

CADEL

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

66 CORSO PRATICO COL COMPUTER

è una iniziativa
FABBRI EDITORI

in collaborazione con
BANCO DI ROMA

e **OLIVETTI**



BATTERY LOW

422188

F4 F5 F6 F7 F8

diretto da **GIANNI DEGLI ANTONI**



FABBRI EDITORI

Cari lettori,
con questo numero sospendiamo per la pausa
estiva l'uscita del
CORSO PRATICO COL COMPUTER
Arriverderci in edicola il 29 agosto con il n. 67.
Buone vacanze!

La redazione

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 PARPELLI

Testi
CLAUDIO PARPELLI, DANIELE MARINI, DOMENICO CAVALLOTTO,
MARCO ANELLI, DIEGO BIASI, Enoteam (ADRIANA BICEGO)

Tavole
Logical Studio Communication
Il Corso di Programmazione e BASIC è stato realizzato da Enoteam
S.p.A., Milano
Computergrafica 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 FENCHI

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

AVVISO AI LETTORI

Con il n. 67 sarà in edicola la
copertina per rilegare il quinto
volume del "Corso pratico col
computer". Prenotatela dal vo-
stro edicolante.

Corso Pratico col Computer - Copyright © sul fascicolo 1985 Gruppo Edi-
toriale Fabbri, Bompiani, Sonzogno, Etas S.p.A., Milano - Copyright ©
sull'opera 1984 Gruppo Editoriale Fabbri, Bompiani, Sonzogno, Etas
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'i-
talia Gruppo Editoriale Fabbri S.p.A., via Mecenate, 91 - Milano - tel. 02/
50951 - Pubblicazione periodica settimanale - Anno II - n. 66 - 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.

SIGNIFICATIVE APPLICAZIONI DELLA TELEMATICA

Disponendo di un calcolatore equipaggiato con le apposite interfacce è possibile risolvere qualsiasi problema di documentazione.

Una delle applicazioni della telematica che negli anni Ottanta ha riscosso notevole successo è quella inerente all'Information retrieval o Ricerca bibliografica.

Cominciamo con il vedere le cause che hanno portato all'esigenza dell'impiego del computer in questo settore.

Esplosione dell'informazione

Dalla fine della seconda guerra mondiale si è assistito al rapido aumento della letteratura scientifica e tecnica, quale ef-

fetto della sempre maggiore importanza che la ricerca riveste nello sviluppo delle società industriali. Il fenomeno, già chiaramente individuabile nel Rinascimento, ha nei giorni nostri raggiunto dimensioni impressionanti tanto che si parla di "esplosione dell'informazione".

Dati interessanti su tale fenomeno si estraggono da un lavoro di Anderla (Anderla, J.G. — Information in 1985 — A forecasting study on information needs and resources. 1973 — OECD — Parigi): per ogni giorno lavorativo vengono pubblicati dai 6000 ai 7000 articoli (o simili), per un totale di circa 2 000 000 di lavori tecnico-scientifici all'anno, con un

I fogli elettronici fanno grandi cose, ma purtroppo impiegano molto tempo nella ricerca e nella registrazione dei dati. Con Dow Jones Spreadsheet Link questo non capita più. Si ha infatti accesso immediato al Dow Jones News/Retrieval e alle sue banche dati, da cui si possono attingere le più aggiornate informazioni finanziarie e di affari. Poi i dati vanno a inserirsi automaticamente nel programma Multiplan.

Control Modems **Communications** **Features** **Data Bases**

DOW JONES **LINK by SOLUTIONS, INC.**

	1	2	5	6	7
1		LAST			
2		PRICE			
3	DJ		COST	VALUE	CHANGE
4	IBM		\$18,075		
5	TAN		\$10,000		
6	AAPL		\$8,000		
7	DEC		\$2,000		
8			\$11,000		

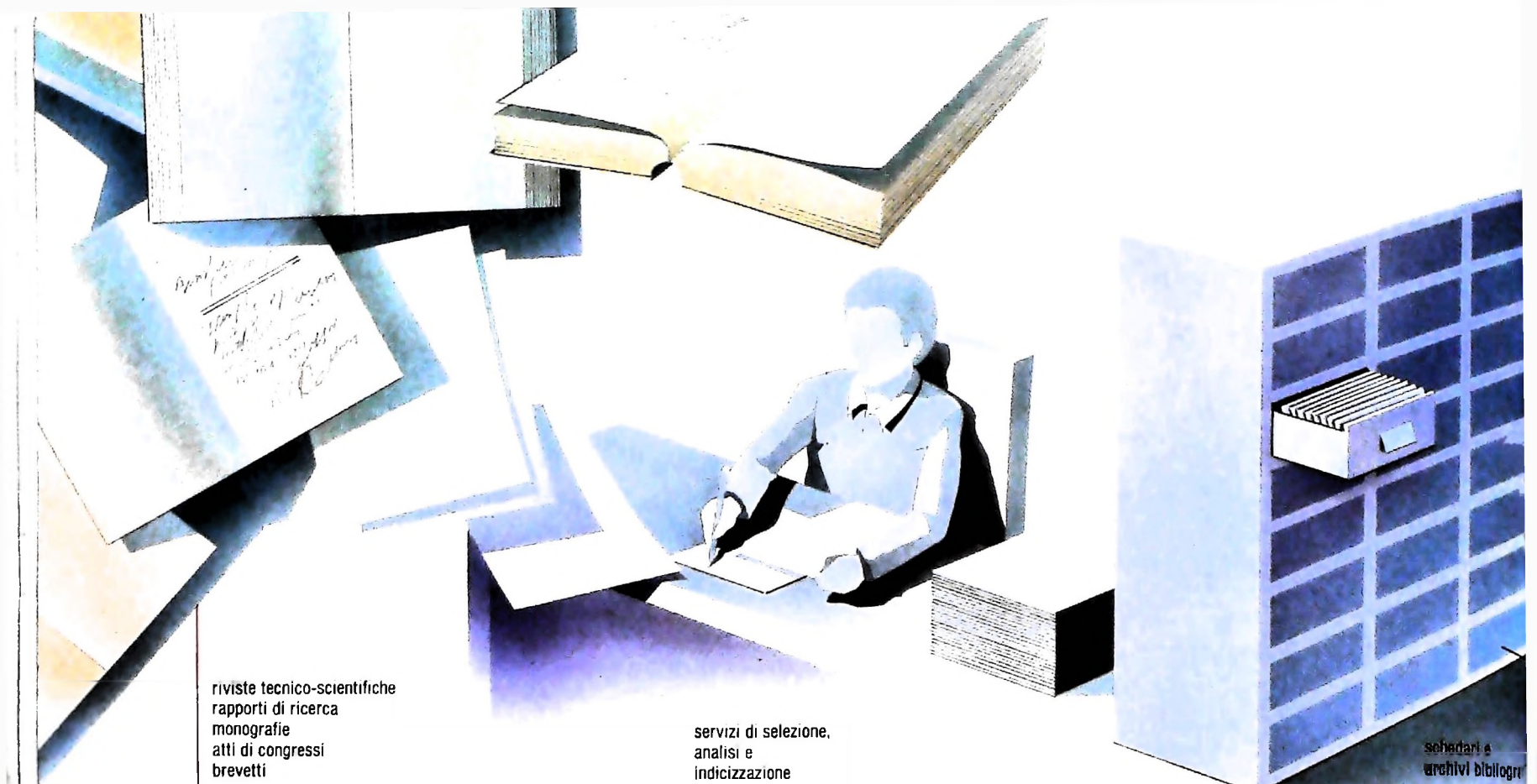
These values will be calculated by Multiplan

These values will be retrieved by Spreadsheet Link from Dow Jones

300 Baud
1200 Baud

The spreadsheet may also contain data you enter yourself.

...ANNOUNCING: Dow Jones Spreadsheet Link on The Macintosh



riviste tecnico-scientifiche
rapporti di ricerca
monografie
atti di congressi
brevetti

servizi di selezione,
analisi e
indicizzazione

schede e
archivi bibliografici

volume complessivo di 5 miliardi di parole. Sempre dallo stesso studio, si viene a conoscenza che il patrimonio mondiale di giornali o riviste scientifiche varia dalle 90 000 alle 140 000 unità, con un tasso di accrescimento intorno al 75% ogni 10 anni.

L'impatto informatico

La metodologia documentaria ha tradizionalmente messo a disposizione dell'utente strumenti di classificazione, recensione e segnalazione della letteratura esistente, al fine di consentire l'identificazione del documento ricercato e del suo reperimento. Si tratta, in maggior parte, di bollettini che periodicamente selezionano il materiale pubblicato in un determinato settore disciplinare e ne pubblicano i dati essenziali (titolo, autore, fonte), fornendo anche il riassunto (abstract) e una lista di parole chiave (keyword) che identificano in modo pertinente i diversi temi trattati (descriptor). Il considerevole aumento delle informazioni disponibili e le sempre maggiori esigenze di quantità di dati e di rapidità di reperimento hanno messo in luce l'inadeguatezza dei sistemi tradizionali suggerendo il ricorso a metodologie più moderne.

L'uso dei bollettini o degli indici comporta spesso una serie di svantaggi, dovuti alla poca maneggevolezza, alla difficoltà di effettuare ricerche con più di due o tre descrittori, alla quantità di tempo necessaria all'analisi dei fascicoli di varie annate e infine alla necessità di avere un archivio cartaceo.

L'informatica ha offerto gli strumenti per affrontare in modo nuovo i problemi di reperimento delle informazioni. La memorizzazione degli archivi di bibliografie e di dati di vario genere su supporti magnetici (nastro o dischi), la loro gestione automatica per mezzo di sofisticati sistemi (retrieval system), la possibilità di accedere e interrogare questi archivi,

immagazzinati in computer, tramite un semplice terminale utilizzando la normale rete telefonica, hanno radicalmente modificato il processo di trasferimento dell'informazione. Verranno esaminati, nei paragrafi che seguono, i diversi elementi che concorrono a realizzare lo scenario dell'informazione in linea e le relazioni che intercorrono tra essi.


Basi di dati

Con l'espressione di BANCA DATI si intende comunemente una collezione omogenea di informazioni memorizzate su di un supporto magnetico e accessibili mediante elaboratore. La definizione appare però piuttosto impropria: dopo un'iniziale e comprensibile confusione di termini, questo particolare tipo di archivio è per ora per lo più chiamato BASE DI DATI, volendo significare che si tratta di DATI che costituiscono la BASE su cui applicare programmi diversi per compiere, mediante elaboratori, operazioni di selezione delle informazioni e di elaborazione dei risultati presentati poi sotto varie forme. In particolare è meglio chiarire che tali strumenti sono di massima BASI DI DATI IN LINEA (on-line database), cioè archivi gestiti da computer e accessibili via terminali collegati attraverso linee telefoniche pubbliche.

