

software MBASIC

L'istruzione PRINT USING

In questo numero ci occuperemo dell'istruzione "PRINT USING" andandone ad analizzare le caratteristiche, che forse non tutti conoscono, magari perché non espressamente citate nel manuale allegato al personal computer.

Innanzitutto vediamo la sintassi dell'istruzione in oggetto:

PRINT USING <espr. di tipo stringa>; <lista di espress.>
dove

<espr. di tipo stringa>

è una stringa opportuna (calcolata oppure esplicita), che consente la formattazione di ciò che è indicato con <lista di espress.>".

Generalmente si può dire che, rispetto ad una semplice PRINT, la PRINT USING introduce come detto una certa formattazione dell'output, intendendo con questo termine l'output di valori numerici con un numero prefissato di cifre prima e dopo la virgola e preceduti o seguiti da appositi simboli oppure l'output di parti di stringhe di caratteri.

Il tutto a seconda del contenuto della <espr. di tipo stringa>" definibile a piacere.

Analizziamo dapprima la formattazione delle stringhe.

La formattazione dei campi alfanumerici

Quando si devono stampare delle stringhe alfanumeriche, le si possono formattare con tre caratteri speciali: vediamo ora i singoli caratteri, un esempio di linea di programma e l'output corrispondente.

— **carattere "!"**: indica che della stringa data deve essere stampato solo il primo carattere. Con

PRINT USING "!"; "MCmicrocomputer";
"Audio Review"

si ottiene in uscita:

MA
e cioè solo le iniziali delle stringhe successive.

— **stringa di "/n spazi/":** indica che della stringa data devono essere stampati 2+n caratteri. Se tra le due "backslash" non si mettono spazi (n=0) allora verranno stampati i primi due caratteri. Se la stringa è più

corta del campo risultante, allora i caratteri stampati vengono giustificati a sinistra, cioè con l'inserimento di spazi bianchi a destra. Ad esempio:

PRINT USING "///"; "MCmicrocomputer"
dà in uscita

MC
mentre l'istruzione (con 5 spazi tra le "backslash")

PRINT USING "/ /"; "MC microcomputer"
fornisce in uscita

MC micr
e cioè 7 caratteri della stringa data.

— **carattere "&":** indica che deve essere stampata correttamente la stringa data e cioè con

PRINT USING "&"; "Audio Review" si ottiene in uscita

Audio Review
e cioè la stringa completa.

Mentre i primi due tipi di formattazione non danno problemi, l'ultimo carattere ("&") può destare qualche perplessità, in quanto apparentemente sembrerebbe inutile.

Invece può essere usato, anzi deve, laddove vogliamo eliminare l'effetto USING dovendo usare però una subroutine di stampa contenente tale istruzione.

Supponiamo ad esempio di avere in un programma una subroutine di stampa tra le cui istruzioni compare una

PRINT USING F\$;A\$;B\$;C\$
dove F\$ è la stringa di formattazione contenuta per l'appunto in una variabile e per-

Stringa	Effetto nell'output
!	Primo carattere delle stringhe
\n spazi\	Primi n+2 caratteri delle stringhe
&	Stringhe complete
#	Posto per una cifra
.	Posto per il punto decimale
+	Segno del numero (a destra o a sinistra)
—	Segno "—" a sinistra, se necessario
**	Asterischi prima del numero
\$\$	Segno del dollaro prima del numero
**\$	Asterischi e dollaro prima del numero
,	Virgola delle migliaia, milioni, ecc.
^ ^ ^ ^	Notazione esponenziale
—	Stampa del carattere successivo

Tabella 1 - Effetto della stringa di formattazione sull'output di stringhe (primi tre casi) e di numeri (tutti i successivi). Per i dettagli rimandiamo senz'altro all'articolo dove sono riportati vari esempi.

ciò preimpostabile prima della chiamata della subroutine.

Supponendo perciò che ad un certo punto vogliamo stampare le stringhe A\$, B\$ e C\$ così come sono, ecco che ci torna utilissimo porre

F\$ = "&"
prima della chiamata della subroutine stessa.

