premete un tasto qualsiasi che non sia Q. Questo tasto forzerà il comando a lasciare e a ritornare al loop di comando.

La stampante e i dispositivi esterni hanno una larghezza di ottanta caratteri e una lunghezza di sessantasei linee, tipica di una stampante con foglio standard. Se avete qualcosa di differente allora è necessario modificare l'Editor Assembler. Potete variare la lunghezza della pagina modificando la procedura della pagina.

Potete variare l'ampiezza dei vari campi modificando il gruppo di costanti COMWIDTH/SYMWIDTH. Il primo byte di tutte queste coppie di costanti definisce un'ampiezza, mentre il secondo definisce l'ampiezza di campo di un dispositivo video. Potete anche cambiare il numero di simboli per linea prodotti durante un ordinamento poiché esiste un tasto specifico per questo intento.

I dispositivi esterno/stampante dovrebbero rispondere ai caratteri di controllo ASCII formfeed (0Ch), Ritorno a capo (0Dh) e avanzamento riga (0Ah). Lo ZIN produce un avanzamento modulo seguito da sessantadue linee di testo per ogni pagina e ogni pagina termina con un ritorno a capo e un avanzamento riga. Il driver del dispositivo esterno viene predisposto per l'uscita sul dispositivo MSX Rs232, cioè la porta seriale, mentre le uscite del driver della stampante sono predisposte per una porta di tipo centronics, cioè la porta parallela che hanno di serie tutti gli Msx. I drivers trattano le stampanti Epson FX 80 così come sono. Se avete qualcosa di insolito c'è uno spazio nel driver per inserire pezzi, per esempio per filtrare LineFeeds.

Il dispositivo video ha un'ampiezza di trentasette caratteri ma quest'è qualcosa che si può cambiare come per la stampante e i dispositivi esterni, sempre che sia disponibile un computer da 80 colonne come l'Msx 2. Notate che i numeri di linea non sono generati sul dispositivo video per l'Assembler/Disassembler a causa della ridotta ampiezza. Il simbolo, l'operando e i campi di commento di un'istruzione Z80 possono avere una lunghezza indefinita. Se necessario lo ZIN troncherà questi campi per ridurli al formato richiesto.

LA TABELLA DEI SIMBOLI

La tabella dei simboli è l'area di memoria usata dallo ZIN per immagazzinare i simboli durante l'Assemblaggio/Disassemblaggio. E' situata fra l'Editor Assembler, cioè lo Zin, e il file di testo. Se volete aumentare il suo formato dovete soltanto cambiare il puntatore di inizio-di-file al nuovo valore richiesto. Ecco come fare:

- 1) Eliminate il file di testo;
- 2) Usate MODIFY per cambiare SOFP;
- 3) Eliminate ancora il file di testo per copiare SOFP in EOFP e corrente;
- 4) Eseguite un assemblaggio per chiudere la tabella dei simboli;
- 5) Usate il WB per scrivere la nuova versione.

Notate che lo ZIN è un programma completamente "elastico", qualsiasi cambiamento fate si riflette nella nuova versione.

LA SINTASSI DELL'ASSEMBLATORE

Lo ZIN si aspetta che le istruzioni del linguaggio assemblato siano costruite secondo la sintassi definita nel manuale di programmazione di linguaggio assembler ZILOG Z80: lo Zin devia da questo standard in un caso particolare: quando vuole EX AF, AF. Potete trovare la lista completa delle istruzioni Z80 in un apposito manuale di Assembler Z80, un buon libro è "Programmazione dello Z80" scritto da Rodney Zaks e pubblicato in Italia dal Gruppo Editoriale Jackson.

Ogni istruzione in assembler può essere divisa in un massimo di quattro campi: Label, Operatore, Operando e Commento.

LABEL

Un'etichetta, cioè Label, è un modo di marcare un'istruzione cosicché le altre istruzioni possano riferirsi a questa. I numeri di linea servono allo stesso scopo nel BASIC, dove per esempio voi potreste usare GOTO 240 per saltare alla linea 240. Il linguaggio assembler vi permette di usare un nome simbolico per un'etichetta. Quando chiamate un'etichetta questa deve essere chiusa con due punti in modo tale che l'assemblatore sappia che è un'etichetta. Una Label deve cominciare con una lettera ma può contenere lettere o numeri dopo di questa. Lo ZIN permette etichette di qualsiasi lunghezza con tutti i caratteri che abbiano un qualche significato. Il registro e i nomi del codice di condizione non possono essere usati come simboli poiché sono considerati identificatori. Qualsiasi tentativo di farlo darà un messaggio di errore come risultato.

OPERATORE

Ci sono sessantasette operatori nel linguaggio As-

sembler Z80 in più lo ZIN ha sette pseudo operatori: END, DS o DEFS, DW o DEFW, DB o DEFB, EQU, ORG e LOAD.

END termina l'assemblaggio e deve essere sempre usato.

DS (Define Storage) salta sopra il numero di locazioni specificate dall'operando.

DW (Define Word) pone l'operando nel file oggetto in ordine inverso, come richiesto dalle istruzioni Z80 che hanno l'ampiezza di una parola.

DB (Define Byte) pone l'operando nel file oggetto alle successive locazioni. Gli operandi sono delimitati da virgole; ogni operando può essere un'espressione con valore inferiore a 256 o una stringa di lettere. Le stringhe di lettere possono avere qualsiasi lunghezza ma non possono essere parte di un'espressione.

EQU (Equate) dà il valore di un operando a un identificatore simbolico.

Qualsiasi identificatore simbolico usato nell'operazione degli operandi deve essere già conosciuto dall'assembler.

Questa regola "nessun riferimento più avanti" è stata studiata per prevenire riferimenti circolari.

ORG (Origin) definisce l'indirizzo iniziale del file oggetto. Questo pseudo-operatore può essere usato ogni volta che è necessario produrre sezioni di codice in diversi indirizzi. La regola del "nessun riferimento anticipato" vale anche per l'operando.

