

CADEL

Spediz. in abbonamento postale GR. 11/70 L. 2.000  
(...)

# 20 CORSO PRATICO COL COMPUTER

421727

F4 F5 F6 F7 F8

diretto da GIANNI DEGLI ANTONI

è una iniziativa  
**FABBRI EDITORI**

in collaborazione con  
**BANCO DI ROMA**

e **OLIVETTI**



BATTERY LOW



**LIBRERIA DI SOFTWARE:  
COME TRASFORMARE  
IL TEMPO DA SOLARE  
IN SIDERALE E VICEVERSA**

**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  
Banca di Messina  
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  
TULLIO CHERSI, ADRIANO DE LUCA (Professore di Architettura del Calcolatore all'Università Autonoma Metropolitana di Città del Messico), GOFFREDO HAUS, MARCO MAIOCCHI, DANIELE MARINI, GIANCARLO MAURI, CLAUDIO PARMELLI, ENNIO PROVERA

Testi  
FABRIZIO CELENTANO, CLAUDIO PARMELLI, GOFFREDO HAUS, EIDOS (TIZIANO BRUGNETTI), 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 FENGHI

Coordinatore settore scientifico  
UGO SCAIONI

Redazione  
MARINA GIORGETTI  
LOGICAL STUDIO COMMUNICATION

Art Director  
CESARE BARONI

Impaginazione  
BRUNO DE CHECCHI  
PAOLA ROZZA

Programmazione Editoriale  
ROSANNA ZERBARINI  
GIOVANNA BREGGÉ

Segretarie di Redazione  
RENATA FRIGOLI  
LUCIA MONTANARI

**AVVISO AI LETTORI**  
Contrariamente a quanto annunciato nel numero precedente la copertina per rilegare il secondo volume del "Corso pratico col computer" sarà in edicola con il n. 22.

Corso Pratico col Computer - Copyright © sul fascicolo 1984 Gruppo Editoriale 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 Gruppo Editoriale Fabbri S.p.A. via Mecenate, 91 - Tel. 50951 - Milano - Distribuzione per l'Italia: A. & G. Marco s.a.s. via Forzezza 27 - Tel. 2526 - Milano - Pubblicazione periodica settimanale - Anno I - n. 20 - esce il giovedì - Spedizione in ab. 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

# LE ALTRE SCIENZE

Esaminiamo ora le applicazioni dell'informatica alla didattica della chimica, della biologia e delle scienze naturali.

L'insegnamento della chimica e, soprattutto, delle scienze naturali viene condotto di solito in maniera prevalentemente qualitativa. Questo non significa solo che si utilizzano scarsamente la fisica e la matematica studiate contemporaneamente, o che già si sanno, ma soprattutto che le informazioni date, numeriche e non numeriche, non vengono correlate fra di loro secondo modelli formulati esplicitamente e rimangono quindi, se mai, nozioni poco utili al fine di orientarsi in un mondo che richiede ragionamenti e scelte basati su correlazioni e ipotesi affidabili.

Non è facile mutare tale situazione. È vero che non occorre molta matematica per dare un'impronta quantitativa a un corso di scienze, insegnando a ragionare per modelli. Ma è anche vero che ciò richiede tempo e la collaborazione di più insegnanti, non facile e dispersiva. Un elaboratore, anche piccolo, può risolvere questa contraddizione. Consente infatti di non perdere tempo con conti e grafici, concentrandosi sul problema, che può venire approfondito coinvolgendo gli allievi nella stesura del programma e nella raccolta dei dati. Non si tratta di giocare con la macchinetta durante l'ora di



HEWLETT-PACKARD

Il calcolatore non è utile solo in laboratorio: anche la didattica delle scienze naturali può trarne vantaggio, sfruttando in particolare la possibilità di costruire modelli quantitativi



biologia, ma di compiere un'operazione didattica importante per varie ragioni.

La prima, quasi banale, consiste nella possibilità di eseguire rapidamente molti esercizi per correlare nozioni disperse e prendere confidenza con i valori assunti dalle grandezze nel mondo reale. La seconda è la possibilità di cominciare molto presto a introdurre, spinti dalla necessità di cercare correlazioni, cioè spiegazioni dei fatti esposti, i concetti di modello e di affidabilità delle valutazioni che questo consente.

Una terza, e fermiamoci qui per verificare con esempi quanto affermato, è l'opportunità di portare nella scuola, oltre che nel gioco imitativo, uno strumento di lavoro, recuperando valori didattici che parevano perduti in molti ordini di scuole.

Certo, l'introduzione dell'elaboratore nella scuola comporta il rischio, comune a tutte le tecnologie didattiche, di rendere più difficile il cammino di chi non è svelto ad appropriarsi dei nuovi modi di lettura delle informazioni se quindi non se ne serve o li usa in maniera dispersiva. Una tecnologia potente, usata imprudentemente, può approfondire i distacchi creati dalla diversa velocità di apprendimento dei singoli, condizionata da fattori fisiologici e ambientali che non dipendono dall'effettiva capacità e volontà di imparare. Se bene usata può contribuire invece a colmarli.

Le possibilità di impiego di un elaboratore personale sono in pratica limitate solo dalla fantasia. Per esempio una macchina in grado di gestire una lista di indirizzi o un'agenda già permette di raccogliere, ordinare e ritrovare informazioni di tutti i tipi, che si potranno poi classificare o correlare con programmi scritti appositamente. Pensiamo alla botanica sistematica, considerata noiosa e antiquata, utile però per mostrare come si siano sviluppate caratteristiche necessarie per la sopravvivenza in certi ambienti. Ecco che, una volta che si sia raccolta una lista di caratteristiche significative, un programma molto semplice, basato su una serie di IF, AND e OR consente di effettuare delle scelte precise e di classificare una pianta seguendo il metodo ben collaudato del Dalla Fior (G. Dalla Fior: *La Flora del Trentino*, Monauli, Trento). Lo stesso si può fare per qualche animale.

Unica precauzione, non appesantire il programma rendendolo troppo analitico. Questo lavoro può servire per fare comprendere che cosa sia veramente necessario per la classificazione e come essa risulti dall'ordinamento di poche caratteristiche essenziali.

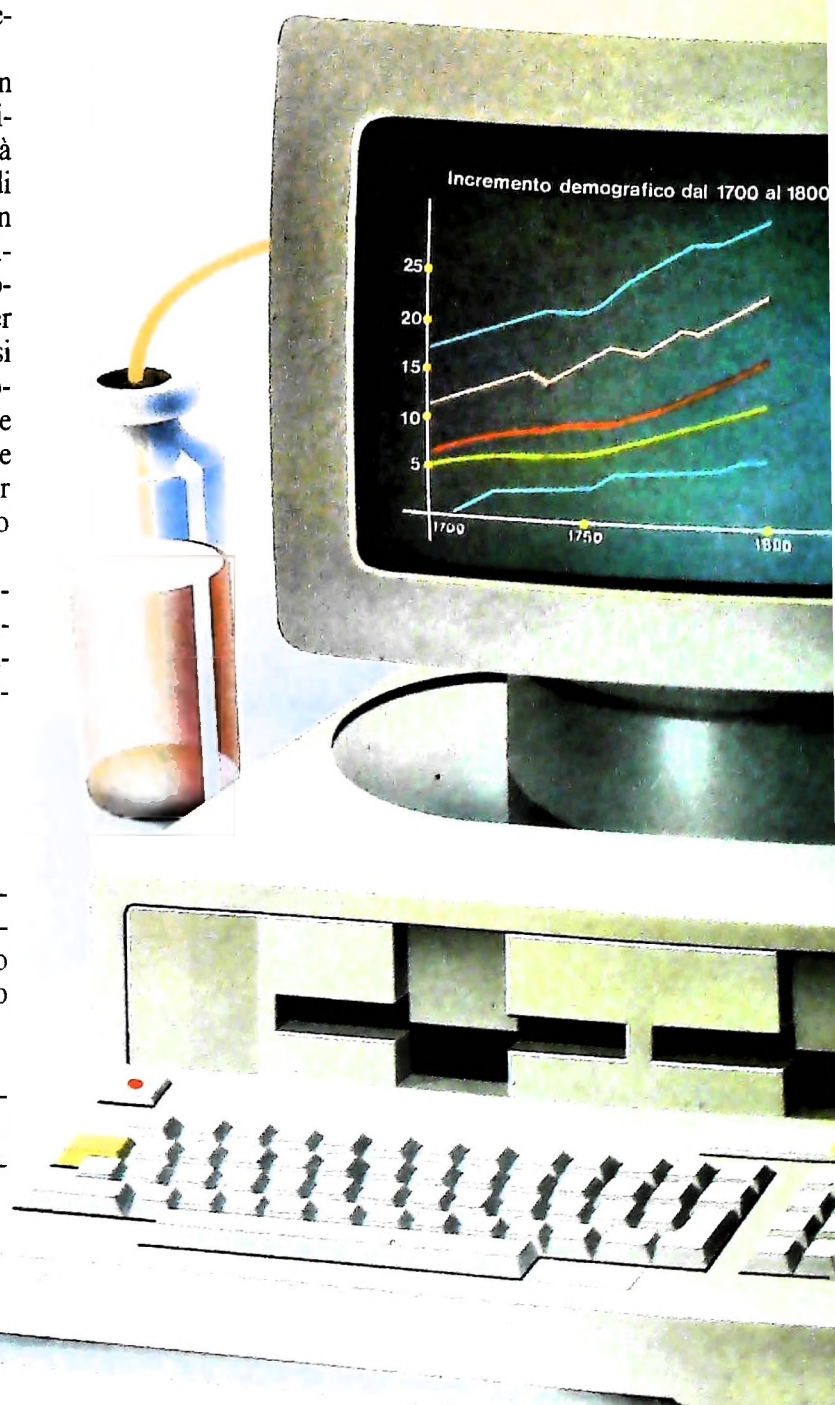
### L'uso di modelli matematici

È sempre opportuno che l'introduzione di modelli matematici si fondi sull'osservazione. Poniamo di voler studiare la crescita di una popolazione di batteri in presenza di nutrimento illimitato. Si può partire da conteggi in funzione del tempo nel classico brodo di fieno, facendo poi due ipotesi:

- 1 — la popolazione raddoppia in un tempo fisso  $\Delta t$
- 2 — la velocità di crescita  $\Delta N / \Delta t$  è proporzionale al numero  $N$  di individui presenti al tempo  $t$ .

