

CADEL

Spediz. in abbonamento postale GR II/70 L. 2.200
(...)

64 CORSO PRATICO COL COMPUTER

è una iniziativa
FABBRI EDITORI

in collaborazione con
BANCO DI ROMA

e **OLIVETTI**

diretto da **GIANNI DIOLI ANTONI**



**FABBRI
EDITORI**

IL BANCO DI ROMA FINANZIA IL VOSTRO ACQUISTO DI M 10 e M 20

Acquisto per contanti

È la formula di acquisto tradizionale. Non vi sono particolari commenti da fare, se non sottolineare che troverete ampia disponibilità presso i punti di vendita Olivetti, poiché, grazie al "Corso pratico col computer", godrete di un rapporto di privilegio.

Il servizio di finanziamento bancario

Le seguenti norme descrivono dettagliatamente il servizio di finanziamento offerto dal Banco di Roma e dagli Istituti bancari a esso collegati:

Banca Centro Sud
Banco di Perugia

Le agenzie e/o sportelli di questi istituti sono presenti in 216 località italiane.

Come si accede al credito e come si entra in possesso del computer

- 1) Il Banco di Roma produce una modulistica che è stata distribuita a tutti i punti di vendita dei computer M 10 e M 20 caratterizzati dalla vetrofania M 10.
- 2) L'accesso al servizio bancario è limitato solo a coloro che si presenteranno al punto di vendita Olivetti.
- 3) Il punto di vendita Olivetti provvederà a istruire la pratica con la più vicina agenzia del Banco di Roma, a comunicare al cliente entro pochi giorni l'avvenuta concessione del credito e a consegnare il computer.

I valori del credito

Le convenzioni messe a punto con il Banco di Roma, valide anche per le banche collegate, prevedono:

- 1) Il credito non ha un limite minimo, purché tra le parti acquistate vi sia l'unità computer base.
- 2) Il valore massimo unitario per il credito è fissato nei seguenti termini:
 - valore massimo unitario per M 10 = L. 3.000.000
 - valore massimo unitario per M 20 = L. 15.000.000
- 3) Il tasso passivo applicato al cliente è pari

- al "prime rate ABI (Associazione Bancaria Italiana) + 1,5 punti percentuali".
- 4) La convenzione prevede anche l'adeguamento del tasso passivo applicato al cliente a ogni variazione del "prime rate ABI"; tale adeguamento avverrà fin dal mese successivo a quello a cui è avvenuta la variazione.
- 5) La capitalizzazione degli interessi è annuale con rate di rimborso costanti, mensili, posticipate; il periodo del prestito è fissato in 18 mesi.
- 6) Al cliente è richiesto, a titolo di impegno, un deposito cauzionale pari al 10% del valore del prodotto acquistato, IVA inclusa; di tale 10% L. 50.000 saranno trattenute dal Banco di Roma a titolo di rimborso spese per l'istruttoria, il rimanente valore sarà vincolato come deposito fruttifero a un tasso annuo pari all'11%, per tutta la durata del prestito e verrà utilizzato quale rimborso delle ultime rate.
- 7) Nel caso in cui il cliente acquisti in un momento successivo altre parti del computer (esempio, stampante) con la formula del finanziamento bancario, tale nuovo prestito attiverà un nuovo contratto con gli stessi termini temporali e finanziari del precedente.

Le diverse forme di pagamento del finanziamento bancario

Il pagamento potrà avvenire:

- ☐ presso l'agenzia del Banco di Roma, o Istituti bancari a esso collegati, più vicina al punto di vendita Olivetti;
- ☐ presso qualsiasi altra agenzia del Banco di Roma, o Istituto a esso collegati;
- ☐ presso qualsiasi sportello di qualsiasi Istituto bancario, tramite ordine di bonifico (che potrà essere fatto una volta e avrà valore per tutte le rate);
- ☐ presso qualsiasi Ufficio Postale, tramite vaglia o conto corrente postale. Il numero di conto corrente postale sul quale effettuare il versamento verrà fornito dall'agenzia del Banco di Roma, o da Istituti a esso collegati.

 **BANCO DI ROMA**
CONOSCIAMOCI MEGLIO.

Direttore dell'opera
GIANNI DEGLI ANTONI

Comitato Scientifico
GIANNI DEGLI ANTONI
Docente di Teoria dell'informazione, Direttore dell'Istituto di Cibernetica dell'Università degli Studi di Milano

UMBERTO ECO
Ordinario di Semiotica presso l'Università di Bologna

MARIO ITALIANI
Ordinario di Teoria e Applicazione delle Macchine Calcolatrici presso l'Istituto di Cibernetica dell'Università degli Studi di Milano

MARCO MAIOCCHI
Professore incaricato di Teoria e Applicazione delle Macchine Calcolatrici presso l'Istituto di Cibernetica dell'Università degli Studi di Milano

DANIELE MARINI
Ricercatore universitario presso l'Istituto di Cibernetica dell'Università degli Studi di Milano

Curatori di rubriche
MARCO ANELLI, DIEGO BIASI, ANDREA GRANELLI, ALDO GRASSO,
MARCO MAIOCCHI, DANIELE MARINI, GIANCARLO MAURI,
CLAUDIO PARMELLI

Testi
CLAUDIO PARMELLI, GIANLUIGI ROCCA, ANDREA GRANELLI,
ALDO GRASSO, Eidos (MARIDA DI CROSTA)

Tavole
Logical Studio Communication
Il Corso di Programmazione e BASIC è stato realizzato da Etnoteam S.p.A., Milano
Computergrafia è stato realizzato da Eidos, S.c.r.l., Milano
Usare il Computer è stato realizzato in collaborazione con PARSEC S.N.C. - Milano

Direttore Editoriale
ORSOLA FENGLI

Redazione
CARLA VERGANI
LOGICAL STUDIO COMMUNICATION

Art Director
CESARE BARONI

Impaginazione
BRUNO DE CHECCHI
PAOLA ROZZA

Programmazione Editoriale
ROSANNA ZERBARINI
GIOVANNA BREGGÈ

Segretarie di Redazione
RENATA FRIGOLI
LUCIA MONTANARI

Corso Pratico col Computer - Copyright (©) sul fascicolo 1985 Gruppo Editoriale Fabbri, Bompiani, Sonzogno, Einaudi S.p.A., Milano - Copyright (©) sull'opera 1984 Gruppo Editoriale Fabbri, Bompiani, Sonzogno, Einaudi S.p.A., Milano - Prima Edizione 1984 - Direttore responsabile GIOVANNI GIOVANNINI - Registrazione presso il Tribunale di Milano n. 135 del 10 marzo 1984 - Iscrizione al Registro Nazionale della Stampa n. 00262, vol. 3, Foglio 489 del 20.9.1982 - Stampato presso lo Stabilimento Grafico del Gruppo Editoriale Fabbri S.p.A., Milano - Diffusione - Distribuzione per l'Italia Gruppo Editoriale Fabbri S.p.A., via Mecenate, 91 - Milano - tel. 02/50951 - Pubblicazione periodica settimanale - Anno II - n. 64 - esce il giovedì - Spedizione in abb. postale - Gruppo II/70. L'Editore si riserva la facoltà di modificare il prezzo nel corso della pubblicazione, se costretto da mutate condizioni di mercato.

RETI LOCALI (III)

Gli standard presenti sul mercato e le principali reti presenti.

Gli standard

La presenza sul mercato di apparecchiature di tipo diverso affacciate su reti di topologia e supporto di trasmissione diversi ha portato alla necessità di definire uno standard che permetta a un numero sempre maggiore di stazioni di interconnettersi e quindi di poter comunicare in modo semplice.

Da alcuni anni è presente un modello di standard a cui si tende sempre più a fare riferimento: il modello OSI (*Open System Interconnection*) definito dall'ISO (*International Standard Organisation*) che è valido per qualsiasi tipo di rete. Sebbene questo non rappresenti ancora uno standard di fatto, in quanto sono ancora in corso battaglie per la definizione precisa di alcune parti di esso, è un punto di riferimento per tutti i produttori che vogliono aprire il loro prodotto a scambi con reti e apparecchiature di altri produttori.

Il modello ISO-OSI stabilisce una segmentazione in sette livelli (figura a destra) del processo di trasmissione di informazioni su di una rete; crea quindi delle aree ben delimitate all'interno delle quali operare e attraverso le quali l'utente dovrebbe essere in grado di interagire con il sistema senza dover conoscere i particolari dei sottostanti livelli.

I diversi livelli dello standard OSI sono:

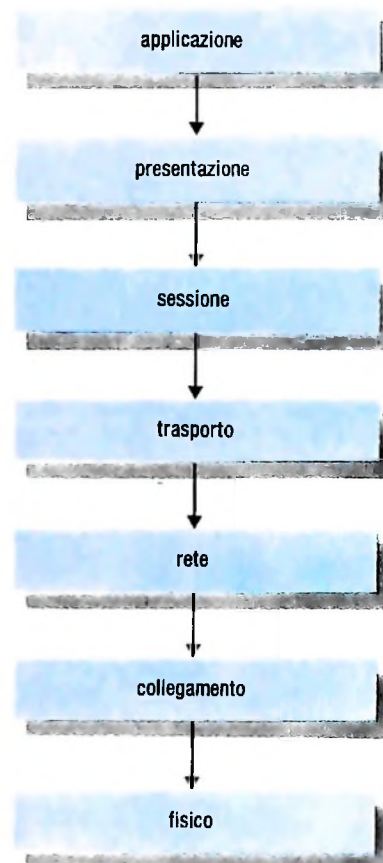
LIVELLO 1 o livello fisico: si riferisce alle specifiche elettriche e meccaniche di una certa apparecchiatura, come per esempio cavi, connettori e circuiti elettrici, e quindi alla struttura propriamente fisica della rete.

LIVELLO 2 o livello di collegamento: comprende i protocolli di base che regolano l'interscambio delle informazioni e si occupa, per esempio, della formazione dei pacchetti che costituiscono il blocco di informazione che deve essere trasmesso e della sincronizzazione degli stessi, nonché della rilevazione di anomalie. Gli schemi di risoluzione dei conflitti di rete (come il CSMA/CD) possono essere collegati a questo livello. In questo livello si controllano anche eventuali errori di trasmissione che vengono però poi corretti da un livello superiore.

LIVELLO 3 o livello di rete: riguarda le funzioni tipiche di rete quali l'instradamento dei messaggi fra le apparecchiature connesse sulla rete o su reti diverse collegate fra loro.

LIVELLO 4 o livello di trasporto: comprende le attività legate alla comunicazione fra due entità e quindi i protocolli responsabili del trasporto *end-to-end* e cioè della trasmissione fra due elementi della rete. Questo livello assicura generalmente un trasporto *sicuro* dei pacchetti.

