

CAPITOLO 1

MODALITA' DI INTERFACCIA E STRUMENTI MATEMATICI

Questo capitolo del testo mostra il semplice passaggio tra l'ambiente MS-DOS versione 6.22 e il Qbasic in esso installato, introduce alle modalità di interfaccia tra il linguaggio Qbasic e l'operatore per consentire l'impiego della tecnica di programmazione oggetto del nostro impegno, fornisce inoltre gli strumenti per la compilazione dei programmi matematici.

1.1 Passaggio al programma operativo Qbasic e il ritorno in MS-DOS

Quando accendiamo il P.C., dopo le verifiche automatiche svolte dal sistema operativo MS-DOS, sullo schermo compare la scritta:

C:\> che indica che il P.C. è pronto a ricevere le istruzioni dell'operatore alla tastiera, a questo punto per entrare in ambiente Qbasic basta digitare la scritta "qbasic" a fianco della precedente come segue:

C:\> qbasic e premere il tasto INVIO (ENTER)

avremo la comparsa sul video della schermata di benvenuto che faremo scomparire premendo il tasto ESC, resterà sullo schermo la finestra blu dentro la quale andremo ad operare in linguaggio Qbasic.

La finestra operativa è corredata da un certo numero di scritte per la gestione del programma che sono disposte, come mostrato in figura 1, sia nella parte alta dello schermo, sia nella parte bassa. Per il nostro lavoro impiegheremo prevalentemente la scritta "File" che compare in alto a sinistra.

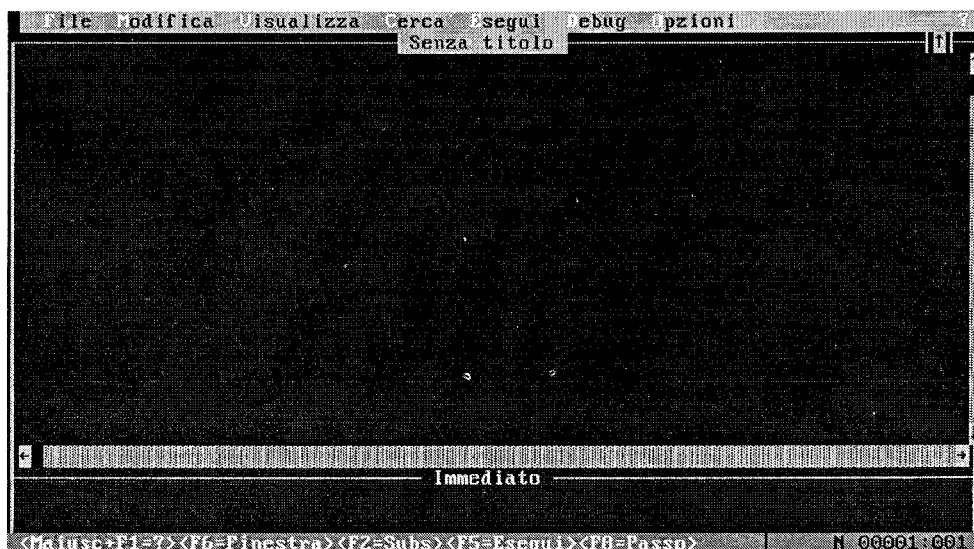


Figura 1
Finestra operativa

Se clicchiamo con il mouse sulla scritta "File" compare nella parte alta dello schermo una lista di operazioni di gestione che possiamo eseguire prima o dopo aver compilato i nostri programmi in Qbasic; queste sono:

Nuovo

Apri....

Salva

Salva con nome ...

Stampa.....

Esci

Il modo di impiego delle diverse operazioni di gestione, ad eccezione della prima e l'ultima, sarà spiegato nel prosieguo del testo quando se ne renderà necessario l'utilizzo.

Se clicchiamo con il mouse sulla scritta **Nuovo** diamo al P.C. l'informazione che il lavoro che ci accingiamo a svolgere è nuovo ed è al momento senza titolo, la conseguenza di questa azione fa scomparire la lista di gestione dallo schermo e lo stesso è pronto ad accogliere il nostro lavoro.

Compare in questo caso, in alto al centro dello schermo, la scritta "Senza titolo".

Per ritornare in MS-DOS basta cliccare con il mouse su **Esci**.

1.2 Le variabili matematiche

Con l'ausilio di poche righe di testo rinfreschiamo alcune nozioni fondamentali presenti senz'altro nel bagaglio culturale di ciascun lettore.

In tutti gli sviluppi matematici, siano essi indirizzati alla soluzione di problemi tecnici, siano essi destinati a studi inerenti l'analisi matematica o la geometria analitica, sono definite **VARIABILI** quelle grandezze, convenzionalmente denotate con le lettere X e Y, che sono suscettibili di assumere valori diversi per descrivere numericamente ogni tipo di processo.