LOAD (Carica) comincia a caricare il codice in memoria all'indirizzo definito dall'operando. L'uso di uno pseudo-operatore ORG successivo disattiva questa operazione, e sarà necessario ristabilire in modo esplicito l'operazione di caricamento.

OPERANDI

In numero di operandi in un'istruzione dipende dall'operatore. Ci sono operatori niladici, monadici e diadici nel set di istruzioni dello Z80. Questi prendono rispettivamente zero, uno e due operandi.

Ci sono tre classi di operandi:

- 1) Registri (A,B,C,D,E,H,L,I,R,HL,DE,BC,AF,IX, IY,SP);
- 2) Codici di condizione (NZ,Z,NC,C,PO,PE,P,M);
- 3) Espressioni numeriche.

Un'espressione numerica è composta da uno o più dei seguenti elementi delimitati dagli operatori Infix Math:

1) Un numero decimale, esadecimale o ottale. Il decimale è la base di omissione per i numeri esadecimali che sono postfixati in "H" e per gli ottali postfixati in "O". I numeri devono cominciare con una cifra, uno zero iniziale è necessario con alcuni numeri esadecimali.

2) Una lettera inclusa in quote singole o doppie.

3) Il carattere \$. Questa variabile imita il Program Counter del programma Run-Time.

4) Un nome simbolico. L'assembler userà il valore associato nel valutare l'espressione.

Gli operatori infix math sono: + Addizione; - Sottrazione; * Moltiplicazione; / Divisione; & AND Logico; . OR Logico. Le espressioni sono valutate strettamente da destra a sinistra con nessun ordine particolare di precedenza. L'aritmetica è un numero intero non segnato di sedici bit. L'overflow sarà ignorato.

COMMENTI

I commenti sono ignorati dall'assembler. Cominciano con un punto e virgola (;) e terminano con un fine-linea.

TRATTAMENTO DEGLI ERRORI

Se l'assemblatore trova un errore di sintassi può accadere uno degli eventi seguenti:

- 1) L'assemblaggio termina;
- 2) Viene visualizzato un messaggio di errore;
- 3) La linea sbagliata viene visualizzata e diventa la linea corrente dell'Editor;
- 4) Il comando loop viene di nuovo registrato.

Adesso potete correggere l'errore e riassemblare. E' impossibile fare un errore di sintassi capace di danneggiare lo ZIN o qualcosa in memoria. I messaggi di errore sono:

UNDEFINED - Avete usato un simbolo non dichiarato

SYMBOL - Avete dichiarato un simbolo di lunghezza zero o avete dimenticato un simbolo necessario con uno pseudo-operatore EQU

RESERVED - Avete cercato di usare una parola riservata a un simbolo

FULL - La tabella dei simboli è piena

DOUBLE SYMBOL - Avete dichiarato più di una volta lo stesso simbolo

EOF - Avete dimenticato END e battuto fine-file

ORG! - Avete dimenticato ORG

HUH? - L'assembler è completamente confuso

OPERANDO - Avete fatto qualcosa di sbagliato con un operando e questo può significare un mucchio di cose! La maggior parte degli errori di sintassi cadono sotto questa categoria così come gli errori di quantità. Questo capita quando tentate di stampare troppo lontano con un salto o con istruzioni per il registro indice.

GOLDEN MELODIES

CHITARRA

ORGANO

TROMBA

“voci” soliste
per le più belle musiche d'amore



3 COMPACT DISC
AL PREZZO DI L. 29.900



CDGN 003
CDGN 012
CDGN 002

Desidero ricevere l'offerta "GOLDEN MELODIES" cod. CD5

Allego assegno ricevuta versamento
+ L. 2.500 quale contributo spese postali

NOME _____ COGNOME _____

VIA _____ N. _____

C.A.P. _____ CITTA' _____

Firma _____

Compilare il coupon allegando ricevuta (o fotocopia) del versamento effettuato sul C/C n. 11319209 intestato a Gruppo Editoriale International Education srl oppure assegno non trasferibile e spedire a:

**Gruppo Editoriale
International Education srl**
viale Famagosta 75
20142-Milano

DISPOSITIVO A DUE STADI ON & OFF

Per meglio capire cosa si intenda quando si parla di un dispositivo a due stadi (on-off) sarà meglio ricorrere ad un esempio.

Immaginiamo una lampadina in un certo istante: essa può essere accesa oppure spenta. Ogni calcolatore è basato su questo principio, ogni suo componente, in un dato istante può essere o non essere attraversato da una carica elettrica.

I due stadi, sono corrisposti al livello fisico da due tensioni differenti come potrebbero essere 0 Volt e 5 Volt oppure una tensione negativa ed una positiva ecc., ecc.

I due stadi sono invece corrisposti a livello logico dalle cifre 0 e 1 appartenenti al sistema binario, sistema di numerazione che si serve esclusivamente delle cifre 0 e 1 per rappresentare tutti i numeri del sistema decimale.

Il binario dà la possibilità di rappresentare in modo molto comodo la situazione interna dei componenti elettronici del calcolatore, quindi per meglio assimilare il suddetto concetto, potremmo dire che al numero binario 1 corrisponde un passaggio di corrente mentre al numero binario 0 non corrisponde un passaggio di corrente.

Questi singoli numeri si chiamano BIT, dall'inglese "Binary digit".

COME RAGIONA UN CALCOLATORE E PERCHÉ?

Se non vi siete mai posti questa domanda, questo è il momento per farlo.

Infatti il modo migliore per capire come è architettato un calcolatore è quello di vedere come lavora ossia come pensa.

Desiderando risolvere un problema molto banale, come quello di riconoscere il nome Claudio in un elenco di altri dieci nomi, si consideri dapprima di assegnare questo problema ad un essere umano.

Per risolverlo, è necessario fornire all'uomo due informazioni: un'istruzione (trova il nome Claudio) e l'elenco dei nomi.

Usando in modo accorto esclusivamente occhi e cervello, l'uomo individuerà rapidamente il nome desiderato.

