

■ Sino a questo punto, gli esempi appena sviluppati hanno tenuto conto, solo in maniera sottintesa, delle unità di misura presenti nei calcoli.

Nel calcolo dell'area della puntata precedente abbiamo utilizzato numeri puri sottintendendo che trattavasi di numeri interi ed appartenenti tutti alla stessa unità di misura. Non sempre però è possibile, conveniente o comodo eseguire a mano preventivamente tutte le conversioni necessarie, per poi poter inserire in numeri così ottenuti nel nostro Rule Sheet.

Ci viene in aiuto un foglio supplementare, presente nel programma (si ricordi quanto abbiamo detto nella prima puntata), che consente di definire unità di misura e conversioni dalle stesse. In tal modo, utilizzando tale foglio, è possibile applicare formule ed equazioni, senza occuparsi eccessivamente di come esse vengano utilizzate nell'ottica del passaggio delle unità di misura. ■

Il Variable Subsheet (sottofoglio di variabili)

Il Variable Subsheet è un foglio che consente di definire le caratteristiche complete ed univoche di ogni variabile contenuta nell'interno delle formule utilizzate nel Rule Sheet. Per ogni variabile inserita nel Rule Sheet e riconosciuta valida dal programma viene immediatamente creato un sottofoglio di variabile contenente la descrizione dettagliata di ogni variabile inserita nelle formule. Prima di scendere in dettaglio nella descrizione del Variable Subsheet è opportuno, ovviamente, individuare i comandi che ci permettono di accedere e di abbandonare tali sottofogli.

Per lavorare su sottofogli di variabile esistono due comandi principali. Il primo, DIVE permette l'accesso ai sottofogli; il secondo, RETURN, ne permette l'uscita. In figura 1 vediamo una descrizione dettagliata del comando DIVE e del contenuto del sottofoglio variabili che da tale comando viene invocato. La figura 2 mostra le relazioni tra le parti descritte nella figura 1.

La figura 3 mostra le caratteristiche di un sottofoglio variabili. In essa vengono poi descritte le tipologie di ogni Entry, ogni riga cioè, che chiede o dà in output un valore, un commento, o descrive certe peculiari caratteristiche della variabile.

Esaminiamo la figura: vediamo in intestazione il nome della variabile. Seguono otto opzioni diverse, otto righe destinate al commento della variabile utilizzata. La prima riga contiene l'opzione status che evidenzia la tipologia della variabile di lavoro. Le opzioni possono essere 3: [B]lank, [I]nput, ed Output [O], dall'ovvio significato. In pratica, lo status evidenzia il tipo di variabile ed il suo uso nell'ambito della formula. La seconda riga (first guess) è destinata ad accogliere il primo dei valori supposti quando viene utilizzato il metodo iterativo per la soluzione del problema.

La terza riga, associate list, consente un'unica opzione: l'inserimento di una L determina l'associazione di questa variabile, della variabile utilizzata e sempre presente in intestazione, con un metodo di risoluzione attraverso lista.

Input value mostra il valore associato con la variabile quando questa stessa viene utilizzata come variabile di input.

Output value funziona allo stesso modo, ma mostra la soluzione del problema. Display unit, mostra le unità di misura della variabile.

Calculation unit mostra le unità di misura e conversione di calcolo utilizzate dal programma, in cui viene trasformata la variabile stessa per lo sviluppo del calcolo.

Comment può contenere commenti illimitati di qualsiasi genere.

Il foglio unità (Unit Sheet)

Il foglio unità fornisce le tipologie di conversione delle unità di misura definite nelle equazioni. Accedere al foglio unità, come d'altro canto in qualsiasi altro foglio, è cosa semplice attraverso il comando Switch o attraverso il comando Select. La figura 4 mostra le diverse tipologie del comando Select in funzione del foglio desiderato.

Il comando Select come si può vedere dalla figura, consente di scegliere, selezionare, il foglio su cui si intende lavorare (è possibile lavorare anche su subfogli). Select è un comando molto più efficiente di Switch in quanto consente di riunire o saltare immediatamente da foglio a foglio senza particolari preoccupazioni circa i valori che man mano si stanno introducendo.

Ma ritorniamo al foglio unità. Entrati in esso col comando Select ci troviamo di fronte ad una finestra del tipo di figura 5.