Iniziando i conteggi al tempo zero, con le due ipotesi si scri-

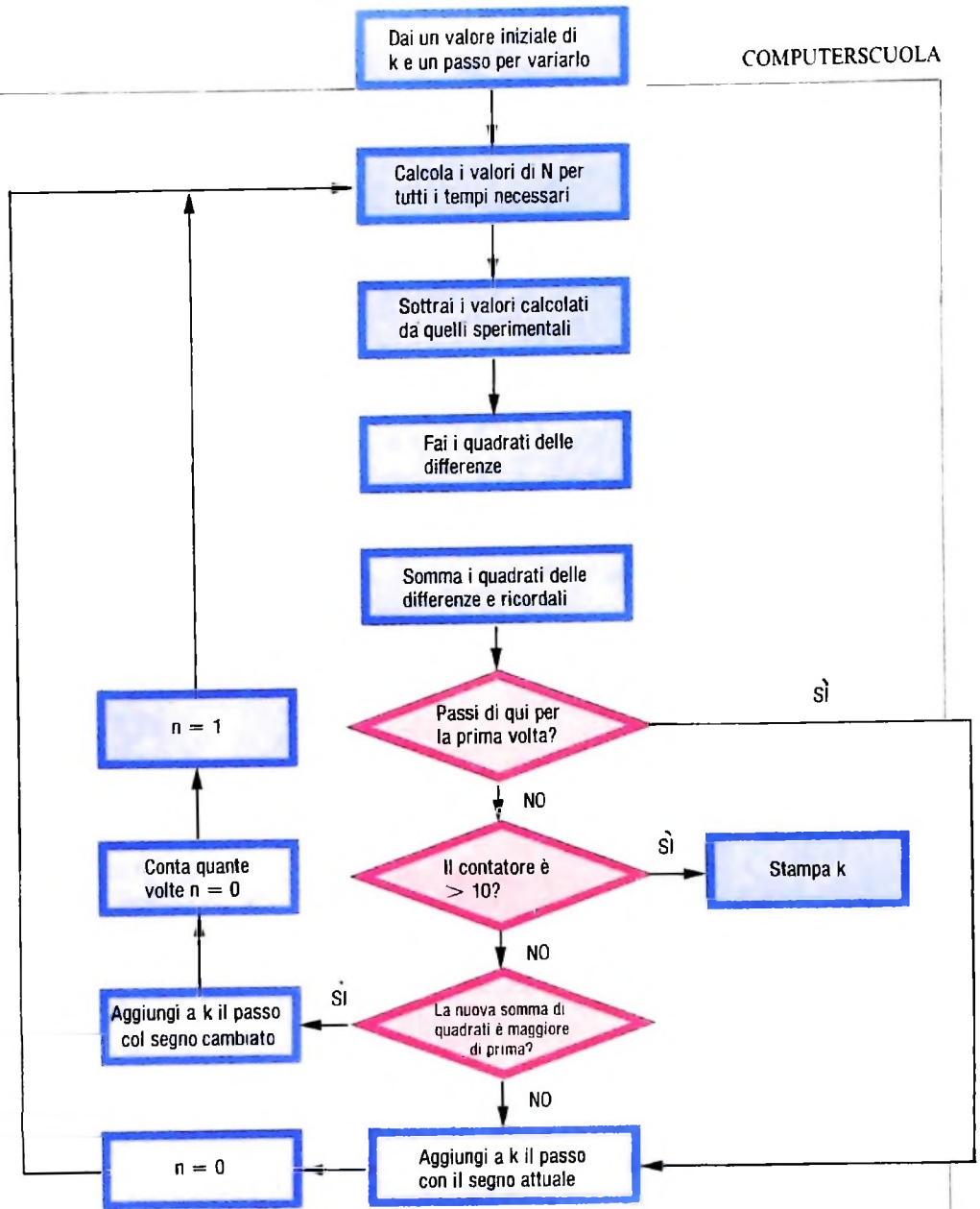
Tempo e popolazione		
tempo	prima ipotesi	seconda ipotesi
$t = 0$	$N_0 = N_0$	$N_0 = N_0$
$t_1 = \Delta t$	$N_1 = 2 N_0$	$N_1 = N_0 + k N_0$
$t_2 = 2\Delta t$	$N_2 = 2 N_1$	$N_2 = N_1 + k N_1$
$t_3 = 3\Delta t$	$N_3 = 2 N_2$	$N_3 = N_2 + k N_2$
.....	.....	.....
$t_n = n\Delta t$	$N_n = 2 N_{n-1}$	$N_n = N_{n-1} + k N_{n-1}$





A sinistra in alto, tabella con due ipotesi per interpretare l'andamento nel tempo del numero di individui di una popolazione:  $k$  è un fattore di proporzionalità positivo (natalità), negativo (mortalità) o somma algebrica dei due.

A destra, diagramma di flusso per un programma di ottimizzazione del valore di  $k$  della seconda ipotesi della tabella. Il controllo sul numero di volte che  $n$  cambia valore serve a impedire che il programma entri in loop oscillando intorno al valore ottimale di  $k$ . La precisione con cui si determina quest'ultimo non è migliore del passo. Sullo schermo del computer in basso a sinistra, grafici dell'andamento demografico dal 1700 al 1800. Qui sotto, rappresentazione di molecole al computer.



vono due programmi basati sul metodo ricorsivo schematizzato nella tabella Tempo e popolazione. Si può anche tracciare sullo schermo l'andamento di  $N$  in funzione del tempo, ponendosi poi varie domande, fra cui:

- 1 — per che valore di  $k$  i due programmi coincidono?
- 2 — esiste una formula generale che esprima i termini che compaiono nella figura?
- 3 — cosa succede se  $k$  diventa negativo e cosa significa questo fatto dal punto di vista biologico?

Già si intravede che un medesimo modello matematico può descrivere fenomeni diversi, come la crescita e la decadenza di una popolazione. Si pongono allora altre domande:

- 1 — esistono altri fenomeni descrivibili con il medesimo modello?
- 2 — come si fa a trovare un valore di  $k$  che adatti il modello ai conteggi effettivamente fatti?

Si tratta di due aspetti di un medesimo problema fondamentale: qual è la potenza descrittiva di un modello, ovvero che rapporto esiste fra il modello e la realtà? Dare una risposta alla prima domanda è facile e aiuta a procurare dati per rispondere alla seconda. Per esempio:

- con un fotometro, trovare come diminuisce la luce trasmessa da una soluzione all'aumentare del suo spessore;
- con il medesimo sistema, trovare come decresce nel tempo la concentrazione di un colorante esposto a radiazione ultravioletta;
- seguire la decomposizione di una sostanza che sviluppa un gas inerte (come l'idrossilammina), misurando l'aumento di volume del gas in una provetta rovesciata in acqua;
- simulare il ricambio di un lago, facendo correre l'acqua del rubinetto in una bacinella con acqua colorata e seguendo come varia nel tempo la concentrazione del colorante.

Sono tutte cinetiche esponenziali, presentate di solito come soluzione di equazioni differenziali, mentre si descrivono altrettanto bene con più immediate equazioni alle differenze finite che si prestano bene alla realizzazione di semplici programmi.

Per trovare il valore ottimale di  $k$ , si può impiegare il principio dei minimi quadrati, che è facile da spiegare con qualche esempio, come si vede nelle illustrazioni. Altrimenti ci si limita a disegnare la curva teorica, variando  $k$  fino a quando questa interpola in maniera soddisfacente i dati sperimentali ottenuti dal brodo di fieno o da una ricerca storica.

Tornando al problema delle popolazioni, si può osservare che  $k$  può essere la somma algebrica di un tasso di nascita (positivo) e di una mortalità (negativa), e si verifica facilmente che la popolazione non può essere stabile se i due tassi sono diversi in valore assoluto. L'insegnante di geografia può azzardare previsioni e interpretazioni di dati storici relativi a qualche paese, trovando facilmente i limiti del modello, dovuti al fatto che il tasso di mortalità varia con l'età.

Il modello esponenziale descritto fin qui ha un campo di applicazioni molto vasto. Descrive infatti fenomeni chimici, fisici, biologici, geologici e demografici. Ma anche altri modelli altrettanto semplici possiedono una potenza descrittiva analoga, come quelli esprimibili con sistemi di equazioni lineari.

Poniamo che in uno stagno vivano tre specie di pesci che si nutrono di insetti, larve e plancton. La specie 1 consuma le tre risorse alimentari in ugual quantità, le specie 2 e 3 consumano insetti in misura uguale alla prima specie, larve in misura rispettivamente doppia e tripla e plancton in misura 3 e 5 volte maggiore. Se la quantità delle tre risorse è rispettivamente di 5000, 10.000 e 15.000 unità giornaliere, il numero degli individui delle tre specie di pesci si ha risolvendo il sistema:

$$\begin{aligned}x_1 + x_2 + x_3 &= 5.000 \\x_1 + 2x_2 + 3x_3 &= 10.000 \\x_1 + 3x_2 + 5x_3 &= 15.000\end{aligned}$$

Questo ha infinite soluzioni. Infatti si osserva che il determinante del sistema è nullo e che tutti i determinanti che si ottengono da esso sostituendo a una delle sue colonne la colonna dei termini noti sono pure nulli. Per sostituzione, si ha:

$$x_1 = x_3 = (5000 - x_2)/2$$

Naturalmente, variando i consumi alimentari delle tre specie la matrice può diventare non nulla e quindi si trova una soluzione unica per il sistema. Sulla medesima falsariga è facile inventare altri problemi ecologici, di produzione di prodotti chimici (usando regole stechiometriche) o demografici.