Le variabili che hanno caratteristiche di indipendenza si dicono **VARIABILI INDIPENDENTI**. Le variabili che hanno caratteristiche di dipendenza da altre variabili si dicono **VARIABILI DIPENDENTI**.

Se scriviamo la **notazione esplicita simbolica** di **FUNZIONE** $Y = f(x)$ indichiamo che x è la variabile indipendente mentre Y , che dipende da x , è la variabile dipendente (la notazione $Y = f(x)$ indica infatti che Y è funzione di x).

Lo stesso concetto di **funzione** si riprende mediante una delle innumerevoli **notazioni esplicite strutturate** quale il calcolo del seno di un arco:

$$Y = \text{Sen } x$$

dove x , variabile indipendente, rappresenta l'argomento (arco od angolo) della funzione trigonometrica, mentre Y rappresenta come varia la funzione seno (variabile dipendente) al variare dell'argomento x .

La notazione esplicita simbolica di una funzione $Y = f(x)$ indica genericamente una qualsiasi funzione di x , mentre la notazione esplicita strutturata si riferisce ad una ben determinata funzione descritta mediante una espressione matematica.

Ricordiamo inoltre che le funzioni possono essere legate a più variabili indipendenti; in questo caso la loro notazione esplicita simbolica assumerà la forma:

$$Y = f(X_1 ; X_2 ; \dots ; X_n)$$

Alcune funzioni dette "parametriche" sono legate tanto alla variabile indipendente quanto a parametri caratteristici che generano le "famiglie di funzioni", la notazione simbolica di una funzione parametrica è del tipo:

$$Y = f(X1 ; X2 ; A)$$

dove con X1 e X2 sono indicate ad esempio due variabili indipendenti e con la lettera A è indicato il parametro. Le funzioni fondamentali che costituiscono le basi della trigonometria e dell'analisi matematica sono dette "funzioni elementari".

Le funzioni necessarie per la soluzione dei più diversi problemi possono essere semplici funzioni elementari, algoritmi formati da più funzioni elementari o da funzioni di funzioni.

1.3 Estensione del concetto di funzione

Per facilitare l'uso del linguaggio matematico di Qbasic è utile estendere i concetti che abbiamo ricordato al paragrafo precedente anche a operazioni di aritmetica elementare che normalmente non sono proposte come funzioni.

Se ad esempio consideriamo l'operazione di somma tra due addendi X1; X2 il risultato si scrive:

$$S = X1 + X2$$

dove gli addendi si possono configurare come variabili indipendenti ed il risultato S della loro somma si può configurare come la variabile dipendente.

La somma indicata si può pertanto scrivere con la notazione esplicita simbolica:

$$S = f(X1 ; X2)$$

intendendo in questo caso che la funzione S è dipendente dalle due variabili X1 e X2 .

Questo semplice modo di ragionare è naturalmente estensibile, sia alle altre tre operazioni elementari quali la sottrazione, la moltiplicazione e la divisione, sia alle operazioni di potenza e di radice quadrata; complessivamente sei funzioni che possono essere rappresentate normalmente mediante la lettera Y come variabile dipendente secondo le seguenti notazioni esplicite strutturate:

| | |
|-----------------|-------------------------------|
| somma | $Y = X1 + X2$ |
| sottrazione | $Y = X1 - X2$ |
| moltiplicazione | $Y = X1 \cdot X2$ |
| divisione | $Y = X1 : X2$ |
| potenza | $Y = X^n$ |
| radice q. | $Y = \sqrt[q]{X} = (X)^{1/q}$ |

1.4 Le istruzioni per il colloquio con Qbasic

Dopo aver avuto accesso alla schermata di lavoro del linguaggio Qbasic, seguendo le indicazioni del paragrafo 1.1, è ora necessario imparare le semplici istruzioni con le quali poter **comandare**, quando di bisogno, **l'ingresso e l'uscita dati** dai nostri programmi per colloquiare con Qbasic.

1.4.1 Le istruzioni di ingresso dati

Prima di digitare le istruzioni di ingresso dati è conveniente digitare sulla schermata l'istruzione:

CLS

Digitando come prima istruzione **CLS** si assicura la pulizia automatica dello schermo in modo che risultati di elaborazioni di calcolo precedenti non compaiano assieme a risultati correnti creando confusione di lettura.

Le **istruzioni di ingresso dati** che, quando necessario, devono precedere il programma di lavoro sono relative alla richiesta che il Qbasic dovrà fare all'operatore affinché possa introdurre, digitandoli, i valori da assegnare alle variabili indipendenti ed ai parametri; questi valori sono alla base di tutte le elaborazioni di calcolo in Qbasic perché in esso vengono sviluppate sempre funzioni di una o più variabili e a volte funzioni parametriche.