LIVELLO 5 o livello di sessione: i protocolli di questo livello

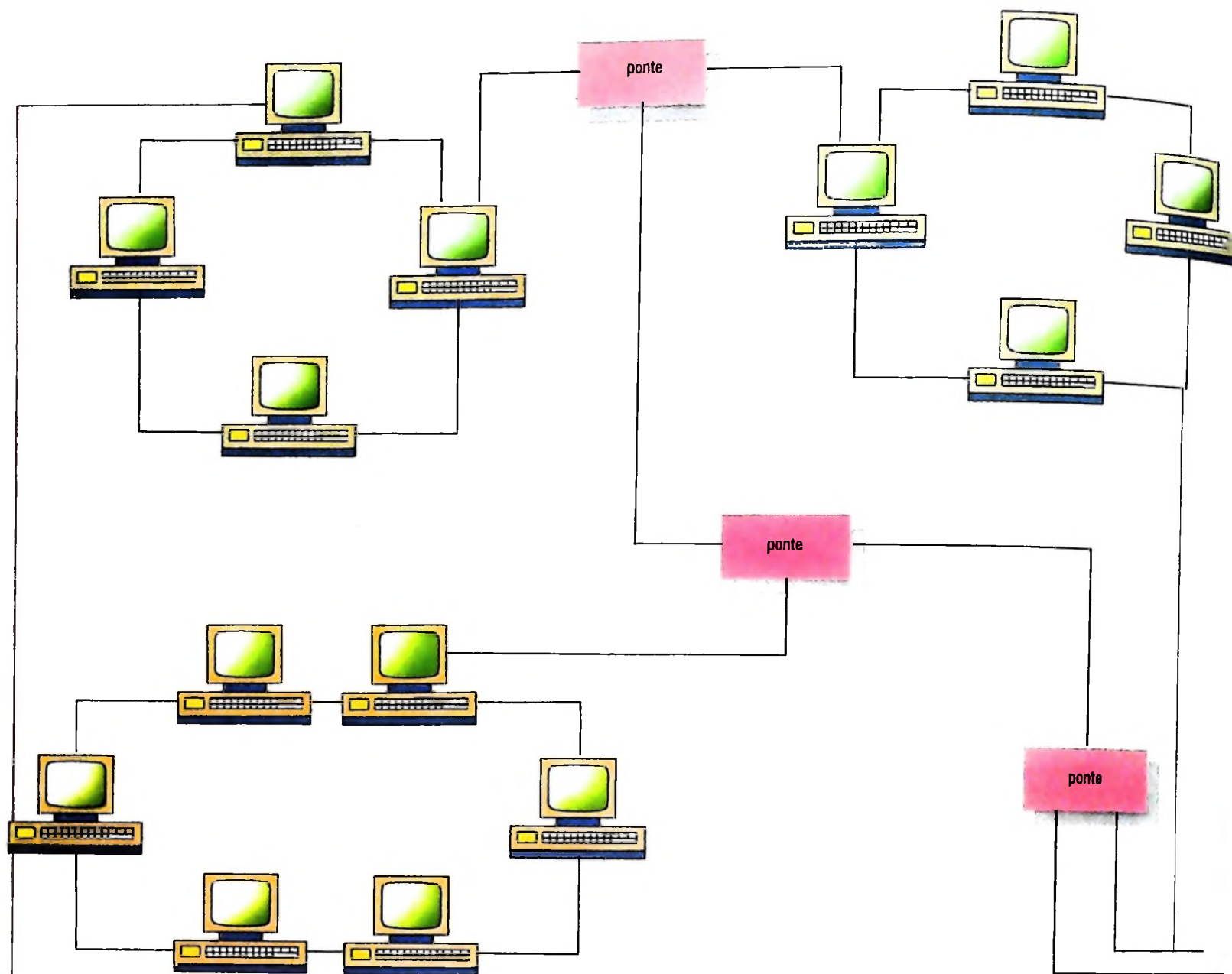


In figura è mostrata la segmentazione in sette livelli del processo di trasmissione di informazioni su di una rete, stabilito dal modello Open System Interconnection che è stato definito dall'International Standard Organisation (si parla di standard ISO OSI.)

sovrintendono alle operazioni di creazione, mantenimento e chiusura del canale di comunicazione e fanno in modo che l'interazione con il sistema sia più semplice; per esempio se due indirizzi al livello precedente sono visti come due numeri, a questo livello possono essere visti come due nomi di stazione.

LIVELLO 6 o livello di presentazione: si occupa dell'adattamento, modifica e traduzione dei dati. Fa in modo che le informazioni trasmesse o ricevute abbiano un determinato formato; si occupa per esempio della trasformazione dei dati e dei comandi contenuti in un file di un formattatore di testi.

LIVELLO 7 o livello di applicazione: corrisponde all'insie-



me dei programmi a disposizione dell'utente per gestire le funzioni di rete. È questo il livello che in generale risente meno della definizione di uno standard in quanto ogni produttore, ogni casa ne definisce uno proprio.

A partire da questa definizione generale di standard, poiché si tratta di uno schema di riferimento e inoltre lo standard è definitivo solo per i primi quattro livelli, ogni produttore sceglie la propria strada di implementazione dei protocolli di rete: possono aversi scelte molto diverse che variano da una implementazione essenzialmente software a una essenzialmente hardware. Nel primo caso è possibile una facile manutenzione e un'alta capacità di seguire i cambiamenti nello standard (un cambiamento di questo richiede solamente la riscrittura di una parte del software) ma si penalizzano le capacità del processore ospite che deve gestire la comunicazione a tutti i livelli. Nel secondo caso invece la realizzazione di una interfaccia con circuiti VLSI può permettere un notevole sfruttamento delle capacità della rete senza per questo sovraccaricare di lavoro il processore centrale: per contro la realizzazione hardware dei protocolli non permette di variarli facilmente in seguito a una mutazione di standard. È peraltro possibile sfruttare entrambe le tecniche e imple-

mentare in hardware i protocolli più standardizzati, cioè di basso livello, e con il software quelli di più alto livello, cioè più facilmente variabili.

Le reti: Ethernet, la rete IBM, AppleTalk

Vediamo le reti locali più diffuse ed in particolare Ethernet, la rete per i PC IBM, e la rete AppleTalk, per i Macintosh. Ethernet è quasi diventato un sinonimo di "rete locale"; si tratta di un marchio registrato della Xerox e la sua prima apparizione risale al 1970. Le sue ottime caratteristiche ne hanno fatto poi uno strumento utilizzato per la connessione di computer su scala locale.

Nel 1980 fu presentata la prima specifica tecnica, release 1.0, di Ethernet assieme con l'annuncio della sua adozione da parte di Xerox, Intel e Digital che l'avrebbero adottata come standard per la trasmissione di dati e di interconnessione fra le loro apparecchiature.

Quasi contemporaneamente a questo annuncio l'IEEE (*Institute of Electrical and Electronics Engineers*) decise di sviluppare alcuni standard per le reti locali, uno dei quali, IEEE

802.3, rispecchiava fedelmente le caratteristiche di base di Ethernet. Si trattava della definizione di una rete locale con una velocità di trasmissione di 10Mb/sec (Megabit = 1 milione di bit al secondo) basata su un cavo coassiale a larga banda con un metodo di accesso basato sul riconoscimento delle collisioni (CSMA/CD). L'IEEE sviluppò altre due definizioni di standard, 802.4 e 802.5, per le reti a bus e quelle ad anello, discusse in uno dei precedenti articoli.

Nonostante lo standard le apparecchiature che utilizzano Ethernet non sono ancora in grado di comunicare dati senza difficoltà in quanto la specifica si riferisce ai livelli più bassi e non a quelli di applicazione per i quali esistono due protocolli generalmente accettati: lo XNS (*Xerox Network System*) e il TCP/IP (*Transmission Control Procedure/Internet Protocol*) che si dividono il mercato e stabiliscono due aree incompatibili fra di loro.

Ogni interfaccia Ethernet costava dai duemila ai tremila dollari alla sua apparizione, ma ora con l'introduzione di interfacce usabili da più stazioni il costo per ogni connessione è sceso a circa 400 dollari. Inoltre l'introduzione di apparecchiature hardware per la gestione dei protocolli di comunicazione ha permesso di ridurre i costi e di facilitare la manutenzione della rete stessa.

La recente introduzione sul mercato della rete IBM e prossimamente di quella Apple hanno messo in crisi la supremazia di Ethernet nel mercato delle reti, ma sono previsti per questa degli aggiornamenti per risolvere alcuni dei suoi più evidenti problemi quali per esempio l'alto costo (è stato proposto l'uso di un cavo coassiale di dimensioni, e quindi capacità, minori utilizzabile però solo su corte distanze).

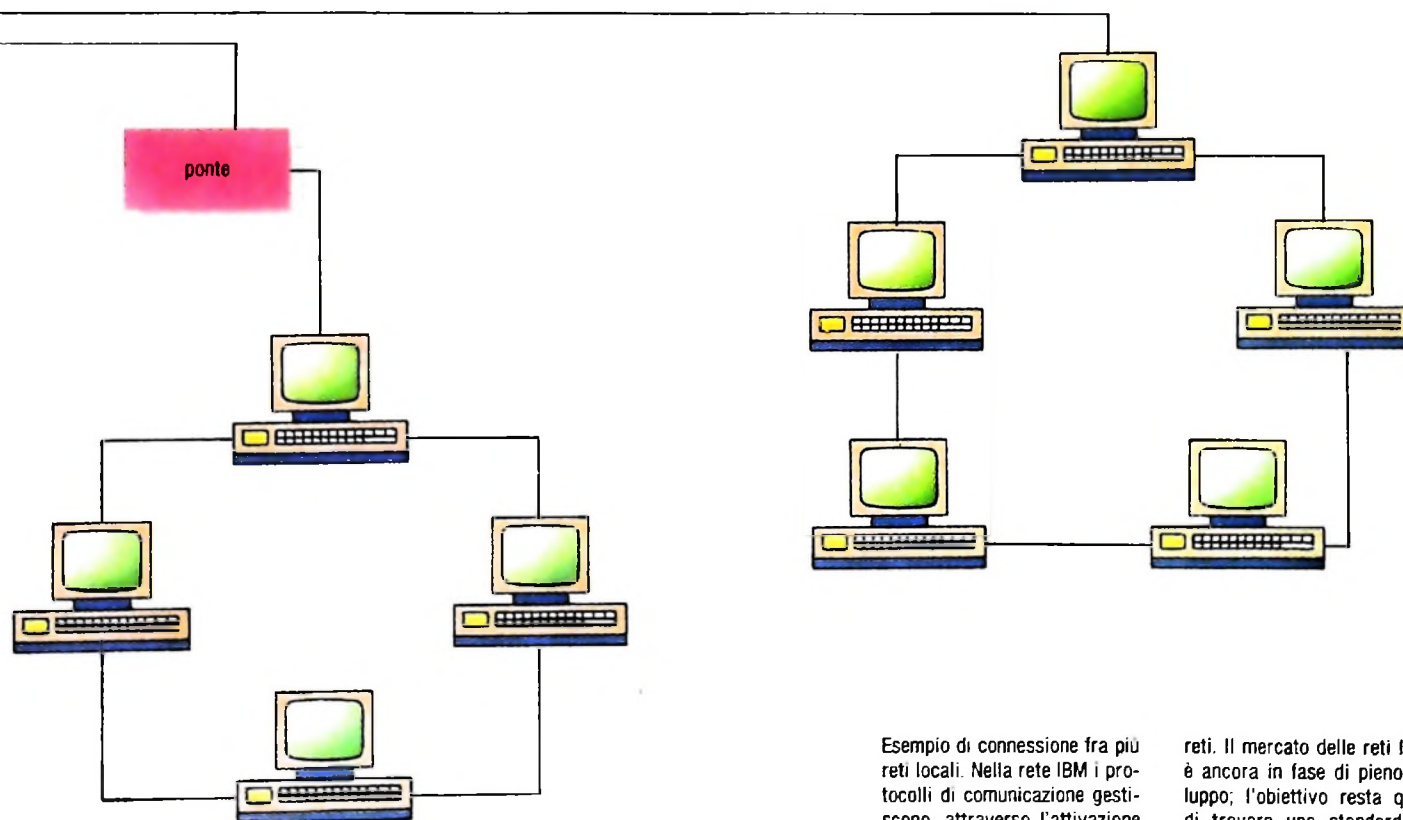
Prendiamo in considerazione la rete IBM.