Per esempio: le attività di un erbivoro della savana si possono classificare in riposo, pascolo e spostamento verso nuovi pascoli o per sfuggire ai predatori. Il consumo energetico per tali attività è, rispettivamente, di 50, 75 e 150 kcal/ora, mentre l'energia ottenuta brucando è pari a 275 kcal/ora (guadagno netto 200 kcal/ora): come si può dividere la giornata fra le tre attività per avere un bilancio energetico in pareggio? Anche in questo caso, il problema è indeterminato, essendovi solo due relazioni fra tre incognite:

$$\begin{aligned}-50t_1 - 150t_2 + 200t_3 &= 0 \\t_1 + t_2 + t_3 &= 24\end{aligned}$$

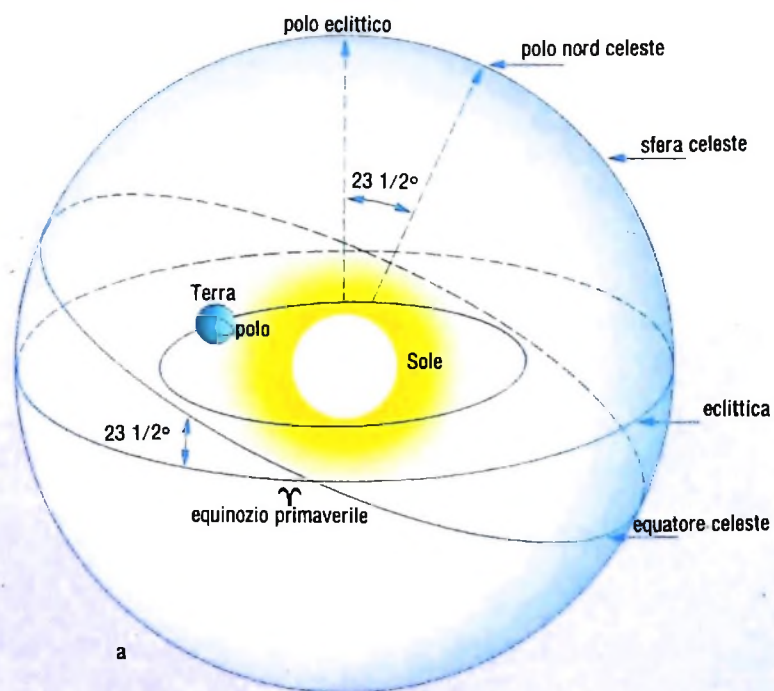
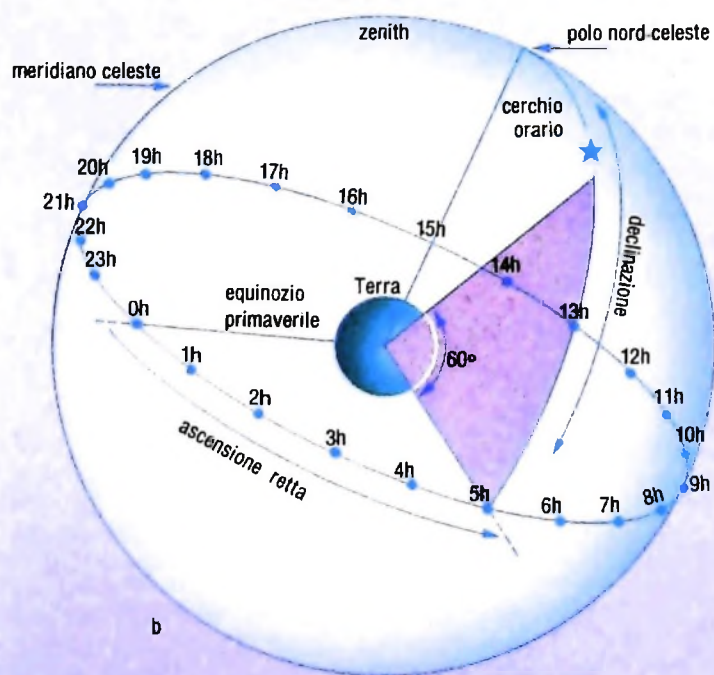
Ma si possono trovare altre relazioni che rendono unica la soluzione, ponendo dei limiti ai tempi dedicati alle tre attività o alle risorse alimentari. La loro ricerca consente di discutere a fondo il problema della sopravvivenza degli animali considerati.

Gli esempi portati sono certamente pochi e limitati a due sole classi di modelli matematici utilizzabili peraltro per descrivere molti fenomeni diversi. Si possono fare anche modelli con sistemi di equazioni alle differenze finite (per descrivere il problema demografico con tassi di natalità e mortalità diversi per classi di età e altri simili) e se poi si possono usare equazioni differenziali il solo limite è posto dalla fantasia. Qualunque cosa si faccia, è chiaro che ciò che si può guadagnare rispetto alla didattica tradizionale è la possibilità di andare oltre le nozioni puramente qualitative, insegnando a inquadrarle in schemi concettuali esprimibili senza grandi difficoltà in modelli matematici che, con l'elaboratore, non è difficile risolvere.



# CONVERSIONE DEL TEMPO

Come trasformare il tempo da solare a siderale e da siderale a solare.



Un osservatore sulla Terra può identificare un oggetto sulla sfera celeste (a) mediante un sistema di coordinate sulla sfera celeste stessa. Tali coordinate (analoghe alla latitudine e longitudine sulla Terra) sono circonferenze che si intersecano formando una griglia e che permettono di determinare l'ascensione retta e la declinazione (b).

Ogni corpo sulla sfera celeste, visto dalla Terra, può essere identificato mediante un sistema di coordinate sulla sfera celeste stessa, circonferenze che si intersecano formando una "griglia", e che danno l'ascensione retta e la declinazione (figura a e b). I poli della sfera celeste, direttamente al di sopra di quelli terrestri, corrispondono ai 90 gradi (nord e sud) di declinazione, mentre a 0 gradi si trova l'equatore celeste (proiezione sulla sfera celeste del piano dell'equatore terrestre). A 0 gradi di ascensione retta si trova il punto di intersezione tra il piano dell'eclittica (piano dell'orbita terrestre) e il piano dell'equatore terrestre da sud a nord: esso prende il nome di equinozio primaverile o primo punto dell'Ariete. In qualsiasi istante il tempo siderale di una località è il tempo trascorso dal più recente passaggio dell'equinozio primaverile dal meridiano. Così come il tempo solare, quello siderale è misurato in ore, minuti, secondi, ma è inferiore rispetto a quello solare di circa 3 minuti e 55, 91 secondi: sul periodo orbitale di 365, 25 giorni, la differenza supera le 24 ore. Per qualsiasi tempo solare il tempo siderale fornisce le ore di

ascensione retta sul meridiano; cioè se il tempo siderale è di 17 ore, le stelle e i pianeti aventi ascensione retta = 17 ore si troveranno sul meridiano a quella particolare ora solare. L'angolo orario di una stella o di un pianeta è la distanza angolare misurata verso occidente lungo l'equatore celeste dal meridiano alla circonferenza oraria della stella (figura b) ed è trovato sottraendo dal tempo solare locale l'ascensione retta della stella. Conoscendo quindi in un particolare istante ambedue le variabili, la posizione di una stella nel cielo è facilmente ottenuta.

Il programma trasforma il tempo solare in tempo siderale, e viceversa, ed è applicabile in qualsiasi punto della Terra; è infatti possibile inserire i parametri locali desiderati (fuso orario, località, longitudine) ogniqualvolta il programma venga utilizzato, oppure è possibile (intervenendo sulla linea 190) memorizzare permanentemente i dati generali prescelti. Nella sua esecuzione il programma richiede l'inserimento della longitudine e del fuso orario (rispetto a Greenwich) della località trattata. Per esempio il fuso orario per l'Italia,

Europa centrale, Camerun è 1; per la Nuova Zelanda 12; per gli Stati Uniti Centrali e il Messico -6; per Gran Bretagna, Spagna, Portogallo 0 e così via (vedi tabella sotto). La longitudine deve essere espressa in frazione di ore e deve essere negativa se la località si trova a est di Greenwich.

## Il programma

Una particolare tecnica si è resa indispensabile per quanto riguarda la visualizzazione dei risultati, una volta che sono stati convertiti in stringhe. Ci si riferisce in particolare alle linee 1140-1141-1142, 1160-1161-1162, da 2081 a 2091.

L'elaborata conversione dei numeri in stringa si è resa necessaria perché la funzione STR\$( ), che appunto converte una variabile numerica in variabile stringa, ritornava una stringa la cui lunghezza era pari al numero di cifre + 1. Questo, in fase di visualizzazione, causava l'allargamento della riga di 10 caratteri, spazi vuoti (è infatti formata da 10 varia-

bili stringa concatenate in 4 gruppi).

L'artificio consiste nell'esaminare la stringa carattere per carattere escludendo dalla stringa finale tutti gli spazi vuoti.

Alla riga 100 viene presentato un altro artificio per rendere più gradevole la visualizzazione sul display:

CHR\$(27) + "p" per mettere in negativo qualsiasi cosa venga visualizzata dopo

CHR\$(27) + "q" per annullare l'effetto precedente.

Particolare attenzione meritano alcuni simboli presenti sul listato, frutto della discrepanza simbolica tra stampante a margherita e tastiera di computer:

il simbolo — è il simbolo > (maggiore di)

il simbolo | è il simbolo < (minore di)

Alle linee 1660, 1670 le scritte h1i e h2i si ottengono così:

premere tasto GRPH e il tasto p contemporaneamente  
premere tasto 1

premere tasto GRPH e il tasto i contemporaneamente  
così anche per la seconda linea, dove invece di 1 bisogna digitare 2.

### Fuso orario con correzione al Tempo Universale

-11	Aleutine, Samoa
-10	Alaska Occidentale, Hawaii
-9	Alaska Orientale
-8	Canada Occidentale, U.S.A. Occidentali
-7	Parte del Canada, Montagne U.S.A., Messico Occidentale
-6	Parte del Canada, U.S.A. Centrali, Messico Orientale
-5	Parte del Canada, U.S.A. Orientali, Perù, Cile, Cuba
-4	Parte del Canada, Brasile Centrale, Paraguay
-3	Brasile Orientale, Groenlandia, Argentina, Uruguay
-2	Azzorre
-1	Madeira
0	Regno Unito, Irlanda, Spagna, Portogallo, Algeria, Marocco
1	Europa centrale, Camerun, Tunisia
2	URSS Occidentale (Mosca), Grecia, Turchia, Israele, Egitto, Giordania, Repubblica Sud-Africana
3	Parte dell'URSS (Gorki), Iraq, Madagascar, Kenia
4	Parte dell'URSS (Sverdlovsk), Iran
5	Parte dell'URSS (Omsk)
5.30	India Ceylon
6	Parte dell'URSS (Novosibirsk), Cina (Tibet), Thailandia
7	Parte dell'URSS (Irkutsk), Cina Centrale, Vietnam, Laos
8	Parte dell'URSS (Yakutsk)
9	Parte dell'URSS (Komsomolsk), Giappone, Corea
10	Parte dell'URSS (Sryanka), Australia Orientale
11	Parte dell'URSS (Ambarchiki), Australia Occidentale
12	Nuova Zelanda

```

70 REM *****
80 REM * Conversione Tempi LST-LMT *
95 REM * *
90 REM *****
100 DEFDBLA-Z
    :F1=0
    :WZ$=CHR$(27)+"p"
    :WW$=CHR$(27)+"q"
147 CLS