Anche il calcolatore, per risolvere il problema necessita di due elementi: l'insieme delle istru-

zioni (che ora chiameremo programma) e la lista dei nomi (dati).

La differenza tra l'istruzione per il ragazzo e quelle per il calcolatore è che quest'ultimo necessita di più precise e numerose istruzioni.

Il programma consiste infatti in una serie di istruzioni che operino su dei dati specifici.

L'uomo per ricordare dei dati fa ricorso alla propria memoria, e analogamente deve fare il computer. Esso possiede infatti una memoria dove poter immagazzinare i dati ma non per poter svolgere calcoli.

Per svolgere i calcoli infatti vi è, diciamo, un reparto chiamato unità aritmetica.

Siccome in questo reparto possono essere svolte anche operazioni logiche (ad esempio confrontare due numeri) il suo nome completo è "ALU" unità aritmetico-logica.

Per spostarsi dall'ALU alla memoria, i dati fanno uso del BUS DATI e del BUS INDIRIZZI assimilabili per semplicità e dei veri e propri autobus adibiti al trasporto dei dati e dei loro indirizzi di memoria.

Tutto ciò è controllato dalla cosiddetta "UC", unità centrale, che è responsabile del movimento dei dati verso l'ALU, delle decisioni relative a quali operazioni aritmetiche o di confronto eseguire e del ritorno dei risultati in memoria centrale.

Tutti questi sono gli unici componenti presenti all'interno di ogni computer che nell'insieme vengono raccolti nella "CPU" ossia Unità Centrale di Processo.

I DISPOSITIVI DI INPUT/OUTPUT

Un dispositivo di ingresso di un calcolatore ha lo scopo di permettere l'introduzione delle informazioni e dei dati nella memoria centrale.

Per fare ciò, il dispositivo di ingresso converte i caratteri alfanumerici (lettere, numeri, simboli e punteggiatura) in formato binario in modo tale che possano essere contenuti nella memoria centrale.

Un dispositivo di uscita ha praticamente lo scopo inverso, ovvero traduce le informazioni immagazzinate in memoria in formato binario, in un formato leggibile dall'uomo.

Vi sono attualmente numerosi tipi di dispositivi di ingresso/uscita: lettori di schede perforate, tastiere, stampanti, monitors, mouses, disk dri-

ves, etc.

Questi dispositivi vengono tutti controllati dall'unità centrale ma siccome non fanno parte della CPU (Unità Centrale di Processo) vengono anche chiamati periferiche del calcolatore.

LA PROGRAMMAZIONE: CONCETTI BASILARI

Ora che a grandi linee siamo a conoscenza di come funziona un calcolatore, possiamo parlare di programmazione o meglio dei concetti elementari e basilari indispensabili per essa.

Per prima cosa si sa che lo scopo di un programma è quello di risolvere un problema.

Risolvere un problema significa ricercare un procedimento risolutivo (d'ora in poi chiamato anche algoritmo) che, eseguito fornisca delle informazioni finali (risultati, dati di uscita) dipendenti da alcune informazioni iniziali (dati di ingresso).

Queste informazioni iniziali e finali devono essere sempre codificabili mediante sequenze finite di simboli dette stringhe (caratteri) e di numeri naturali detti dati.

E' bene chiarire bene la differenza sostanziale fra dati e informazioni: i primi non costituiscono di per sè informazione, ma la rappresentano quando riescono a portare qualche nuova conoscenza a chi riceve questi dati (in questo caso l'esecutore che attende i dati iniziali da una parte e l'utente che aspetta di vedere i risultati dall'altra).

Chiarita quindi questa differenza, parleremo genericamente di <dati> distinguendo, se necessario, fra dati di ingresso, dati intermedi e dati di uscita.

I dati intermedi sono quelli che possono essere creati o trasformati dall'esecutore della procedura risolutiva durante il calcolo, cioè durante l'esecuzione della procedura risolutiva o algoritmo.

USO INTERATTIVO E NON INTERATTIVO DI UN SISTEMA DI CALCOLO

Come abbiamo già detto, esiste una figura chiamata utente che introduce i dati nel sistema di

calcolo e attende i risultati.

Il suddetto utente può lavorare in due modi:

A) modo non interattivo: l'utente inserisce i dati di ingresso tutti in una volta e preleva i risultati tutti in una volta.

B) modo interattivo: l'utente introduce una parte dei dati, attende un primo risultato, in base al quale decide quale nuovo ingresso fornire e così via fino al termine del colloquio: esiste quindi una interazione fra l'utente e la macchina.

Quest'ultima soluzione è ormai permessa da tutti i nuovi sistemi di calcolo in commercio ed è, per fare un semplice esempio quella utilizzata in tutti i videogames.

I sistemi interattivi sono sicuramente preferibili per la loro notevole immediatezza nelle risposte e sono gli unici possibili in quelle applicazioni in cui il tempo ha un ruolo determinante (applicazioni in tempo reale).

Per meglio capire cosa si intenda per applicazione in tempo reale sarà opportuno fare un esempio

Immaginiamo un sistema di calcolo che con appropriate apparecchiature elettriche debba sondare le radiazioni presenti nei pressi di una centrale nucleare.

E' ovvio che la risposta del calcolatore debba essere immediata e ripetuta ogni frazione di secondo per evitare un eventuale emergenza nucleare.

Dovendo giocare a scacchi con un sistema interattivo è sufficiente che l'utente comunichi di volta in volta la propria mossa perché il sistema conosce già lo stato della scacchiera e le informazioni riguardanti la propria strategia.