La IBM ha recentemente introdotto una rete, la PCNetwork, e ne ha annunciata una seconda che verrà introdotta sul mercato successivamente. Per lo sviluppo della PCNetwork la IBM si è rivolta alla Sytek che ha prodotto una versione semplificata della propria rete, la LocalNet. La LocalNet è una delle reti più diffuse vantando circa 500 installazioni la più ampia delle quali, secondo la Sytek, connette circa 20 000 stazioni ed è una rete che sfrutta un cavo coassiale con tecnologia a larga banda su cui sono definiti numerosi canali ciascuno dei quali gestisce centinaia di stazioni.

Per la IBM la Sytek ha ridotto la enorme capacità della sua rete, eccessiva per una rete di PC, e ne ha fatto una versione con un solo canale allargando la banda base a 2 Mb/sec. In questa rete la maggior parte dei compiti di gestione è devoluta a una interfaccia hardware intelligente che permette di non sovraccaricare la CPU del PC IBM che con una velocità di 4.77 MHz non poteva certo essere in grado di gestire la rete senza mostrare un forte decadimento di prestazioni.

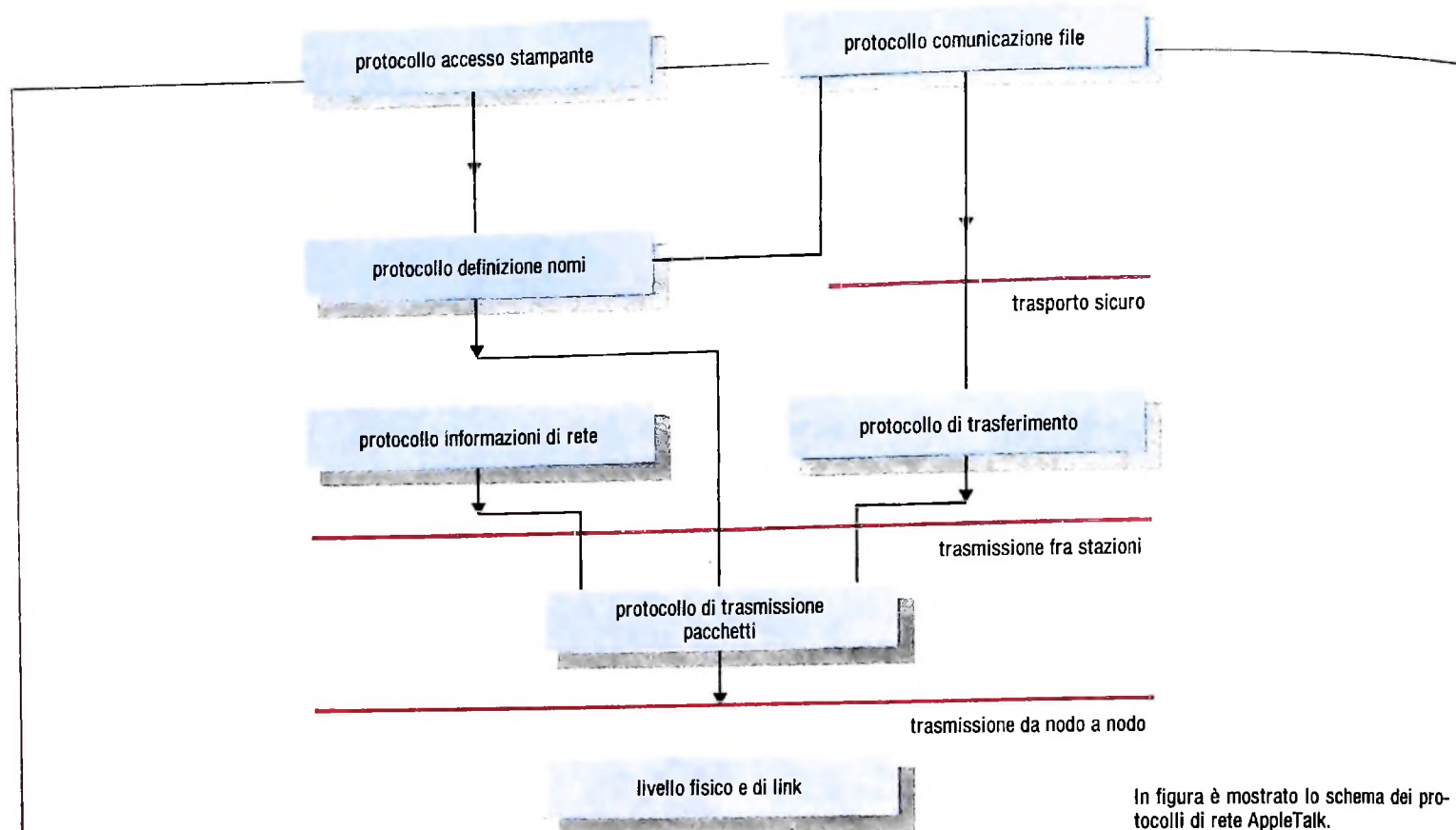
Attraverso l'uso di circuiti dedicati e di un proprio processore l'interfaccia Sytek, una delle più raffinate fra quelle esistenti, è in grado di gestire i primi 5 livelli della specifica ISO-OSI lasciando così al processore solamente la gestione dei livelli di presentazione e di applicazione.

I protocolli di comunicazione sono in grado di gestire, attraverso l'attivazione di opportuni ponti, la trasmissione di informazioni ad altre reti interconnesse il che fa prevedere lo sviluppo di reti del tipo della figura di queste pagine. Nonostante le molte e favorevoli caratteristiche di questa rete la strategia IBM a lungo termine si basa sulla seconda rete di



Esempio di connessione fra più reti locali. Nella rete IBM i protocolli di comunicazione gestiscono, attraverso l'attivazione di opportuni ponti, la trasmissione di informazioni ad altre

reti. Il mercato delle reti locali è ancora in fase di pieno sviluppo; l'obiettivo resta quello di trovare uno standard che consenta uno scambio di informazioni veramente libero.



prossima introduzione. Questa è una rete Token Passing (vedi precedenti articoli) configurata ad anello che rispecchia in pieno quelle che sono le definizioni dello standard IEEE 802. Una delle caratteristiche principali di questa rete sarà l'utilizzo, come supporto di trasmissione, di semplici doppietti schermati simili a quelli telefonici il che comporterà un costo di installazione particolarmente basso e una manutenzione piuttosto semplificata. In questa rete ogni stazione avrà una interfaccia intelligente per gestire i protocolli del Token Ring ed esisteranno dei concentratori su cui passerà il traffico diretto anche ad altre reti interconnesse.

Esaminiamo ora la rete AppleTalk.

Il concetto sottostante alla nuova rete locale Apple rappresenta una sostanziale novità: invece di produrre una rete con una grande larghezza di banda per connettere un grande numero di PC su di una ampia area la Apple ha prodotto una rete di basso costo con una velocità di trasmissione non alta adatta a connettere piccoli gruppi di PC.

L'AppleTalk si basa sulla distribuzione dell'intelligenza alle apparecchiature locali e sullo sfruttamento totale delle capacità di trasmissione della rete. Insieme alla rete è stata presentata una stampante laser che incorpora un potente microprocessore, più di un Mbyte di RAM e mezzo Mbyte di ROM. Il cuore di AppleTalk è costituito dal processore Zilog 8530 SCC che gestisce le comunicazioni sincrone e asincrone a una velocità di 230.4 Kb/sec e il supporto fisico di trasmissione è un doppietto che è connesso passivamente a un computer o a una periferica.

Una rete AppleTalk può connettere fino a 32 stazioni su di una distanza massima di 300 metri ed esternamente è composta da un cavo, molto piccolo, e da una scatoletta da cui escono due connettori che vanno attaccati al cavo che si inse-

risce nel computer. L'inserimento del cavo nella presa sul computer è l'unico passo che bisogna fare per inserire, o disinserire, una nuova stazione sulla rete e si può capire quale sia la differenza da un approccio tradizionale che in un caso simile richiede la presenza di un esperto.

Funzionalmente è basata sullo standard ISO-OSI (figura in alto) e lo schema di accesso alla rete è il CSMA/CA (CA = Collision Avoidance) che significa che tutti i trasmettitori aspettano fino a quando la rete non è libera e il tempo di attesa è determinato dalla generazione di un numero casuale che dipende anche dal traffico presente sulla rete stessa.

La rete Apple, sulla base della non alta velocità di trasmissione, codifica le informazioni che passano su di essa in modo che queste la occupino il meno possibile, sia in termini di spazio che di tempo, lasciando alle unità periferiche intelligenti la maggior parte dei compiti di traduzione e codifica.

Conclusioni

Con questo articolo abbiamo concluso la panoramica sulle reti locali e abbiamo dato uno sguardo sia al problema degli standard, sia ad alcune reti particolari, una delle quali, AppleTalk, sembra fatta in modo da essere più semplice e facilmente utilizzabile delle altre. D'altra parte esistono degli approcci tecnicamente sofisticati, e quindi più complessi quali Ethernet, che permettono una interconnessione fra stazioni più grandi che richiedono quindi una capacità di trasmissione più alta.

In definitiva il mercato delle reti locali appare in fase di crescita ed espansione ed ancora alla ricerca di uno standard che permetta uno scambio di informazioni senza vincoli.

M10 - TOOL (I)

I primi di una serie di cinque programmi utili per risolvere problemi di programmazione.

Renumberer

È il primo di una serie di programmi di utilità: permette di rinumerare un programma BASIC che sia stato salvato in formato ASCII sotto forma di file testo, quindi con suffisso .DO.

Oltre alle linee di programma, verranno rinumerati anche tutti i riferimenti contenuti in una linea (per esempio i GOSUB, GOTO, RESTORE ecc.), mentre non verranno presi in considerazione, e quindi non modificati, i riferimenti contenuti in linee REM.

Prima di effettuare l'operazione di Renumber, è necessario salvare il programma in formato ASCII con le linee ordinate sequenzialmente.

Per ottenere l'ambiente richiesto, è sufficiente caricare il programma con suffisso .BA in BASIC e quindi impartire il comando SAVE nomefile.DO.

Dopo la videata iniziale, il programma Renumberer presen-

terà l'elenco dei file memorizzati in M10 (in RAM) e quindi visualizzerà la domanda:

Nome del file .DO da rinumerare:

alla quale bisognerà rispondere indicando il nome del file testo esattamente come visualizzato. A questo punto verrà richiesto di inserire la nuova prima riga da cui partire (minimo 0). Se non si specifica il numero di linea, il programma assumerà automaticamente il valore 0, e l'incremento di riga (se non viene specificato, il programma assume 10 come incremento standard).

Al termine delle richieste, il programma provvederà alla rinumerazione del file, il tempo impiegato è proporzionale alla grandezza del programma da elaborare.

Terminata l'operazione, è conveniente, per risparmiare memoria, caricare il file .DO, salvarlo in BASIC (con suffisso .BA) e quindi cancellare il file testo.

Da ricordare: i numeri di linea inferiori a 0 e superiori a 65529 non vengono presi in considerazione.