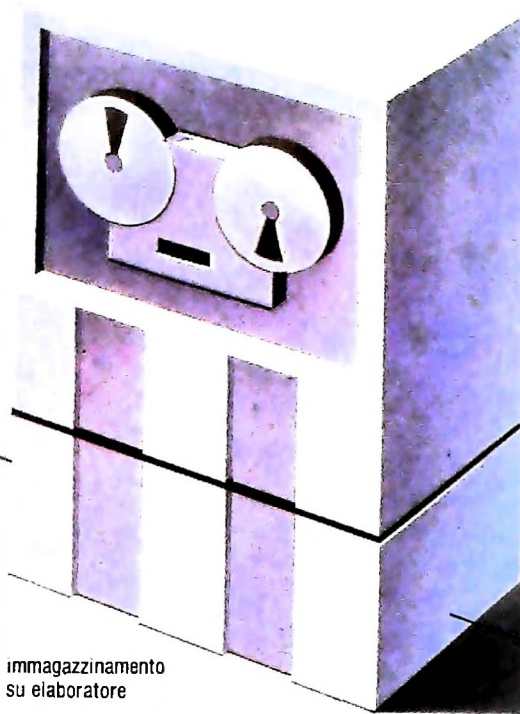
I supporti informativi

Senza entrare nei dettagli sulla composizione di un sistema di Information retrieval e sul software che opera al suo interno, è bene esaminare, seppur superficialmente, la struttura delle informazioni che esso contiene, cioè le modalità con cui queste informazioni sono dislocate sui supporti magnetici. Si è sempre parlato, finora, di supporti magnetici per identi-

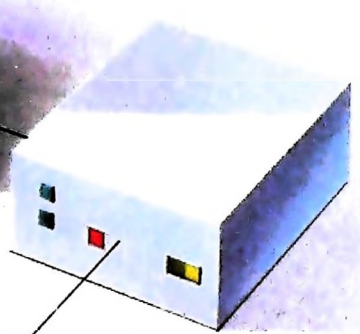
Nelle illustrazioni di queste pagine: itinerario di costruzione e di funzionamento di una base di dati bibliografica. Archivi di bibliografie e di dati vengono memorizzati su supporti magnetici (nastro o disco). La loro gestione è automatica, e avviene attraverso sistemi sofisticati. Esiste la possibilità di accedere a questi archivi, immagazzinati in computer, tramite un semplice terminale utilizzando la rete telefonica.



memorizzazione
su supporto
magnetico



immagazzinamento
su elaboratore



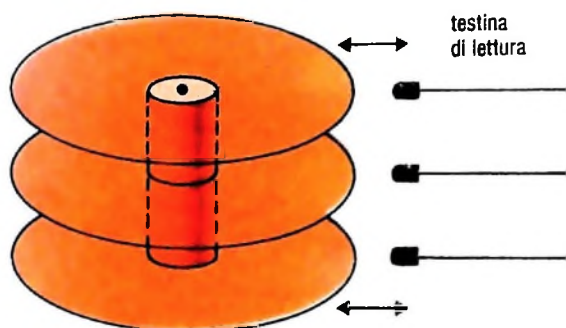
collegamento
alle reti
trasmissioni dati



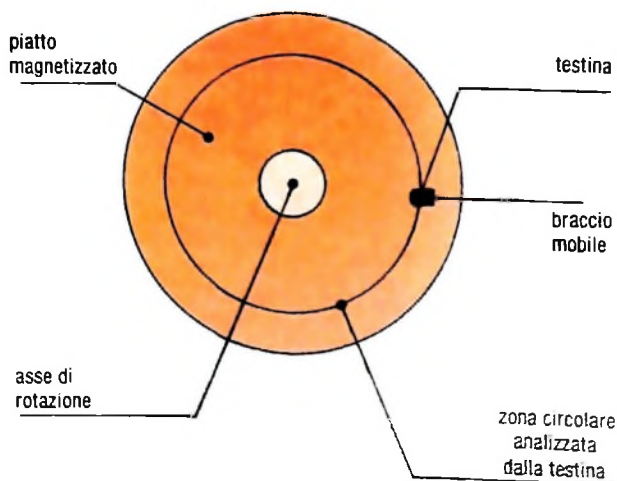
terminali
utenti

ficare quei mezzi destinati a contenere i dati da fornire a un elaboratore, o da esso generati, così come i supporti cartacei vengono da secoli dedicati alla registrazione delle informazioni da fornire agli uomini o da questi generate. Allo stato attuale della tecnologia del settore, tali supporti magnetici sono essenzialmente rappresentati dai dischi.

Una unità a dischi magnetici è formata dal supporto magnetico vero e proprio e dal meccanismo di accesso a esso. Il supporto è costituito da un insieme di piatti le cui superfici sono rivestite di materiale ferromagnetico che ruotano insieme, costantemente e ad alta velocità, attorno a un unico asse. Per reperire un'informazione in questo supporto, viene adoperato un dispositivo con delle testine di lettura simili a quelle usate nei registratori musicali, montate però su bracci metallici affiancati l'uno all'altro come i denti di un pettine e in grado di penetrare nell'intercapedine esistente tra i singoli piatti (figura sotto).



Il pettine contenente le testine può infatti essere traslato in modo da avvicinarsi o allontanarsi dall'asse di rotazione, e a ogni sua posizione, ciascuna testina si troverà in corrispondenza di una zona di disco di forma circolare, che va ruotando ad alta velocità. Pertanto, posizionando il complesso bracci-testine, si seleziona un insieme di sottili circonferenze di materiale magnetico, ognuna delle quali è esplorata da una singola testina (figura sotto). Le singole circonferenze



sono chiamate PISTE e l'insieme esplorato contemporaneamente (cioè l'insieme delle piste di ugual raggio) viene detto CILINDRO.

Mediante tale suddivisione, il disco viene reso indirizzabile; l'indirizzo di un dato sarà appunto costituito dal numero del cilindro e dal numero della pista in cui esso è registrato. Il tutto si svolge in un tempo che si aggira intorno a poche decine di millesimi di secondo.

L'introduzione dei dischi magnetici è stata di fondamentale importanza per la ricerca dell'informazione; i loro progenitori, i nastri magnetici, infatti, potevano rintracciare un dato in un tempo dell'ordine di diverse decine di secondi o anche di qualche minuto.

Ci si può rendere conto facilmente che tali tempi erano assolutamente incompatibili con qualsiasi tipo di servizio di ricerca di dati on-line. Immaginiamo, infatti, che su un determinato archivio occorra un minuto per reperire un'informazione: se le singole richieste di accesso all'archivio giungessero con un ritmo costante di una al minuto, ognuna di esse riceverebbe la risposta con il minimo tempo di attesa possibile per il sistema: un minuto, nel nostro esempio.

Ma se le richieste giungessero con una frequenza maggiore, per esempio 10 al minuto, allora si verificherebbe il tipico fenomeno di formazione di code, che porterebbe i tempi di servizio a limiti inaccettabili: infatti, in ogni minuto, una richiesta verrebbe trattata e 9 rimarrebbero in attesa.

La riduzione dei tempi di accesso a qualche decina di millesimi di secondo, ottenuta con i dischi, non deve quindi essere interpretata come una caratteristica di media importanza; se un archivio ha un tempo di accesso di 50 millesimi di secondo, vuol dire che esso può sopportare fino a 20 interrogazioni al secondo da parte della collettività degli utenti, senza che si formino code di lunghezza crescente.

Cosa c'è dentro le basi di dati

Sparsi per il mondo circa 200 distributori (conosciuti con il termine inglese Host computer) offrono attualmente la possibilità di accedere direttamente a oltre 1400 basi dati che differiscono tra loro per finalità, tipo di informazioni offerte, forma dell'output, argomento, area geografica e periodo coperto, frequenza di aggiornamento e altro.

Esistono insomma basi di dati per tutti i gusti: in genere, ciascuna contiene informazioni su un settore molto specifico, quali metallurgia, alluminio, tecnologia dell'industria tessile, notizie economiche riguardanti l'industria, matematica, fisica, chimica, musica, psicologia, testi di una gazzetta ufficiale e altri ancora.

Vari sono i tentativi di classificare le basi dati secondo il tipo di informazioni offerte. Tra essi meritano di essere ricordati quelli dell'EUSIDIC (Associazione Europea dei Servizi Informativi) e della CUANDRA su cui si forma lo schema di classificazione adottato dalla commissione tecnica dell'ISO (International Organization for Standardization) che ha elaborato un fondamentale dizionario della informazione e della documentazione.

PERIFERICHE DI CONTROLLO

Attribuiscono al computer un ruolo più attivo nella vita quotidiana.

Se è dotato di un apposito convertitore un calcolatore, come abbiamo già visto, è in grado di accettare dati e informazioni espressi in forma analogica, cioè in una forma molto più congeniale alla comprensione da parte di un essere umano.

Questo semplice fatto incrementa notevolmente le possibilità operative della macchina, consentendo applicazioni come il controllo di strumenti di misura e le rappresentazioni grafiche. Oltre ai convertitori analogico-digitali esistono però altri tipi di periferica, che attribuiscono al computer un ruolo molto più attivo in quella che potremmo definire la vita quotidiana.

Il tipo più semplice di queste interfacce è solitamente rappresentato da una scheda a circuiti, sulla quale sono montati alcuni relè.

Grazie ad apposite interfacce i relè sono visti dal calcolatore come singole periferiche che presentano due stati: "aperto" e "chiuso".

Tramite appositi comandi il calcolatore stesso può inoltre variare attivamente lo stato dei relè, aprendoli o chiudendoli. A ciascuno di questi relè può quindi essere connesso un circuito elettrico, anche di potenza elevata, che sarà o meno attivato a seconda della posizione assunta dal relè corrispondente.

In questo modo il computer viene trasformato in una specie di quadro elettrico, capace di accendere o spegnere, tramite comandi da tastiera, una serie di apparecchiature elettriche.

Questo tipo di interfaccia trova ampio uso nell'industria, e particolarmente nel controllo dei processi produttivi.

Un caso particolare di questo impiego è rappresentato dal "Controllo Numerico" (CN) in cui un elaboratore, interfacciato a una o più macchine utensili, è capace di produrre un pezzo di forma anche assai complicata sulla base delle istruzioni contenute in un apposito programma.

Il vantaggio principale del Controllo Numerico è rappresentato dal fatto che qualsiasi modifica al ciclo produttivo viene eseguita semplicemente modificando il programma di partenza e non, come avviene nelle normali macchine utensili, riprogettando l'unità produttiva.

Se alla "scheda a relè" che abbiamo appena visto vengono aggiunti anche alcuni convertitori, capaci di digitalizzare grandezze analogiche provenienti dall'esterno, avremo realizzato uno strumento in grado di svolgere praticamente qualsiasi compito che coinvolga l'uso di macchine elettriche.

Collegando i convertitori ad appositi sensori disporremo in-

fatti di un sistema capace di raccogliere dati dall'esterno e di reagire alle loro variazioni in base alle istruzioni contenute nel programma residente in quel momento in memoria.

Periferiche di questo tipo esistono sia per i grossi sistemi che per i personal o gli home computer, con differenze anche notevoli in termini di costo e di complessità.

Per quanto riguarda i piccoli elaboratori il linguaggio in cui vengono scritti i programmi è solitamente il BASIC.

Tenendo presente che i valori che rappresentano lo stato dei sensori di input e dei relè di output sono normalmente contenuti in apposite locazioni di memoria, accessibili e modificabili con i comandi PEEK e POKE, un programma che insegna al calcolatore a controllare tramite un apposito sensore di umidità se i fiori devono essere innaffiati potrebbe essere:

```

5 REM inizializzazione
10 UMIDITÀ = PEEK (32564)
20 POMPA = 28621
30 ACCESO = 1
40 SPENTO = 0
50 REM controllo e azione
60 IF UMIDITÀ < 50 THEN POKE POMPA, ACCESO
70 IF UMIDITÀ > = 50 THEN POKE POMPA, SPENTO
80 GOTO 60
90 END

```

Il programma controlla continuamente, tramite un loop senza fine, lo stato del sensore di umidità, il cui valore è memorizzato nella locazione 32564, e, in caso di umidità inferiore a un dato valore, provvede ad accendere la pompa che trasporterà l'acqua, per spegnerla nuovamente quando il sensore avrà segnalato un grado di umidità soddisfacente.