```

```

190 ZN=1
    :L$="MILANO"
    :LO=-.6127
200 PRINTWZ$;"Condizioni per:";WW$;" ";L$
210 PRINTWZ$;" fuso orario: ";WW$;ZN
220 PRINTWZ$;" longitudine: ";WW$;LO
230 PRINT
240 PRINTWZ$;"vuoi effettuare variazioni? ";WW$
245 AS=INKEY$

```



```

:IFA$=""THEN245
250 IFA$="N"THEN330
260 IFA$| "S"THENPRINT"risposta non corretta"
:GOTO240
270 CLS
280 PRINTWZ$;LOCALITA'
: ";ww$:INPUTL$
290 PRINTWZ$;"fuso orario:";ww$
:INPUTZN$
300 ZN=VAL(ZN$)
310 PRINTWZ$;"LONGITUDINE:";ww$
:INPUTLO$
320 LO=VAL(LO$)
330 CLS
340 Z=ZN
350 IFF1=1THEN820
360 A$=""
520 CLS
530 PRINT" ESECUZIONE PER CALCOLO DEL: "
531 PRINT$80," Tempo Solare Medio Locale ... LMT "
532 PRINT$160,"Tempo Siderale Locale...LST:"
:INPUTT$
540 IFT$="LMT"THEN570
550 IFT$="LST"THEN570
570 PRINT" ";WZ$;"INSERIMENTO
DATA:";ww$
580 INPUT"ANNO (AAAA):";YD$
:Y=VAL(YD$)
650 INPUT"MESE (MM):";MD$
:M=VAL(MD$)
670 INPUT"GIORNO (GG):";DD$
:D=VAL(DD$)
710 PRINT"vuoi vedere le condizioni locali?"
715 A$=INKEY$
:IFA$=""THEN715
720 IFA$="S"THEN760
740 IFA$="N"THEN800
760 PRINTWZ$;"LOCALITA' .....";ww$;L$
770 PRINTWZ$;"fuso orario .....";ww$;ZN
780 PRINTWZ$;"LONGITUDINE .....";ww$;LO
800 PRINT"vuoi cambiare?"
805 A5$=INKEY$
:IFA5$=""THEN805
810 IFA5$="S"THEN F1=1
:GOTO270
820 F1=0
830 GOSUB1650
840 IFT$="LST"THENU$="LMT"
:GOTO860
850 U$="LST"
860 GOSUB 1900
870 PRINTWZ$;"numero dei calcoli da eseguire:";ww$
:INPUTNO
880 PRINTWZ$;"intervallo di tempo considerato:";ww$
:INPUTIN
890 CLS
910 EP=722895
:GOSUB1950
:DE=DG
:ND=DE-EP
930 PRINTTAB(8);"CALCOLO RICHIESTO PER ";T$
935 PRINT" "
936 LINE(4,12)-(235,52),1,B
940 TC=.065753
950 PRINT$84,"DATA : ";Y;" ";M;" ";D
960 PRINT$124,"LUOGO: ";L$;" ";WZ$;" fuso ";ww$;" ";
ZN-LGC/15
970 PRINT$164,"LONG.: ";LO

```

```

980 PRINT$204,"INTERVALLO DI TEMPO: ";IN;" ore"
985 FORI=3000TO1STEP-1
:NEXT
987 CLS
990 PRINT "=====
1000 PRINT" ";U$;" ";T$
1010 PRINT" ORA DEC HR/MI/SE ORA DEC HR/MI/SE"
1020 PRINT"=====
1030 FORK=1TONO
1040 H1=INT(T1)
:M1=INT((T1-H1)*100)
1050 S1=INT(((T1-H1)*100)-M1)*100
1060 GOSUB1250
1070 IFINT(T2)_24THENT2=T2-24
:GOTO1080
1080 H2=INT(T2)
:M2=INT((T2-H2)*100)
1090 S2=INT(((T2-H2)*100)-M2)*100
1100 H3=INT(T2)
:M3=(T2-INT(T2))*60
:H3$=STR$(H3)
1110 S3=(M3-INT(M3))*60
:S3=INT(S3)
:S3$=STR$(S3)
1120 M3=INT(M3)
:M3$=STR$(M3)
:GOSUB2050
1130 H1$=STR$(H1)
:M1$=STR$(M1)
:S1$=STR$(S1)
:P$="."
1140 HA$=H1$+P$+M1$+S1$
1141 D8$=HA$
:HA$=""
:FORI=1TOLEN(D8$)
:D9$=MID$(D8$,I,1)
:IFD9$|_" "THENHA$=HA$+D9$
1142 NEXT
1150 H2$=STR$(H2)
:M2$=STR$(M2)
:S2$=STR$(S2)
1160 HB$=H2$+P$+M2$+S2$
1161 E1$=HB$
:HB$=""
:FORI=1TOLEN(E1$)
:E2$=MID$(E1$,I,1)
:IFE2$|_" "THENHB$=HB$+E2$
1162 NEXT
1163 R8$=RR$+"/"+IM$+"/"+ES$
1165 R9$=H3$+"/"+M3$+"/"+S3$
1170 PRINTK;TAB(2);HA$;TAB(13);R8$;
1180 PRINTTAB(22);HB$;TAB(31);R9$
1190 T1=T1+IN
1200 IFT1_24THENT1=T1-24
:ND=ND+1
:GOTO1200
1210 NEXTK
1220 GOTO1450
1250 GC=11.927485
1270 TC=.065711
1280 IFT$="LMT"THEN1380
1300 T2=TC*ND+GC+(((ZN+T1)/24)*TC)+T1
1320 T2=T2+(.0656667*LG)
1330 IFT2_24THENT2=T2-24
:GOTO1410
1340 IFT2|-24THENT2=T2+24
:GOTO1340
1350 IFT2|OTHENT2=T2+24

```



```

1360 RETURN
1380 T2=T1-(TC*ND+GC)-(ZN/24*TC)
1400 T2=T2-(.0656667*LG)
1410 IFT2|OTHEHT2=T2+24
      :GOTO1410
1420 IFT2_24THEHT2=T2-24
      :GOTO1420
1430 T2=T2-(T2/24*TC)
1440 RETURN
1450 PRINT"=====
1459 A$=""
1460 PRINT"ULTERIORI CALCOLI?:"
1461 A$=INKEY$
      :IFA$=""THEN1461
1480 IFA$="N"THEN1630
1490 IFA$| "S"THEN 1460
1500 H1$=""
      :M1$=""
      :S1$=""
      :HA$=""
      :H2$=""
      :M2$=""
      :S2$=""
      :HB$=""
      :T1=0
      :T2=0
1520 CLS
1525 A$=""
1530 PRINT"DESIDERI ULTERIORI CALCOLI DI: ";T$
1540 A$=INKEY$
      :IFA$=""THEN1540
1550 IFA$="N"ANDT$="LST"THEHT$="LMT"
      :GOTO1580
1560 IFA$="N"ANDT$="LMT"THEHT$="LST"
      :GOTO1580
1570 IFA$| "S"THENPRINT"risposta non conforme"
      :GOTO1525
1580 CLS
1590 A$=""
1591 PRINT"VUOI CAMBIARE LA DATA?:"
1592 A$=INKEY$
      :IFA$=""THEN1592
1600 IFA$="N"THENZN=Z
      :GOTO710
1610 IFA$| "S"THENPRINT"risposta non conforme"
      :GOTO1590
1620 ZN=Z
      :GOTO570
1630 CLS
1640 END
1650 CLS
1655 PRINTWZ$;" DESIDERI INSERIRE IL TEMPO";WW$
1660 PRINT$80," h1i in frazione di ore"
1670 PRINT$160," h2i in ore,min,sec"
1691 PT$=INKEY$
      :IFPT$=""THEN1691
1700 PT=VAL(PT$)
1710 IFPT=1THEN 1750
1720 IFPT=2THEN 1820
1730 GOTO1650
1740 IFT$="LMT"THEHT$="LST"
1750 PRINT"Insersch il tempo di partenza"
1760 PRINT" in frazione di ore: "
1780 INPUTT1
1790 IFT1_23.9999THENPRINT"risposta non conforme"
      :GOTO1780
1810 GOTO1890
1820 PRINT" Insersch il tempo di partenza"

```

```

1821 PRINT"      in ore,minuti,secondi"
1850 INPUTHR,MI,SE
1870 IFHR_23 ORMI_59ORSE_59.99THENPRINT"inserimento
      non conforme"
      :GOTO1850
1880 T1=HR+MI/60+SE/3600
1890 RETURN
1900 LG=(ZN*15)-LO
1910 IF LG|OTHEHT1930
1920 ZN=ZN+ABS(LG/15)
      :GOTO1940
1930 ZN=ZN+LG/15
1940 RETURN
1950 '
1960 IFM_3THEN2020
1980 DG=365*Y*D
1990 DG=DG+((M-1)*31)+INT((Y-1)/4)-INT((.75)*INT
      ((Y-1)/100+1))
2000 RETURN
2020 DG=365*Y+D+((M-1)*31)-INT(M*.4+2.3)
2030 DG=DG+INT(Y/4)-INT((.75)*INT((Y/100)+1))
2040 RETURN
2050 '
2060 RR=INT(T1)
      :RR$=STR$(RR)
2061 E3$=RR$
      :RR$=""
      :FORI=1TOLEN(E3$)
      :E4$=MID$(E3$,I,1)
      :IFE4$| " " THENRR$=RR$+E4$
2062 NEXT
2070 IM=(T1-INT(T1))*60
      :ES=(IM-INT(IM))*60
2080 ES=INT(ES)
      :IM=INT(IM)
2081 ES$=STR$(ES)
      :IM$=STR$(IM)
2082 E5$=ES$
      :ES$=""
      :FORI=1TOLEN(E5$)
      :E6$=MID$(E5$,I,1)
      :IFE6$| " " THENES$=ES$+E6$
2083 NEXT
2084 E7$=IM$
      :IM$=""
      :FORI=1TOLEN(E7$)
      :E8$=MID$(E7$,I,1)
      :IFE8$| " " THENIM$=IM$+E8$
2085 NEXT
2086 Q1$=H3$
      :H3$=""
      :FORI=1TOLEN(Q1$)
      :Q2$=MID$(Q1$,I,1)
      :IFQ2$| " " THENH3$=H3$+Q2$
2087 NEXT
2088 Q3$=S3$
      :S3$=""
      :FORI=1TOLEN(Q3$)
      :Q4$=MID$(Q3$,I,1)
      :IFQ4$| " " THENS3$=S3$+Q4$
2089 NEXT
2090 Q5$=M3$
      :M3$=""
      :FORI=1TOLEN(Q5$)
      :Q6$=MID$(Q5$,I,1)
      :IFQ6$| " " THENM3$=M3$+Q6$
2091 NEXT
2095 RETURN

```



## Lezione 19

**Ancora sulle strutture di controllo: la struttura case**

Abbiamo visto finora una sola struttura di controllo per la selezione e cioè la struttura IF...THEN...ELSE. Tale struttura permette di evidenziare due cammini di programma a fronte della valutazione di una condizione, che può fornire due risultati: vero o falso. Spesso, però, la condizione da valutare non ha solo due possibili valori, bensì N che richiedono quindi N differenti trattamenti. Un esempio di situazioni di questo tipo è la descrizione del comportamento da tenere davanti a un semaforo, che possiamo descrivere, usando una struttura di controllo Pascal, nel modo seguente:

```

CASE SEMAFORO OF
  VERDE: procedere;
  ROSSO: arrestarsi;
  GIALLO: procedere con cautela
END

```

dove SEMAFORO è una variabile di tipo COLORE.