RENUMBERER

by C.V.P.

```

16 RV$=CHR$(27)+"p"
   :NV$=CHR$(27)+"q"
17 CLS
   :LINE(32,5)-(207,18),1,BF
18 PRINT$55,RV$;"Renumberer";NV$
19 LINE(29,2)-(210,21),1,B
20 PRINT$204,"Premi un tasto per continuare..."
21 M$=INKEY$
   :IFM$=""THEN21
60 CLEAR500
70 GOSUB960
80 CLEAR500
   :IFMAXFILES<1THENMF=1
   :MAXFILES=1ELSEMF=0
90 PRINT
100 LINEINPUT"Nome del file .DO da rinumerare:";F$
105 IFF$=""THEN100
110 PRINT
120 PRINT"Inserire nuova prima linea:";
   :GOSUB2000
   :N=VAL(R$)

```



```

140 PRINT
150 PRINT"Inserire incremento di linea:";
    :GOSUB2000
    :L=VAL(R$)
160 IFL<=0THENL=10
162 IFN<0THENN=0
165 CLS
    :PRINT$132,"Fase di Renumber"
170 PRINT$212,"Attendere Prego.."
175 ONERRORGOTO3000
180 OPENF$FORINPUTAS1
    :I=1
190 LINEINPUT$1,A$
200 A=A+LEN(A$)
210 IFEOF(1)THEN230
220 I=I+1
    :GOTO190
230 CLOSE1
240 OPEN"I.DO"FOROUTPUTAS1
250 PRINT$1,I,N,L,MF,F$
260 CLOSE1
270 CLEARA+300
280 OPEN"I.DO"FORINPUTAS1
290 INPUT$1,I,N,L,MF,F$
300 CLOSE1
310 KILL"I.DO"
320 DIMA$(I),B$(I)
330 OPENF$FORINPUTAS1
340 FORJ=1TOI
    :LINEINPUT$1,A$(J)
    :NEXTJ
350 CLOSE1
360 FORJ=1TOI
    :FORK=1TOLEN(A$(J))
370 S$=LEFT$(A$(J),1)
380 IFASC(S$)<58ANDASC(S$)>47THENB$(J)=B$(J)+S$
    :A$(J)=RIGHT$(A$(J),LEN(A$(J))-1)ELSENEXTJ
390 IFJ>1THEN410
400 NEXTK
    :NEXTJ
410 FORJ=1TOI
    :S$=""
    :FO=0
    :FORK=1TOLEN(A$(J))+253
420 IFK LEN(A$(J))THEN560
430 IFMID$(A$(J),K,1)=""ANDFO=0THEN560
440 IFMID$(A$(J),K,1)=" "THEN550
450 IFMID$(A$(J),K,1)=CHR$(34)ANDFO=0THENFO=1
    :GOTO550
460 IFMID$(A$(J),K,1)=CHR$(34)ANDFO=1THENFO=0
    :GOTO550
470 IFFO=1THEN550
480 S$=S$+MID$(A$(J),K,1)
    :S$=RIGHT$(S$,4)
490 IFRIGHT$(S$,3)="REM"ANDFO=0THEN560
500 IFSS$="DATA"ANDFO=0THEN560
510 IFSS$="GOTO"ORS$="THEN"ORS$="ELSE"ORS$="OSUB"ORS$="SUME"ORS$="TORE"THENK1=0
    :GOSUB630
520 IFRIGHT$(S$,3)="RUN"THENK1=0
    :GOSUB630
530 IFSS$="EDIT"ORS$="LIST"THENK1=0
    :GOSUB630
540 IFSS$="LETE"THENK1=0
    :GOSUB630
550 NEXTK
560 NEXTJ
570 OPENF$FOROUTPUTAS1
580 FORJ=1TOI

```



```

590 PRINT f1, RIGHT$(STR$(J*L+N-L), LEN(STR$(J*L+N-L))-1); A$(J)
600 NEXT J
610 CLOSE 1
620 BEEP
    : IF MF=1 THEN MAXFILES=0
    : MENU ELSE MENU
630 IF K+1 > LEN(A$(J)) THEN RETURN ELSE IF K1=0 THEN K1=K+1 ELSE K1=K1+1
640 IF K1 > LEN(A$(J)) THEN PRINT "Errore in linea "; B$(J)
    : RETURN
650 S$=MID$(A$(J), K1, 1)
660 IF S$=" " AND K1+1 > LEN(A$(J)) THEN 880
670 IF S$=" " THEN 630
680 IF S$="," AND MID$(A$(J), K1-1, 1)="," THEN K=K1
    : GOTO 630
690 IF S$="," THEN 780
700 IF S$="-" THEN 830
710 IF ASC(S$) <= 47 OR ASC(S$) >= 58 THEN 750
720 IF ASC(S$) > 47 AND ASC(S$) < 58 THEN Z$=Z$+S$
730 IF K1=LEN(A$(J)) THEN 880
740 GOTO 630
750 IF Z$="" THEN S$=""
    : RETURN
760 GOSUB 910
770 A$(J)=LEFT$(A$(J), K)+Z$+RIGHT$(A$(J), LEN(A$(J))-K1+1)
    : K=K+LEN(Z$)
    : Z$=""
    : RETURN
780 IF Z$="" THEN S$=""
    : GOTO 630
790 GOSUB 910
800 A$=LEFT$(A$(J), K)+Z$+","+RIGHT$(A$(J), LEN(A$(J))-K1)
810 K=K+LEN(Z$)+1
    : Z$=""
    : S$=""
    : K1=0
820 A$(J)=A$
    : GOTO 630
830 IF Z$="" THEN S$=""
    : RETURN
840 GOSUB 910
850 A$=LEFT$(A$(J), K)+Z$+"-"+RIGHT$(A$(J), LEN(A$(J))-K1)
860 K=K+LEN(Z$)+1
    : Z$=""
    : S$=""
    : K1=0
870 A$(J)=A$
    : GOTO 630
880 IF Z$="" THEN K=K1
    : RETURN
890 GOSUB 910
900 A$(J)=LEFT$(A$(J), K)+Z$
    : Z$=""
    : RETURN
910 FORM=1 TO I
    : IF VAL(Z$)=VAL(B$(M)) THEN Z$=RIGHT$(STR$(M*L+N-L), LEN(STR$(M*L+N-L))-1)
    : GOTO 950
920 NEXT M
930 PRINT "Errore in nuova linea "; J*L+N-L
940 PRINT "(Vecchia linea "; B$(J); "): Linea "; Z$; " non trovata"
950 RETURN
960 CLS
970 FILES
980 A=65024
990 FOR J=1 TO 8
1000 B=PEEK(A+I)
1010 C$(J)=C$(J)+CHR$(B)
1020 NEXT J
1030 FOR J=1 TO 8

```

```

1040 D=INSTR(1,C$(J),"I .DO")
      :IFD<>0THEN1070
1050 NEXTJ
1060 RETURN
1070 SOUND2000,100
      :PRINT
      :PRINT"Rinominare file I.DO,altrimenti"
      :PRINT"si cancella!"
      :S
      :
1080 MENU
      :END
2000 INPUTR$
2005 IFR$=""THEN2100
2010 FORM=1TOLEN(R$)
2020 M$=MID$(R$,M,1)
2030 IFASC(M$)<48ORASC(M$)>57THEN2100
2040 NEXTM
2050 RETURN
2100 PRINT"Inserire un numero INTERO come"
      :PRINT"numero di linea od incremento di linea."
2120 GOTO2000
3000 BEEP
      :IFERR<>52THENRESUME20
3010 CLS
      :PRINT"***ERRORE** Il file da rinumerare"
      :PRINT"non esiste!"
3030 GOTO40

```

Modifica file .DO

Il seguente programma risolve parecchi problemi quando si tratta di effettuare, in un listato, modifiche di cui si conosce l'entità e il valore, ma non la corretta localizzazione.

Esempio classico, quando si deve adattare un programma che permette l'uscita su video in modo da consentirne l'uscita su stampante: bisogna quindi modificare tutti i PRINT in LPRINT.

Fino a ora, bisognava avere un listato del programma, identificare le linee contenenti l'istruzione PRINT e modificarle una a una.

Con il seguente programma l'operazione, a volte noiosa, di ricerca e sostituzione manuale, viene abolita, permettendo di

effettuare più modifiche solo premendo un tasto.

Un'unica raccomandazione va fatta, ed è che il programma da manipolare deve essere prima salvato in formato ASCII, quindi con suffisso .DO.

Lanciato il programma MODIFICA, verrà presentato l'elenco dei file presenti in RAM e richiesto di inserire il nome, senza suffisso, del file sul quale operare le modifiche. Soddisfatta la richiesta, il programma proseguirà richiedendo qual è la parola da sostituire e quale la nuova parola da inserire.

Ricordarsi, in questa fase, che la differenza tra lettere maiuscole e minuscole è importante (MILANO è diverso da MILANO) quindi inserire la parola correttamente.

La sostituzione è totale, nel senso che tutte le parole uguali a quella indicata verranno sostituite con la parola specificata.

MODIFICA FILE.DO

by C.V.P.

```

16 RV$=CHR$(27)+"p"
   :NV$=CHR$(27)+"q"
17 CLS
   :LINE(32,5)-(207,18),1,BF
18 PRINT$52,RV$;"Modifica FILE.DO";NV$
19 LINE(29,2)-(210,21),1,B
20 PRINT$204,"Premi un tasto per continuare..."

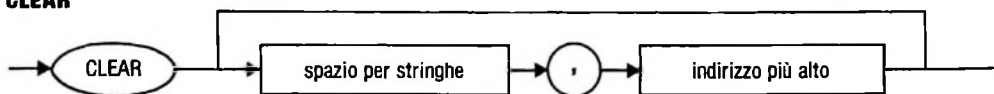
```


Comandi BASIC disponibili su M10

Storicamente, il BASIC è nato come sistema che divideva le ISTRUZIONI, da far sempre precedere da un numero di linea e costituenti i programmi, dai COMANDI, cioè ordini eseguiti direttamente, senza numero di linea, e operanti in genere sui programmi (come SAVE, RUN ecc.). Di fatto l'evoluzione del BASIC ha eliminato tale distinzione, cosicché è possibile eseguire in modo DIRETTO (cioè senza numero di linea) quasi tutte le istruzioni, come pure è possibile inserire comandi nei programmi, opportunamente preceduti da numero di linea.

Manteniamo qui la divisione tra istruzioni e comandi più per seguire una tradizione storica e per connotare diversamente le operazioni che operano su programmi, piuttosto che per una reale differenza operativa.

CLEAR



Ripulisce la memoria dati, inizializzando a 0 le variabili numeriche e a stringa nulle quelle stringhe. Lo "spazio per stringhe" è la quantità di memoria in caratteri destinata alle stringhe; l'"indirizzo più alto" è la massima localizzazione di memoria destinata ai programmi BASIC.

CLOAD

Vedi LOAD.

CLS



Ripulisce il display.

CONT

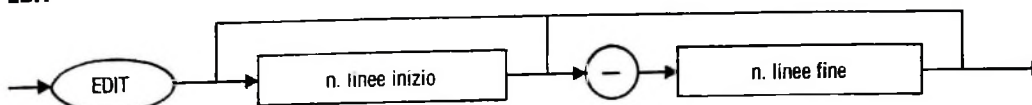


Permette di continuare l'esecuzione di un programma interrotta dall'istruzione STOP.

CSAVE

Vedi SAVE

EDIT



Cambia la modalità operativa sul programma corrente, trasformandolo in un testo di caratteri, e rendendo possibili tutte le operazioni di editing messe a disposizione del comando TEXT. La presenza dei numeri di linea specifica quale porzione di programma si vuole modificare (eventualmente una sola linea).