La complessità del programma di controllo/azione è ovviamente limitata solamente dalle capacità del programmatore e dall'hardware della macchina impiegata, consentendo per esempio la realizzazione di un controllo dell'impianto di riscaldamento di questo tipo, ovviamente impossibile con un termostato tradizionale:

- alle ore 7.30 porta la temperatura di bagno e cucina a 22 gradi
- alle ore 7.50 accendi la caffettiera elettrica
- alle ore 8.00 accendi la radiosveglia
- dalle ore 9.00 alle ore 17.00 mantieni la temperatura della casa sui 18 gradi

— se la temperatura sale oltre i 22 gradi apri le finestre
 — alle ore 17.00 accendi il riscaldamento e fai in modo che per le 18.00 la temperatura sia di 20 gradi in tutta la casa.
 L'esecuzione di questo programma, per quanto complesso, non richiede la continua attenzione da parte del computer. Possono quindi essere inserite nel corpo principale alcune subroutine che, periodicamente, controllino un eventuale segnalatore di incendi, comandino un forno a microonde e così via. Periferiche di questo tipo sono molto usate anche dai ferromodellisti più evoluti che le impiegano, simulando in modo perfetto la realtà, per controllare gli scambi e il traffico sui propri plastici.

Dal punto di vista della programmazione questo rappresenta forse il compito più complesso che una interfaccia di questo tipo è chiamata a svolgere in quanto la gestione del traffico ferroviario, sia pure in scala uno a cento, richiede l'impiego di modelli matematici molto complessi e di vere e proprie "tabelle delle decisioni".

Per chiarire meglio i concetti visti prima esaminiamo ora più in dettaglio il VicRel, una periferica di controllo molto diffusa realizzata dalla Handic per il VIC 20.

Questo accessorio è costituito da una scheda dotata di LED e morsetti che viene inserita sul retro del calcolatore, similmente a quanto avviene per tutte le altre schede di espansione, ed è dotata di 20 morsetti a vite.

I primi dodici morsetti definiscono il positivo e il negativo di sei relè di output, mentre gli ultimi quattro rappresentano gli ingressi di due sensori esterni.

Altri due morsetti consentono all'installatore di prelevare di-

rettamente dall'alimentazione del computer cinque volt in corrente continua.

Lo stato di attivazione di ogni circuito è segnalato da un apposito LED presente sull'interfaccia stessa. A differenza di altre schede dello stesso tipo ogni relè non possiede la propria locazione di memoria, ma lo stato di tutti i circuiti (o meglio la configurazione dei vari relè) è rappresentato da un valore che risiede nella locazione 37136. L'insieme dei sei relè può infatti essere assimilato a un numero binario composto da sei bit, ciascuno dei quali può assumere un valore di "zero" e di "uno" a seconda dello stato in cui si trova.

Un sistema di questo genere può assumere un numero finito di configurazioni, e precisamente 64 (due alla sesta potenza), cioè da "000000" (tutto spento) a "111111" (tutto acceso).

Questo metodo di controllo risparmia memoria nei confronti del sistema "una locazione un interruttore" descritto prima, ma risulta di impiego meno immediato in quanto un relè può essere comandato singolarmente solamente intervenendo sulla configurazione di bit del numero memorizzato nella locazione 37136.

Questo intervento può essere fatto solo tramite operatori booleani del tipo AND e OR.

Per attivare tramite il VicRel il relè numero uno sarà quindi necessaria l'istruzione:

POKE (37136), PEEK (37136) OR 1

mentre per disattivare il medesimo interruttore senza modificare lo stato degli altri bisognerà fornire al computer il seguente comando:

POKE (37136), PEEK (37136) AND (63-1)

Un possibile uso domestico del computer: tramite periferiche di controllo e un adatto programma, si possono innaffiare automaticamente i fiori. Un sensore segnala quando l'umidità scende al di sotto di un valore prestabilito: viene allora attivata la pompa che trasporta l'acqua e che verrà spenta quando il sensore segnalerà che il grado di umidità è sufficiente. Possono inoltre essere inserite alcune subroutine che controllano periodicamente un eventuale segnalatore d'incendi, un forno a microonde ecc.





Attualmente non sono disponibili sistemi di "home automation" già pronti, ciascun utente deve pertanto, dopo l'acquisto dell'interfaccia, provvedere alla stesura dei programmi e alla realizzazione dei collegamenti.

L'unica eccezione è rappresentata da alcuni sistemi "dedicati", che impiegano interfacce a relè e sensori per il comando di sistemi di sicurezza.

Un antifurto dotato dei poteri decisionali di un computer, infatti, rappresenta un ostacolo formidabile per un eventuale ladro ed è inoltre in grado di ridurre grandemente il problema dei falsi allarmi, simulando il discernimento di un essere umano. Un rumore di natura ignota ha infatti un significato ben diverso di giorno piuttosto che di notte, mentre un colpo

violento su di una persiana può essere causato dal vento o da un ragazzino e venire quindi ignorato, a meno che non si ripeta dopo un breve intervallo.

L'unico inconveniente rappresentato dall'impiego di computer per il controllo di apparecchiature esterne in ambienti domestico o professionale è costituito dal fatto che, mentre sta girando il programma di controllo, il calcolatore non può svolgere nessun altro compito.

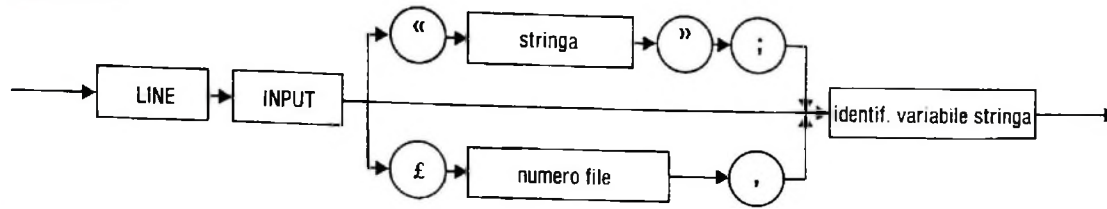
L'avvento di personal a basso costo capaci di esecuzione contemporanea di più programmi, lascia però prevedere uno sviluppo notevole di questi sistemi, che con tutta probabilità risiederanno in permanenza in una parte della memoria del computer, lasciando libero il resto della potenza di calcolo.

Oggi esistono sistemi che impiegano interfacce a relè e sensori per il controllo di sistemi di sicurezza. In questo modo un antifurto è veramente in grado di esercitare una funzione di controllo "attiva" e intelligente.



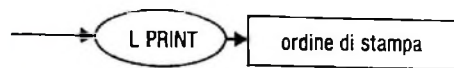
Istruzioni Basic disponibili su M10 (II)