FINE SECONDA PARTE

IL LIBRO DEL MESE

COMPUTER CRIME, VIRUS, HACKERS

Metodi di indagine e strumenti di prevenzione – Realizzato da: IPACRI - Istituto per l'Automazione delle Casse di Risparmio Italiane

Edito da: Buffetti Editore

Pagine: 310 – Prezzo: 28.000 lire

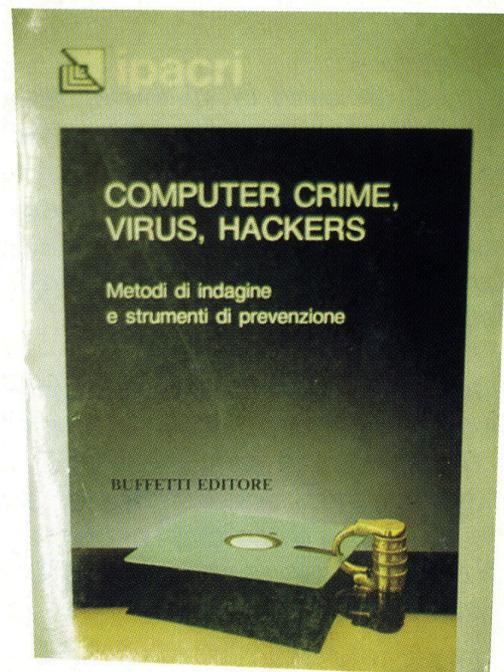
Anche questa volta parliamo di un libro dedicato ai virus, ma questa volta si tratta più di un rapporto sulla situazione italiana che di un libro descrittivo delle caratteristiche dei virus. Ma non solo virus: anche computer crime e furto elettronico.

I rischi connessi ai crimini perpetrati a mezzo del computer non possono essere più sottovalutati.

Sempre più frequentemente si registrano attacchi alla sicurezza dei dati, dei sistemi e delle reti, non solo a fini economici, ma spesso a scopi dimostrativi o ideologici.

I problemi in gioco sono talmente vasti da interessare le aree più diverse, dagli aspetti tecnici legislativi, agli assicurativi, per cui specialmente negli ultimi tempi si assiste a sempre più ampi sforzi per sistematizzare gli argomenti e fornire soluzioni organizzative e tecniche realmente adeguate.

Per dare uno sguardo d'insieme alla materia del computer crime e fornire lo spunto per ulteriori riflessioni e dibattiti, il Club sul Computer Crime ha organizzato il seminario internazionale nel corso del quale sono state concretamente



mostrate le intrusioni degli "hackers" in alcune reti, l'azione dei virus, nonché le difese atte a diminuire o vanificare questi attacchi criminali.

Questo libro contiene gli atti del seminario, cioè tutto ciò che è stato esposto dai relatori.

Importantissimo è proprio l'apporto di questi esperti che con le loro relazioni offrono un'apertura sulle dimensioni del fenomeno nel nostro paese.

Il libro tende a evitare una terminologia prettamente tecnica ed è anzi fin troppo "politichese", forse un po' noioso, ma si possono trovare dati decisamente interessanti.

Tutto l'insieme non è male, ma si vede che è più una registrazione di un seminario che un libro vero e proprio.

In conclusione dobbiamo dire che il Club sul Computer Crime è un'associazione fondata da diversi istituti bancari del nostro Paese nata con lo scopo di combattere in ogni modo il crimine elettronico e la diffusione dei virus.

Nel libro sono contenute tutte le informazioni relative a questa associazione e il modo per entrare in contatto con questa e altre associazioni.

Più di un anno fa un terremoto scosse il mondo dell'informatica, fino ad allora frasi come 'lotta alla pirateria', 'tutela del software', 'rispetto dei diritti d'autore', erano pure e semplici asserzioni, buone intenzioni non incidenti sul malcostume, non solo individuale ma aziendale, diffuso nel nostro paese.

L'Italia era tra i primi tre paesi al mondo dove si copiava di più, è stato calcolato che nel 1987 per ogni dieci personal Ibm e compatibili venduti solo 3.5 pacchetti sono stati acquistati legalmente e sembra ovvio che l'acquisto di un PC non è realizzato per usare solo l'Ms-Dos e il basic forniti con la macchina.

Questo e la perdita globale superiore ai quattro miliardi di dollari all'anno hanno portato alla creazione della BSA la Business Software Alliance da parte di sei tra i più grossi produttori americani di pacchetti applicativi, tutti grandi nomi: Microsoft, Lotus, Wordperfect, Autodesk, Aldus e Ashton-Tate.

Un agguerrito gruppo di avvocati che scelsero come primo campo di battaglia proprio il nostro Paese causando il terremoto in un mercato ormai quasi legittimato a copiare a più non posso.

Nella rete caddero subito la Montedison e altre società del gruppo di Gardini, seguite da Enichem e altre società chimiche.

In ognuna di queste aziende gli ufficiali giudiziari rilevarono situazioni limite che la stessa BSA non si aspettava.

L'esempio Montedison vale per tutti, nei soli uffici della sede centrale di Milano più del novanta per cento del software esaminato risultava riprodotto illegalmente.

Fin da subito la BSA è stata disposta a raggiungere un accordo prima della convocazione in aula con le aziende citate in giudizio, e ad oltre un anno di distanza nei tribunali non si è visto ancora nessuno e nessun compromesso è stato

reso noto.

Tornando al caso Montedison sappiamo, grazie ai programmatori che l'hanno fatto, che il direttore dei sistemi informatici ha fatto in modo che in ogni disco rigido di ogni PC della società sia stata data vita a due partizioni: la prima di cui è responsabile la Montedison e che contiene solo programmi originali, e la seconda di cui è responsabile il singolo utente che, ovviamente, non sarà mai perseguito legalmente come può essere fatto con una grande società.

Non è di poca importanza il fatto che la stessa Montedison abbia fatto inserire nella partizione personale dell'utente anche programmi copiati illegalmente.

Quando si è parlato di azioni legali molte aziende si sono destate improvvisamente dal comodo torpore e sono iniziati i sudori freddi: vedere arrivare un ufficiale giudiziario pronto a incriminare ogni atto di illegalità informatica non è certo piacevole, meglio fare ammenda di propria iniziativa.