Il suddetto costrutto prevede tre differenti situazioni (il semaforo può essere verde, rosso o giallo) per ciascuna delle quali è indicato il comportamento da tenere. Un altro esempio di struttura CASE è il seguente:

```

CASE MESE OF
  11,4,6,9: GIORNI:=30;
  2: GIORNI:=28;
  ELSE GIORNI:=31
END

```

che può essere facilmente riconosciuto come la nota filastrocca: “trenta giorni ha novembre con april, giugno e settembre, di ventotto ce n'è uno, tutti gli altri ne han trentuno”.

In questa forma la struttura presenta due caratteristiche aggiuntive: la possibilità di indicare un cammino d'esecuzione per tutti i valori diversi da quelli indicati (clausola ELSE) e la possibilità di elencare N valori differenti da trattare tutti nello stesso modo.

Solo i più moderni linguaggi di programmazione mettono a disposizione questa struttura di controllo: per tutti gli altri si è costretti a ricorrere a una successione di strutture IF innestate. Così l'algoritmo che calcola i giorni del mese risulterà il seguente:

```

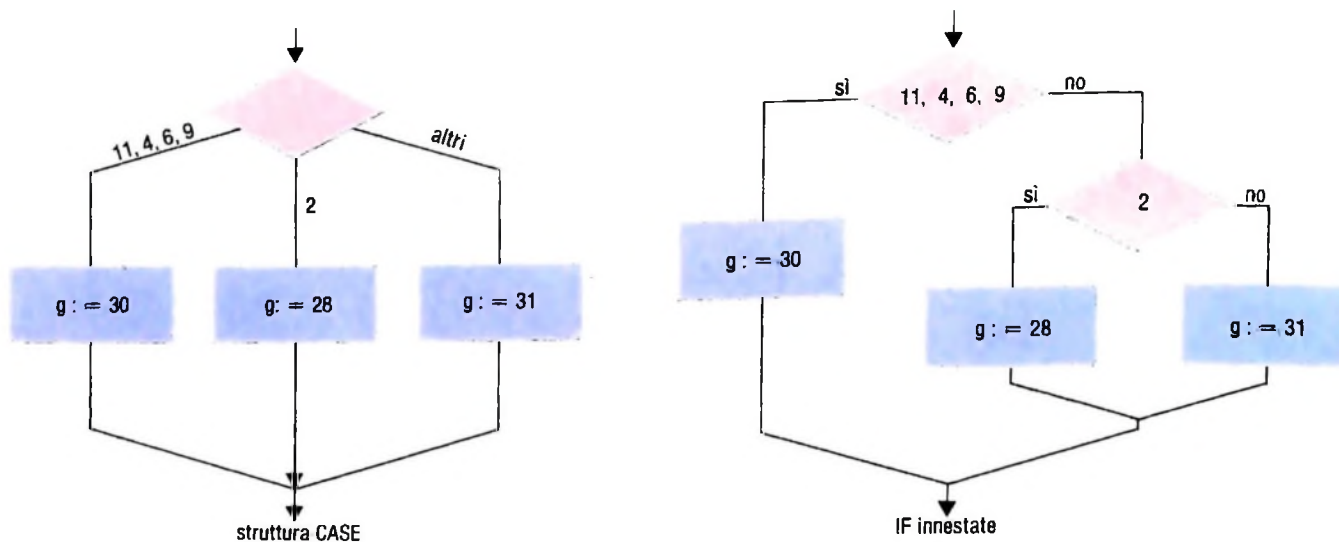
IF MESE=11 OR MESE=4 OR MESE=6 OR MESE=9 THEN
  GIORNI:=30
ELSE
  IF MESE=2 THEN
    GIORNI:=28
  ELSE
    GIORNI:=31

```

che evidenzia però una distinzione in livelli gerarchici in realtà estranea al proble-



ma. Se infatti schematizziamo le due strutture otterremo i seguenti risultati rispettivamente per la CASE e la IF innestate:



dove si nota come la struttura CASE preveda un solo controllo sul valore di una condizione a tre valori, mentre l'algoritmo espresso con IF innestate prevede un primo controllo che verifica se la variabile assume i valori 11,4,6,9 e in caso contrario un secondo controllo che verifica il valore 2.

### La struttura case in BASIC

Il linguaggio BASIC non mette a disposizione la struttura CASE, costringendoci così a ricorrere alle IF innestate. Per alcune classi di problemi è però utilizzabile l'istruzione ON...GOTO ogniqualvolta l'espressione da verificare prevede valori numerici a partire da 1.

Il programma BASIC per il calcolo dei giorni di un mese risulterà quindi:

```

10 INPUT "Numero mese" : M
15 ' Case M of
20 ON M GOTO 30,40,30,50,30,50,30,30,50,30,50,30
30 ' 1,3,5,7,8,10,12 : G:=31
32 LET G=31
35 GOTO 60
40 ' 2 : G:=28
42 LET G=28
45 GOTO 60
50 ' Else
52 LET G=30
60 ' Fine case

```

Si osservi come siamo stati costretti a ripetere più volte la stessa etichetta, poiché ci sono 3 vie a fronte di 12 casi; in una simile situazione è forse più comoda (anche se meno coerente con la natura del problema) la struttura IF innestata:

```

10 INPUT "Numero mese" : M
15 ' If M=11 or M=4 or M=6 or M=9 then

```

```

20 IF M<>11 AND M<>4 AND M<>6 AND M<>9 GOTO 50
30 LET G=30
40 GOTO 100
50 ' Else
52 ' If M=2 then
55 IF M<>2 GOTO 80
60 LET G=28
70 GOTO 90
80 ' Else
85 LET G=31
90 ' Endif
100 'Endif

```

Si noti che il valore che determina lo smistamento del controllo può essere un'espressione come:

```
10 ON A+(5,3-D)*SQR(E) GOTO 20,30,40
```

### Istruzioni GOSUB

#### SINTASSI



#### SEMANTICA

L'istruzione trasferisce il controllo all'istruzione con numero di linea che segue "GOSUB", memorizzando in un'opportuna zona di memoria interna il numero di linea dell'istruzione che segue, per permettere il ritorno alla fine dell'esecuzione del sottoprogramma.

### Istruzione RETURN

#### SINTASSI



#### SEMANTICA

L'istruzione conclude un sottoprogramma e restituisce il controllo all'istruzione successiva all'ultimo GOSUB eseguito.



L'espressione viene valutata e quindi TRONCATA (cioè se il risultato è 2.7, si sceglierà la seconda via).

Esiste inoltre l'analoga istruzione ON...GOSUB che consente di richiamare il sottoprogramma indicato a fronte del valore dell'espressione indicata. Abbiamo già visto esempi d'uso della ON...GOTO quali la gestione delle funzioni offerte da un menù, che espressa con la struttura CASE risulterebbe così:

#### CASE FUNZIONE OF

- 1: calcolaareaquadrato;
- 2: calcolaarearettangolo;
- 3: calcolaareatrapezio;
- 4: calcolaareapentagono;
- ELSE: calcolaareacerchio

END

dove per ogni funzione del menù viene richiamato un corrispondente sottoprogramma che esegue la funzione richiesta.

In BASIC la parte di programma che si occupa della selezione di un comando a fronte di un menù, riscritta con la ON...GOSUB, risulta così:

```
10 INPUT "Funzione richiesta":F
20 ON F GOSUB 100,200,300,400,500
30 GOTO 1000 ' Fine programma
100 ' Calcolo area quadrato
190 RETURN
200 ' Calcolo area rettangolo
290 RETURN
300 ' Calcolo area trapezio
390 RETURN
400 ' Calcolo area pentagono
490 RETURN
500 ' Calcolo area cerchio
590 RETURN
1000 ' Fine programma
```

che richiama il sottoprogramma della linea 100 se F vale 1, il sottoprogramma della linea 200 se F vale 2 e così via.

### Cosa abbiamo imparato

In questa lezione abbiamo visto:

- la struttura di controllo CASE secondo il modello PASCAL;
- la realizzazione di tale struttura in BASIC;
- l'istruzione BASIC ON...GOSUB;
- la sintassi completa delle istruzioni GOSUB e RETURN.

# LE FUNZIONI MUSICALI

Esaminiamo ora l'applicazione musicale più adatta per i piccoli calcolatori: l'elaborazione dei testi musicali.

I processi e le funzioni musicali possono essere ben modellati mediante gli strumenti descrittivi messi a disposizione dai linguaggi di programmazione. In questo e in un prossimo capitolo vedremo come realizzare le funzioni musicali più comuni, esemplificando mediante il BASIC dell'M10.

Descriviamo le funzioni musicali nella forma  $b = f(a)$  in cui  $b$  e  $a$  sono sequenze di entità musicali (note, parametri acustici, temi melodici, accordi ecc.) ed  $f$  è la funzione che, applicata alla sequenza  $a$ , produce la sequenza  $b$ ; possiamo definire funzioni che:

— producono sequenze di entità di un certo tipo in dipendenza da sequenze di entità del medesimo tipo (ad esempio, rovesciamenti di sequenze di durate);

— producono sequenze di entità di un certo tipo in dipendenza da sequenze di entità di più di un tipo (ad esempio, il riconoscimento di una tonalità);

— producono sequenze che mantengono l'ordine delle sequenze di partenza (cioè il primo elemento delle sequenze prodotte dipende dal primo delle sequenze di partenza, il secondo dal secondo e così via; ad esempio, la trasposizione di una sequenza di altezze);

— producono sequenze di dimensione diversa dalla dimensione delle sequenze da cui dipendono (ad esempio, funzioni di selezione o di generazione di sequenze); e così via, con elaborazioni di sequenze di qualsiasi tipo.