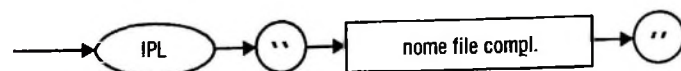
FILE



Visualizza i file memorizzati in RAM. Essi sono caratterizzati dai seguenti suffissi:

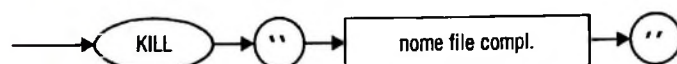
- .BA programma BASIC
- .CO programma in linguaggio macchina
- .DO testo

IPL



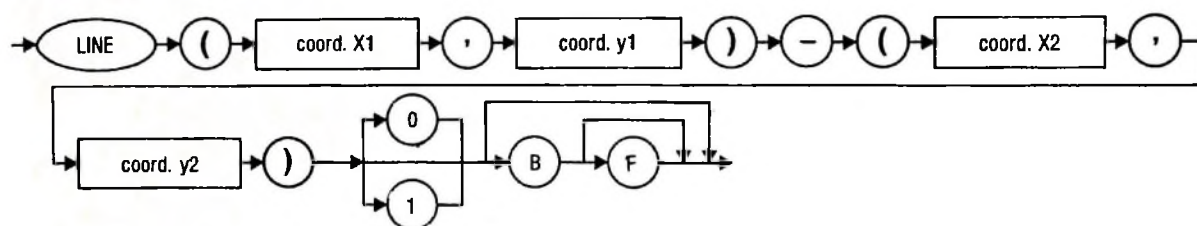
Acronimo di "Initial Program Load" (Caricamento di programma iniziale); provoca l'esecuzione senza ulteriore comando del file (deve essere completo di suffisso).

KILL



Distrugge un file in RAM. Il nome deve essere completo del suffisso (vedi FILE). Non può essere cancellato il file attualmente in uso.

LINE

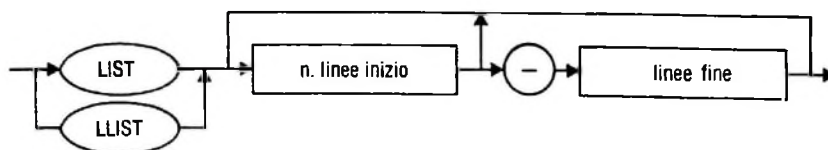


Traccia sullo schermo una linea dal punto specificato dalle prime coordinate a quello specificato dalle seconde (x compreso tra 0 e 239, y tra 0 e 63); se l'opzione B è specificata, viene tracciato un rettangolo la cui diagonale è costituita dal segmento specificato.

L'opzione 0 traccia in bianco su fondo scuro, mentre l'opzione 1 traccia in scuro.

L'opzione F riempie completamente il rettangolo.

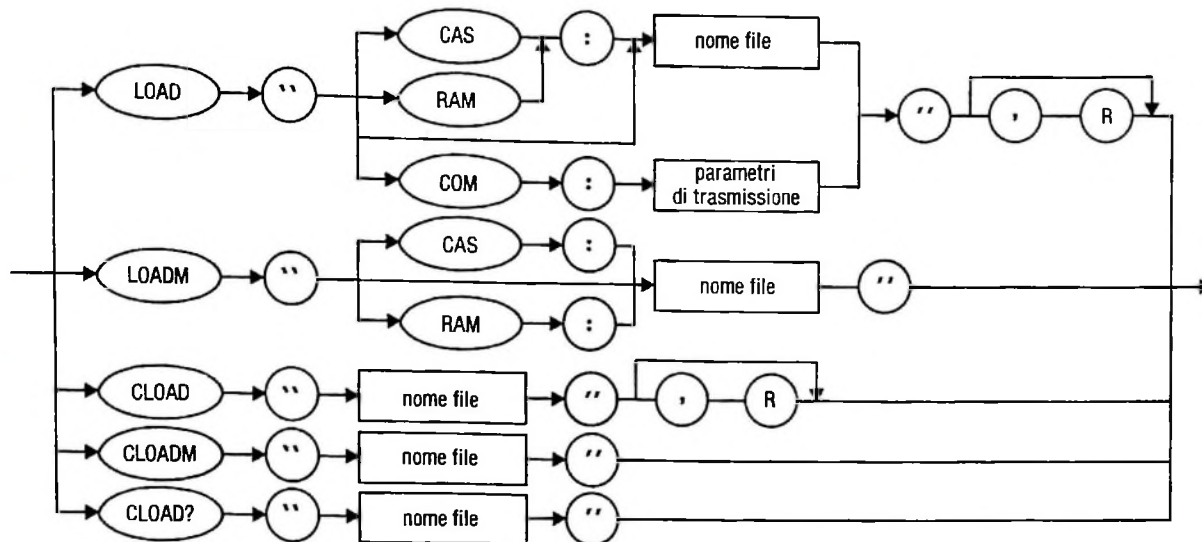
LIST



Lista il programma sullo schermo o su stampante (con LLIST), eventualmente per una parte specificata dai numeri di linea (anche una sola linea).

LLIST

Vedi LIST

LOAD

Carica un programma nell'area di esecuzione. L'opzione R innesca automaticamente l'esecuzione dopo il caricamento.

Il caricamento avviene:

da cassetta con CAS

con CLOAD

da RAM con RAM

senza specifica

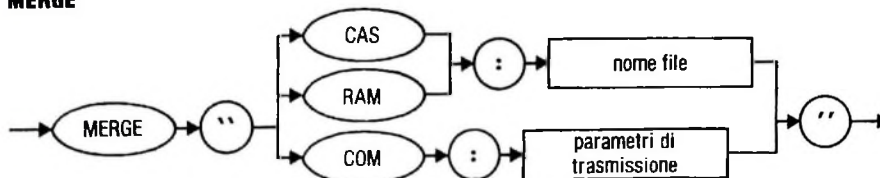
da linea con COM: in questo caso è necessario fornire i parametri di comunicazione (vedi istruzione OPEN per i dettagli)

Il caricamento con LOADM e CLOADM è relativo a programmi in linguaggio macchina, che verranno inseriti all'indirizzo specificato all'atto della loro memorizzazione con SAVEM.

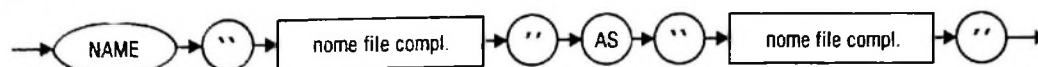
CLOAD? è riferito a programmi che sono stati appena memorizzati su cassetta, e ne verifica la corretta registrazione, rileggendo la cassetta e confrontando il contenuto con quello in memoria. In caso di errore di registrazione, viene segnalato il fallimento della verifica, indicando che l'operazione di SAVE deve essere ripetuta.

MENÙ

Richiede l'abbandono dell'ambiente BASIC per ritornare al menù principale dell'M10.

MERGE

Fonde il programma attualmente in memoria (che deve essere sotto forma di caratteri, e non di programma BASIC, ottenibile via SAVE), con quello specificato da cassetta, da memoria o da linea. Le linee vengono mescolate secondo il loro naturale ordinamento; per eventuali linee doppie vengono privilegiate quelle non in memoria.

NAME

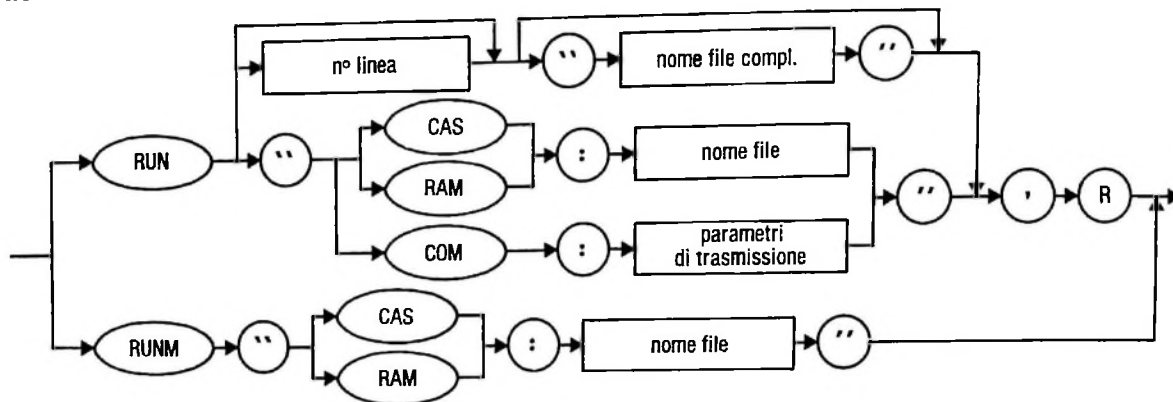
Ridefinisce il nome di un file, con il secondo nome citato. I nomi specificati devono essere completi di suffisso, che non può cambiare dal nuovo nome a quello vecchio.

NEW



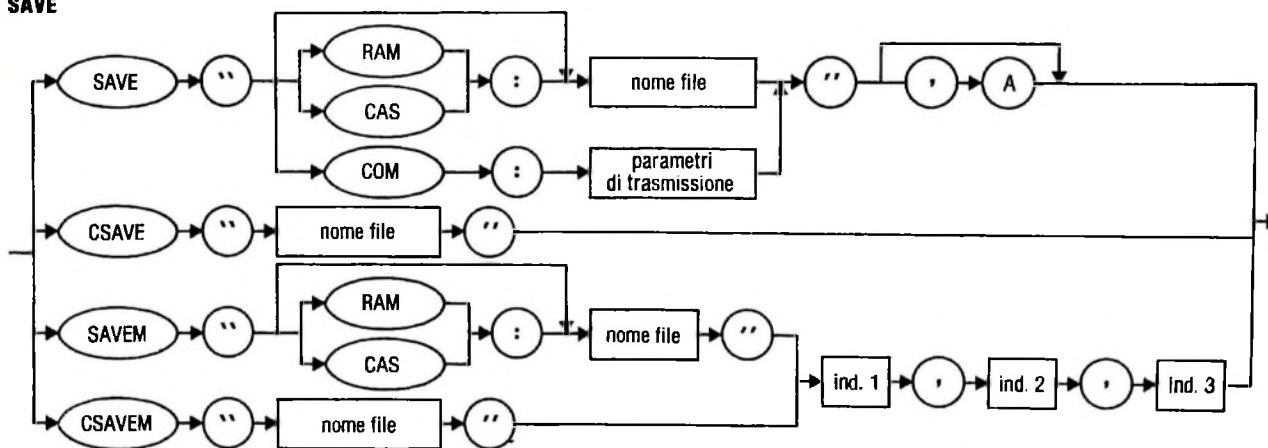
Cancella il programma attualmente presente nella memoria di lavoro.

RUN



Esegue un programma nella memoria di lavoro. L'opzione R mantiene aperti i file. Il programma, se non è già in memoria di lavoro, vi viene caricato distruggendone eventuali presenti. Il caricamento può avvenire da cassetta (CAS), da linea COM, nel qual caso è necessario specificare i parametri di trasmissione; per dettagli vedere l'istruzione OPEN, o da memoria (RAM). L'assenza di indicazioni del programma da eseguire fa eseguire quello in memoria; l'assenza di indicazione di provenienza lo fa prelevare da RAM. In questi due casi è possibile specificare il numero di linea da cui iniziare l'esecuzione. RUNM permette l'esecuzione di programmi registrati in linguaggio macchina.