LINE INPUT

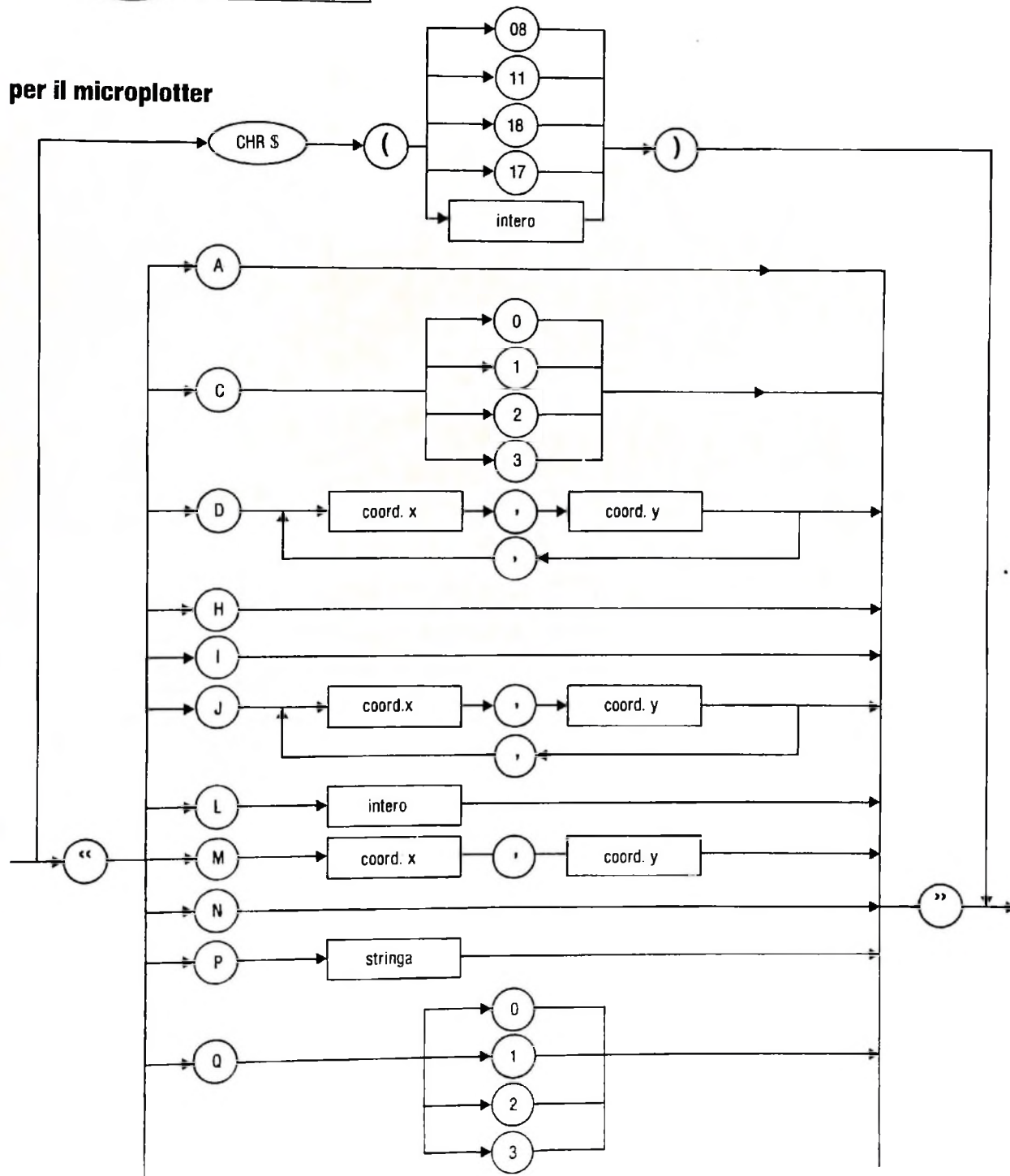


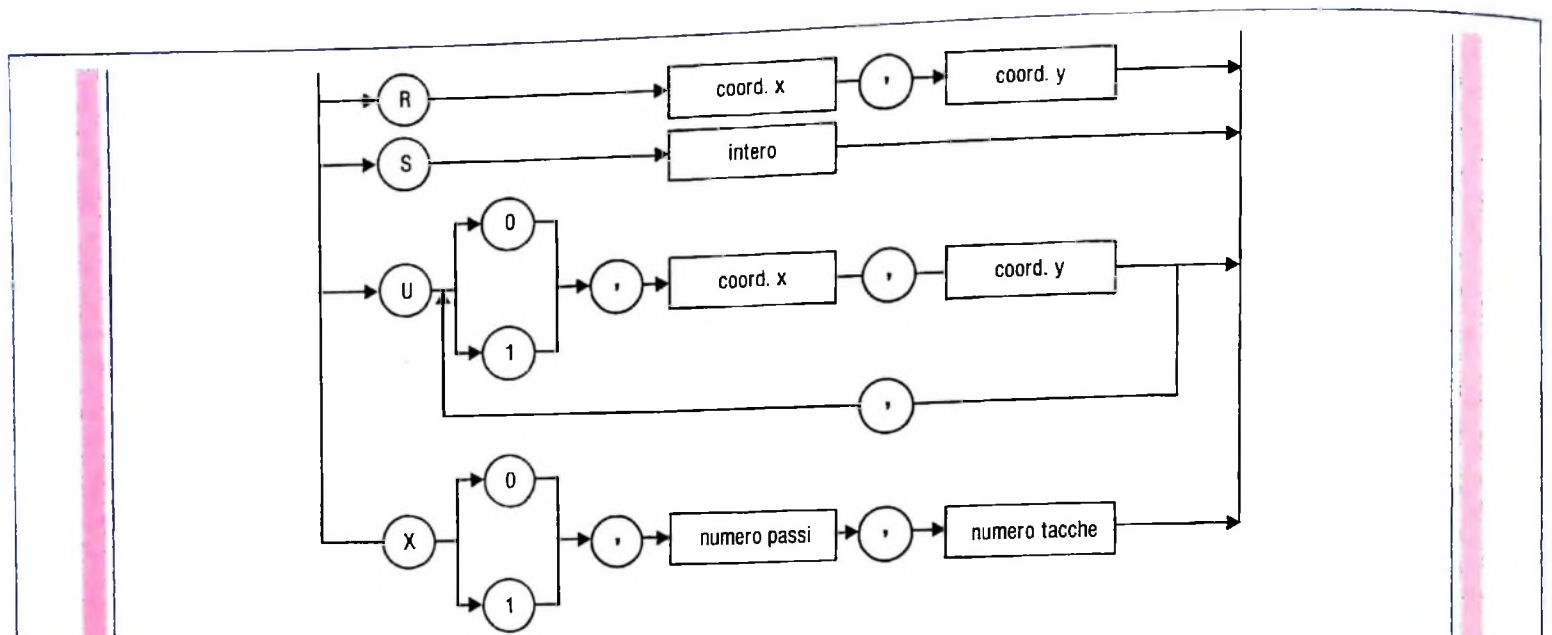
Funziona come l'istruzione INPUT, ma opera su una sola variabile stringa, e legge un'intera linea, incurante di eventuali separatori: la linea termina con la pressione del tasto ENTER.

L PRINT



per il microplotter



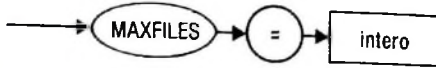


Visualizza dati su una stampante o su un microplotter. Per i dettagli, vedi PRINT.
Per il microplotter valgono le seguenti osservazioni:

- .CHR\$(08) ritorna di un passo
- .CHR\$(11) fa un'interlinea all'indietro
- .CHR\$(17) seleziona il modo testo
- .CHR\$(18) seleziona il modo grafico
- .CHR\$ applicato a un intero che è il codice di un carattere stampabile, stampa il carattere corrispondente
- .A seleziona il modo testo
- .C seleziona il colore del pennino (0= nero, 1=blu, 2=verde, 3=rosso)
- .D traccia segmenti dalla posizione attuale, toccando tutti i punti di cui sono fornite le coordinate (le coordinate devono essere comprese tra -999 e 999)
- .H posiziona il pennino all'origine
- .I definisce la posizione attuale del pennino come nuova origine
- .J funziona come il comando D, ma le coppie definiscono spostamenti anziché coordinate
- .L definisce il tipo di linea: un parametro 0 fa tracciare linee continue, il parametro 1 linee tratteggiate con tratteggio breve, il parametro 15 linee tratteggiate con tratteggi lunghi; tutti i valori tra 1 e 15 sono ammessi
- .M muove il pennino alle coordinate indicate (comprese tra -999 e 999), senza tracciamento
- .N stampa un carattere definito dall'utente (vedi U)
- .P stampa la stringa che segue
- .Q definisce la direzione di stampa:
 - 0 orizzontale da sinistra a destra
 - 1 verticale dall'alto in basso
 - 2 orizzontale capovolta da destra a sinistra
 - 3 verticale dal basso in alto
- .R sposta il pennino delle quantità indicate (comprese tra -999 e 999), senza tracciare linee
- .S specifica la dimensione dei caratteri; l'intero che segue può variare tra 0 e 63 e al crescere del valore cresce la dimensione del carattere; al valore 1 corrispondono 40 caratteri per riga
- .U permette la definizione di un carattere specificato dall'utente, mediante una successione di comandi, ciascuno dei quali prevede:
 - 0 pennino alzato (senza traccia)
 - 1 pennino abbassato (traccia)
 - coppia di coordinate (tra -7 e 7) a cui spostare il pennino
 un carattere così definito può comporsi al più di 15 di tali comandi.
- X permette la stampa di assi cartesiani, con le seguenti modalità:

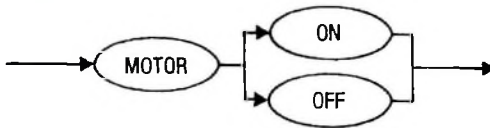
0 per l'asse orizzontale delle ascisse
 1 per l'asse verticale delle ordinate
 n. passi indica il numero di passi di spostamento del pennino che devono essere presenti tra una tacca e la successiva dell'asse da tracciare
 n. tacche definisce di quante tacche deve essere composto l'asse da tracciare

MAXFILES



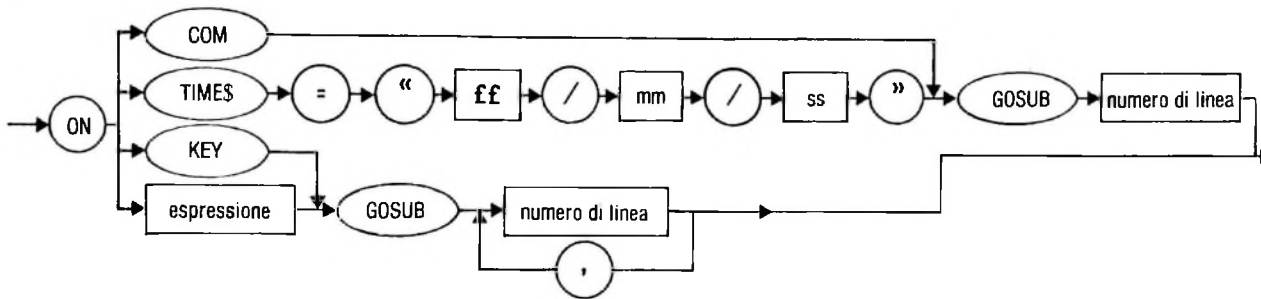
Permette di specificare il massimo numero di file (al più 15) che si intendono aprire contemporaneamente.

MOTOR



Permette il controllo di un registratore a cassetta da programma: l'opzione OFF disabilita i tasti manuali del registratore, mentre ON li abilita.

ON...GOSUB



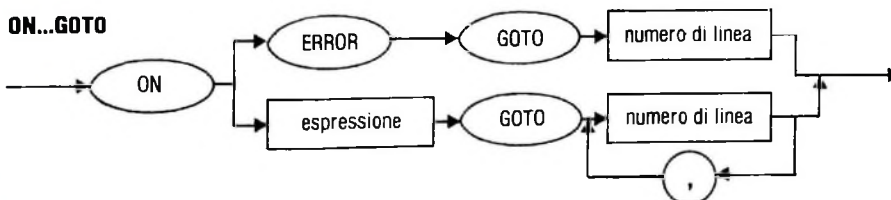
ON COM GOSUB viene usata congiuntamente all'istruzione COM. Infatti, nel caso in cui sia stata eseguita una COM ON il controllo del programma passa al sottoprogramma indicato qualora dalla linea di comunicazione siano giunti dati. Vedi COM.

L'uso di ON..GOSUB con la variabile TIMES permette di eseguire un sottoprogramma a una determinata ora.

L'uso di ON KEY GOSUB permette di specificare fino a 8 righe di sottoprogramma, una per ogni tasto funzione. Quando un tasto funzione viene premuto, il programma salta alla linea corrispondente, disabilitando automaticamente il tasto funzione. Esso viene riabilitato all'esecuzione della RETURN (a meno di un'esecuzione intermedia KEY OFF).

L'uso più classico dell'istruzione per realizzare selezioni multiple invia al sottoprogramma indicato in posizione I nella lista, se I è il valore (troncato) dell'espressione che compare nell'istruzione. Se il valore dell'espressione è nullo o maggiore del numero di alternative offerte (ma sempre minore di 255), l'esecuzione prosegue con l'istruzione successiva.

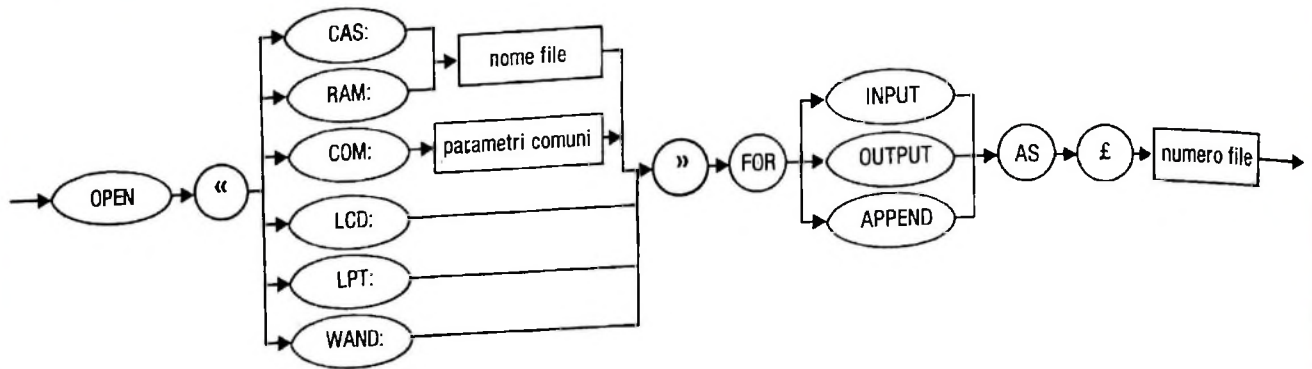
ON...GOTO



L'istruzione ON ERROR specifica la linea a cui saltare nel caso in cui un errore sia occorso (vedi ERROR).

L'istruzione ON con l'espressione trasferisce il controllo alla linea il cui numero è indicato in posizione I, se I è il valore dell'espressione (troncato). Se l'espressione ha valore nullo o maggiore del numero di alternative offerte (ma minore di 255), l'esecuzione prosegue con la linea successiva.

OPEN



parametri di comunicazione



PARAMETRO	VALORI POSSIBILI	SIGNIFICATO
Velocità di trasmissione (v)	1	75 baud
	2	110 baud
	3	300 baud
	4	600 baud
	5	1200 baud
	6	2400 baud
	7	4800 baud
	8	9600 baud
	9	19200 baud
Lunghezza parola (l)	6	6 bit
	7	7 bit
	8	8 bit
Parità (p)	O	Odd (dispari)
	E	Even (pari)
	N	No (senza) parità
	I	Ignora parità
Bit di stop (s)	1	1 bit di stop
	2	2 bit di stop
Stato della linea (x)	E	Enable (abilità)
	D	Disable (disabilità)

L'istruzione apre file, associando a essi un numero di riferimento che verrà successivamente usato da altre istruzioni, come INPUT, PRINT, CLOSE ecc.

Se si tratta di file su cassetta o in RAM, è possibile effettuare la OPEN per leggere (INPUT), per scrivere ex novo (OUTPUT) o per scrivere in aggiunta a ciò che il file contiene già (APPEND).

LCD apre il display di M10 trattandolo come se fosse un file.

LPT apre una stampante, trattandola come un file.

WAND apre il lettore di codici a barre, trattandolo come se fosse un file.