In pochi giorni una smisurata quantità di ordini sono pervenuti a tutti i distributori di software, il fenomeno non è sembrato solo emotivo: è stato sicuramente più accentuato per alcuni prodotti che non per altri e ora si può vedere come una crescita del volume d'affari ci sia effettivamente stata.

Prima dell'estate '89 la BSA e l'Assoft, l'Associazione Italiana dei Produttori di Software, introducono il "Codice di Comportamento dell'Utente di Programmi per Computer" con lo scopo di dissuadere l'azienda acquirente di programmi da permettere, ignorare, assecondare o incoraggiare, la copiatura illegale di software.

Tra i primi ad aderire ci sono stati il CNR e la SGS Thomson mentre FIAT e COMIT, cioè la Banca Commerciale Italiana, hanno dichiarato di condividere la filosofia dell'iniziativa ma senza sottoscriverne il codice.

PIRATERIA

Novembre '89:

il ministro di Grazia e Giustizia Vassalli porta al Consiglio dei ministri una proposta di legge per la tutela del software; la regolamentazione che per anni è stata auspicata e che è sempre mancata.

Gli entusiasmi sono alle stelle fra produttori e distributori e nessuno vuole polemizzare sugli aspetti più macchinosi del progetto.

Con il passare del tempo la riflessione si è imposta e un certo malcontento inizia a serpeggiare, si contesta l'introduzione di un elemento burocratico protagonista della nuova regolamentazione: il marchio SIAE.

Ammesso che si trovi il modo di marchiare e contraddistinguere il software efficacemente - un gruppo di studio dell'ente sta facendo i salti mortali per capire come - nella speranza che alla legge segua una norma attuativa, la preoccupazione è quella dell'aumento dei tempi e dei costi di consegna e in più esistono dei seri problemi sulla reale utilità del bollino.

Poco male se la legge - tanto agognata e tanto applaudita alla sua presentazione - a Settembre '90 non è stata messa all'ordine del giorno per la discussione in nessuno dei due rami del Parlamento, c'è chi addirittura nel settore si domanda se una legge sul software serva davvero.

L'orientamento della Magistratura è quello di considerare il software come un'opera dell'ingegno e dell'intelletto umano, da garantire così con la normativa sui diritti d'autore.

Ma pur con l'avallo della Corte di Cassazione, arrivato nel 1986, l'uso della normativa sui diritti d'autore non è vincolante, mentre la legge proposta sancisce proprio la validità dell'orientamento che la Magistratura ha tenuto in questi anni.

Se la situazione legislativa è in stallo, su un altro fronte sono state prese delle contromosse.

Proprio la Montedison per reagire a una campagna da lei giudicata scandalistica, e per evitare il più possibile danni alla propria immagine ha preso l'iniziativa e insieme ad altri grandi utenti italiani, tra cui Fiat, Enimont, Iri, Stet, Pirelli, Barilla, Rcs Editori e Mondadori, ha dato vita a quella che è stata impropriamente chiamata la Lega degli Utenti Italiani.

Lo scopo è quello di ridisegnare un nuovo tipo di approccio alla trattativa commerciale da proporre ai distributori e produttori di software.

Per ora viene proposto un decalogo e più avanti un convegno-incontro, organizzato dalla Lega, tutte le parti in causa potranno portare le obiezioni e le proposte che ritengono utili.

Che dire di quest'ultimo anno di lotta ai pirati?

Prima di tutto non si hanno sufficienti elementi per decretare una vittoria o una sconfitta poiché non si sa come e quanto sia stato intaccato il fenomeno.

Il volume d'affari è certamente cresciuto, con delle accentuate sproporzioni a seconda dei marchi e dei prodotti, ma se nell'87 la perdita, a causa della copiatura illegale, in valore nel nostro paese secondo la BSA era circa di mezzo miliardo di dollari, le cifre recuperate risultano al di sotto della metà dei danni.

Ritardi, tentennamenti, ripensamenti anche nelle strategie commerciali verso tutti i tipi di utenti, rischiano di compromettere quei pochi risultati fin qui ottenuti.



Da questo numero cercheremo di offrirvi delle informazioni su programmi e prodotti vari che nascono per il nostro sistema.

A quanto pare è nato un virus anche per il nostro MSX. Sembra che in Giappone esista una versione Msx Dos di un virus che si riproduce abbastanza facilmente. Purtroppo non sappiamo molto di più ma appena possibile vi faremo sapere qualcosa.

Quello che vedete nella foto è l'Hard Disk per Msx. Completo di interfaccia SCSI, cioè quella in dotazione a MacIntosh e Atari, l'Hard Disk che vedete ha una capacità di 21 Megabyte ma ci sono anche altri modelli che arrivano a 49 Megabyte. Gli Hard Disk e le interfacce SCSI complete di software di gestione sono prodotte in Germania dalla HSH Computervertrieb GmbH, Rombergstrasse 16, D-4715 Davensberg, West Germany - Tel. xx49-2593/6168 - Fax. xx49-2593/7234. I prezzi in lire vanno dal 1.100.000 a 1.500.000 lire circa. In Italia lo potete trovare da: Msx Club Italia - Casella Postale 34 - 20075 Lodi Centro - MI.

Sempre dal Giappone abbiamo visto un gioco favoloso: YIS. Il pacchetto comprende un grossissimo manuale e una videocassetta con un film-cartone animato necessaria per imparare a giocare. La videocassetta VHS, in giapponese, permette di entrare nella storia da veri protagonisti. Nei prossimi numeri una recensione completa.

Dalla Corea è arrivato un prototipo di Msx 2 abbastanza strano: si chiama Victory ed è prodotto dalla Daewoo, nota multinazionale coreana, si tratta di una macchina tipo console senza tastiera ma con il solo ingresso cartucce in cui inserire i cartridge gioco. Nella confezione due keypad tipo Nintendo che permettono di giocare. La macchina, rosa e bianca, è praticamente una console Sega che, come non tutti sanno, è a tutti gli effetti un computer Msx 2. Nei prossimi numeri una recensione e molte foto.