Come si vede il foglio contiene 4 colonne. La prima [from] assegna un nome all'unità da convertire; la seconda [to] assegna il nome dell'unità cui è destinata la conversione. La terza colonna [multiply] indica il valore di conversione tra le due variabili; infine la quarta colonna [add offset], comunque non sempre utilizzata, indica un valore da aggiungere alla variabile da convertire perché assuma l'esatto valore del modulo di conversione (si pensi, ad esempio, alla conversione gradi C/F).

L'uso del foglio di unità ci pare talmente ovvio e banale da non necessitare di ulteriori chiarimenti. È possibile comunque definire in esso range di unità di conversione abbastanza elastiche, tali cioè da consentire l'inserimento di mutazione di variabile abbastanza potente.

Riorganizzazione dello schermo

Qualche breve parola circa la riorganizzazione dello schermo. Generalmente abbiamo visto, anche nella organizzazione degli esempi che abbiamo appena fornito, che è possibile e talvolta addirittura necessario ridurre la superficie delle due finestre per poter avere sempre a portata d'occhio e di mano tutto quanto contenuto in un'altra. Esiste un comando, in TK!Solver, che consente di designare nel modo più conveniente lo schermo di lavoro su cui si dovrà operare. Questo comando, che va sotto il nome di window, possiede una serie di opzioni che possiamo vedere in figura 6. Attraverso tale comando è possibile selezionare particolari fogli od abbinare sullo schermo maschere desiderate senza tener conto del default proposto dal programma. La figura mostra in maniera dettagliata la tipologia d'uso del comando window. Nessun problema per l'uso stesso. C'è solo da dire che,

COMANDO DIVE simbolo [>]

Il comando DIVE accede a certe sutilsie a seconda del punto in cui è posizionato il cursore.

foglio d'origine	effetto
variable sheet	mostra il sottofoglio relativo alla variabile su cui è situato il cursore
variable subsheet	scende ancora nella gerarchia, mostrando il sottofoglio LIST
list sheet	mostra il sottofoglio LIST
user function sheet	mostra il relativo sottofoglio
user function subsheet	mostra il sottofoglio LIST
plot sheet	seleziona il relativo sottofoglio LIST
table sheet	mostra il sottofoglio LIST

COMANDO RETURN simbolo [<]

Ritorna al foglio di partenza in cui era stato selezionata l'opzione [>]

Figura 1

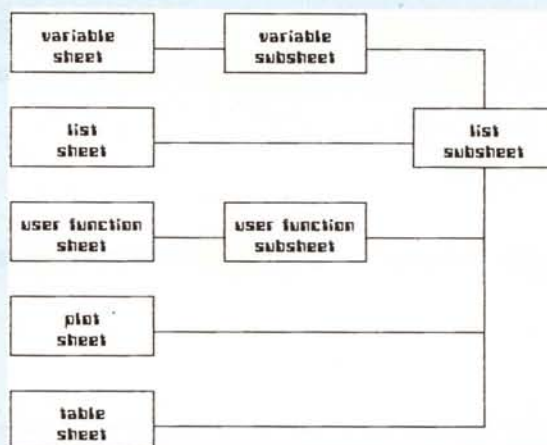


Figura 2

COMANDO SELECT simbolo [=]

Il comando SELECT consente di mostrare e lavorare su uno specifico foglio. Al simbolo [=] va abbinata una nuova lettera, che consente l'opportuna scelta; le opzioni possibili sono:

= U	Variable sheet
= R	Rule sheet
= U	Unit sheet
= G	Global sheet
= L	List sheet
= F	user Function sheet
= P	Plot sheet
= T	Table sheet

Figura 3

TK!Solver © by Software Arts

(s) Status: 114/

Variable:

Status:
 First Guess:
 Associated List:
 Input Value:
 Output Value:
 Display Unit:
 Calculation Unit:
 Comment:

Rule Sheet

5 Rule

Figura 4

mediante la prima delle opzioni mostrate, è possibile assegnare lo schermo tutto ad una finestra. Questo può essere particolarmente utile nel caso di finestre abbastanza complesse, quale, generalmente, è possibile vedere nel Rule Sheet, quando vengono introdotte formule ed equazioni di particolare impegno.