Il numero delle variabili indipendenti e dei parametri che possono essere introdotti non ha limitazioni pratiche; se ad esempio si devono impiegare due variabili e due parametri: X1, X2, A, C, si dovranno digitare nell'ordine le seguenti quattro istruzioni separate da (invio):

```
INPUT " X1 " ; X1  
INPUT " X2 " ; X2  
INPUT " A " ; A  
INPUT " C " ; C
```

Ciascuna istruzione contiene la scritta di richiesta **INGRESSO DATI (INPUT)**, la scritta che definisce la variabile (" X1 "), ad esempio, che permette, al momento della richiesta da parte di Qbasic, di far comparire sul video la lettera stessa accompagnata da un punto interrogativo, la scritta (; X1) che impone al valore che sarà digitato di inserirsi nel programma di calcolo previsto.

E' utile a questo punto fare un semplice esempio pratico con il P.C. su quanto abbiamo detto: supponiamo pertanto di voler iniziare un programma di lavoro, che non necessariamente dobbiamo ora digitare, che richieda l'introduzione del valore di una variabile indipendente X1 e del parametro B, inizieremo a digitare in sequenza e premere il tasto invio come segue:

```
CLS (invio)  
INPUT " X1" ; X1 (invio)  
INPUT " B" ; B (invio)
```

dopo l'ultimo (invio) premeremo il tasto **F5** per il passaggio obbligato all'esecuzione delle tre istruzioni che abbiamo digitato. A seguito della pressione su F5 scompariranno le istruzioni digitate in campo blu e comparirà, in campo nero, la richiesta di introduzione del valore della variabile indipendente con la scritta:

X1 ?

collocata nella parte alta a sinistra dello schermo.

Supponiamo ad esempio che il valore da assegnare ad X1 sia 9.5, digiteremo tale valore a fianco di X1? come segue:

X1 ? 9.5 (invio)

a seguito del comando invio si avrà la comparsa della scritta in B:

B ?

Con questa azione il valore 9.5 è stato introdotto in un ipotetico programma in Qbasic, e la seconda scritta ci chiede il valore che vogliamo assegnare al parametro B; se questo deve essere per esempio -3.75 digiteremo tale valore ottenendo sullo schermo:

B ? - 3.75 (invio)

premendo (invio) il valore -3.75 verrà introdotto nell'ipotetico programma in Qbasic, contemporaneamente comparirà in fondo allo schermo la scritta (**premere un tasto per continuare**), si preme nuovamente **F5** e si ritorna alla schermata blu di partenza in cui sono ancora presenti le istruzioni digitate all'inizio. La sequenza può essere ripetuta quante volte sia necessario per l'introduzione di altre serie di valori che la variabile indipendente ed il parametro debbano assumere.

Con questo esercizio abbiamo preso confidenza con le modalità di introduzione dei valori delle variabili indipendenti e dei parametri che possono essere richiesti da un qualsiasi tipo di programma di lavoro.

1.4.2 Le istruzioni di uscita dati

Le istruzioni di **uscita dati** da un programma collocate, quando necessario, alla fine dello stesso hanno lo scopo di rendere all'operatore i risultati dei calcoli eseguiti dal programma di lavoro.

I risultati dei calcoli sono naturalmente i valori assunti dalle variabili dipendenti in funzione dei valori assegnati alle variabili indipendenti ed ai parametri.

Il numero delle funzioni elaborate nell'ambito di un programma di calcolo non ha limitazioni pratiche; se ad esempio sono state implementate nel programma di lavoro sette diverse funzioni i risultati di sette diverse elaborazioni numeriche devono essere presentate all'operatore, ciascuna distinta dalle altre mediante lettere o combinazioni di lettere e numeri diversi.

1.4.2.1 Le istruzioni di uscita dati per presentazione in video

Le notazioni delle funzioni possono essere ad esempio tre Y1, Y2, Y3, e le istruzioni necessarie, da collocare alla fine del programma di lavoro, per rendere possibile la lettura sul **video** dei risultati dovranno essere digitate come segue:

```
PRINT " Y1 = " ; Y1
PRINT " Y2 = " ; Y2
PRINT " Y3 = " ; Y3
```

Ciascuna istruzione contiene la scrittura di richiesta di stampa del valore della variabile dipendente (**PRINT**), contiene inoltre la scrittura che definisce la variabile che dovrà essere presentata sul video con il segno di uguale, ad esempio (" Y1 = "), infine la scrittura (; Y1) che permette al programma di andare a prelevare i valori da presentare nell'ambito del complesso di calcolo di Y1. Così come fatto per le istruzioni di INPUT costruiamo un esempio di applicazione pratica per le istruzioni di PRINT.