Finalmente, nei prossimi numeri saremo in grado di offrirvi la prima recensione italiana dell'Msx 2+. Il sistema è davvero potente e siamo riusciti a provare quello prodotto dalla Sony. La grafica è incredibile e un'idea potete farvela dalle foto che vedete. Il computer è davvero potente: ha

un'uscita video RGB, 80 kb di ROM per il Basic versione 3, 16 Kb di ROM per il DiskBasic, 16 Kb di ROM per il TurboBasic e molto altro.

I prezzi in lire partono dalle circa 700.000 lire del PANASONIC Msx 2+ al 1.300.000 lire circa del Sanyo 2+.

L'importatore europeo è:

MSX CENTRUM - Witte de Withstraat 27 - 1057 XG Amsterdam - Nederland - Tel. 020-167058 - Fax. 020-167058.

In Italia potete averlo tramite l'Msx Club Italia - Casella Postale 34 - 20075 Lodi Centro - MI.

Da alcune riviste straniere abbiamo appreso che in Giappone è arrivato un ordine di 20.000 macchine Msx 2+ dall'Unione Sovietica. Perestroika MSX?

Tra i vari indirizzi che siamo in grado di darvi c'è quello di una società giapponese tramite la quale comprare hardware e software direttamente dal Giappone, ovviamente non possiamo garantire nulla.

L'indirizzo è:

Global English Network, Mr. Robin Von Hoegen, Osari 555, Nishizucho, 410-35 Shizuoka-Ken, Giappone.

Indirizzi utili più vicini a noi sono: Mauber Electronics - 95, DB Saint German, F-75005 Paris, Francia; sempre in Francia: Msx Video Center - 89 BIS, Rue de Charenton, F-75012 Paris; in Spagna: LASP, ALFONSO I' 28 - 50003 ZARAGOZA.

Tra le molte interfacce viste all'estero c'è un acceleratore per Msx 2 che porta la velocità da 3.58 Mhz a circa 7 Mhz e costa circa 80.000 lire. La stessa ditta offre una specie di Ramdisk velocissimo che permette di lavorare molto velocemente con dati di qualsiasi tipo tutto in Msx Dos. Costa circa 125.000 lire in versione 256K oppure 65.000 lire nella versione 64k.

L'indirizzo è:

MK Public Domani, Libellendans 30, 2907 RN Capelle a/d IJssel - Nederland - Tel. 010 / 4581600.

Questa società possiede anche una libreria di software di pubblico dominio di circa 2400 titoli e propone anche diverse interfacce Msx come Schede di Sintesi Vocale (Speech Cards), interfacce Hard Disk SCSI, buffer per stampante e un'espansione da 4 Megabyte di Ram.

I prodotti di questa società dovrebbero essere importati in Italia dall'Msx Club Italia probabilmente da Ottobre in poi.

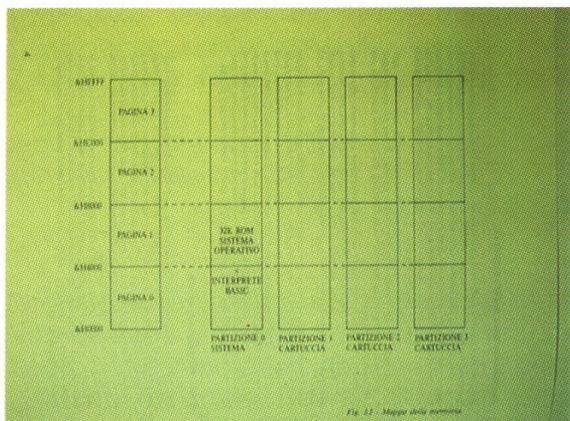


Fig. 11 - Mappa della memoria

Da questo numero ha inizio una nuova serie di articoli che vi porteranno dentro l'Msx, alla scoperta dei segreti di questa nostra Cenerentola degli Home Computer. Parleremo delle caratteristiche della macchina, della mappa di memoria, di come gestire video e suono e delle routine della Rom.

INTRODUZIONE

Lo standard MSX è una specifica introdotta a metà del 1983 dalla Microsoft Inc.. Esso comprende il linguaggio BASIC, il sistema operativo e alcuni connettori esterni, come porta joystick, interfaccia stampante e cartuccia Rom. Tutti i modelli - tranne lo Spectravideo SVI e lo Yamaha CX5M - sono dotati di alimentazione interna. Sono rigidamente specificate alcune caratteristiche non richieste nel sistema base: una porta parallela tipo Centronics per stampante, una seconda porta per il joystick, un'interfaccia RS-232C - disponibile internamente con alcuni modelli Toshiba HX - e un secondo alloggiamento per cartucce di memoria - è il caso del Sanyo e del Philips NMS8280.

Il sistema operativo MSX-DOS usa per i dischi il medesimo formato dell'MS-DOS, si trova con formattazioni da 360kb o da 720kb, e richiede un minimo di 64Kb di memoria RAM. L'unità centrale di elaborazione (CPU) è uno Z80-A, o equivalente, che funziona a 3.58 Mhz, affiancato da un processore video Texas Instruments TMS 9129 chiamato VDP. Questo chip ha 16k di Ram dedicata al video e consente di definire fino a 32 sprite, ciascuno di un solo colore. Tutti gli sprite appaiono davanti al piano di visualizzazione dei caratteri. Il VDP genera l'unico segnale di interruzione del sistema. Questo avviene al termine di ogni scansione del video - cioè ogni 1/50 di secondo circa - e viene usato, per esempio, per analizzare lo stato della tastiera.

Lo Z-80 ha una capacità di indirizzamento diretta pari ad un massimo di 64Kb, cioè 8 bytes per 8 bytes, ma nei sistemi con più di 64Kb la memoria viene gestita a banche da 64 Kb. All'accensione la Rom di sistema occupa i 32K di memoria ad indirizzo più basso. Viene inoltre fornito un minimo aggiuntivo di 8Kb di Ram di memoria utente. In pratica, il metodo di gestione della memoria comporta un'espansione di quest'ultima Ram aggiuntiva ad almeno 16kb, ma quasi tutti i sistemi ne forniscono 32kb o addirittura 64kb.