Principi fondamentali di sviluppo di modelli in TK!Solver

Le modalità di sviluppo di un modello in TK!Solver sono abbastanza simili per i diversi problemi che, appunto, utilizzando tale programma, è possibile affrontare. In effetti qualunque problema risolvibile da TK!Solver può essere ridotto ad una serie di operazioni elementari che possono essere così definite ed elencate:

- definizione del problema.
- descrizione matematica del problema.
- creazione di un modello TK! destinato a risolvere un problema.
- verifica del modello.
- soluzione del modello.
- se le operazioni precedenti hanno avuto successo documentazione del modello.
- interpretazione della soluzione.

Non è questo né l'ordine né il numero esatto delle operazioni da seguire. Ci renderemo conto, nello sviluppo dei modelli, che alcune di queste fasi possono essere omesse. Ciononostante, per chiarezza d'e-

sposizione, e per comprensione delle documentazioni che verranno eventualmente fornite ad altri utilizzatori, appare necessario seguire uno standard e, questo, oltre tutto proposto dai costruttori del programma, sembra essere lo standard più efficace. Ad esso ci rifaremo, quindi, anche noi, nelle note che seguiranno.

Definizione del problema

Il primo passo nella costruzione di un modello TK!Solver è la perfetta ed univoca definizione del problema. Questo vuol dire rendere nella maniera più chiara possibile, utilizzando eventualmente anche uno schema scritto, la problematica affrontata evidenziando gli obiettivi, i fatti conosciuti, le relazioni ed i sottintesi, che, ovi per una persona, possono non esserli per una macchina. Per aiutarci nella descrizione del problema utilizzeremo un esempio tratto direttamente del manuale che accompagna il programma.

Immaginiamo di aver chiesto un prestito di 5 milioni ad una banca, con un interesse del 20% annuo. Il prestito dovrà essere restituito in due anni e la banca caricherà sul prestito delle spese di istruttoria di L. 100.000. Quanto sarà la rata mensile necessaria per il pagamento del prestito?

Definire il problema significa ricercare le conoscenze di partenza relative al problema stesso ed individuare i risultati richiesti.

Ovviamente, in un problema semplice come questo, il risultato, l'unico obiettivo ricercato, è a quanto ammonta il quantitativo di danaro da pagare mensilmente per restituire la somma richiesta in prestito. I dati invece a disposizione sono: importo del prestito, interesse, spese di istruttoria, e termini di pagamento dell'interesse stesso.

Descrizione matematica del problema

Una volta definite le variabili in gioco ed i valori di partenza, la fase successiva è quella di individuare il problema e di trasformarlo in una equazione o gruppo di esse. In analogia a quanto dice il manuale d'istruzione, si tratta della fase più delicata del problema. Si tratta, cioè, di individuare la migliore costruzione, il migliore progetto di soluzione che consenta di raggiungere il risultato ottenuto nel modo più efficiente e, perché no, elegante. D'altro canto la ricerca del miglior metodo di sviluppo matematico è patrimonio sicuramente non di TK!Solver, ma certamente della matematica generale.

Conoscendo il problema è possibile pervenire alla soluzione attraverso una serie di formule. Queste possono essere già note all'operatore, per acquisito bagaglio personale di conoscenze; possono essere recuperate da libri, riviste (per il problema che ci interessa qualcosa è stato detto in numeri

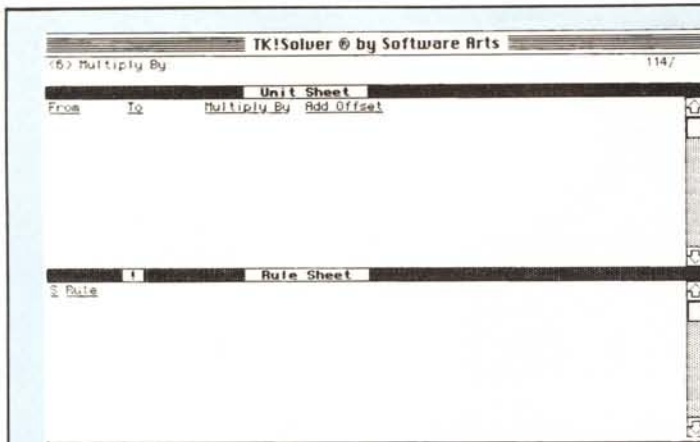


Figura 5

richiesta di prestito	_____	E. 5.000.000
spese per istruttoria	_____	E. 100.000
capitale da restituire	_____	E. 5.100.000
interesse applicato	_____	20%
importo da restituire	_____	?
rata mensile	_____	?