Un esempio di applicazione potrebbe essere la gestione di tabulati o prospetti dove la lunghezza di un campo alfanumerico potrebbe variare da una colonna all'altra del tabulato.

La formattazione di campi numerici

Quando, analogamente al caso delle stringhe, bisogna stampare dei valori numerici, con un ben preciso numero di cifre per problemi di incolonnamento o con inserzione automatica di alcuni speciali simboli, ecco che si possono usare fino a dieci tipi di stringhe di formattazione, in genere formate da uno, due o più caratteri.

Anche in questo caso indicheremo la stringa di formattazione ed un esempio di istruzione con il relativo output.

— **carattere "#"** - Il simbolo "#" (chiamato a seconda dei gusti "diesis", "cancelletto", "canale", "number", ecc.) è usato per indicare la singola posizione di stampa di una cifra. Nel caso in cui il numero da stampare possiede meno cifre di quelle specificate dai "diesis", allora tali cifre vengono (correttamente!) giustificate a destra e cioè precedute da un numero opportuno di spazi bianchi.

È questa senz'altro l'opzione della PRINT USING più conosciuta dai programmatori e utilizzata appunto per incolonnare numeri ottenendo il naturale incolonnamento a destra in contrapposizione all'innaturale incolonnamento a sinistra proprio dei computer.

— **carattere "."** - Il "punto" può essere inserito in una qualsiasi posizione di un campo numerico, seguito e preceduto da un numero desiderato di "diesis". In questo caso l'incolonnamento avverrà tenendo conto delle cifre decimali. Una cosa che forse non tutti sanno è che, se necessario, i valori decimali vengono arrotondati; inoltre, se il valore non ha la parte intera, allora viene stampato comunque uno "0" seguito dal punto decimale e dalle cifre prestabilite.

Ad esempio il programmino

```
10 A$ = "#.#.#"
20 PRINT USING A$; 31.41
30 PRINT USING A$; 1.3
40 PRINT USING A$; 5.119
50 PRINT USING A$; .55574
```

darà in uscita la seguente tabella di valori

31.41
1.30
5.12
0.56

dove, negli ultimi due casi si è avuto un arrotondamento del valore numerico.

— **carattere "+"** - Il segno di addizione, posto all'inizio o alla fine della stringa

COMPUTER GAME

by B&V

COMPUTERS ED ACCESSORI

Viale Roma, 168
47100 FORLÌ

☎ 0543/67078 - 52190

SPEDIZIONI IN TUTTA ITALIA

PREZZI IVA COMPRESA

COMMODORE

COMMODORE 64	Telefonare
REGISTRATORE	L. 55.000
COMMODORE + REGISTRATORE	Telefonare
DRIVE 1541	L. 510.000
STAMPANTE MPS 803	L. 490.000
MONITOR COLORI	L. 540.000
INTERFACCIA CENTRONICS	L. 120.000
INTERFACCIA RS 232	L. 77.000
MODEM CBM 64	L. 295.000
BACK UP - COPIA PERFETTAMENTE CASSETTE	
TRA 2 REG. COMMODORE	L. 64.000

SINCLAIR

SINCLAIR QL	L. 1.195.000
SPECTRUM	L. 460.000
SPECTRUM 48K	L. 360.000
CARTUCCE MICRODRIVE	L. 12.000
ESPANSIONE 32K	L. 65.000
MODEM SPECTRUM	L. 295.000
INTERFACCIA CENTRONICS QL	L. 110.000
INTERFACCIA CENTRONICS SPECTRUM	L. 100.000
INTERFACCIA 2 SINCLAIR + KEMPSTON	
+ CURSOR JOYSTICK	L. 80.000
KEMPSTON	L. 25.000
INTERFACCIA JOYSTICK PROG.	L. 80.000
GRAPHIC PEN SPECTRUM	L. 55.000
MODEM QL	L. 285.000
CAVO RS 232 QL	L. 35.000