COM apre la linea di comunicazione, trattandola come un file; in tal caso è necessario fornire i vari parametri che caratterizzano le regole della comunicazione, come specificato nella tabella riportata.

Il numero del file deve essere compreso tra 1 e quello specificato con l'istruzione MAXFILE (vedi).

Non è possibile aprire con una modalità un file già aperto con un'altra.

I LINGUAGGI APPLICATIVI

Nati negli anni Sessanta, hanno fornito la base di confronto per evidenziare le carenze dei linguaggi tradizionali.

Lo sviluppo tecnologico ha permesso negli ultimi anni di abbassare enormemente i costi dell'hardware e di aumentarne le prestazioni; si è così sbilanciata la proporzione dei costi tra hardware e software (figura in basso) e si è quindi evidenziata la necessità di una nuova generazione di linguaggi per la produzione di software di basso costo e di facile manutenzione, nonché per controllo del parallelismo che le nuove tecnologie permettono.

Infatti negli ultimi anni, particolarmente a partire dalla pubblicazione dell'articolo di J. Backus, si è sviluppato l'interesse per i linguaggi di programmazione "non procedurali" (detti anche "non imperativi" o "non tradizionali"). Con questi termini si intende raggruppare un insieme abbastanza eterogeneo di strumenti di programmazione dalle caratteristiche generali notevolmente differenti da quelle dei linguaggi "procedurali", come Fortran, Algol e Pascal, che hanno avuto una grande diffusione fino a oggi.

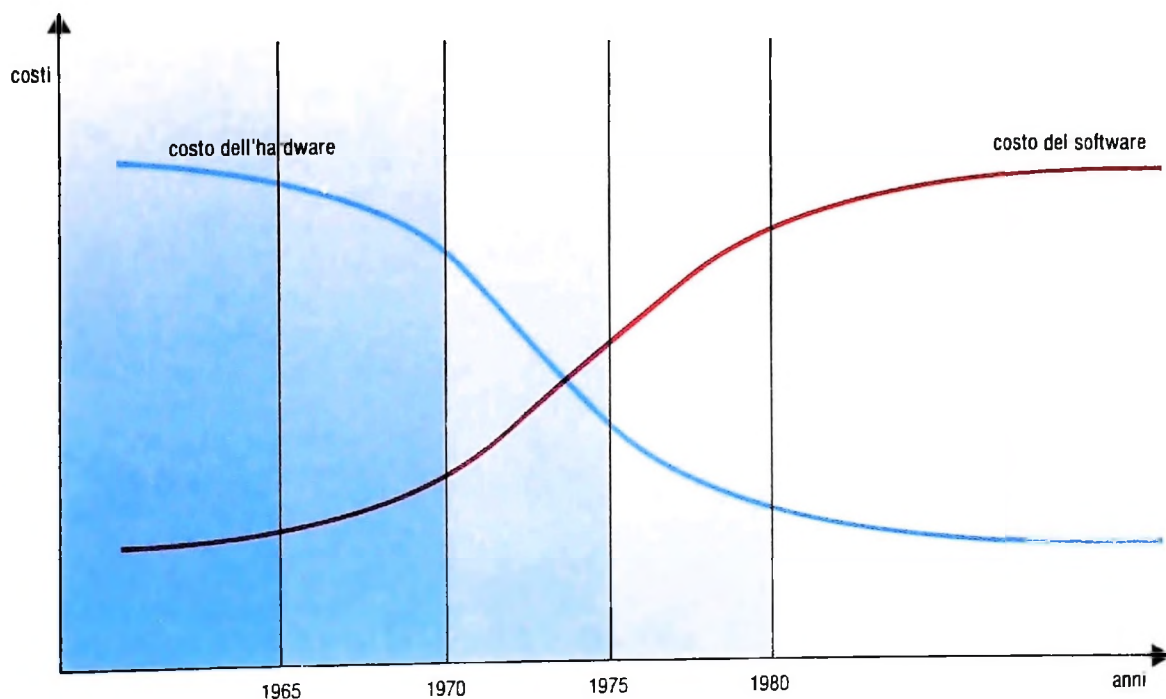
I linguaggi tradizionali e non: pregi e difetti

All'interno dei linguaggi non tradizionali si possono riconoscere alcune classi, come per esempio i linguaggi funzionali, quelli object oriented e i linguaggi logici di cui si è parlato in precedenti articoli.

Le motivazioni che hanno portato allo studio dei linguaggi non procedurali sono principalmente due: da un lato considerazioni sull'evoluzione del software, dall'altro gli sviluppi della tecnologia in campo hardware.

Nell'ambiente software si è evidenziato che quell'insieme di idee maturate negli ultimi anni Sessanta e raggruppate sotto il nome di "programmazione strutturata" sono insufficienti a ridurre drasticamente il costo della produzione di programmi, probabilmente perché propongono cambiamenti non abbastanza radicali delle tecniche di programmazione tradizionale. Al tempo stesso nell'ambiente hardware si stanno stu-

Nel grafico è mostrato l'andamento dell'evoluzione del rapporto fra costo dell'hardware e del software nel corso degli ultimi vent'anni.





Architettura della macchina di Von Neumann. La struttura è costituita da tre parti distinte: un unico processore, una memoria lineare e un canale di collegamento (definito da Backus "collo di bottiglia") che permette il passaggio di una parola alla volta.

diando nuove architetture e quindi si impone la necessità di nuovi metodi di programmazione in grado di sfruttare la possibilità di un alto livello di parallelismo nelle macchine che si è aperta con lo sviluppo della tecnologia VLSI.

Nell'ambito dei linguaggi non procedurali verrà esaminata la classe dei linguaggi funzionali o applicativi, alcuni dei quali nacquero negli anni Sessanta e hanno fornito la base di confronto per evidenziare le carenze dei modelli procedurali.

Esaminiamo ora brevemente queste critiche ai linguaggi imperativi (o tradizionali) nei loro punti fondamentali:

1) I linguaggi imperativi sono stati ideati pensando al modello di macchina di Von Neumann e ne rispecchiano la struttura, costituita da tre parti distinte: un unico processore, una memoria lineare e un canale di collegamento tra questi (definito da Backus il "collo di bottiglia") che permette il passaggio di una parola alla volta. Il compito del programma è quello di organizzare il traffico di informazione attraverso questo collo di bottiglia: nei linguaggi imperativi quest'ultimo è diventato una strozzatura intellettuale in quanto ha legato i programmatori a ragionare "una parola alla volta" invece di pensare in termini di unità concettuali più vaste.

2) I linguaggi imperativi mancano di utili proprietà matematiche. Si possono notare delle differenze sostanziali tra i linguaggi tradizionali e la notazione matematica: in primo luogo la matematica è statica, è quindi difficile assimilare una funzione matematica al concetto di una procedura, che può dare risposte diverse ogni volta che viene chiamata, in dipendenza dallo stato del sistema che non è determinabile dalla definizione della procedura; in secondo luogo c'è in matematica un uso consistente dei nomi per le variabili, che in un contesto denotano sempre lo stesso valore, mentre il concetto di variabile nei linguaggi imperativi è associato a quello di cella di memoria che quindi può assumere valori diversi nel corso dell'elaborazione; infine in matematica ci si può appoggiare al concetto di uguaglianza che permette la sostituzione di espressioni che denotano lo stesso valore, mentre nei

linguaggi imperativi l'uguaglianza di due espressioni è dinamica, poiché dipende dall'evoluzione della computazione, e quindi la sostituibilità delle espressioni ha validità limitata.

3) I linguaggi tradizionali stanno crescendo più nelle dimensioni delle specifiche che nel potere espressivo. Viene dedicata molta attenzione ai particolari sintattici e alla definizione di nuove caratteristiche utilizzabili in casi particolari sottilmente diversificati, viene appesantita così la base concettuale del linguaggio di infrastrutture non essenziali, limitando la possibilità di organizzare i programmi come composizione di unità concettuali ben definite che permettano di esprimere sinteticamente un algoritmo.

4) Un algoritmo può essere visto come un insieme parzialmente ordinato di computazioni; un linguaggio convenzionale ne impone la rigida sequenzializzazione. Questa operazione è a carico del programmatore, che deve anche provvedere all'organizzazione dei risultati intermedi delle sotto-computazioni e alla loro memorizzazione per mezzo dell'assegnamento. L'uso di queste variabili per i risultati intermedi fa sì che l'apparenza del programma sia spesso diversa da quella dell'algoritmo da cui deriva ed è necessario "eseguirlo" mentalmente per comprenderne il significato, in quanto l'esame statico del testo è scarsamente significativo.

Alla base di tutti questi problemi vi sono comunque i due concetti strettamente correlati di variabile vista come cella di memoria e di assegnamento. In un linguaggio imperativo una variabile non ha alcuna corrispondenza con l'ente matematico, ma diventa semplicemente una delle parti che costituiscono lo "stato della computazione" che viene via via modificato da una sequenza di assegnamenti. Ogni computazione quindi si basa su questo stato e a sua volta lo modifica, in questo senso si dice che la computazione avviene per "side effect". L'assegnamento è il collo di bottiglia dei linguaggi di programmazione e fa ragionare in termini di singoli valori esattamente come una macchina di Von Neumann fa con i propri registri. Inoltre l'assegnamento è la causa della asim-

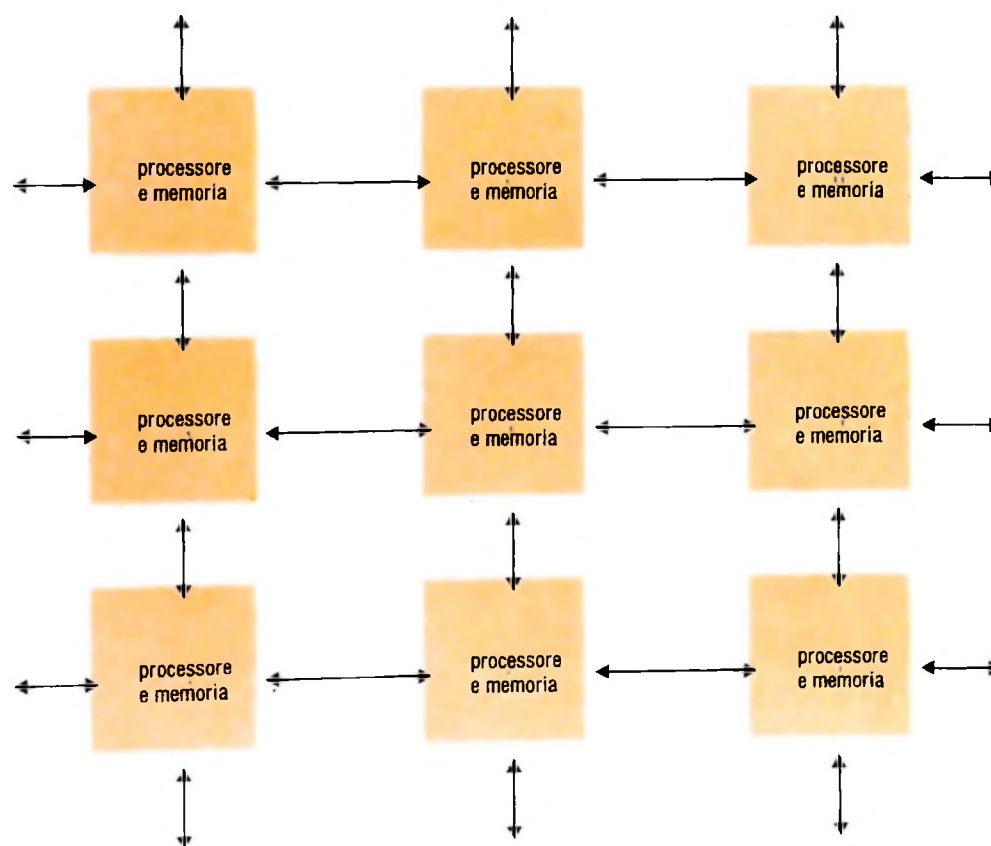
metria semantica dei linguaggi di programmazione. Esiste infatti il mondo delle espressioni alla destra dell'istruzione di assegnamento che gode delle proprietà matematiche di cui si è parlato (se non vengono rese inutili dai side effect) e il mondo delle variabili alla sinistra che invece crea i side effect e fa perdere al linguaggio le proprietà matematiche.