Supponiamo che un ipotetico programma abbia già calcolato i valori delle tre variabili dipendenti e questi siano Y1 = 3.6 Y2 = -1.45 Y3 = 0.45, per inserire nel nostro esempio questi tre valori, per poter impiegare le istruzioni finali di PRINT, è necessario digitare, prima delle tre istruzioni di PRINT, i valori stessi come se fossero tre righe di istruzioni così come è sotto riportato, avendo cura di premere il tasto (invio) dopo aver digitato ciascun valore di Y e dopo ciascuna delle tre istruzioni di PRINT.

```
Y1 = 3.6           (invio)
Y2 = -1.45        (invio)
Y3 = .45          (invio)      (il programma segue nella pag. 12)
```

```
PRINT " Y1 = " ; Y1 (invio)
PRINT " Y2 = " ; Y2 (invio)
PRINT " Y3 = " ; Y3 (invio)
```

dopo aver digitato quanto sopra premeremo il tasto **F5** per passare all'esecuzione delle tre istruzioni di PRINT. A seguito della pressione su **F5** scompariranno le istruzioni ora scritte su campo blu e compariranno, su campo nero, i risultati attesi nella seguente forma:

```
Y1 = 3.6
Y2 = -1.45
Y3 = .45
```

comparirà inoltre sulla parte bassa del video la scritta (**premere un tasto per continuare**).

Premendo **F5**, od un altro tasto a piacere, si tornerà alla schermata blu di programma.

E' opportuno osservare che nell'esercizio ora eseguito le prime tre righe digitate portano di fatto le identiche scritte che compaiono dopo aver premuto **F5**, ciò può trarre in inganno dando la sensazione che le tre righe in questione vengano trasferite per la lettura per il solo fatto di averle in precedenza digitate. Ciò non corrisponde a realtà, infatti se nell'esercizio omettiamo le altre tre righe relative alle istruzioni di PRINT e premiamo **F5** si ha la scomparsa delle righe rimaste ma nessun dato viene scritto in campo nero sul video, dove compare soltanto la scritta (**premere un tasto per continuare**).

1.4.2.2 Le istruzioni di uscita dati per la presentazione su stampante

Le istruzioni di fine programma che preparano la presentazione dei dati all'operatore mediante scrittura degli stessi su stampante, sono simili alle precedenti descritte per la presentazione video; la differenza tra le due consiste, per la stampa, nella anteponizione della lettera **L** all'istruzione **PRINT**.

Avremo di conseguenza che il complesso delle sei istruzioni dell'esercizio di paragrafo 1.4.2.1, per essere provato con stampante, deve essere scritto come segue:

```
Y1 = 3.6 (invio)
Y2 = -1.45 (invio)
Y3 = .45 (invio)
LPRINT " Y1 = " ; Y1 (invio)
LPRINT " Y2 = " ; Y2 (invio)
LPRINT " Y3 = " ; Y3 (invio)
```

dopo aver digitato quanto sopra premeremo il tasto **F5** per passare all'esecuzione delle tre istruzioni di LPRINT. A seguito della pressione su **F5** scompariranno le istruzioni ora scritte su campo blu e saranno scritti sulla stampante i risultati attesi nella forma:

```
Y1 = 3.6
Y2 = -1.45
Y3 = .45
```

Premendo **F5** si tornerà alla schermata blu di programma .

1.4.2.3 La presentazione contemporanea dei dati su video e stampante

La presentazione dei valori delle variabili dipendenti è possibile contemporaneamente sia su video che su stampante, per far ciò si devono digitare sia le istruzioni PRINT che le istruzioni LPRINT così come riportato nell'esercizio seguente:

```
Y1 = 3.6           (invio)
Y2 = -1.45        (invio)
Y3 = .45          (invio)
PRINT " Y1 = " ; Y1 (invio)
PRINT " Y2 = " ; Y2 (invio)
PRINT " Y3 = " ; Y3 (invio)
LPRINT " Y1 = " ; Y1 (invio)
LPRINT " Y2 = " ; Y2 (invio)
LPRINT " Y3 = " ; Y3 (invio)
```

dopo aver digitato quanto sopra premeremo il tasto **F5** per passare all'esecuzione delle istruzioni di PRINT e LPRINT. A seguito della pressione su **F5** scompariranno le istruzioni scritte su campo blu e saranno scritti sul video e sulla stampante i risultati attesi nella forma:

```
Y1 = 3.6
Y2 = -1.45
Y3 = .45
```

Premendo **F5** si tornerà alla schermata blu di programma .

1.5 L'uso del tasto invio

Nei paragrafi precedenti abbiamo visto che il tasto invio deve essere premuto dopo aver digitato ciascuna istruzione, per evidenziare ciò abbiamo avuto cura di scrivere accanto ad ogni istruzione tale dizione raccolta tra parentesi (invio); dato che questa operazione è sempre necessaria, nel prosieguo del testo non indicheremo più, per semplicità, questa informazione che sarà ricordata dal lettore.