L'interprete BASIC è in grado di utilizzare solo 32kb di Ram - indipendentemente da eventuali Ram aggiuntive sulle schede - dei quali 28kb circa disponibili per la memorizzazione dei programmi. Il Basic MSX è il Basic Standard Microsoft (versione 4.5) con alcune estensioni, soprattutto per le elaborazioni grafiche e musicali. Le principali caratteristiche comprendono un editor a schermo intero, comandi guidati da interruzioni e una precisione di 14 cifre sulle variabili di tipo default.

La capacità di produrre suoni sono affidate ad un Generatore di Suoni Programmabile (PSG) General Instruments AY-3-8910, che consente l'utilizzo di tre voci su un'estensione di tre ottave. Il PSG inoltre gestisce l'input da joystick. La specifica base per lo standard Msx indica una sola porta standard per joystick di tipo D, cioè con due tasti di fuoco distinti, ma tutti gli Msx prodotti fino ad oggi ha due porte gioco.

Una PPI (Interfaccia Periferica Programmabile) 8255 Parallel I/O viene usata per selezionare i bank di memoria che saranno visti dal processore in quattro "alloggiamenti", o "partizioni", di 16Kb e per la scansione della tastiera. Se viene fornita una porta per stampante, che a tutti gli effetti è in dotazione a tutti gli Msx prodotti, è anch'essa servita dall'8255. Una seconda estensione opzionale sottoposta a standard è un'interfaccia RS-232C. I componenti LSI necessari per quest'ultima sono il chip d'interfaccia di comunicazione 8251 e un temporizzatore ad intervallo programmabile 8253.

L'ORGANIZZAZIONE DELLA MEMORIA

Lo Z80 può leggere e scrivere dati da e in un'area di memoria fino a sessantaquattro kilobyte. Il progetto MSX consente di scegliere questi 64K byte in una memoria più ampia. Questo banco di memoria è composto da quattro "partizioni" principali, o slot, ciascuna delle quali può contenere fino a 64K. La selezione della memoria viene gestita in blocchi, o pagine, di 16k ciascuno. L'area di memoria indirizzabile dallo Z80 consiste quindi in quattro di queste pagine da 16K. La porta A del PPI 8255 viene usata per determinare da quale delle suddette partizioni viene fornito ciascun blocco da 16k. I due bit meno significativi indicano quale tra esse fornisce il bloc-

co da 0 a 16k e così via per i bit di ordine maggiore. Si noti che un blocco può essere visto dal processore solamente alla stessa locazione che esso occupa nella partizione principale, vale a dire, per esempio, che il blocco della quarta partizione che copre gli indirizzi da 48k a 64k può apparire al processore solo a questi indirizzi e non da 0 a 16 o da 32 a 48k. Le partizioni principali sono numerate da 0 a 3. Nella partizione 0 trovano posto i 32k della Rom del sistema operativo e dell'interprete. La maggior parte degli elaboratori Msx è dotata di 64k di Ram. Quest'ultima è di solito posta in un'unica partizione. Per esempio, il Sanyo MPC-100 usa a questo scopo il numero 3 mentre il Toshiba HX-10 usa la partizione numero 2. Una di quelle che restano è per la cartuccia nastro, mentre l'altra può essere configurata sia per un'altra cartuccia sia per una espansione del bus. La memorizzazione dei programmi Basic inizia dall'indirizzo 32768.

LE PORTE DI INGRESSO/USCITA

Oltre a poter indirizzare 64k di memoria, lo Z80 può effettuare delle operazioni di ingresso e uscita su 256 porte di otto bit ciascuna. Le porte dalla 128 alla 255 sono assegnate alle componenti del sistema e alle eventuali espansioni: RS-232C, stampante, processore video, generatore suoni, penna luminosa, etc. Le restanti sono riservate. Il NMI (Interrupt Non Mascherabile) non è utilizzabile poiché il vettore locazione 66H viene usato dall'Msx Dos. Le operazioni di ingresso e uscita dovrebbero essere effettuate usando le procedure della Rom residente, poiché lo standard non garantisce che le locazioni di sistema siano le stesse sulle varie macchine. Un'eccezione può essere fatta per il processore video, dove può rivelarsi necessario un trasferimento rapido di dati.

INTERFACCIA CASSETTE

Lo standard Msx non richiede la presenza di un'unità nastro dedicata. La velocità di trasferimento di default è di 1200 baud, cioè 1200 bit al secondo, ma può essere elevata a 2400. La rilevazione e l'adeguamento ad una velocità differente da quella standard avviene automaticamente. La connessione esterna è una basetta DIN a 8 pin con la possibilità di controllo del motore del nastro. La modulazione avviene per mezzo di uno spostamento di frequenza sotto il controllo del software. I livelli ottimali per la registrazione e per la riproduzione vengono determinati dal registratore a cassette. Spesso, però, è necessario un livello assai vicino al massimo. Il motore dell'unità nastro può essere controllato per mezzo dell'istruzione MOTOR del Basic. Sono consentiti tre tipi di file:

Programmi memorizzati su cassetta. I comandi relativi sono CLOAD, CSAVE e CLOAD? per verifica-

re. File in formato ASCII, per i quali si usa SAVE e LOAD. Non è invece disponibile un comando per la verifica. I file programma ed i file Ascii possono essere mischiati.

Immagine del contenuto della memoria, per le quali si usa BSAVE e BLOAD. Anche in questo caso non è disponibile un comando di verifica.