SAVE



Permette di memorizzare il programma attualmente in memoria di lavoro. La memorizzazione può avvenire su RAM (se è specificato RAM o se non è specificata la destinazione), su cassetta (se è specificato CAS o se è usato CSAVE), o verso una linea (se è specificato COM: in tal caso è necessario specificare i parametri di trasmissione: vedi l'istruzione OPEN per dettagli). SAVEM e CSAVEM registrano programmi in linguaggio macchina; ind1, ind2 e ind3 sono indirizzi relativi alle locazioni in cui caricare i programmi (comandi LOADM o RUNM). L'opzione A permette la registrazione dei programmi sotto forma di testo invece che di programma BASIC, operabili quindi con comandi MERGE.


```

21 M$=INKEY$
   :IFM$=""THEN21
30 DEFINT A-Z
40 MAXFILES=2
   :CLEAR2048
50 DIMS1$(19),S2$(19)
60 ONERRORGOTO460
100 CLS
   :FILES
110 PRINT
   :F1$=""
   :ER=0
   :PRINT"    Su quale file operare la modifica:"
   :INPUTF1$
120 IFF1$=""THENMENU
130 OPENF1$FORINPUTAS1
140 IFERTHEN110
150 F2$=""
   :ER=0
   :PRINT
   :PRINT" Verrà creato un nuovo file .DO"
   :PRINT"quale nome desideri?"
   :INPUTF2$
170 I=INSTR(F2$,".")IFITHENF2$=LEFT$(F2$,I-1)
180 IFF2$=""THENF3$="CH$$$$"ELSEF3$=F2$
190 OPENF3$FOROUTPUTAS2
210 ONERRORGOTO0
220 N=-1
230 FORI=0TO19
240 S1$(I)=""
   :PRINT
   :PRINT"Quale parola in "F1$".DO vuoi"
   :INPUT"modificare";S1$(I)
260 IFS1$(I)=""THEN310
270 S2$(I)=""
   :PRINT
   :PRINT"La parola ";S1$(I);" deve essere"
   :INPUT"sostituita con:";S2$(I)
290 N=I
300 NEXTI
310 IFN<0THENCLOSE
   :KILLF3$+".DO"
   :STOP
   :MENU
320 IFEOF(1)THEN410
330 LINEINPUTF1$,L$
340 SL=LEN(L$)
350 FORI=0TON
360 J=INSTR(1,L$,S1$(I))
   :IFJ=0THEN380
370 PRINTF2$,LEFT$(L$,J-1);S2$(I);
375 L$=MID$(L$,J+LEN(S1$(I)))
   :GOTO360
380 NEXTI
390 IFSL=255THENPRINTF2$,L$;ELSEPRINTF2$,L$
400 GOTO320
410 CLOSE
420 IFF2$=""THENMENU
430 I=INSTR(F1$,".")
   :IFI=0THENF1$=F1$+".DO"
435 I=0
   :I=INSTR(F3$,".")
   :IFI=0THENF3$=F3$+".DO"
440 KILLF1$
   :NAMEF3$ASF1$
450 MENU
460 ER=-1
   :RESUMENEXT

```

GIOCHI DA TAVOLO

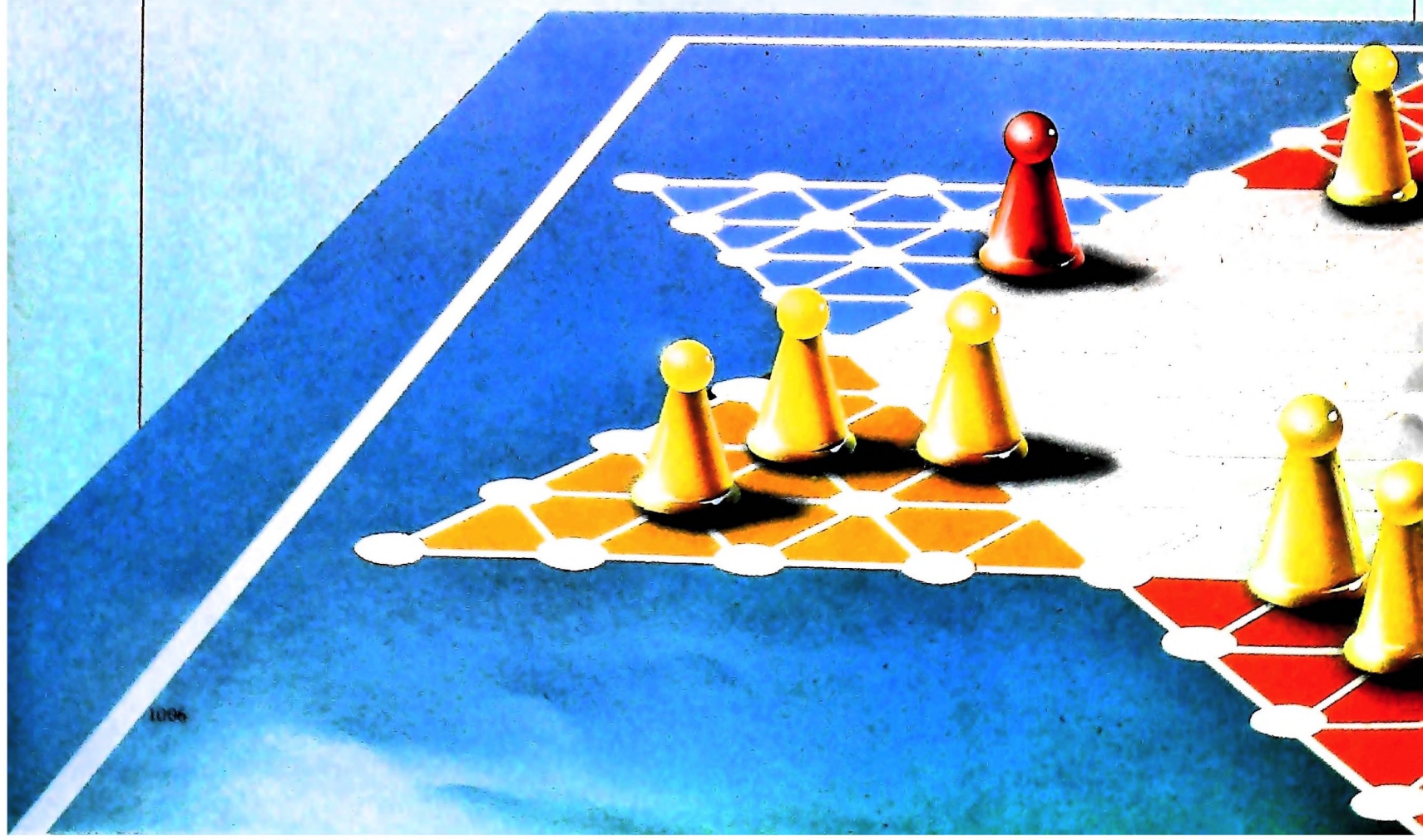
Board game: ovvero quando l'avversario è il computer.

Il termine "board game" designa generalmente due tipi di gioco: la versione elettronica di celebri giochi da tavolo (tombola, scacchi, dama, monopoli ecc.) e la ricostruzione schermica bidimensionale di battaglie famose o immaginarie ("board war game"), per la quale vengono messi a disposizione dei giocatori armi e mezzi di rifornimento corrispondenti a valori supposti reali in termini di massa, capacità di movimento, autonomia, potenza di tiro ecc.

Del primo tipo di gioco, il game più diffuso è certamente quello degli scacchi, che conta trasposizioni elettroniche e tutte di grande interesse: diversi livelli di abilità, dimostrazione di tutti gli eventuali sviluppi del gioco legati a ogni singola mossa, visualizzazione del ragionamento sviluppato dal computer con effetto "ralenti", rifiuto immediato delle mosse irregolari.

Del secondo, ha avuto più fortuna la versione immaginaria, la proiezione in stati o futuribili o fantasticati nel passato. Parliamo allora di due esempi molto riusciti: "Reversi" e "Conquest of the world".

Ci sono giochi che, apparentemente, sembrano avere qualcosa di incompleto, sembrano quasi una sorta di ripiego destinato a ingannare la noia. Molti si adattano solo in attesa di meglio, fino all'arrivo di qualche partner che permetta loro di cambiare questo piacere senza riscontri, isolato, con un gioco realmente disputato. Ma ecco che a un tratto succede qualcosa di imprevisto: comincia a manifestarsi una segreta voglia di competizione, un desiderio di misurarsi con la riflessione, la speranza di riuscire la prossima volta là dove abbiamo appena sbagliato. "Reversi" è uno di questi giochi; le sue regole e il suo meccanismo sono fondamentalmente semplici. Si tratta infatti di "chiudere", fra le proprie, le pedine dell'avversario; bisogna tessere una specie di ragnatela dentro la quale catturare i "pezzi" nemici e farli propri. "Reversi" (conosciuto in alcune nazioni anche con il nome di "Othello") si avvicina, come modello strategico, agli Scacchi e alla Dama ma le sue origini sembrano risalire al Go, un gioco orientale di grande bellezza e di profondi significati, incentrato com'è sul "controllo del territorio". L'elettronica,



poi, esalta questi giochi perché è in grado di dotarli di facoltà strategico-previsionali autonome, tali che risulta più esaltante giocare a "Reversi" contro il computer che contro un avversario in carne e ossa. Prezioso è il tasto di "preview" che consente di provare le mosse prima di eseguirle definitivamente.

"Conquest of the world" descrive un ipotetico conflitto mondiale dei nostri giorni. A metà strada fra il tradizionale gioco di società (Monopoli) e il war game con mappe e soldatini da un lato, e il computer game dall'altro, questa sfida internazionale propone negoziati, alleanze, conquiste, trattative diplomatiche, conflitti grandiosi e tattiche spregiudicate. Ogni giocatore ha l'impressione di sedere al tavolo delle superpotenze e di decidere, con una mossa, la sorte di migliaia e migliaia di uomini; ci sono infatti a disposizione sottomarini dotati di missili, caccia micidiali, poderosi carri armati. Le diverse nazioni sono raggruppate in zone geopolitiche con relative forze potenziali; la loro redistribuzione determina la vittoria finale. "Conquest of the world", come altri board game complessi, non è facile da manovrare, soprattutto perché è spostato su piani diversi: il gioco elettronico va sempre integrato e cioè riportato sulla mappa che i concorrenti hanno a disposizione. Ma una volta che ci si è impraticati, il game regala grandi soddisfazioni.

Citazione d'obbligo, infine, per "Tic Tac Toe" il game reso celebre dal finale di "Wargames", l'ormai mitico film sulla generazione elettronica. Questo gioco, che in Italia si chiama "Tris", consiste nell'apporre una crocetta o un cerchio per volta su un tabellone nel tentativo di sistemarne almeno tre

uno accanto all'altro.

Vince chi costringe l'avversario a fare una mossa sbagliata perché altrimenti il gioco non ha necessariamente un vincitore: ne sa qualcosa il supercomputer di "Wargames" che deve ammettere che nella vita (nella guerra) ci sono situazioni che non contemplano un vincitore. La versione elettronica offre un effetto tridimensionale animando notevolmente la povertà della scena del "Tic Tac Toe".

Il Programma

Biglie è un semplice solitario board-game ideato da Anthony Rizzolo. Si gioca su di una board lineare composta da 9 buchi e si hanno a disposizione due tipi di biglie, rappresentate rispettivamente con "+" e "*"; all'inizio tutte le biglie di un tipo sono da una parte e quelle dell'altro tipo dall'altra parte. Scopo del gioco è, nel minor numero possibile di mosse, invertire l'ordine di partenza. Le mosse permesse sono di due tipi:

- sposta la biglia nel buco adiacente
- fai "saltare" la biglia nel buco vicino; si può saltare al massimo una biglia

Ogni mossa viene segnalata al computer introducendo, nell'ordine, l'indirizzo in cui è la biglia da spostare (1..9) e l'indirizzo in cui la si vuole spostare. La parte più complicata del programma comprende le procedure che controllano la legalità della mossa. Insieme al listato viene presentato un frammento di esecuzione del gioco.