1.6 Esercitazione completa di programmazione

In questo paragrafo riportiamo una semplicissima ma completa esercitazione di programmazione in linguaggio Qbasic che può essere provata dal lettore sul P.C. Si fissa come obiettivo la elaborazione numerica della funzione somma indicata nel paragrafo 1.3:

$$Y = X1 + X2$$

questa funzione ha il vantaggio, su altre, che può essere scritta in linguaggio Qbasic così come si scrive comunemente in modo algebrico.

Si stabilisce che la funzione debba essere computata per le seguenti serie di coppie della variabile indipendente:

| X1 | X2 |
|-------|-------|
| 3 | 7 |
| - 6.7 | 1.5 |
| 1.95 | - 4.3 |

(si osservi che la scrittura dei numeri decimali deve essere espressa con il punto di separazione e non con la virgola; con la stessa espressione saranno presentati i valori della variabile indipendente Y)

La stesura del programma inizia con le istruzioni di cui al paragrafo 1.4.1, a queste segue la funzione somma ed in chiusura l'istruzione di cui al paragrafo 1.4.2.1 così come sotto indicato:

```
CLS
INPUT " X1 " ; X1
INPUT " X2 " ; X2
Y = X1 + X2
PRINT " Y = " ; Y
```

dopo l'ultimo (invio) premiamo il tasto **F5** per il passaggio obbligato all'esecuzione delle istruzioni che abbiamo digitato. A seguito della pressione su **F5** scompaiono le istruzioni digitate in campo blu e compare, in campo nero, la richiesta di introduzione del valore della prima delle variabili indipendenti con la scritta:

X1 ?

che andiamo a completare digitando il primo valore di $X1 = 3$:

X1 ? 3

a seguito del comando invio si ha la comparsa della nuova richiesta:

X2 ?

Con questa azione il valore 3 è stato introdotto nel programma in Qbasic e la seconda scritta ci chiede il valore che vogliamo assegnare a $X2$ che nella nostra tabella è stato fissato in $X2 = 7$. Digitiamo tale valore e otteniamo sullo schermo:

X2 ? 7

premiamo (invio) ed il valore 7 viene introdotto nel programma in Qbasic, contemporaneamente compare il primo valore calcolato della variabile dipendente Y nella forma:

Y = 10

ed in fondo allo schermo la scritta (**premere un tasto per continuare**), si preme nuovamente **F5** e si ritorna alla schermata blu di partenza in cui sono ancora presenti le istruzioni digitate all'inizio. La sequenza si ripete per la coppia $X1 = -6,5$; $X2 = 1,5$ con il risultato:

$Y = -5$

ed ancora per la coppia $X1 = 1,9$; $X2 = -4,3$ con il risultato:

$Y = -2,35$.

La banalità dell'operazione e dei risultati non devono condurre il lettore ad una sottovalutazione del lavoro svolto; è opinione dell'autore che un approccio molto semplice alle problematiche della programmazione di carattere matematico serva a preparare la via ai più complessi processi di calcolo che nel prosieguo del testo dovranno essere affrontati.

1.7 Commenti ai programmi

E' molto utile, a volte indispensabile, affiancare alle istruzioni o a gruppi di istruzioni un commento scritto che riporta il significato delle istruzioni stesse.

Se in un programma molto semplice il commento può risultare superfluo, nel caso di programmi complicati rappresenta l'unico mezzo con il quale, a distanza di tempo, si può ricostruire la filosofia di programmazione seguita che è caratteristica peculiare e distintiva di ciascun programma.

Il commento al programma è cosa estremamente semplice; si tratta di affiancare all'istruzione digitata il significato dell'istruzione stessa separandolo dalla prima mediante il segno ' disponibile sul tasto in cui è presente anche il simbolo (?) (tastiera italiana).

Con queste informazioni procediamo, a solo titolo dimostrativo e non per necessità, al commento del programma presentato nel paragrafo 1.6.

| (programma) | (commento) |
|-------------------|--|
| CLS | ' istruzione per la pulizia dello schermo |
| INPUT " X1 " ; X1 | ' istruzione per l'ingresso della prima variabile indipendente |
| INPUT " X2 " ; X2 | ' istruzione per l'ingresso della seconda variabile indipendente |
| Y = X1 + X2 | ' funzione da elaborare |
| PRINT " Y = " ; Y | ' istruzione per la presentazione video del valore della ' variabile dipendente |

Come si vede per l'ultima istruzione sono state richieste due righe di commento; in questo caso ciascuna riga deve essere preceduta sempre dal segno ' ; **se il commento richiede più righe queste devono essere sempre precedute dal segno ' .**

E' importante osservare che il commento dei programmi non altera minimamente i risultati di questi ma porta ad un allungamento dei tempi di esecuzione degli stessi da parte del P.C. Naturalmente questo fenomeno non è assolutamente avvertibile dall'operatore; ci sono però casi in cui i programmi dedicati ad elaborazioni di dati che pervengono al P.C. dall'esterno possono risentire delle perdite di tempo dovute ai commenti, in questi casi particolari potrà essere necessario eliminare i commenti stessi dal programma che diventa operativo.