MODALITA' DI VISUALIZZAZIONE

Il TMS 9129A è in grado di visualizzare un piano video con un massimo di 15 colori più il trasparente indipendentemente da qualsiasi sprite. Sono disponibili quattro modalità di visualizzazione, due per i testi e due per la grafica:

- 1) Modalità testo con 40*24 caratteri, due colori e nessun sprite, matrice 6*8 punti per la visualizzazione del carattere, un insieme di 256 caratteri disponibili.
- 2) Modalità testo con 32*24 caratteri, 16 colori, matrice carattere di 8*8 punti, insieme di 256 caratteri disponibili.
- 3) Modalità grafica ad alta risoluzione con 32*24 caratteri e 16 colori. Si tratta essenzialmente di una modalità simile a quella per i testi, con 32*24 caratteri e con un insieme di caratteri su 768 celle per permettere uno schermo "bit-mapped", più l'informazione extra del colore.
- 4) Modalità multicolore, 16 colori, ciascun blocco di 4*4 pixel può essere di diverso colore. Non sono disponibili i caratteri.

La modalità di visualizzazione viene selezionata usando l'istruzione Basic SCREEN oppure andando a modificare i tre bit che definiscono la modalità in due degli otto registri a sola scrittura del VDP (M3, bit 6 del registro 0, M1 ed M2, bit 3 e 4 del registro 1). In tutte le modalità, tranne in quella per i testi a 40 colonne, il colore del bordo può essere definito indipendentemente. Il VDP scandisce i 16kb della memoria Ram dedicata al video.

Questa RAM è svincolata dallo spazio di memoria dello Z80 e può essere letta o scritta dalla CPU solo attraverso il VDP.

Quest'ultimo non fornisce alcuna funzione per lo scroll dello schermo.

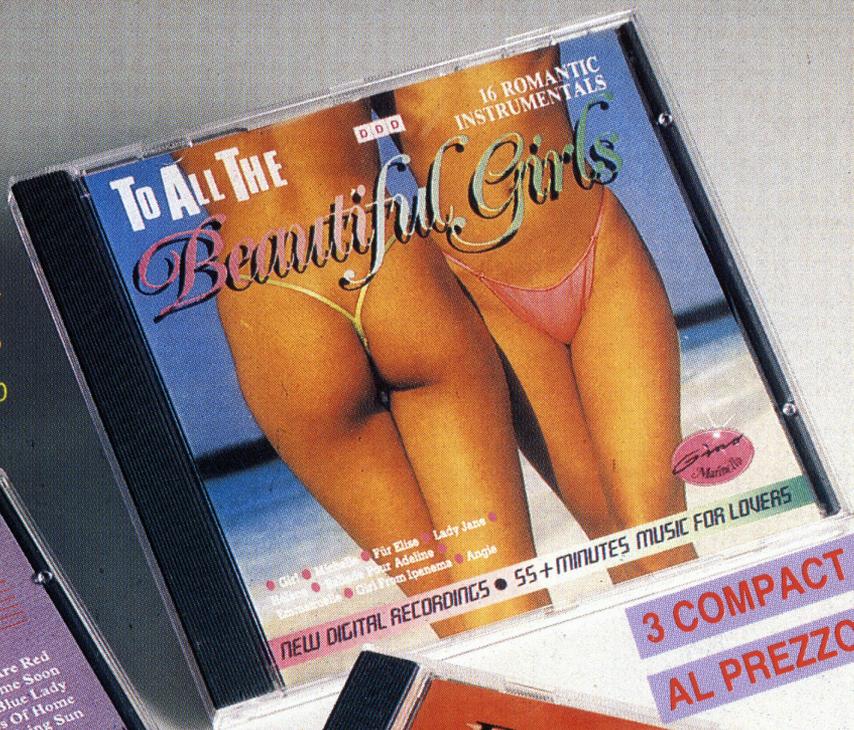
L'istruzione Basic per pulire lo schermo CLS è in qualche modo inusuale in quanto funziona in tutte le modalità di visualizzazione.

Questo è tutto per la prima parte.

Nel prossimo numero introdurremo il VDP e gli integrati del suono e poi passeremo ad analizzare il Basic Standard Microsoft dell'MSX.

MUSIC FOR LOVERS

- ENDLESS NIGHT CD 01034
- BEAUTIFUL GIRLS CD 01035
- SUNLIGHT MELODIES DGC 1020



3 COMPACT DISC
AL PREZZO DI L. 29.900

**Scopri nuove magiche
atmosferae con le più belle
melodie d'amore**

SERIE "STRUMENTALI"

Desidero ricevere l'offerta "MUSIC FOR LOVERS 2" cod. CD9		Compilare il coupon allegando ricevuta (o fotocopia) del versamento effettuato sul C/C n. 11319209 intestato a Gruppo Editoriale International Education srl oppure assegno non trasferibile e spedire a: Gruppo Editoriale International Education srl viale Famagosta 75 20142 Milano
Allego	assegno <input type="checkbox"/> ricevuta versamento <input type="checkbox"/> + L. 2.500 quale contributo spese postali	
NOME	COGNOME	
VIA	N.	
C.A.P.	CITTÀ	
Firma		

SPLENDIDI INEDITI DEI MOSTRI SACRI DEL JAZZ

○ CHARLIE PARKER CDJJ 610
○ BENNY GOODMAN CDJJ 609
○ COUNT BASIE CDJJ 604
○ SIDNEY BECHET CDJJ 603

○ DIZZY GILLESPIE CDJJ 606
○ DUKE ELLINGTON CDJJ 602
○ LIONEL HAMPTON CDJJ 605

7
COMPACT DISC
AL PREZZO DI
L. 84.000



Desidero ricevere l'offerta "JAZZ" codice CD7
Allego assegno ricevuta versamento
+ L. 2.500 quale contributo spese postali

NOME _____ COGNOME _____
VIA _____ N. _____
C.A.P. _____ CITTÀ _____
Firma _____

Compilare il coupon allegando ricevuta (o fotocopia) del versamento effettuato sul C/C n. 11319209 intestato a Gruppo Editoriale International Education srl oppure assegno non trasferibile e spedire a:

**Gruppo Editoriale
International Education srl**
viale Famagosta 75
20142 Milano

JAZZ