AMSTRAD

AMSTRAD CON MONITOR COLORI	L. 960.000
AMSTRAD CON MONITOR F. VERDI	L. 580.000

OFFERTE SPECIALI

SINCLAIR POKET TV	L. 280.000
TELEVISORE SCHERMO PIATTO	
STAMPANTE 80 COLONNE	
GRAFICA FOGLIO SINGOLO	
INTERFACCIA CENTRONICS	L. 620.000
PROGRAMMATORI DI EPROM	
PER TUTTI I COMPUTER	
CON RS 232	L. 340.000
(2716 + 2732 + 2764 + 27128 +27256)	
MONITOR FOSFORI VERDI	L. 150.000

CONDIZIONI DI VENDITA :

Spese postali fisse L. 6.000.
Pagamento a mezzo vaglia postale
o contrassegno (ORDINE SCRITTO).
Garanzia 3 mesi su tutta la merce.

software

MBASIC

di formattazione (costituita ad esempio da "diesis" e ".") fa sì che corrispondentemente venga stampato, prima o dopo il numero, il suo segno positivo o negativo che sia (attenzione!).

Ad esempio

PRINT USING "+##.#"; -1.1.,33
fornisce in uscita

-1.1 +0.3

— **carattere "-"** - Il segno "meno", a differenza del precedente, impone la stampa di un segno "meno" (solo se il numero è negativo) subito dopo il numero stesso. Strano, no?! Ad esempio l'istruzione (il "-" va posto in fondo)

PRINT USING "##.#-"; -1.1.,33
dà un output del tipo

1.10— 0.33

— **stringa "**"** - Un doppio asterisco all'inizio di un campo di formattazione forza la stampa di asterischi al posto di eventuali spazi, nel caso che il numero sia formato da meno cifre che non il campo prefissato. I due stessi asterischi indicano, ove necessario, la posizione per altre due cifre. Ma vediamo degli esempi.

Con l'istruzione

PRINT USING ".*##.#";3.14,-.77.855
otterremo

.*3.1 —0.8 855.0

— **stringa "\$\$"** - Il doppio dollaro consente di stampare "prima" del numero formattato il carattere "dollaro", considerando che anche in questo caso i "due dollari" indicano due ulteriori cifre del campo di formattazione.

L'istruzione

PRINT USING "\$\$##.#";123.44
fa stampare in uscita

\$123.4

— **stringa "\$*\$"** - L'insieme di due asterischi e del dollaro all'inizio della stringa di formattazione fa sì che si ottenga l'effetto combinato delle due precedenti stringhe e cioè il riempimento con asterischi e la stampa del singolo "dollaro" subito a sinistra del numero formattato. Per non complicarci ulteriormente le idee lasciamo agli abili lettori la sperimentazione di questa stringa.

— **carattere ","** - Siamo arrivati alla "fatidica" virgola, che gli anglosassoni si ostinano ad usare, insieme al punto, in maniera esattamente opposta al nostro modo. In pratica ponendo una virgola subito a sinistra del "punto decimale" nella stringa di formattazione si ottiene la stampa automatica di una virgola separatrice delle migliaia, dei milioni, ecc. Se invece la virgola viene posta alla fine della stringa di formattazione, allora verrà stampata alla destra del numero, come parte integrante del numero stesso.

Ad esempio

PRINT USING "###.#,##";5555.5
fornisce in uscita il valore

5,555.50

— **stringa "AAAA"** - La stringa formata da quattro caratteri "accento circonflesso" o

"freccia in su" specificano, se posti alla destra della stringa di formattazione, che il numero verrà stampato in formato esponenziale. In particolare i quattro simboli "~" (in inglese "carat") specificano che verrà stampata la stringa "E+xx" dove "xx" è l'esponente calcolato correttamente in base alle posizioni delle cifre intere e decimale stabilite dalla solita stringa di formattazione. Ancora una volta non riteniamo di portare un esempio di applicazione in quanto ormai la stringa di formattazione può contenere sia il "~" che il segno "+" a sinistra o a destra, sia il segno "-" a destra: elencare tutti i casi possibili servirebbe solo a complicare le idee...

— **carattere "_"** - Il carattere "sottolineatura" (in inglese "underscore") ha un effetto molto piacevole, anche se non molto noto: in particolare il carattere "_" all'interno della stringa di formattazione fa stampare così com'è il "prossimo carattere" della stringa stessa.