Possiamo classificare le funzioni musicali in una varietà di modi infinita; per comprendere meglio facciamo riferimento alla più usuale prassi compositiva.

Poniamo, ad esempio, di operare su sequenze di parametri di altezza o di durata, che memorizziamo nella forma di vettori; i vettori  $Ax$  contengono sequenze di altezze; i vettori  $Dy$  contengono sequenze di durate. I valori delle altezze sono codificati numerando i tasti di una tastiera di pianoforte: 1 è il suono più basso, 2 il suono del tasto successivo verso destra cioè un semitono sopra, e così via. Le durate sono codificate con valori proporzionali alla durata fisica del suono.

È bene ricordare che non dobbiamo uscire dalla tastiera! Quindi, dobbiamo considerare tutte le funzioni come definite su un campo di valori finito e mediante operazioni modulari; cioè, ad esempio, se continuiamo a trasporre verso l'acuto anche di un solo semitono arriveremo presto alla fine della tastiera: per definire correttamente le funzioni musicali dovremo considerare questo aspetto. Se la tastiera fosse di cinque ottave avremmo 60 tasti e i valori di altezze dovrebbero essere compresi tra 1 e 60; se innalzassimo l'altezza 58 di 12 semitoni dovremmo dare come risultato l'altezza 10

(58 + 12 - 60) o viceversa se abbassassimo di 10 semitoni l'altezza 4 dovremmo dare come risultato l'altezza 54 (4 - 10 + 60).

## Trasposizione

La trasposizione delle altezze della sequenza  $A1 = A1(0), \dots, A1(n)$  di  $i$  semitoni verso l'acuto produce  $A2$  mediante l'operazione:

$$A2 = T(i, A1) = A1(0) + i, \dots, A1(n) + i$$

Analogamente, la trasposizione verso il grave di  $i$  semitoni:

$$A2 = T(-i, A1) = A1(0) - i, \dots, A1(n) - i$$

Notiamo che, se avessimo codificato le altezze in una relazione lineare (in proporzione diretta o inversa) con la frequenza, come facciamo con l'istruzione SOUND, la stessa operazione sarebbe stata:

$$A2 = T(i, A1) = A1(0) / 1.059463^i, \dots, A1(n) / 1.059463^i$$

verso l'acuto, e verso il grave:

$$A2 = T(-i, A1) = A1(0) * 1.059463^i, \dots, A1(n) * 1.059463^i$$

## Inversione speculare

L'inversione speculare degli intervalli consiste nel rovesciamento degli intervalli "a specchio": un intervallo verso l'acuto diventa verso il grave e viceversa; questa funzione mantiene la dimensione delle sequenze (cioè  $A2$  è costituito dallo stesso numero di elementi di  $A1$ ) e mantiene l'ordine, nel senso che  $A2(k)$  è funzione di  $A1(k)$ ,  $A2(k+1)$  è funzione di  $A1(k+1)$  e così via.

Descriviamo formalmente, fissato il grado dell'inversione (cioè rispetto a quale "specchio" effettuare l'operazione di inversione), l'inversione speculare come l'operazione seguente:

$$A2 = I(i, A1) = i - (A1(0) - i), \dots, i - (A1(n) - i)$$

ovvero

$$A2 = I(i, A1) = 2 * i - A1(0), \dots, 2 * i - A1(n)$$



## Retrogradazione

Questa funzione consiste nel rovesciamento temporale di una sequenza (tipicamente di altezze); si tratta quindi di una funzione che mantiene la dimensione, ma non l'ordine delle sequenze; formalmente:

$$A2 = R(A1) = A1(n), A1(n-1), \dots, A1(1), A1(0)$$

Nell'illustrazione (pagina 316) è mostrato un esempio di serie musicale tratto dalle *Variazioni per orchestra*, op. 31 di A. Schönberg: alla serie sono applicate le funzioni canoniche della dodecafonia: inversione speculare, retrogradazione, retrogradazione dell'inversione speculare (ovvero inversione speculare della retrogradazione).

## Combinazione di funzioni

Come si vede nell'illustrazione, la letteratura musicale suggerisce l'applicazione combinata (composta) di più funzioni musicali ad una certa sequenza. La composizione più frequente è appunto la retrogradazione dell'inversione speculare, ma naturalmente possiamo realizzare composizioni di funzioni musicali a piacere. Le più comuni sono: retrogradazione composta con inversione speculare:

$$A2 = I(i, R(A1)) = R(I(i, A1)) = 2 * i - A1(n), \dots, 2 * i - A1(0)$$

retrogradazione composta con trasposizione:

$$A2 = T(i, R(A1)) = R(T(i, A1)) = A1(n) + i, \dots, A1(0) + i$$

Notiamo che non sempre l'ordine di applicazione delle funzioni porta allo stesso risultato; quando questo accade (come negli esempi finora visti) si dice che le funzioni sono commutative. Più in generale, invece, dovremo porre attenzione all'ordine con cui applichiamo determinate funzioni; ad esempio, la concatenazione di due sequenze retrogradate è differente dalla retrogradazione della sequenza concatenazione delle due originali, cioè  $R(A B)$  è diverso da  $R(A) R(B)$ , infatti  $R(A B) = B(m), \dots, B(0), A(n), \dots, A(0)$ , mentre  $R(A) R(B) = A(n), \dots, A(0), B(m), \dots, B(0)$ .

## Progressione della selezione

Se invece di codificare tutti i tasti di una tastiera musicale avessimo codificato solo i tasti corrispondenti ad una tonalità (ad esempio, i tasti bianchi per il *do maggiore*), in modo che 1 fosse la *tonica* più bassa, 2 la *seconda* più bassa, 3 la *mediante* più bassa ecc., avremmo potuto definire, con un formalismo del tutto analogo a quello della trasposizione, la funzione di progressione (non modulante):

$$A2 = P(i, A1) = A1(0) + i, \dots, A1(n) + i$$

## Birthday in Basic

Se osserviamo lo spartito di *Birthday* possiamo notare alcune caratteristiche strutturali della parte di basso:

- la struttura ritmica si ripete ciclicamente ed è costituita da una sequenza di cinque figure di un ottavo, una di un quarto e due di un ottavo; sono quindi tutte figure di un ottavo tranne la sesta di ogni battuta;
- il frammento melodico della prima battuta è la base strutturale della sequenza di altezze della parte di basso secondo il seguente schema:

A A A A B B A A C C A A

dove A è il frammento melodico della prima battuta, B quello della quinta battuta, C quello della nona battuta;

c) B è lo stesso frammento melodico di A trasposto verso il grave alla tonalità della *sottodominante*;

d) C è lo stesso frammento melodico di A trasposto verso il grave alla tonalità della *dominante*.

Ecco allora che, sulla base di queste considerazioni, possiamo costruire un programma come il seguente:

```
010 CLEAR
020 REM "Birthday"
(Lennon-Mc Cartney)
dall'album bianco dei Beatles
040 dim a(6)
100 REM caricamento altezze
110 a(0)=10544
120 a(1)=a(0)
130 a(2)=8368
140 a(3)=7032
150 a(4)=5918
160 a(5)=6269
170 a(6)=a(3)
800 REM introduzione del tempo
810 input "tempo (da 0 a 60):"tmp
820 y=64/2^tmp
910 REM scelta del numero di
ripetizioni del ritornello
920 input "quante ripetizioni":rip
930 for k=1 to rip
940 gosub 1000
950 next k
960 sound a(0),y#3
970 end
1000 REM struttura del testo
musicale
1010 gosub 2000
1020 gosub 2000
1030 gosub 3000
1040 gosub 2000
1050 gosub 4000
1060 gosub 2000
1070 return
```



```

2000 REM sequenza sulla tonica
2010 for j=0 to 1
2020 for i=0 to 6
2030 if i=5 then sound a(i),y*2 else
se sound a(i),y
2040 next i
2050 next j
2150 return
3000 REM sequenza sulla
sottodominante
3010 for j=0 to 1
3020 for i=0 to 6
3030 x=a(i)*3/2
3040 if i=5 then sound x,y*2 else
sound x,y
3050 next i
3060 next j
3150 return
4000 REM sequenza sulla dominante
4010 for j=0 to 1
4020 for i=0 to 6
4030 x=a(i)*4/3

```

```

4040 if i=5 then sound x,y*2 else
4050 next i
4060 next j
4150 return
sound x,y

```

In questo programma il vettore a contiene il frammento melodico della prima battuta; la struttura del testo musicale è costituita dal richiamo delle subroutine 2000-2150, 3000-3150, 4000-4150 (rispettivamente: A, B, C); le altezze vengono generate scandendo il vettore a in un ciclo FOR; per la struttura A, mediante assegnamento della componente di a; per B, mediante l'assegnamento 3030 in cui la componente a(i) è abbassata di un intervallo di quinta (sottodominante); per C mediante l'assegnamento 4030 in cui la componente a(i) viene abbassata di un intervallo di quarta (dominante); le durate sono sempre di un ottavo tranne per la sesta componente di ogni struttura, in cui l'ottavo viene raddoppiato (2030, 3040, 4040). Questo ciclo è innestato entro un altro ciclo FOR che ripete due volte (A A o B B o C C). Inoltre, abbiamo inserito la possibilità di ripetere per un numero di volte determinato l'intera struttura, sotto il controllo del ciclo FOR 930-950, dipendente dal parametro Rip; in chiusura, abbiamo aggiunto una tonica di durata tre ottavi:







La serie originale su cui sono basate le *Variazioni per orchestra* op. 31 di Arnold Schönberg e le tre serie da essa derivate con i tipici processi dodecafonici della retrogradazione e dell'inversione.

### Selezione

La funzione di selezione caratterizza procedimenti di *analisi* oppure di *sintesi contestuale*; ad esempio, se vogliamo aggiungere ad una certa sequenza di altezze il *centro della sequenza*, definito come l'altezza semisomma della altezza più grave e dell'altezza più acuta, possiamo ottenerlo mediante la seguente funzione:

$$c = C(A1) = (\min(A1(0), \dots, A1(n)) + \max(A1(0), \dots, A1(n))) / 2$$

Più in generale, possiamo ottenere da una certa sequenza una sottosequenza in base a criteri da noi definiti: ad esempio, un elemento ogni due oppure tutti gli elementi che godono di una certa proprietà o altri ancora.

### Generazione

Quando parliamo di funzioni generative intendiamo funzioni che permettono di generare una sequenza a partire da sequenze di dimensione inferiore; su queste funzioni si basano i metodi di tipo linguistico-grammaticale. Possiamo, ad esempio, definire funzioni come le seguenti:

- a)  $f1(a) = a$
- b)  $f2(a) = b a c$
- c)  $f3(x a) = x b a c$
- d)  $f4(a y) = b a c y$
- e)  $f5(a) = b c$

e così via.