Figura 7

COMANDO WINDOW simbolo [/4]

Il comando WINDOW seleziona una o due finestre per l'output sullo schermo. Nel secondo caso è possibile, tramite lo stesso comando, l'abbinamento degli SHEET desiderati.

opzione effetto

I (2)	mostra una finestra singola (doppia). Il foglio in cui era contenuto il cursore diviene il foglio presente nella finestra
U	seleziona il foglio VARIABLE
R	seleziona il foglio RULE
U	Seleziona il foglio UNIT
G	seleziona il foglio GLOBAL
L	seleziona il foglio LIST
F	Seleziona il foglio User FUNCTION Sheet
P	seleziona il foglio PLOT Sheet
T	seleziona il foglio TABLE Sheet

Figura 6

► capitale = prestito - spese istruttorie
Figura 8

$$\text{pagamento} = \frac{i/100 * \text{capitale} / n}{1 - (i/(100*n)) * (-1)^{-a*n}}$$

$$\text{totale pagato} = \text{pagamento} * n * a$$

$$i = \text{totale} - \text{prestito}$$

dove:

- pagamento è la rata mensile
- i è il tasso d'interesse (in%)
- a è il numero di anni di pagamento
- n è il numero di pagamenti per anno

passati nella rubrica MCArbitri), od addirittura potrebbe essere il caso di affrontare personalmente il problema cercando una soluzione (caso non raro né disprezzabile).

Le 4 equazioni di figura 8 definiscono le relazioni matematiche tra variabili note e sconosciute nel problema.

Creazione di un modello

Scrivendo un modello TK! occorre tener conto di un fatto. Una maschera TK!Solver non è solo un set di equazioni collegate brutalmente insieme. Le equazioni devono invece essere tutte vere contemporaneamente e in relazione alle variabili precedentemente espresse. Le formule devono essere basate sugli stessi assunti e non contraddirsi tra di loro. In pratica, ed in altre parole, esse devono descrivere aspetti, congruenti, dello stesso problema. La seconda considerazione di cui tener conto quando si scrive una equazione riguarda il metodo che il programma usa per risolvere una equazione stessa. Esiste un diretto rapporto, piuttosto stretto, tra il modo con cui si scrive una equazione e quello con cui il programma tenta di risolverlo. Inizialmente il programma tenta, preferenzialmente, per la risoluzione della formula, un meccanismo che va sotto il nome di direct solver (risolutore diretto). In pratica, per risolvere una equazione con direct solver, il programma sostituisce i valori conosciuti delle variabili nella o nelle equazioni proposte e manipola le

equazioni algebricamente, così come fa un comune operatore, per isolare e risolvere le equazioni stesse rispetto ad una singola variabile. Talvolta non è possibile risolvere per una sola incognita, e il tipo di soluzione con direct solver può dare un errore. Se ciò accade, occorre, se si desidera ancora continuare con questa metodologia di soluzione, controllare accuratamente le equazioni inserite, per cercare di stabilire dove è la causa dell'errore. Generalmente, a livello di regola generale, occorre soprattutto verificare che:

- l'incognita appaia una sola volta nella equazione.

- Tutte le variabili, ad eccezione dell'incognita, abbiano un valore assegnato di input o un valore ottenibile dalla soluzione, attraverso direct solver, di un'altra equazione.