Il modello strategico del gioco degli scacchi, come pure quello della dama, deriva da antichi giochi orientali. E, con affascinante continuità, rivive oggi, nell'era dell'elettronica, in "Reversi", uno dei più riusciti board game.




```

10 DIM BOARD$(9)
20 s=0 : vuo$=""
25 piu$="+" : per$="*"
30 FOR i=1 TO 4
40 BOARD$(i) = piu$
50 NEXT
55 BOARD$(5)=vuo$
60 FOR i=6 TO 9
65 BOARD$(i)=per$
70 NEXT
100 GOSUB 800
110 s=s+1
120 PRINT "mossa ";s;" ";
130 INPUT m,m1
140 IF m<=9 AND m>=1 AND m1<=9 AND m1>=1 THEN 180
150 PRINT "mossa illegale !!!"
160 GOTO 120
180 REM controlla la correttezza della mossa
190 IF m+1=m1 OR m-1=m1 THEN 250
195 IF m=9 THEN 220
200 IF m=1 THEN 240
205 IF BOARD$(m+1)=vuo$ OR BOARD$(m-1)=vuo$ THEN 150
210 GOTO 248
220 IF BOARD$(m-1)=vuo$ THEN 150
225 GOTO 248
240 IF BOARD$(m+1)=vuo$ THEN 150
248 IF m+2<>m1 AND m-2<>m1 THEN 150
250 IF BOARD$(m)<>vuo$ THEN 270
260 PRINT "Niente nella casella ";m
265 GOTO 120
270 IF BOARD$(m1)=vuo$ THEN 300
280 PRINT "casella ";m1;" occupata "
290 GOTO 120
300 BOARD$(m1)=BOARD$(m)
310 BOARD$(m)=vuo$
315 s1=0 : s2=0
320 FOR i=1 TO 4
325 IF BOARD$(i)=per$ THEN s1=s1+1
330 NEXT
335 FOR i=6 TO 9
340 IF BOARD$(i)=piu$ THEN s2=s2+1
345 NEXT
348 IF s1=4 AND s2=4 THEN 400
350 GOTO 100
400 PRINT "Hai vinto in ";s;" mosse "
450 GOTO 999
800 REM stampa la BOARD
810 FOR i=1 TO 9
820 PRINT BOARD$(i);
830 NEXT
840 PRINT " ";
850 RETURN
999 END

```

++++	****	mossa 1 :	6	5
++++*	***	mossa 2 :	4	6
+++	*+***	mossa 3 :	5	4
++++	+***	mossa 4 :	3	5
++	*+***	mossa 5 :	4	3
+++	+***	mossa 6 :	4	3
Niente nella casella 4				
mossa 6 : 3 6				
mossa illegale !!!				
mossa 6 : 2 4				
+	+*****	mossa 7 :	3	2
+	*+*****	mossa 8 :	4	3
++	+*****	mossa 9 :	5	4

IL BASIC DESIGN EIDOMATICO

Una disciplina di grande importanza
nell'approccio grafico contemporaneo.



A lato, quadro di Max Bill:
"Variation 1" del 1936
(Collezione dell'autore).

Il nome trova le sue giustificazioni nel fatto che questa disciplina vuole essere una "base" costruttiva dell'attività progettuale, e non ha quindi nessun riferimento con l'acronimo BASIC (Beginners All-purpose Symbolic Instruction Code), riferito all'omonimo linguaggio di programmazione.

Il Basic Design è uno degli argomenti più vivacemente dibattuti dell'educazione al design.

Le origini del Basic Design risalgono al Bauhaus, la scuola d'arti applicate di Weimar fondata nel 1915 e considerata la matrice di tutte le scuole di design.

È al Bauhaus infatti che per la prima volta si assiste alla comparsa del "corso propedeutico", un corso di base della durata di sei mesi che precedeva gli anni di specializzazione.

Attraverso questo training gli studenti venivano introdotti in modo sperimentale ai concetti di proporzione, scala, ritmo, luce, ombra e colore tramite semplici esperienze con materiali e strumenti di ogni tipo.

Questi esercizi avevano l'obiettivo di sviluppare e maturare

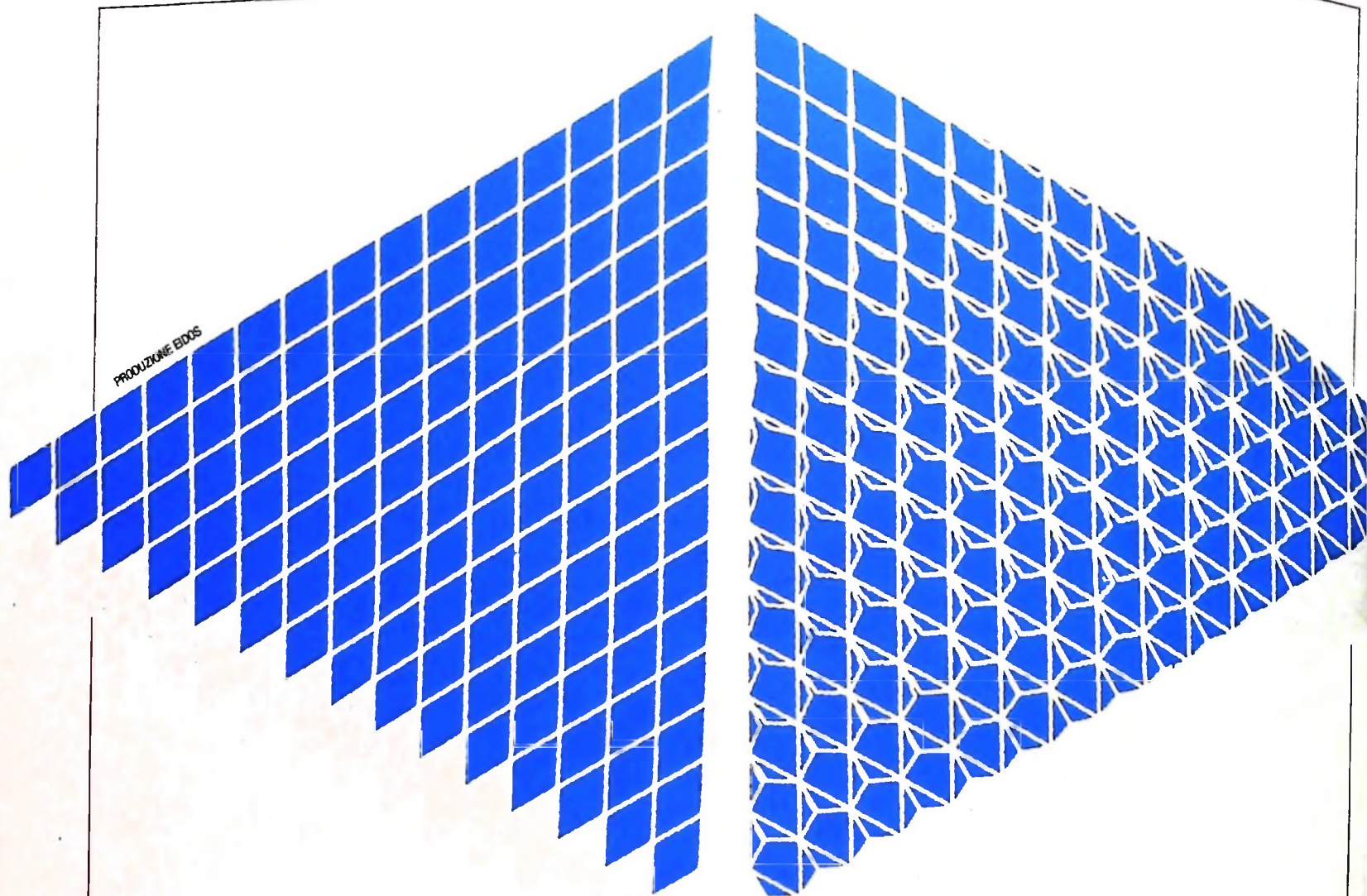
la sensibilità e le idee degli allievi, permettendo loro di affinare le proprie attitudini.

In questo senso ebbero enorme importanza i corsi tenuti da Itten, Klee e Kandinsky, da cui furono trattati i problemi fondamentali della composizione lineare e cromatica, e quelli di Albers e Moholy-Nagy, che si occuparono in particolare dei rapporti forma/materia, arte/tecnologia.

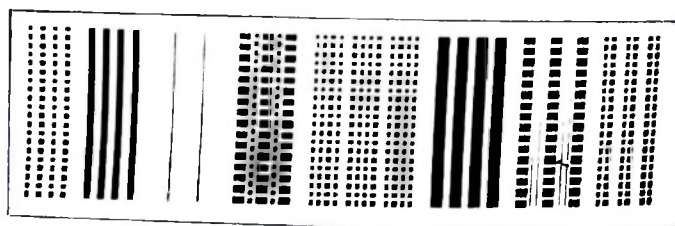
Quello del Bauhaus rappresenta quindi il primo tentativo di strutturare l'avvicinamento al processo creativo in senso razionalistico: il progettista doveva acquisire una conoscenza scientifica sia dei fenomeni ottici sia delle teorie che avrebbero poi guidato e condizionato le sue scelte formali.

Tale tentativo fu ripreso alla Hochschule für Gestaltung di Ulm, fondata nel 1954 da Max Bill, che rappresenta in un primo momento la continuazione della formula del corso propedeutico al Bauhaus, da cui Bill proveniva.

L'ingresso alle tre diverse sezioni di Architettura, Disegno Industriale, Comunicazioni visive e informazione è precedu-



Sopra: la prima sequenza di immagini si riferisce alla riproduzione al computer di un "parquet" di W. Huff. Il "parquet", secondo la definizione fornita dallo stesso Huff è una configurazione finita composta da varie parti che formano una superficie piena, priva di sovrapposizioni o di buchi. La struttura di partenza è una griglia di forme geometriche che subisce dei mutamenti e si deforma fino a dare origine ad altre forme. A lato: esercitazione "antiprimadonna": consiste nel realizzare una figura con bande di testura diverse in equilibrio tra loro, senza elementi dominanti otticamente. Il fine didattico di questo esercizio è quello di abituare l'occhio alla valutazione di elementi eterogenei.