Questi, secondo alcuni studiosi, sono i limiti dei linguaggi tradizionali che hanno causato negli ultimi anni una sempre maggiore difficoltà di produrre software affidabile e facilmente manutenibile e che soddisfacesse le crescenti esigenze qualitative delle applicazioni.

Oltre ai problemi nello sviluppo del software negli ultimi anni si è assistito a un impressionante calo dei costi dell'hardware unito, grazie alla tecnologia VLSI, alla possibilità di realizzare sistemi sempre più complessi e dalle architetture estremamente differenti da quelle tradizionali. Questa evoluzione tecnologica è stata finora sfruttata per la produzione di hardware tradizionale capace di grandi prestazioni, ma ha anche stimolato la ricerca di nuovi strumenti concettuali per l'ideazione e il controllo di modelli computazionali sostanzialmente differenti da quello di Von Neumann. All'inizio della produzione dei computer il processore e la memoria venivano costruiti con due tecnologie diverse e il primo era di gran lunga il più costoso dei due; ora sia l'uno che l'altra

possono essere prodotti con la stessa tecnologia e la differenza di costi non è più apprezzabile. Quindi sono crollate le ragioni che consigliavano la costruzione di macchine uniprocessor ed è ragionevole pensare allo sviluppo di computer con una rete di processori addetti alla computazione. Inoltre così facendo si possono aumentare le prestazioni. La velocità di elaborazione di una macchina di Von Neumann è limitata dalla banda passante del canale tra memoria e processore. Per aumentarne la velocità si deve migliorare la tecnologia con cui sono costruiti i componenti e c'è un limite a questo processo. In una architettura multiprocessor invece è possibile, in linea di principio, aumentare la velocità aggiungendo nuovi processori alla rete. Il problema dell'utilizzo delle enormi potenzialità di queste architetture è duplice: in primo luogo si devono trovare dei metodi di programmazione che sfruttino questa potenziale concorrenza; in secondo luogo è necessario che questi linguaggi siano trasparenti nei confronti dell'hardware cui sono destinati, ovvero che non costringano il programmatore a gestire esplicitamente l'organizzazione dei vari processi concorrenti. Questo può avvenire solo con un linguaggio intrinsecamente asincrono.

Il rinato interesse per i linguaggi funzionali è motivato dal fatto che essi sembrano avere molte delle caratteristiche di cui si sente la mancanza nei linguaggi procedurali e che si



In figura è riportato un esempio di architettura non tradizionale. La memoria e la capacità di calcolo sono distribuite su più unità comunicanti fra di loro.

possono facilmente adattare ad architetture parallele. La mancanza di assegnamento nei linguaggi funzionali elimina tutte le difficoltà pratiche e formali a esso connesse. In questi linguaggi si può effettivamente parlare di funzioni in senso matematico e le variabili hanno un significato statico nel loro contesto. Inoltre essi poggiano su modelli come il lambda calcolo di Church o la logica combinatoria di Curry, che sono decisamente più raffinati e dotati di precise proprietà formali che non il modello di Von Neumann. Queste caratteristiche, unite alla completa mancanza di dipendenze dallo stato della computazione, che è un concetto assente nei linguaggi funzionali, permettono di ragionare e di dare significato a un programma in termini puramente statici senza dover tenere conto della storia del sistema che lo valuta. La definizione della semantica denotazionale di un linguaggio applicativo è generalmente concisa ed elegante, mentre per i linguaggi imperativi una tale descrizione è un compito di difficile risoluzione e sicuramente di scarsa utilità poiché può fornire minori indicazioni sui programmi che non l'esame diretto dei programmi stessi.

All'interno della classe dei linguaggi funzionali si possono notare alcuni orientamenti di particolare interesse. Il lavoro di P. Landin proponeva lo schema generale di una famiglia di linguaggi chiamata ISWIM da cui deriva anche il linguaggio che presenteremo, nonché il SASL di Turner. In questi linguaggi le funzioni sono espresse in lambda-notazione e il loro modello di valutazione deriva dal lambda calcolo.

Un altro gruppo di linguaggi è quello proposto da J. Backus

in una sua fondamentale pubblicazione, fra cui FP, FFP e AST. Questi linguaggi sono basati sul calcolo combinatorio e i programmi sono composizioni di funzioni che si applicano direttamente agli argomenti senza l'uso di lambda-variabili. La ricerca di Backus va nella direzione di una "algebra dei programmi" che permetta di ragionare sugli stessi e di verificarli in termini del linguaggio stesso.

Un'altra classe di linguaggi funzionali è nata dallo sviluppo di architetture non tradizionali. Fra questi vi sono i linguaggi data-flow e i linguaggi reduction, per lo più legati ai prototipi delle macchine di cui sono il linguaggio di sistema.

Conclusioni

Nonostante la loro eleganza e semplicità strutturale, i linguaggi funzionali, sebbene fossero già noti da tempo, non hanno mai riscosso un grande successo se non nel ristretto ambito della ricerca per la difficoltà e inefficienza della loro implementazione sulle macchine esistenti, che meglio si adattano ai linguaggi convenzionali. Paradossalmente, nonostante la loro fama di inefficienza, i linguaggi funzionali vengono studiati per poter ottenere quell'incremento di prestazioni che i linguaggi tradizionali non sembrano permettere. Gli studi finora effettuati non permettono ancora di esprimere un giudizio definitivo sui linguaggi applicativi e si devono ancora raccogliere molti elementi per poter stabilire le potenzialità di questi linguaggi.

Glossario

POKE - un'istruzione che effettua l'operazione opposta rispetto all'istruzione PEEK: consente, cioè, di inserire uno specifico valore direttamente in una specifica locazione di memoria della macchina.

Protocollo - un insieme di regole per la trasmissione di dati: ambedue i calcolatori che si trovano alle estremità di una linea di trasmissione debbono utilizzare lo stesso protocollo per controllare il flusso delle informazioni.

Sintesi del parlato - la capacità, da parte di una macchina, di tradurre un testo in una emissione sonora comprensibile. È uno dei campi di ricerca più interessanti, per la realizzazione di nuove forme di interfaccia fra calcolatore e utente, che utilizzino direttamente il linguaggio parlato.

Software integrato - un insieme di programmi che svolgono funzioni diverse, ma che possono avere in comune i dati su cui operano e che hanno una interfaccia comune verso l'utente. Sui calcolatori personali, i pacchetti di software integrato solitamente comprendono le funzioni di scrittura testi, data base, tabellone elettronico, comunicazione, cioè le funzioni più comuni nelle applicazioni personali, a cui si affiancano a volte anche funzioni di grafica, ma in linea di principio qualunque funzione potrebbe entrare a far parte di un pacchetto integrato. Il vantaggio dell'integrazione sta nella messa in comune degli archivi dati, con la possibilità quindi di trasferire dati, senza doverli reinscrivere, da una funzione all'altra.

Spooling - una tecnica per aumentare l'efficienza delle operazioni di input e output che chiamano in causa dispositivi relativamente lenti. Una stampante, per esempio, è un dispositivo lento: mentre si stampa una lunga serie di dati, l'unità centrale resta per la maggior parte del tempo inoperosa. Per aumentare l'efficienza, i dati da stampare possono venire inviati a un dispositivo intermedio (disco o apposito dispositivo di spooling, dotato di memoria RAM), lasciando libera l'unità centrale per altre elaborazioni.

Traccia - un'area di memorizzazione magnetica, di forma circolare, su un disco. Il numero delle tracce presenti su un dischetto può variare, in funzione delle caratteristiche del supporto fisico stesso e del tipo di formato che ciascun calcolatore impone.

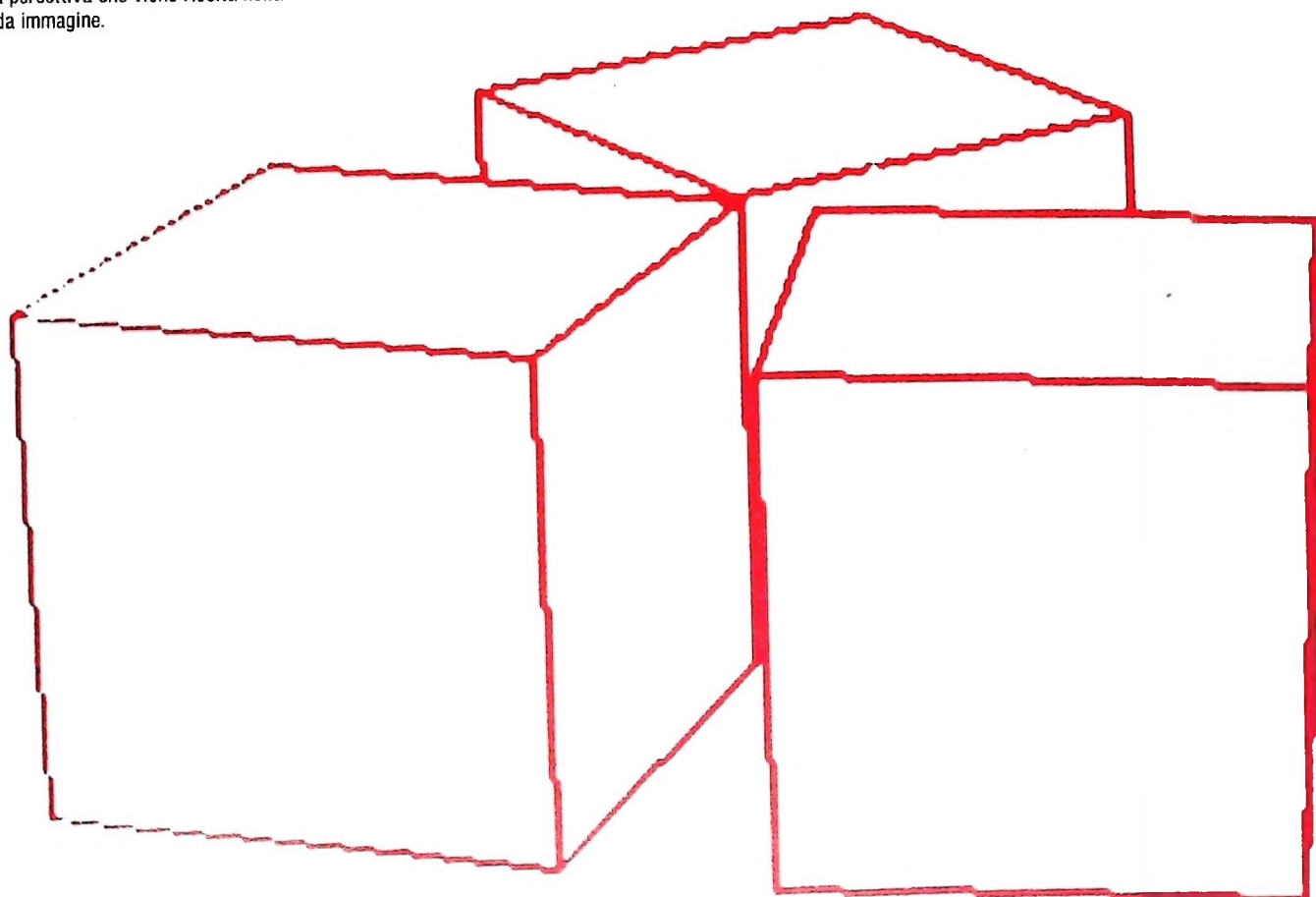
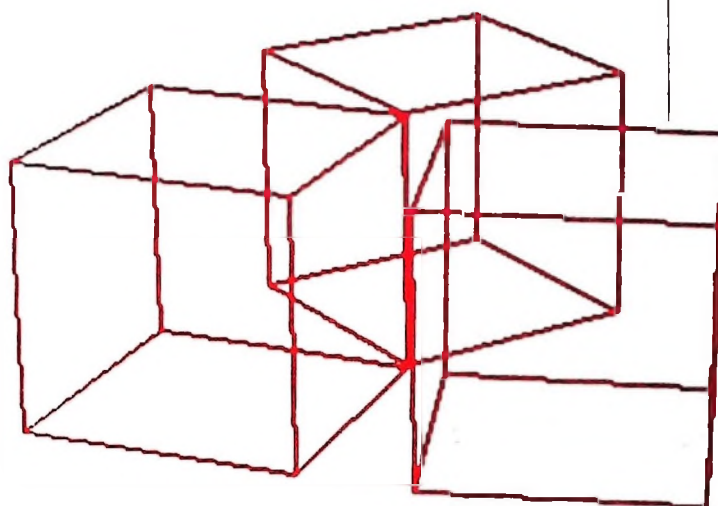
UART - sigla di Universal Asynchronous Receiver/Transmitter (Ricevitore/trasmittitore asincrono universale). Un dispositivo che trasforma dati paralleli in forma seriale per la trasmissione lungo un'interfaccia seriale, o viceversa. L'UART si rende necessario perché la CPU trasferisce le informazioni in parallelo sul bus dei dati e degli indirizzi, mentre in genere i dispositivi periferici lavorano in modo seriale. L'UART è universale nel senso che può elaborare informazioni per tutti i tipi di dispositivi, ciascuno dei quali richiede una particolare velocità di trasferimento, un particolare tipo di controllo di parità, un particolare metodo per separare un byte di informazione dall'altro. La denominazione di "asincrono" deriva dal fatto che la distribuzione nel tempo degli impulsi di trasferimento è diversa da quella dell'unità centrale.