A seconda del tipo di funzione che definiamo, otteniamo regole più o meno potenti, più o meno libere, più o meno dipendenti dal contesto eccetera.

### Trasformazioni funzionali in forma generale

Se vogliamo guardare al problema delle funzioni musicali in modo del tutto generale, diciamo allora che una sequenza A2 è ottenibile dalla sequenza A1 mediante la funzione f se vale la relazione:

$$A2 = f(A1)$$

In particolare, se la funzione f mantiene l'ordine degli elementi allora vale la relazione più restrittiva:

$$A2 = f(A1) = f(A1(0), \dots, A1(n)) = f(A1(0)), \dots, f(A1(n))$$

Il discorso che abbiamo impostato ci permette di guardare alla descrizione dei fenomeni musicali come ad un linguaggio in continua evoluzione, in cui definiamo di volta in volta le *operazioni* che permettono di realizzare nuove idee musicali. Ogni volta che definiamo una funzione arricchiamo il patrimonio del nostro *linguaggio musicale*, ne aumentiamo la potenza descrittiva. Dobbiamo porre in evidenza una cosa, infatti: l'impostazione funzionale ci permette non solo di descrivere l'informazione musicale con strumenti descrittivi aderenti alla concezione musicale che l'ha generata, ma risparmiamo anche spazio, cioè compattiamo la descrizione; effetto collaterale non trascurabile quando ci troviamo a dover fare i conti con la memoria di un piccolo elaboratore!

# LE TRASFORMAZIONI DI ROTAZIONE

Insieme alle trasformazioni di scala costituiscono il gruppo delle cosiddette trasformazioni lineari.

Riprendiamo in questo capitolo la trattazione delle trasformazioni geometriche affrontando le "rotazioni", le quali consentono di modificare l'angolazione delle figure che costituiscono l'immagine sullo schermo. Come le trasformazioni di scala e di traslazione, le rotazioni verificano un'importante proprietà: sono *invertibili*.

Questo significa che, data una matrice di trasformazione (di rotazione, di scala o di traslazione), è sempre possibile determinare una seconda matrice che esegue la trasformazione inversa, ossia permette di ritornare alla figura originaria. Non staremo qui a illustrare la teoria delle matrici inverse, che richiederebbe l'introduzione del concetto di "determinante di una matrice", ma ci limiteremo a dare qualche esempio di rotazione e a fornire alcune indicazioni su come costruire la figura trasformata.

## Rotazione

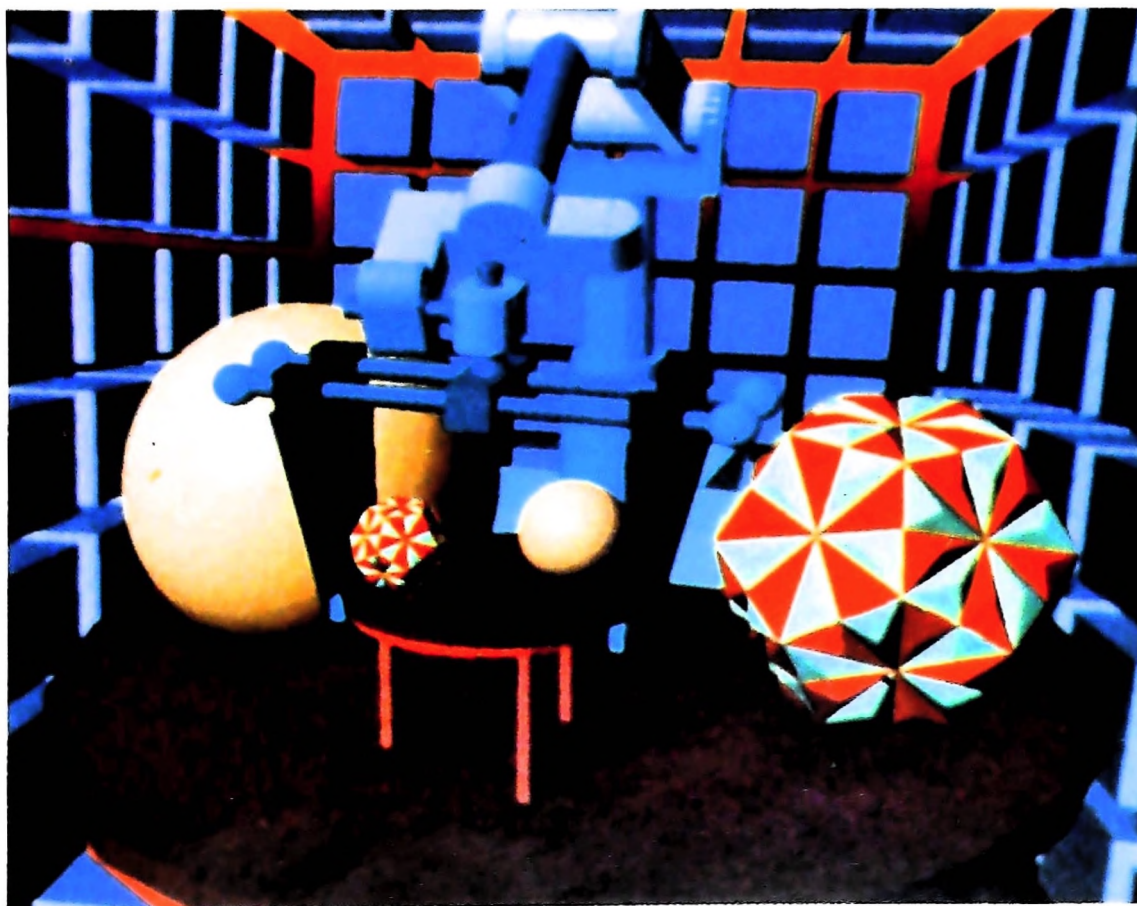
Per ruotare di un angolo  $\alpha$  *in senso orario* un punto  $(x, y)$  intorno all'origine del sistema di coordinate, bisogna fare ricorso alle funzioni seno e coseno che sono già note dalla trigonometria.

Per calcolare le coordinate del punto trasformato in funzione delle coordinate di partenza, valgono le seguenti equazioni:

$$\begin{aligned}x' &= x \cos \alpha + y \sin \alpha; \\y' &= -x \sin \alpha + y \cos \alpha\end{aligned}$$

Per esempio, il triangolo di vertici:

$$(20, 0), (60, 0), (40, 100)$$





ruotato di 45 gradi in senso orario rispetto all'origine, diventa:

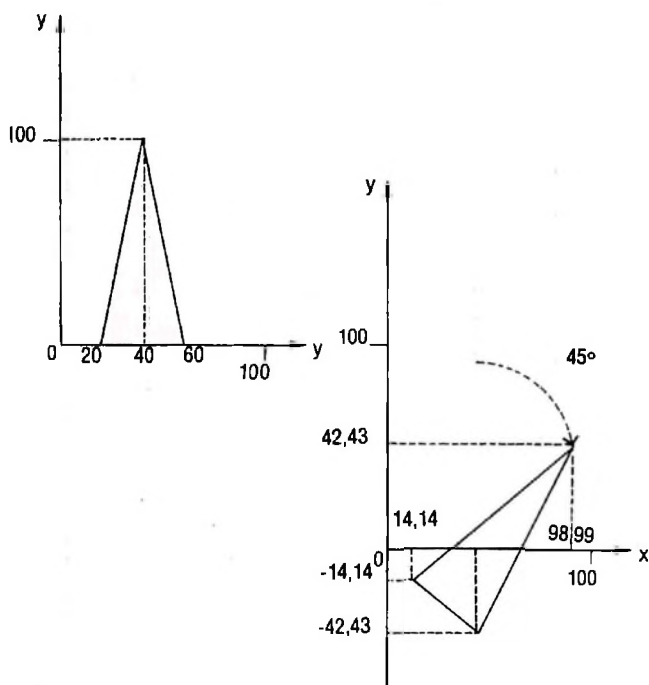
$$(14.14, -14.14), (42.43, -42.43), (98.99, 42.43).$$

Ricordiamo che:  $\sin 45 = \cos 45 = 0.707$

Bisogna qui specificare molto chiaramente che queste equazioni di trasformazione possono essere utilizzate solo per rotazioni attorno all'origine del sistema di riferimento.

La rappresentazione matriciale delle rotazioni in senso orario rispetto all'origine è la seguente:

$$[x' y' 1] = [x y 1] \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



Rotazione di 45° di un triangolo. Nella pagina precedente, tre solidi geometrici vengono traslati e sottoposti a una trasformazione di scala in tre dimensioni.

Sono presenti procedure di clipping (per il tavolo grande), sorgente di luce, prospettiva, eliminazione di superfici nascoste.

Al solito, risolvendo il prodotto righe per colonne si verifica che questa formulazione è equivalente alle due equazioni viste sopra, inoltre anch'essa è valida soltanto per rotazioni attorno all'origine.

Come si fa allora per ruotare un'immagine attorno ad un punto qualsiasi?

Supponiamo per esempio di dover ruotare in senso orario di  $\alpha$  gradi il punto  $(x, y)$  intorno ad un generico punto che indichiamo  $(P_x, P_y)$ . Le trasformazioni di rotazione viste sopra possono essere applicate solo nell'origine, quindi, per prima

cosa, bisognerà traslare i punti in modo tale che  $(P_x, P_y)$  venga a coincidere con l'origine, ossia:

$$[x' y' 1] = [x y 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -P_x & -P_y & 1 \end{bmatrix}$$

A questo punto, è possibile applicare la rotazione:

$$[x'' y'' 1] = [x' y' 1] \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Da ultimo, bisognerà applicare la traslazione contraria alla precedente, per riportare il punto nella posizione corretta:

$$[x''' y''' 1] = [x'' y'' 1] \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ P_x & P_y & 1 \end{bmatrix}$$

Il punto  $(x''', y''')$  risulta essere così il risultato della trasformazione desiderata.

Naturalmente, è possibile concatenare le tre matrici di trasformazione utilizzate senza alterare il risultato finale. Quindi, ogniqualevolta si desidera ruotare una figura intorno a un generico punto, bisognerà avere l'accortezza di traslare l'intera figura fino a far coincidere il punto di rotazione con l'origine, applicare la matrice di rotazione e poi effettuare la traslazione contraria.