1.8 La rivelazione degli errori di compilazione

In Qbasic è implementato un sistema per la rivelazione degli errori che l'operatore può commettere durante la digitazione delle diverse istruzioni che compongono un programma.

Se durante la compilazione del programma si commette un errore qualsiasi, dalla mancanza di una parentesi, ad un errore di sintassi, allo scambio tra una virgola ed un punto ed altro, al primo (invio) che si preme compare, sovrapposta al testo digitato, una finestra con indicazioni accompagnata da un rettangolino luminoso sopra o nei pressi della presenza dell'errore.

Per avere un'idea della situazione conseguente ad uno degli innumerevoli errori che si possono commettere supponiamo di aver digitato l'istruzione non corretta

```
INPUT " A = " . A
```

invece dell'istruzione corretta

INPUT " A = " ; A

dove l'errore consiste nell'aver digitato un punto invece che un punto e virgola prima della lettera A. In questo caso appena premeremo (invio) per passare a digitare l'istruzione seguente comparirà sullo schermo un rettangolino luminoso sul punto errato ed una finestra con le seguenti indicazioni:

Era atteso : ; o ,

< OK> < GUIDA>

Le informazioni date nella finestra sono molto chiare: dove si è acceso il rettangolino luminoso era atteso un punto e virgola o una virgola.

A questo punto l'operatore può procedere alla correzione indicata dopo aver rimosso la finestra indicatrice mediante la pressione del tasto **ESC**.

L'esempio che abbiamo mostrato, che si suggerisce di ripetere al P.C., è uno dei più semplici ed evidenti, in molti altri casi purtroppo la ricerca dell'errore, rimarcato dal sistema di rivelazione, richiede molta pratica all'uso della "Guida per ricerca errori" che è richiamata nella finestra; guida alla quale si può accedere cliccando con il mouse sulla scritta < Guida > della finestra stessa.

1.9 Gli strumenti matematici disponibili in Qbasic

Prima di impiegare le metodologie di programmazione Qbasic è necessario conoscere gli strumenti matematici che questo tipo di linguaggio è in grado di offrire.

Le funzioni matematiche in linguaggio Qbasic devono essere, a volte, espresse secondo una simbologia diversa da quella ordinaria comunemente impiegata in trigonometria od in analisi matematica, per tale ragione sono proposte di seguito, per le funzioni elementari più usate, le corrispondenze tra le due simbologie. Sono infine trattate le memorie di calcolo che sono alla base dei programmi in Qbasic.

Le funzioni di uso poco corrente non sono riportate, per casi particolari si dovrà consultare il testo classico HANDBOOK OF MATHEMATICAL FUNCTIONS di Abramowitz -Stegun.

La simbologia che è indicata in **grassetto** è la stessa che dovrà essere digitata mediante la tastiera nelle fasi di compilazione dei programmi di lavoro, la simbologia in caratteri normali è quella comunemente impiegata nell'ambito della matematica ordinaria.

Le funzioni che sono in dotazione del linguaggio Qbasic si dicono "implementate", le funzioni che si ottengono mediante manipolazioni matematiche a programma si dicono "implementabili".

Le serie delle funzioni elementari, raccolte nei successivi paragrafi, serviranno come ausilio e riferimento nella stesura dei vari programmi di lavoro, a questo scopo sono numerate progressivamente per poter più facilmente richiamarle in seguito.

1.9.1 Le funzioni aritmetiche

Le funzioni aritmetiche impiegate in Qbasic sono:

Simbologia ordinaria

Simbologia Qbasic

1) (radice quadrata)

(NB. Nel testo per indicare la radice quadrata verrà sempre impiegato l'esponente frazionario 1/2)

$$y = \sqrt{x} = (x)^{1/2}$$

$$y = \text{SQR}(x)$$

2) (potenza di n qualsiasi)
 $y = x^n$ $y = (x)^n$

3) (moltiplicazione)
 $y = a \cdot b$ $y = a * b$

4) (divisione)
 $y = a : b = a / b$ $y = a / b$

5) (somma)
 $y = a + b$ $y = a + b$

6) (differenza)
 $y = a - b$ $y = a - b$

7) (valore assoluto)
 $y = |x|$ $y = ABS(x)$

1.9.1.1 Gli operatori aritmetici relazionali

Le corrispondenze simboliche relative agli operatori aritmetici relazionali sono:

Simbologia ordinaria Simbologia Qbasic

8) (maggiore di)
 $a > b$ $a > b$

(maggiore od uguale)

$a \geq b$ $a > = b$

9) (minore di)
 $a < b$ $a < b$

(minore od uguale)

$a \leq b$ $a < = b$

(diverso)

$a \neq b$ $a < > b$

1.9.2 Funzioni trigonometriche

Le funzioni trigonometriche che sono implementate od implementabili in Qbasic prevedono l'impiego di argomenti (valori angolari) espressi in radianti; nel caso in cui gli argomenti debbano essere espressi in gradi sessagesimali è necessario trasformarli in radianti con la seguente espressione:

$$\text{argomento in radianti} = (0.017453293 \cdot A^\circ)$$

dove A° è l'argomento in gradi sessagesimali.

Se per esempio si deve calcolare il seno di 45° (sessagesimali) il corrispondente argomento in radianti è:

$$x = 0.017453293 \cdot 45^\circ = 0.78539818$$

La corrispondenza tra le funzioni trigonometriche espresse in simbologia ordinaria e simbologia Qbasic implementata è la seguente:

| Simbologia ordinaria | Simbologia Qbasic |
|----------------------|-------------------|
|----------------------|-------------------|

10) (seno di x)
 $y = \text{Sen } x$

$$y = \text{SIN}(x)$$

11) (coseno di x)
 $y = \text{Cos } x$

$$y = \text{COS}(x)$$

12) (tangente di x)
 $y = \text{Tang } x$

$$y = \text{TAN}(x)$$

La corrispondenza tra le funzioni trigonometriche espresse in simbologia ordinaria e implementabili in simbologia Qbasic si ottiene:

| Simbologia ordinaria | Simbologia Qbasic |
|----------------------|-------------------|
|----------------------|-------------------|

13) (secante di x)
 $y = \text{Sec } x$

$$y = 1 / \text{COS}(x)$$

14) (cosecante di x)
 $y = \text{Cosec } x$

$$y = 1 / \text{SIN}(x)$$

15) (cotangente di x)
 $y = \text{Cot } x$

$$y = 1 / \text{TAN}(x)$$

Nella simbologia Qbasic gli argomenti devono essere messi sempre tra parentesi (x).

1.9.5 Le funzioni iperboliche

Le funzioni iperboliche sono implementabili in Qbasic mediante il seguente prospetto:

Simbologia ordinaria

Simbologia Qbasic

21) (coseno iperbolico di x)

$$y = \text{ch } x$$

$$y = (\text{EXP}(x) + \text{EXP}(-x)) / 2$$

22) (seno iperbolico di x)

$$y = \text{sh } x$$

$$y = (\text{EXP}(x) - \text{EXP}(-x)) / 2$$

23) (tangente iperbolica di x)

$$y = \text{th } x$$

$$y = (\text{EXP}(x) - \text{EXP}(-x)) / (\text{EXP}(x) + \text{EXP}(-x))$$

1.9.6 Funzioni speciali

Rientrano nella categoria delle funzioni speciali quelle che non sono comprese nelle tipologie precedenti; queste funzioni sono implementabili in Qbasic:

Simbologia ordinaria

Simbologia Qbasic

24) (seno di x su x)

$$y = \frac{\text{Sen } x}{x}$$

$$y = \text{SIN}(x) / x$$

25) (gaussiana)

$$y = e^{-a x^2}$$

$$y = \text{EXP}(-a * x ^ 2)$$

26) (fattoriale)

$$y = x !$$

$$y = 1*2*3*4* * x$$

1.9.7 Le sommatorie algebrica e di funzione

1.9.7.1 La funzione sommatoria algebrica

La funzione sommatoria algebrica è un algoritmo di notevole interesse matematico che è facilmente implementabile in Qbasic. Esponiamo la corrispondenza simbolica ordinaria e Qbasic riservandoci il commento del programma in fase di esercitazione:

Simbologia ordinaria

Simbologia Qbasic

27) (sommatoria algebrica)

$$y = \sum_{x=p}^{x=n} x$$

```
FOR x=p TO n STEP s
  k = x
  y = y + k
NEXT x
```

1.9.7.2 La funzione sommatoria di funzione

La funzione sommatoria di funzione, che gioca un ruolo fondamentale in numerose applicazioni tecniche e matematiche, prevede il calcolo della somma progressiva dei valori che una funzione $f(x)$ assume al variare di "x", a passi di "s", dal valore "x = p" al valore "x = n". La sommatoria di funzione è implementabile in Qbasic mediante una routine di calcolo che viene qui indicata senza commenti al programma; in fase applicativa la routine sarà commentata:

Simbologia ordinaria

Simbologia Qbasic

28) (sommatoria di funzione)

$$y = \sum_{x=p}^{x=n} f(x)$$

```
FOR x=p TO n STEP s
  k = f(x)
  y = y + k
NEXT x
```