ELIMINAZIONE DELLE LINEE E DELLE FACCE NASCOSTE

Come risolvere un'ambiguità percettiva dell'immagine disegnata col computer.

Il primo problema che la ricerca del realismo deve risolvere è l'eliminazione di ogni ambiguità che sussiste in una figura disegnata con il computer. Osservando la classica figura del cubo disegnata con la tecnica a "fil di ferro" (quella che con terminologia anglosassone è chiamata *wire frame*), è impossibile decidere quale sia la forma cui la rappresentazione è riferita: è un cubo visto dall'alto o dal basso? È una scatola vuota con una faccia aperta o una gabbia di fili di ferro? Per risolvere l'ambiguità è necessario che il sistema di elaborazione disponga di un insieme di informazioni sufficienti a decidere che cosa è visibile e che cosa no.

Le due immagini rappresentano la progressiva eliminazione delle linee e delle facce nascoste di tre cubi. Si osservi come la mancata rimozione delle facce nascoste generi nell'osservatore un'ambiguità percettiva che viene risolta nella seconda immagine.



PRODUZIONE EIDOS

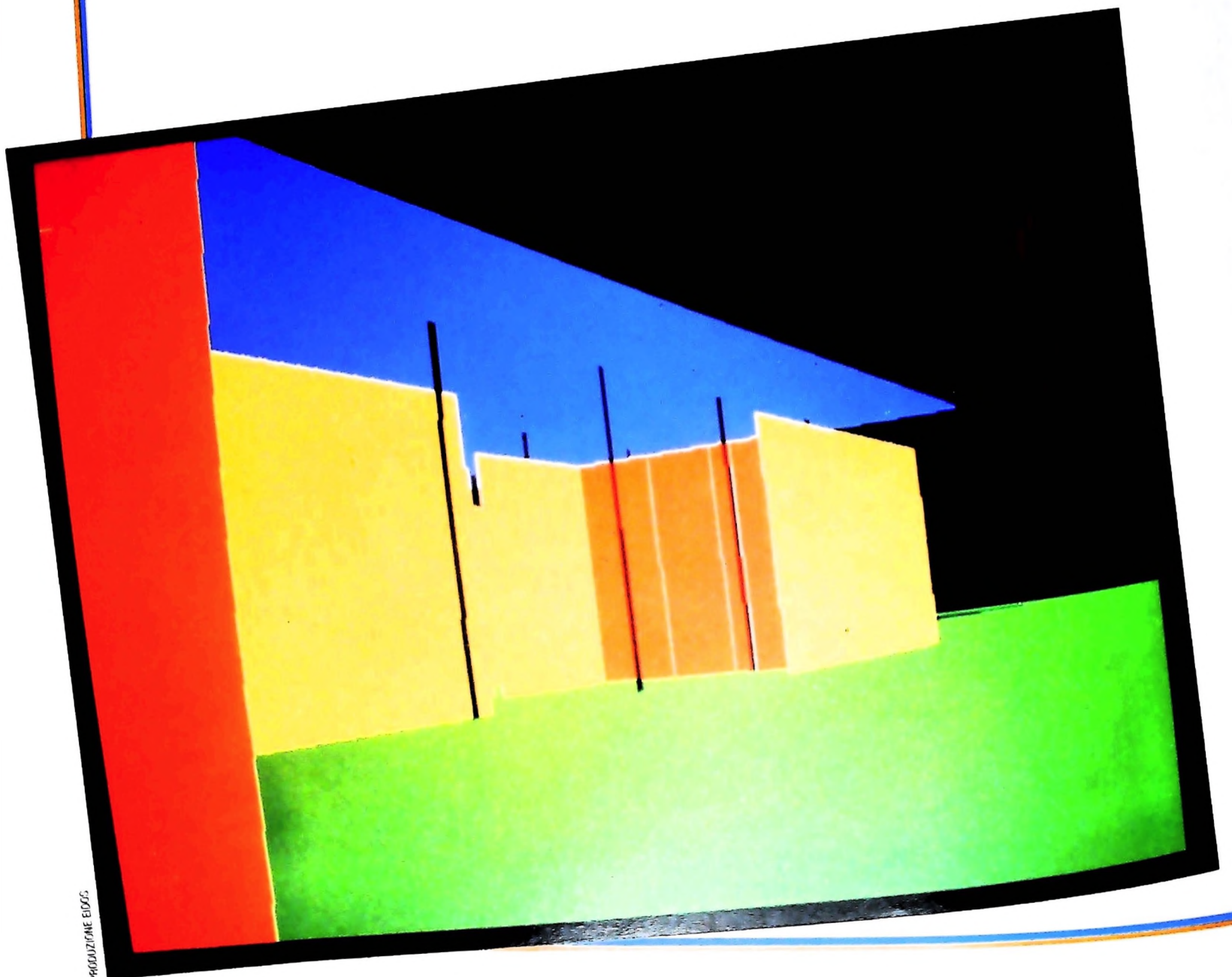
L'algoritmo di Galimberti e Montanari

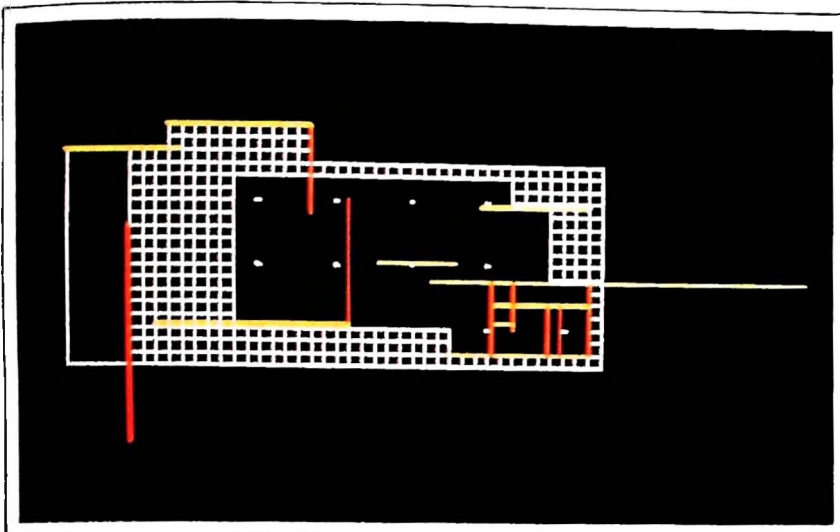
Questo approccio classico costituisce un importante contributo italiano all'eidomatica. Ideato nell'ambito della tesi di laurea del professor Ugo Montanari, venne pubblicato nel 1969 sulla prestigiosa rivista *Communications of the A.C.M.*, e costituì, all'epoca, una delle poche soluzioni "esatte" al problema della eliminazione delle linee che non sono visibili in una proiezione prospettica di una figura poliedrica.

Questo algoritmo, particolarmente adatto alla costruzione di immagini per il plotter, procede secondo tre passi fondamentali. Per comprenderli, è bene ricordare che una figura (o più in generale una scena poliedrica tridimensionale) viene descritta, per il riconoscimento e la eliminazione delle linee nascoste in una data proiezione, mediante informazioni di tipo geometrico e di tipo topologico. Le prime sono condensate nelle coordinate dei vertici dei poligoni che compongono le facce dei poliedri, le seconde sono le relazioni che legano vertici, spigoli e facce, costituendo forme convesse o concave, dotate di "buchi" o connesse. Pertanto la descrizione dei poliedri da analizzare dovrà comprendere, oltre ai vertici, una descrizione degli spigoli e delle facce che compongono ogni singolo poliedro di una scena comunque complessa.

Date queste informazioni, i passi dell'algoritmo sono così schematizzati:

- 1) per ogni spigolo si decide se esso appartiene a un diedro convesso oppure concavo;
- 2) tutte le linee che sono nascoste da un volume vengono escluse. Se a questo punto l'oggetto da rappresentare è convesso, allora l'algoritmo si ferma automaticamente tracciando solo le linee rimanenti (ovvero quelle non mascherate dal volume stesso);
- 3) tutti gli spigoli che non sono mascherati da volumi vengono analizzati successivamente. Per ogni spigolo si analizza uno dei due estremi e gli si attribuisce un "tipo" caratterizzato dall'insieme delle facce che lo mascherano. Quindi partendo da tale punto si esamina l'intero spigolo, e questo insieme di facce mascheranti viene modificato calcolando le intersezioni del segmento proiettato sul piano di proiezione con ogni altro segmento. Alla fine di questo passo è noto il "tipo" dell'altro estremo del segmento. Infine, sulla base del "tipo" così determinato si procede al tracciamento dei segmenti che risultano completamente visibili. A tale scopo, dato un punto intersezione di due segmenti sul piano di proiezione, esso sarà la proiezione di due punti, appartenenti ai due spigoli che, proiettati, danno luogo all'inter-