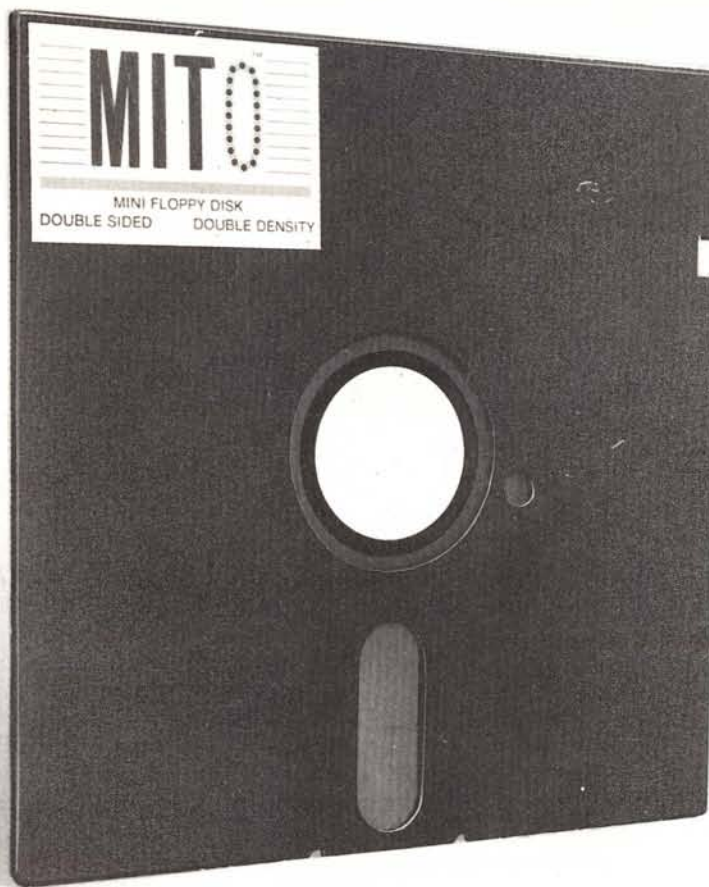
- Infine che l'incognita non sia argomento di una funzione non reversibile (in altre parole non sia funzione di un operatore la cui funzione inversa non è riconosciuta da programma).

Esistono certe regole informali per evitare di cadere negli errori di non soluzione attraverso direct solver. Generalmente la regola migliore, valida non solo in informatica, ma nella vita in genere, è quella di tenere le cose più ordinate possibile. Nel nostro caso, in particolare, appare necessario mantenere, per quanto possibile, le formule piuttosto semplici, eventualmente spezzandole in formule successive. General-

mente questa operazione è possibile, e non c'è da vergognarsi certo nello scrivere formule semplici, anche di pochi operatori. È certamente molto più semplice verificare modelli e cercare errori in formule brevi o che facciano riferimento ad altre già testate in precedenza. È lo stesso discorso della modularità dei linguaggi «C», «Forth» ed altri nei confronti del vecchio Basic confusionario ed arruffone.

Una volta creato un modello destinato a risolvere un problema, è necessario trasformare le equazioni in una forma che il Rule Sheet possa accettare. Questo va fatto secondo una serie di regole, presenti nel manuale del programma, ed una serie di convenzioni necessarie non solo a rendere trasparenti al programma le nostre equazioni, ma a consentirci di organizzare in maniera razionale le nostre regole per una successiva verifica. È opportuno, comunque, dire che queste regole formali sono piuttosto ridotte come numero, intuitive, semplici e del tutto logiche. Alcune, come la proibizione d'uso di certi caratteri (sottolineature, caratteri speciali) sono comunque già note, essendo generalmente gli stessi caratteri già proibiti in altri programmi e linguaggi. Accanto a tali regole esistono suggerimenti, ovi ma mai banali. L'uso di nomi e variabili significative e di modeste gerarchie di parentesi è fin troppo utile per non poter essere consigliato. E l'abbondare nei commenti non sarà mai troppo consigliato!

MC



LA PERFEZIONE DIVENTA MITO

MITO - 5 1/4" Floppy 48 TPI
Doppia Faccia - Doppia Densità
Garantito al 100% - Velocità di
registrazione 5800 BPI
600.000 bytes unformatted.

le misure della perfezione

RECOVERY SERVICE - Un nostro servizio esclusivo. Cosa è il Recovery Service? È uno scudo a protezione del vostro lavoro. Se per un incidente qualsiasi: macchie di caffè, di cioccolato o impronte, il vostro disk dovesse danneggiarsi, la MICROFORUM è in grado di recuperare i dati senza alcun esborso da parte vostra.



La MICROFORUM MANUFACTURING INC.
è interessata all'ampliamento della propria rete distributiva.
Per qualsiasi contatto scrivere anche in italiano.