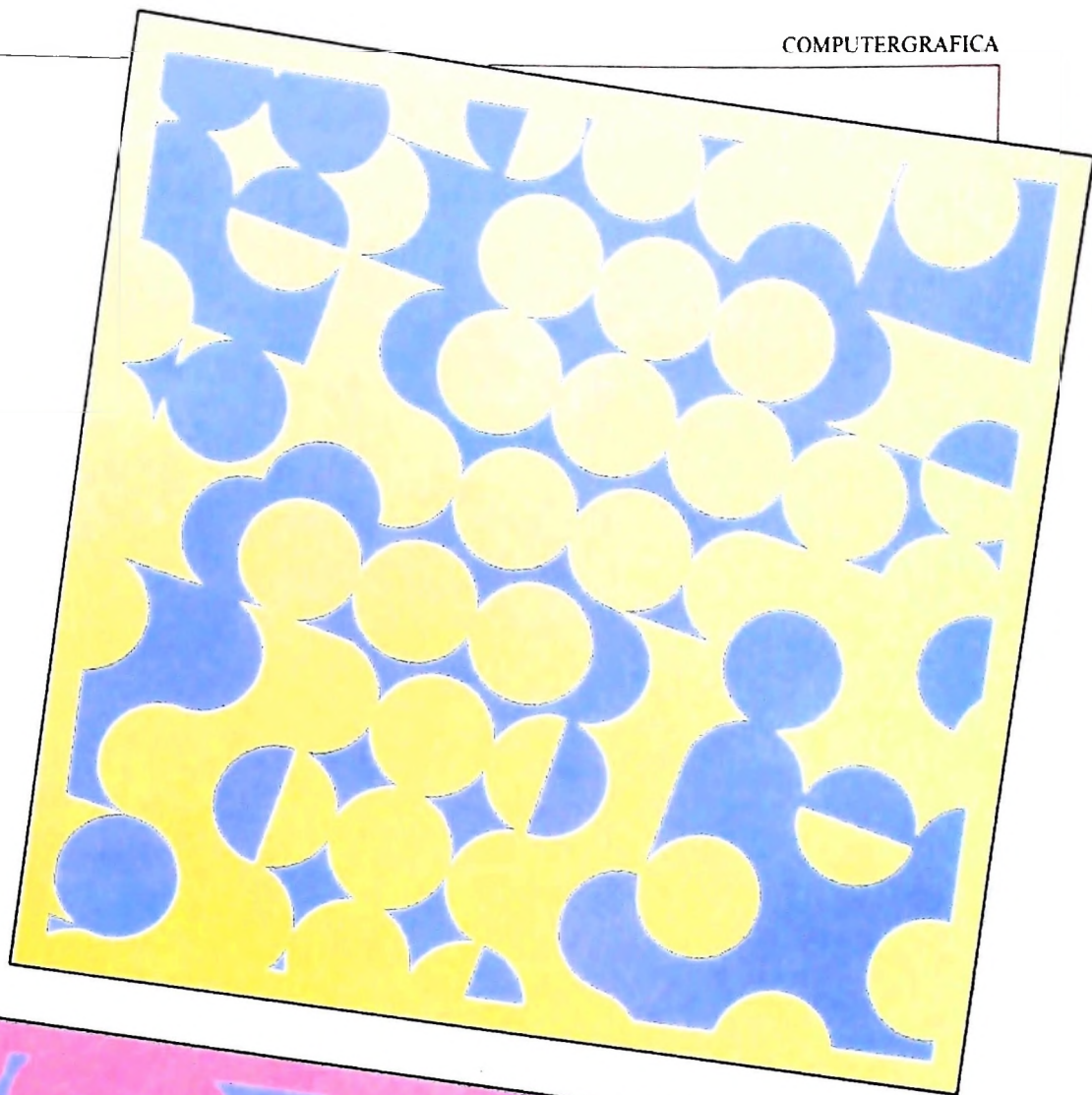


to da un primo anno di base indifferenziato. In seguito con T. Maldonado l'impostazione del training si rinnova acquisendo sempre di più una connotazione scientifica, attraverso l'introduzione di materie come topologia, simmetria, psicologia della forma. Il corso di base si estende via via a tutti gli anni di studio e diventa specialistico in rapporto ai diversi indirizzi. Con gli interventi teorici e didattici di T. Maldonado, W. Huff e G. Bonsiepe il Basic Design acquista definitivamente il valore di disciplina scientifica autonoma, con la duplice funzione metodologica e propedeutica.

Dalla scuola di Ulm in poi si assiste al sorgere di diversi tipi di Basic, che partendo dalle esperienze precedenti si evolvono e si aggiornano in rapporto ai nuovi strumenti e alle nuove tecnologie.

L'orientamento più recente del Basic riguarda appunto il computer e le problematiche inerenti all'utilizzo di questo strumento, sia per quanto riguarda la progettazione visiva che la produzione delle immagini. La prima grossa differenza è avvertibile infatti già nell'approccio metodologico alla progettazione che in questo caso è strettamente vincolato alle possibilità e alle peculiarità di base dello strumento.

A destra: "Structure", di Z. Sykoras. In basso: "Parole di quattro moduli", opera dello spagnolo Manuel Barbadillo. L'artista utilizza il più comune fra i dispositivi d'uscita dell'elaboratore, e cioè la stampatrice veloce, che consente di ottenere interessanti effetti visivi. Il più delle volte essa viene usata per costruire un modello sul quale si basa l'ulteriore lavoro manuale dell'artista. In genere si cerca di combinare alcune forme elementari con la ricerca di modificazioni sempre nuove e originali.



L'effettiva opportunità dell'utilizzo del computer si rivela quindi particolarmente nella produzione di immagini ottenute modificando o ripetendo strutture morfologiche elementari: composizioni modulari e simmetriche, progressioni, deformazioni ecc. È tipico infatti dell'elaboratore procedere attraverso l'iterazione su insiemi di istruzioni, e questa caratteristica di fondo consente di ottenere, con facilità e in tempi molto brevi, questo tipo di configurazioni.

L'utente si trova di fronte a diverse possibilità: può rielaborare al computer le esercitazioni "classiche" di Basic, ottenendo molto spesso un'enfaticizzazione degli effetti percettivi raggiungibili, oppure ideare nuove configurazioni, intervenendo sui dati geometrici della forma e sulle possibili combinazioni cromatiche.

L'impiego del computer per alcune di queste esercitazioni offre l'indiscutibile vantaggio di intervenire sulle immagini (sia a livello cromatico che morfologico) praticamente in tempo reale. Si evitano in questo modo i tempi piuttosto lunghi dell'esecuzione, e si garantisce molto più spazio al momento dell'ideazione e della creazione, ottenendo il risultato auspicabile di trasformare artisticamente il mezzo tecnico.

L'aspetto più entusiasmante del Basic Design applicato al computer (Basic Design Eidomatico) riguarda infatti la possibilità di realizzare veramente fino in fondo anche progetti finora confinati nella sfera dell'"intuitivo" dai problemi di esecuzione manuale.

Dal punto di vista propedeutico inoltre l'effetto ottenuto da questo tipo di esercitazioni è duplice: da un lato si prospettano nuove possibilità e strategie grafiche legate al mezzo-computer per risolvere problemi strutturali-morfologici e cromatici; dall'altro, nel contempo, si verifica la funzione educativa rispetto alla fruizione percettiva dell'immagine creata.

Il Programma

Di seguito viene fornito il listato di un programma per M10 che riproduce un esercizio di Basic Design. Si tratta di una composizione modulare ottenuta ripetendo più volte sul display l'elemento di partenza, cioè il quadrato. Il fatto che esistano delle parti riempite produce un effetto percettivo di tridimensionalità.

```
10 REM***Basic Design*****
20 REM***Esercizio pratico*****
25 CLS
30 GOSUB 100*Definizione costanti*****
40 GOSUB 500*Disegno dei moduli*****
50 GOSUB 1000*Stampa Titolo*****
60 F1$="":F1$=INKEY$:IF F1$="" THEN 60
70 MENU
80 END
100 REM***Definizione costanti*****
110 LA=45*lunghezza lato
120 SP=5* distanza fra i segmenti
```

```
130 NQ=5* numero segmenti
140 MN=5* numero moduli da disegnare
150 XP=5* coord. x partenza
160 YP=0* coord. y partenza
170 RETURN
180 *****
200 REM***definizione modulo base*****
201 LT=LA
202 PX=X:PY=Y
203 REM***Quadrati concentrici*****
205 FOR I=1 TO NQ
210 LINE(PX,PY)-(PX+LT,PY+LT),1,B
220 LT=LT-(2*SP)
230 PX=PX+SP:PY=PY+SP
250 NEXT I
255 *****
260 REM***Riempimento aree*****
270 PX=X:VX=X
280 PY=Y+LA:VY=Y+LA
290 LT=LA:VT=LA
300 FOR KK=1 TO NQ-2
310 FX=PX+LT:FY=PY-LA
315 REM***Riempimento orizzontale*****
320 FOR I=1 TO SP
330 LINE (PX,PY)-(FX,PY),1
340 PY=PY-1:FX=FX-1
350 NEXT I
360 *****
370 REM***Riempimento verticale*****
380 FOR I=1 TO SP
390 LINE (VX,VY)-(VX,VY-VT),1
400 VX=VX+1:VT=VT-1
410 NEXT I
420 *****
425 REM***Aggiornamento vertici*****
430 LT=LT-(4*SP):VT=LT
440 PX=X+(KK*SP*2):VX=PX
450 PY=Y+LA-(KK*SP*2):VY=PY
460 NEXT KK
470 RETURN
480 *****
500 REM***Disegno dei moduli*****
510 X=XP
520 Y=YP
530 FOR M=1 TO MN
540 GOSUB 200
550 X=X+LA
560 Y=YP
570 NEXT M
580 *****
1000 REM***Stampa il messaggio finale**
1010 PRINT @280,"
        Battere <un tasto> per finire":
1020 BEEP
1030 RETURN
1040 *****
```


— UN NUOVO MODO DI USARE LA BANCA.

Conto corrente più

TANTI PENSIERI IN MENO CON IL CONTO CORRENTE "PIÙ" DEL BANCO DI ROMA.

Essere cliente del Banco di Roma vuol dire anche essere titolari del conto corrente "più". Un conto corrente più rapido: perché già nella maggior parte delle nostre filiali trovate gli operatori di sportello che vi evitano le doppie file.

Più comodo, perché potete delegare a noi tutti i vostri pagamenti ricorrenti: dai mutui all'affitto, dalle utenze alle imposte.

Più pratico, perché consente l'utilizzo del sistema di prelievo automatico Bancomat e l'ottenimento della carta di credito.

Inoltre un servizio utilissimo, soprattutto per imprenditori e commercianti denominato "esito incassi", consente di avere comunicazione dell'eventuale insolvenza entro solo cinque giorni dalla scadenza. Una opportunità veramente speciale.

Più sicuro, perché con una minima spesa potrete assicurarvi contro furti e scippi mentre vi recate in banca o ne uscite.

Veniteci a trovare, ci conosceremo meglio.

 **BANCO DI ROMA**
CONOSCIAMOCI MEGLIO.



Olivetti M10 vuol dire disporre del proprio ufficio in una ventiquattrore. Perché M10 non solo produce, elabora, stampa e memorizza dati, testi e disegni, ma è anche capace di comunicare via telefono per spedire e ricevere informazioni. In grado di funzionare a batteria oppure collegato all'impianto elettrico, M10 mette ovunque a disposizione la sua potenza di memoria, il suo display orientabile a cristalli liquidi capace anche di elaborazioni grafiche, la sua tastiera professionale arricchita da 16 tasti funzione.



Ma M10 può utilizzare piccole periferiche portatili che ne ampliano ancora le capacità, come il micro-plotter per scrivere e disegnare a 4 colori, o il registratore a cassette per registrare dati e testi, o il lettore di codici a barre. E in ufficio può essere collegato con macchine per scrivere elettroniche, con computer, con stampanti. Qualunque professione sia la vostra, M10 è in grado, dovunque vi troviate, di offrirvi delle capacità di soluzione che sono davvero molto grandi. M10: il più piccolo di una grande famiglia di personal.

PERSONAL COMPUTER OLIVETTI M10

L'UFFICIO DA VIAGGIO



Anche in leasing con Olivetti Leasing.

olivetti

Per informazioni rivolgetevi al vostro rappresentante Olivetti M10 Personal Computer o al nostro ufficio Olivetti M10 Personal Computer, Via Meravigli 12, 20121 Milano, Italia.

1015 COGNOME
1016 COGNOME
1017 COGNOME
1018 COGNOME
1019 COGNOME
1020 COGNOME
1021 COGNOME
1022 COGNOME
1023 COGNOME
1024 COGNOME
1025 COGNOME
1026 COGNOME
1027 COGNOME
1028 COGNOME
1029 COGNOME
1030 COGNOME