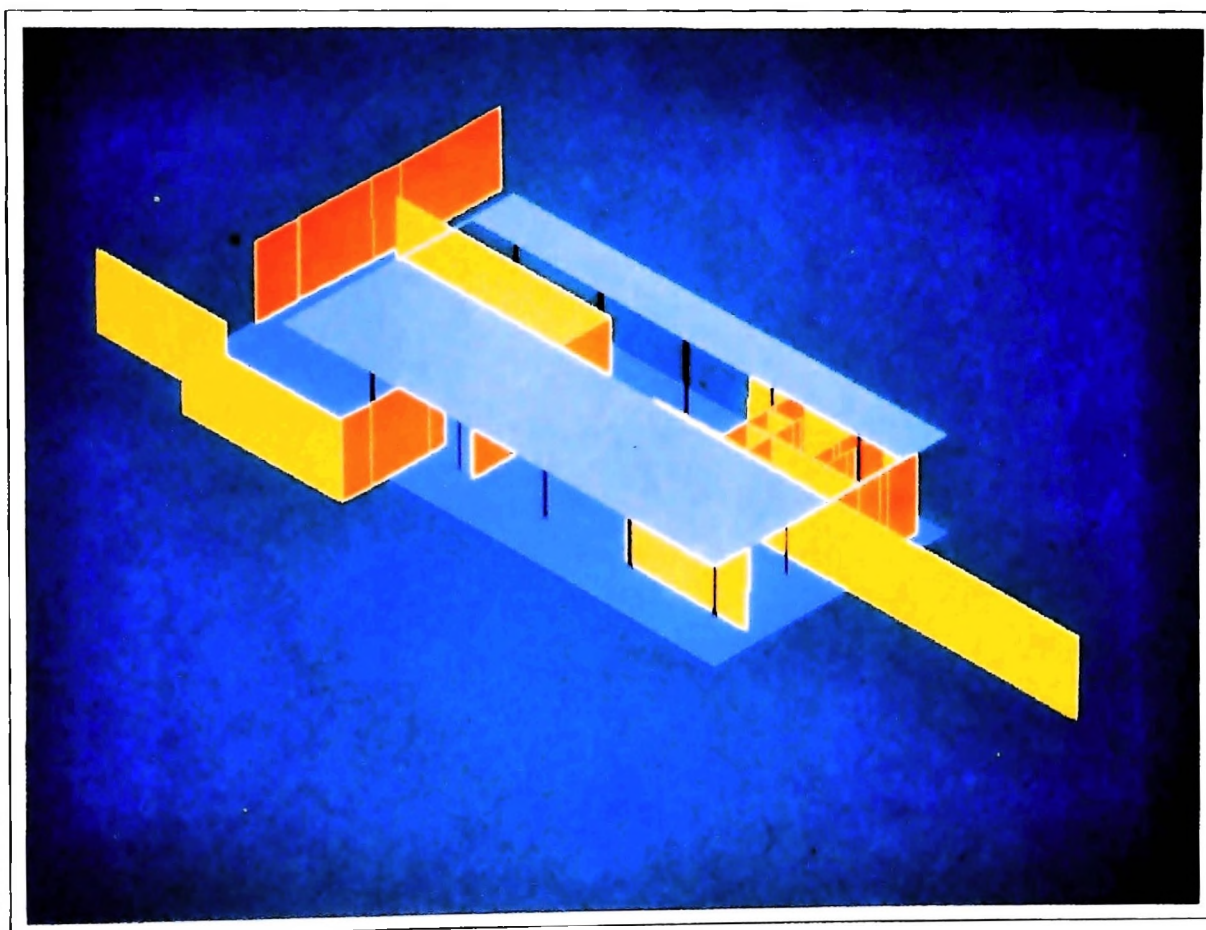


sezione; si tratta quindi di decidere quale di questi due estremi è vincente, ovvero si trova più vicino all'osservatore: tale punto determina la priorità dello spigolo di appartenenza.

Il vantaggio più significativo di questo algoritmo consiste in un aspetto: si tratta di un algoritmo "esatto", al limite delle capacità di rappresentazione numerica dell'elaboratore. Usato dal punto di vista computazionale, l'algoritmo di Galimberti e Montanari è relativamente costoso, in quanto la complessità cresce approssimativamente come il quadrato del numero di spigoli della scena analizzata.

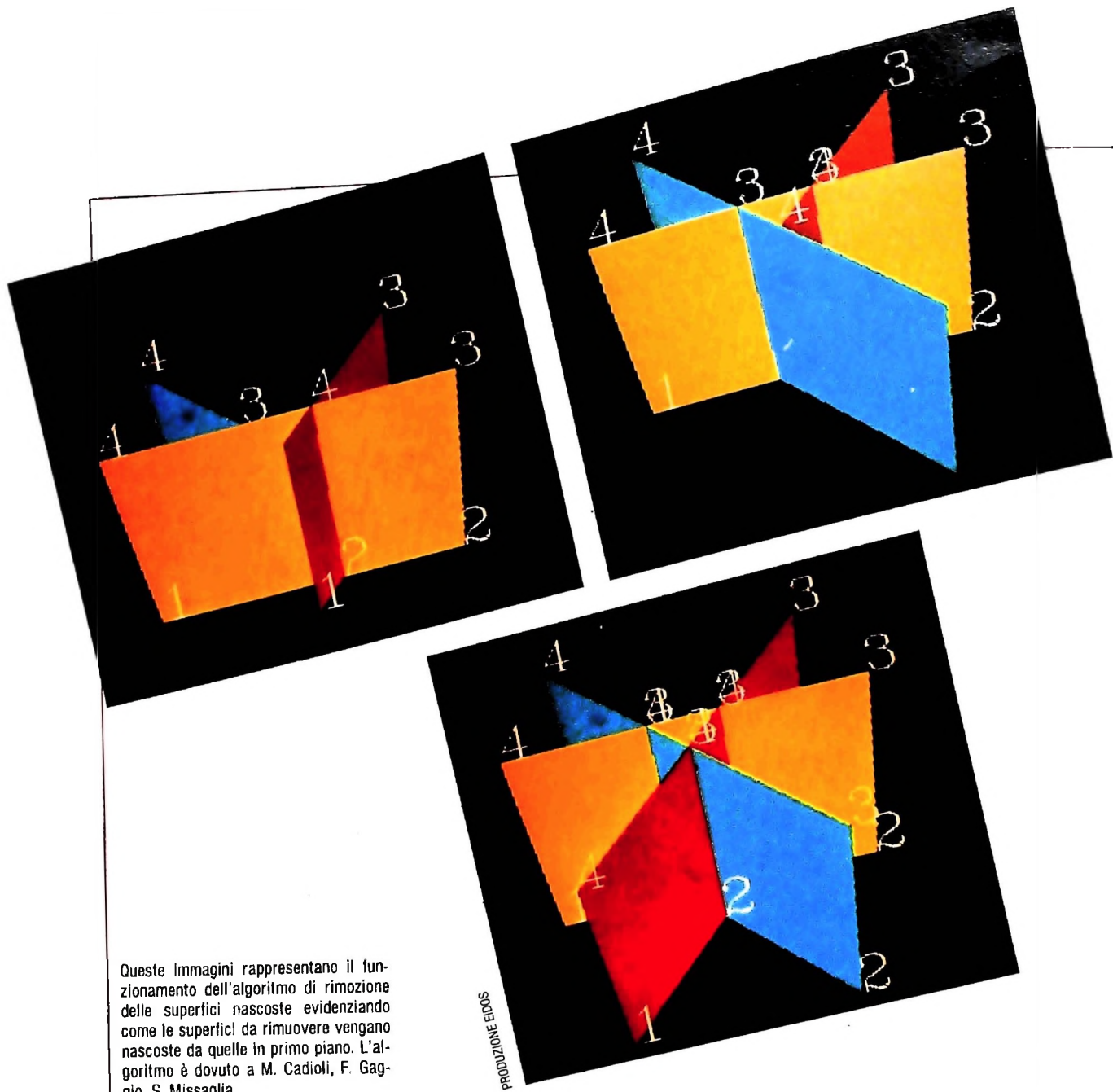
Altri algoritmi, come lo Z-Buffer, hanno complessità costante pari al numero di pixel del display, mentre quello "del pittore" è legato più al numero di facce che a quello di spigoli, ottenendo così una complessità notevolmente inferiore.

Considerazioni sulla complessità degli algoritmi di rimozione di linee e superfici nascoste devono comunque essere espresse tenendo presente gli scopi. Spesso, per certe applicazioni, il costo computazionale può assumere un valore irrisorio di fronte all'esigenza di ottenere risultati esatti: proprio in questi casi l'algoritmo di Galimberti e Montanari risulta tra i più validi.



La visualizzazione della casa di Mies van der Rohe, progettata per l'Esposizione di Berlino del 1931, è stata ottenuta mediante l'algoritmo "del pittore". Questo tema ha cimentato anche Galimberti e Montanari, che hanno adottato lo

stesso soggetto per illustrare i risultati del loro algoritmo. Le immagini riprodotte in queste due pagine sono state realizzate da Chiara Negri e Marco Cadioli con la collaborazione di Attilio Marcolli.



Queste immagini rappresentano il funzionamento dell'algoritmo di rimozione delle superfici nascoste evidenziando come le superfici da rimuovere vengano nascoste da quelle in primo piano. L'algoritmo è dovuto a M. Cadioli, F. Gaggio, S. Missaglia.

L'eliminazione delle linee e delle superfici nascoste è quindi in primo luogo un problema che richiede la definizione dei dati di partenza in modo completo. Per decidere se una linea è visibile oppure no, è infatti necessario sapere se è coperta da una faccia opaca, perciò non è più sufficiente descrivere una figura in termini di segmenti che compongono poligoni, ma occorre introdurre la nozione di faccia, ovvero di "catena" di segmenti (poligoni) chiusa, e disposti tutti sullo stesso piano. Questa informazione consente di ricavare dalla descrizione di una forma geometrica anche proprietà topologiche, essenziali nella soluzione del problema posto.

La soluzione automatica del problema, una volta che le informazioni necessarie siano disponibili, può essere impostata in vari modi, secondo che si analizzi la struttura di una scena, composta di poliedri e facce piane, nelle sue condizioni reali, oppure che si analizzi la sua proiezione sullo schermo del computer. In entrambi i casi occorre prima di tutto calcolare tutte le trasformazioni geometriche essenziali e creare una scena costituita da poliedri trasformati secondo la proiezione prospettica desiderata. Successivamente la scena viene analizzata per decidere che cosa è visibile e che cosa no.

Un approccio importante è quello basato sull'algoritmo di

Galimberti e Montanari, mediante il quale la soluzione viene trovata analizzando i rapporti che intercorrono tra ogni spigolo e ogni faccia della scena data: il risultato è accurato, ma pecca per un'elevata complessità computazionale.

Altri algoritmi sfruttano la nozione di "coerenza", secondo la quale le proprietà specifiche di una scena non si presentano in modo casuale: se un punto è coperto da una faccia è molto probabile che tutti i punti "abbastanza" vicini a questo siano coperti dalla stessa faccia. Tale proprietà consente di impostare algoritmi come quello di Warnock, in cui la scena, proiettata sullo schermo del computer, viene suddivisa in regioni in modo ricorsivo, e ciascuna regione viene analizzata per decidere se segmenti e facce si coprono oppure no.

Infine un algoritmo di realizzazione piuttosto facile e di complessità relativamente bassa è quello cosiddetto "del pittore" in cui l'elaboratore dispone in ordine le varie facce di una scena. L'ordinamento è basato sulla vicinanza di ogni faccia al punto di osservazione. Deciso l'ordinamento delle facce, esse vengono disegnate e colorate in modo progressivo, sicché le facce più vicine coprono con il loro colore quelle più distanti, proprio come fa un pittore che, sulla tela, ripassi con nuovi colori parti già dipinte in precedenza.

— 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**
CONSCIAMOCI MEGLIO.



Olivetti M10 vuol dire disporre del proprio ufficio in una ventiquattre. 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

Il più piccolo personal computer Olivetti M10 è in grado di scrivere, stampare, memorizzare dati e testi, 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.

OLIVETTI
 VIA
 CARICATI
 TELEFONO