1.9.8 Le funzioni della geometria analitica

Sono riportate in questo paragrafo le funzioni della geometria analitica che sono implementabili in Qbasic:

Simbologia ordinaria

Simbologia Qbasic

29) (retta)

$$Y = m x + n$$

$$Y = m * x + n$$

30) (parabola)

$$Y = a x^2 + b x + c$$

$$Y = a * x ^ 2 + b * x + c$$

31) (circonferenza)

$$Y = +/- (r^2 - x^2)^{1/2}$$

$$Y = \text{SQR} (r^2 - x^2)$$
$$Y = - \text{SQR} (r^2 - x^2)$$

32) (iperbole)

$$Y = \frac{1}{x}$$

$$Y = 1 / x$$

33) (ellisse)

$$Y = +/- (b : a) (a^2 - x^2)^{1/2}$$

$$Y = (b / a) * \text{SQR} (a^2 - x^2)$$
$$Y = - (b / a) * \text{SQR} (a^2 - x^2)$$

34) (parabola con asse orizzontale)

$$Y = +/- (2 p x)^{1/2}$$

$$Y = \text{SQR} (2 * p * x)$$
$$Y = - \text{SQR} (2 * p * x)$$

1.10 L'impiego delle memorie di calcolo

Nello sviluppo delle routine di programma si impiegano numerose "memorie di calcolo".

L'uso delle memorie di calcolo è fondamentale nella composizione dei programmi, con esse infatti si possono implementare complessi algoritmi matematici e importanti sistemi grafici.

L'uso delle memorie nel contesto della stesura di un programma in Qbasic è automatico ed estremamente semplice:

-Se digitiamo A = .3759 abbiamo automaticamente assegnato alla memoria da noi denominata A il valore di .3759.

-Se digitiamo B1 = A abbiamo imposto che il contenuto della memoria A sia caricato anche nella memoria B1.

-Se si digita K = 0 si azzerò il contenuto della memoria K.

-Se digitiamo a(23) = -13.89 abbiamo assegnato alla memoria a(23) il valore di -13.89.

Come si è visto le diverse memorie si possono identificare a piacere con lettere o con combinazioni di lettere e numeri:

Non vi è una limitazione pratica al numero delle memorie che possono essere utilizzate in un programma.

Le memorie di calcolo sono volatili, soltanto la struttura del programma nel quale sono state inserite consente il loro impiego con i valori voluti; se in un programma si è posto ad esempio Q = -3.756 un altro programma che dovesse impiegare la memoria Q troverebbe Q = 0.

Le memorie non possono essere rappresentate da soli numeri, la scritta T632 individua una memoria volatile, alla scritta 632 non è associabile nessuna locazione di memoria.

Le memorie possono indifferentemente contenere dati fissi, tipo C = 72.329, come dati variabili quali, ad esempio, quelli risultanti da un calcolo di una funzione $R = f(x)$, dove nella memoria R transitano, uno dopo l'altro, i valori assunti da $f(x)$.

Il trasferimento dei dati tra memorie può essere in cascata:

$F = S$

$Y = F$

$E = Y$

in questo modo il valore contenuto nella memoria S viene a trovarsi anche nella memoria E.

1.11 Specificazione in merito alla stesura dei programmi nel testo

La stesura dei programmi nel testo non può essere contenuta sempre nell'ambito di una pagina a causa sia delle necessità tipografiche, sia delle dimensioni di alcuni programmi .

Onde evitare al lettore spiacevoli errori di digitazione al P.C. si stabilisce di indicare la prosecuzione di un programma nella pagina successiva con il simbolo & posto all'estremità destra dell'ultima riga di programma o di commento.

1.12 Condizioni di blocco nell'esecuzione di un programma

Un programma, formalmente corretto, può bloccarsi in esecuzione se i dati in esso introdotti creano condizioni anomale.

I casi più frequenti nel calcolo numerico sono denunciati dalla comparsa sullo schermo delle seguenti indicazioni:

Divisione per zero

la scrittura compare, con il blocco del programma, quando il denominatore di una frazione si azzerà a seguito dell'introduzione di un dato.

Overflow

la scrittura compare, con il blocco del programma, quando un valore calcolato eccede la capacità numerica specificata .

Chiamata di funzione illegale

la scrittura compare, con il blocco del programma, quando ad esempio si chiede il computo del logaritmo di un numero negativo.

Le scritte menzionate si rimuovono pigiando il tasto **ESC**.

Rimossa la scritta si riavvia il programma bloccato con le quattro seguenti azioni consecutive:

1) si clicca su **File**

2) si clicca su **Apri...**

3) si pigia nuovamente il tasto **ESC**

4) si pigia il tasto **F5** .