Subito un esempio:

PRINT USING "_(##.#-);33.44
fa stampare in output
(33.44)

Lasciamo ancora una volta al lettore la prova di caratteri vari subito dopo "-", per ottenere stampe di effetto.

— **carattere "%"** - In realtà non è un carattere da porre nella stringa di formattazione, ma compare "naturalmente" in output allorché il numero da stampare ecceda lo spazio riservato. Appare anche quando l'arrotondamento di un numero che altrimenti non darebbe problemi, genera invece un numero che non "ci sta" nella stringa data.

Capita infatti che usando l'istruzione

PRINT USING "##.#";555.111
si ottiene

%555.11

Come pure, nel caso in cui si ha

PRINT USING ".##";.9999
ecco apparire in uscita

%1.000

Si è visto dunque che la PRINT USING si comporta in alcuni casi molto intelligentemente: notevolissimo è l'arrotondamento, che non costringe l'operatore a complesse istruzioni di stringa per avere un numero decimale approssimato. Prima di terminare ci sono da aggiungere due importantissime considerazioni.

La prima è che vi è un limite al numero di cifre prefissabili con la stringa di formattazione, limite posto a 24, oltre il quale l'MBASIC segnalerà un

"ILLEGAL FUNCTION CALL".

L'altra considerazione è che tutto quanto detto finora vale anche per l'istruzione LPRINT USING, con la sola differenza che l'output stavolta si avrà sulla stampante e non sul video ed assumendo una lunghezza della linea di stampa pari a 132 caratteri. Con questo concludiamo questa puntata ed inoltre forniamo uno specchietto riassuntivo di tutto quanto esposto finora, in forma comprensibilmente succinta.

MC

**PER IL TUO
COMMODORE 64**

EASY COMPUTING

Ora EASY COMPUTING
ti dà una mano per far funzionare
al meglio il tuo COMMODORE 64.
Una organizzazione amica ed efficace
famosa in Europa, e da oggi anche in Italia.

EASY COMPUTING ti offre la più vasta gamma di prodotti originali per il COMMODORE 64, tradotti in italiano, per un immediato utilizzo, sia nel campo professionale che nel tempo libero. Con il vantaggio di ricevere tutta la documentazione relativa al programma che ti interessa direttamente a casa tua. Basta compilare il coupon o scrivere direttamente a EASY COMPUTING - Via A. Bertani 24 - 50137 Firenze.

Questi i principali programmi che EASY COMPUTING ha selezionato per te:

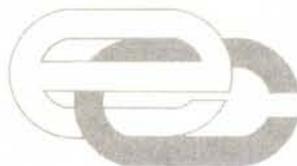
SUPERSOFT - MUSIC MASTER, BUSICALC 2, BUSICALC 3, TOOLKIT, VICTREE, ZOOM, INTERDICTOR PILOT, MIKRO ASSEMBLER e una scelta di VIDEOGAMES intelligenti.

ABACUS - ZOOM PASCAL, SUPER DISK UTILITIES, SCREEN GRAPHICS, ULTRABASIC, SYNTHY 64, VIDEOBASIC, GRAPHICS DESIGNER, TAS, CADPAK, CHARTPAK.

VIZA - VIZASPELL, VIZAWRITE.

ANIROG - Per la prima volta in Italia decine di videogames originali, considerati come i più elaborati e affascinanti del mercato europeo.
OXFORD PASCAL, HARDCOPY.

HARDWARE - SUPERSKETCH, VIDEO GRAPHIC DIGITISER, LIGHT PEN, 4 SLOT MOTHERBOARD.
INTERFACCE: SERIELINK/RS, SERIELINK, CENTROSERIAL, PRINTLINK, etc.



EASY COMPUTING
VIA A.BERTANI 24 FIRENZE

Sono interessato a ricevere il catalogo generale EASY COMPUTING, gratuitamente e senza impegno, al seguente indirizzo:

Nome _____
Cognome _____
Indirizzo _____
Città _____ CAP _____
Professione _____
Tel. _____

MC