### Alcune considerazioni

Prima di passare a vedere qualche applicazione sulle trasformazioni geometriche, vogliamo fare alcune brevi considerazioni.

Il problema delle trasformazioni è, come già si è detto nella precedente lezione, un problema geometrico, pertanto in questo contesto una figura può venire considerata come un insieme di punti nello spazio. Ogni trasformazione applicata a questo insieme di punti produrrà un nuovo insieme di punti, ossia la figura trasformata. Ovviamente l'insieme di punti della figura originaria non è caotico, ma esisteranno varie relazioni che legano i punti fra loro a formare elementi geometrici base, come vertici e lati.

Occorrerà perciò che il tipo di trasformazione definita conservi le relazioni tra i punti e soprattutto che la figura risultante dalla trasformazione conservi le proprietà più generali della figura originale, come per esempio le misure o gli angoli tra i vari segmenti. La conservazione di queste proprietà garantisce che la figura trasformata venga percepita da chi la osserva nello stesso modo, permettendo così il riconoscimento della forma dell'oggetto di cui la figura è una rappresentazione.

Le trasformazioni che abbiamo considerato fino a questo punto possiedono la proprietà di conservare alcune importanti relazioni delle figure su cui vengono applicate. In parti-

colare, la traslazione e la rotazione sono trasformazioni *isometriche*, ovvero conservano le misure associate alla figura; la trasformazione di scala non è evidentemente isometrica, pur conservando gli angoli fra i vari segmenti.

### Un esempio di applicazione

Diamo ora il listato di un programma che realizza le tre trasformazioni geometriche viste finora. La struttura e un commento al programma vengono forniti nel riquadro di questa stessa lezione, alla pagina successiva.

```

5 REM PROGRAMMA SULLE TRASFORMAZIONI
6 REM GEOMETRICHE BIDIMENSIONALI
10 CLS
12 PIGRECA=3.1415
15 INPUT "NUMERO DEI VERTICI";N
20 DIM X(N),Y(N)
30 FOR I=1 TO N
40 PRINT "VERTICE";I":"
50 INPUT X(I),Y(I)
60 NEXT I
70 CLS
80 FOR I=2 TO N
90 X1=X(I-1) : Y1=Y(I-1)
95 X2=X(I) : Y2=Y(I)

```

Elaborazione al calcolatore di un insieme di sfere colpite da una sorgente di luce. L'immagine è stata realizzata utilizzando procedure per simulare la prospettiva, eliminare la luce nascosta e creare l'effetto di riflessione luminosa.

```

98 GOSUB 2000
100 NEXT I
110 X1=X(N) : Y1=Y(N)
112 X2=X(1) : Y2=Y(1)
115 GOSUB 2000
120 INPUT "TRASLA(T),RUOTA(R),SCALA(S),FINE(F)"
;A$
121 GOSUB 1500
122 IF A$="F" THEN 4000
123 INPUT "CANCELLA SCHERMO (S/N)";S$
124 GOSUB 1500
125 IF S$="S" THEN CLS ELSE IF S$="N" THEN
GOSUB 1500
127 IF S$<>"N" AND S$<>"S" THEN 1100
129 IF A$="T" THEN 140 ELSE IF A$="R" THEN 500
ELSE IF A$="S" THEN 1000 ELSE IF
A$="F" THEN 4000
130 PRINT "ERRORE DI INPUT"
132 GOSUB 1500
135 GOTO 120
136 REM
137 REM
138 REM ROUTINE DI TRASLAZIONE
140 INPUT "TX E TY";TX,TY
145 GOSUB 1500
150 FOR I=1 TO N
160 X(I)=X(I)+TX : Y(I)=Y(I)+TY
170 NEXT I
190 GOTO 80
193 REM
194 REM
500 REM ROUTINE DI ROTAZIONE

```





```

510 INPUT "PUNTO DI ROTAZIONE";PX,PY
513 REM VISUALIZZA IL PUNTO DI ROTAZIONE
515 PSET(PX,PY)
520 GOSUB 1500
530 REM TRASLAZIONE NELL'ORIGINE
540 FOR I=1 TO N
550 X(I)=X(I)-PX : Y(I)=Y(I)-PY
560 NEXT I
565 INPUT "ANGOLO";A
567 GOSUB 1500
570 REM TRASFORMAZIONE IN RADIANTI
580 B=A/180
585 C1=COS(B*PIGRECA) : C2=SIN(B*PIGRECA)
610 FOR I=1 TO N
620 Z=X(I)
630 X(I)=Z*C1+Y(I)*C2
640 Y(I)=-Z*C2+Y(I)*C1
650 NEXT I
670 REM TRASLAZIONE CONTRARIA
680 FOR I=1 TO N
690 X(I)=X(I)+PX : Y(I)=Y(I)+PY
700 NEXT I
710 GOTO 80
800 REM
810 REM
1000 REM ROUTINE PER SCALA
1010 INPUT "SX,SY";SX,SY
1015 GOSUB 1500
1020 FOR I=1 TO N
1030 X(I)=X(I)*SX : Y(I)=Y(I)*SY
1040 NEXT I
1050 GOTO 80
1060 REM
1061 REM
1100 PRINT "ERRORE DI INPUT"
1110 GOSUB 1500
1120 GOTO 123
1300 REM
1310 REM
1500 REM SUBROUTINE PER ELIMINARE LE
1510 REM SCRITTE SULLO SCHERMO
1520 PRINT@ 0, "
1530 PRINT@ 0, " ";
1540 RETURN
1900 REM
1910 REM
2000 REM SUBROUTINE CLIPLINE
2010 XS=0 : XD=239 : YB=0 : YA=63
2050 GOSUB 3050 :REM CODIFICA ESTREMI
2100 REM PROCEDURA PRINCIPALE
2110 IF C1$="" AND C2$="" THEN 2700
2120 IF LEFT$(C1$,1)=LEFT$(C2$,1) OR RIGHT$(C1$,1)=RIGHT$(C2$,1) THEN 2710
2130 C$=C1$
2140 IF C$="" THEN C$=C2$
2150 IF LEFT$(C$,1)="S" THEN 2500 ELSE IF LEFT$(C$,1)="D" THEN 2550 ELSE IF RIGHT$(C$,1)="B" THEN 2600 ELSE IF RIGHT$(C$,1)="A" THEN 2650
2160 IF C$=C1$ THEN X1=X : Y1=Y ELSE X2=X : Y2=Y
2170 GOSUB 3050
2180 GOTO 2100
2490 REM CALCOLO DELLE INTERSEZIONI DEL
2491 REM SEGMENTO DI LINEA CONSIDERATO
2492 REM CON I PROLUNGAMENTI DEI LATI
2493 REM DELLO SCHERMO.
2500 Y=Y1+(Y2-Y1)*(XS-X1)/(X2-X1)
2510 X=XS
2520 GOTO 2160
2550 Y=Y1+(Y2-Y1)*(XD-X1)/(X2-X1)
2560 X=XD
2570 GOTO 2160
2600 X=X1+(X2-X1)*(YB-Y1)/(Y2-Y1)
2610 Y=YB
2620 GOTO 2160
2650 X=X1+(X2-X1)*(YA-Y1)/(Y2-Y1)
2660 Y=YA
2670 GOTO 2160
2700 LINE (X1,Y1)-(X2,Y2)
2710 RETURN
3050 REM ASSEGNAMENTO CODICI DEI VERTICI
3055 C1$="" : C2$=""
3060 IF X1<XS THEN C1$="S" ELSE IF X1>XD THEN C1$="D"
3070 IF Y1<YB THEN C1$=C1$+"B" ELSE IF Y1>YA THEN C1$=C1$+"A"
3080 IF X2<XS THEN C2$="S" ELSE IF X2>XD THEN C2$="D"
3090 IF Y2<YB THEN C2$=C2$+"B" ELSE IF Y2>YA THEN C2$=C2$+"A"
3100 RETURN
4000 END

```

### Comento al programma

Il programma elabora soltanto figure geometriche chiuse, delle quali vengono richiesti in input il numero e le coordinate dei vertici in modo interattivo. Va fatto rilevare che il BASIC di M10 consente di definire gli array anche in maniera dinamica: infatti nell'istruzione 20 vengono definiti due array di N elementi, ove N è fornito da tastiera e rappresenta il numero dei vertici che compongono la figura.

Un programma di trasformazioni geometriche deve essere dotato di una subroutine di clipping. Qui abbiamo utilizzato in maniera pressoché integrale quella vista nella relativa lezione. La CLIPLINE viene richiamata nel ciclo di FOR in cui si disegnano i segmenti della figura (dalla 80 alla 100), e mediante le istruzioni 90 e 95 si riescono a conservare anche le coordinate dei vertici che vengono sottoposti al clipping. Le istruzioni 110 e 112 servono per tracciare il segmento che congiunge il vertice n-esimo al primo, segmento che non rientra nel ciclo di FOR precedente.

Il programma è poi strutturato in quattro routine: per la traslazione, per la rotazione, per la scala ed infine per cancellare la prima riga di caratteri, che viene utilizzata per il dialogo interattivo con l'utente.

Una possibilità presente nel programma è quella di conservare la figura originale insieme a quella trasformata (istr. 123).

Molti sono gli sviluppi che si potrebbero apportare: per esempio: costruire una subroutine che consenta di ottenere l'output anche sul plotter, definire

una viewport all'interno della quale riprodurre la figura, concatenare fra loro più trasformazioni. È un buon esercizio per lo studioso cimentarsi nella realizzazione di queste subroutine. Diamo qui qualche suggerimento per uscire su plotter.

Per poter ottenere la doppia possibile uscita su video o su plotter, bisognerà dapprima inserire una richiesta specifica, del tipo:

```
INPUT "Plotter o Video (P/V)"; U$
```

Poi sarà necessario creare una routine di disegno delle linee nella CLIPLINE che esegua la sola 2700 se l'uscita deve essere su video, oppure una opportuna sequenza di istruzioni se l'uscita deve essere su plotter. Dovranno cioè comparire:

```
LPRINT CHR$(18), per la modalità grafica del plotter
```

```
LPRINT DX1, Y1, X2, Y2, per tracciare il segmento
```

più le eventuali richieste di spostamento "a penna su" (MX, Y) o di cambio colore.

Bisognerà inoltre, per uscire su plotter, modificare gli assegnamenti della window di clipping (istr. 2010).

— 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.

Più esclusivo, perché potete usufruire del servizio Voxintesi, attraverso il quale chiedere direttamente al nostro elaboratore il saldo del vostro conto corrente con una semplice telefonata: in qualsiasi ora come in qualsiasi giorno, anche festivo.

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 microplotter 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**