

Brevi note sull'uso del formato Fortran per ingressi ed uscite dei dati

Formato libero

Le istruzioni Fortran di ingresso ed uscita dati, a formato libero, sono le più semplici da utilizzare quando serve unicamente prendere visione dei risultati e non si desidera fissare (per i numeri floating point), nè il numero di cifre dopo il punto di radice nè il tipo di rappresentazione (punto fisso oppure forma esponenziale).

Pertanto con tale formato, ci si limita a segnalare le variabili (il cui tipo è stato definito nella parte dichiarativa) per le quali tramite lettura si vuole assegnare un valore, ovvero per le quali si vuole visualizzare in uscita il valore memorizzato.

Gli esempi che seguono sono fatti per ingressi da tastiera ed uscite su video, ma **tutto resta valido anche quando si lavora con i files.**

Esempio:

```
CHARACTER*20 titolo
REAL*8 X, Y, Z
INTEGER N, M, L
.....
READ *, titolo, L, Z      oppure READ (*,*) titolo, L, Z
READ *, N, X, M, Y      oppure READ (*,*) N, X, M, Y
PRINT *, titolo, L, Z   oppure WRITE (*,*) titolo, L, Z
PRINT *, N, X, M, Y     oppure WRITE (*,*) N, X, M, Y
```

Con questo esempio, in ingresso (da tastiera), sarà sufficiente inserire due righe di dati, separati da almeno uno spazio bianco, ciascuna contenente i dati per le variabili elencate (attenzione che, se si vuole inserire una stringa di caratteri con degli spazi bianchi è obbligatorio inserirla tra apici!). Il tipo di dati inserito deve essere corretto, altrimenti si ottiene una segnalazione di errore del tipo:

```
invalid number: incomprehensible list input
apparent state: unit 5 (unnamed)
last format: list io
lately reading direct formatted external IO
Aborted
```

oppure

```
invalid integer: read unexpected character
apparent state: unit 5 (unnamed)
last format: list io
lately reading direct formatted external IO
Aborted
```

Un possibile ingresso potrebbe essere

```
PROVA_programma  -10  1.234567891234
52 1345.65435467  -1   0.536153215353e-12
```

e la corrispondente uscita su video

```
PROVA_programma      -10  1.23456789
52 1345.65435 -1  5.36153215E-13
```

oppure

```
'Questo e' un titolo' 10 1.25e10
1 15.76524453e-7 -50 1234567891234567.1234567
```

e l'uscita

```
Questo e' un titolo 10 1.25E+10
1 1.57652445E-06 -50 1.23456789E+15
```

Si noti che, indipendentemente da come sono stati inseriti i dati e da come sono memorizzati, in uscita il formato libero assume per default come posizionare i dati. Per i numeri floating point decide se utilizzare o meno la notazione esponenziale (con la prima cifra significativa alla sinistra del punto di radice) ed anche il numero di cifre decimali che vengono visualizzate.

Formato assegnato dall'utente

Per quanto riguarda le **istruzioni di lettura**, non è particolarmente essenziale indicare esplicitamente il formato, e quindi si può utilizzare il formato libero.

Invece, spesso, vi è la necessità di indicare con maggior dettaglio come si vogliono ottenere i **dati in uscita** (principalmente per i numeri floating point). Infatti, ad esempio, se utilizziamo la doppia precisione reale, può essere opportuno ottenere in uscita almeno 15-16 cifre significative per i valori che rappresentano le approssimazioni delle soluzioni, oppure un formato esponenziale con poche cifre significative dopo il punto di radice per i valori che rappresentano gli errori (dove è solo l'ordine di grandezza che è importante), oppure ancora avere delle uscite ben incolonnate.

I principali descrittori di formato che si possono usare sono:

- Iw (w colonne per rappresentare un numero intero, compreso il segno; es: I5)
- Fw.d (w colonne per rappresentare un numero reale nella forma a punto fisso, con d cifre dopo il punto decimale; es: F20.15)
- Ew.d (w colonne per rappresentare un numero reale nella forma esponenziale ovvero numeroE+nn oppure numeroE-nn, con d cifre dopo il punto decimale; es: E10.2)
- Gw.d (w colonne per rappresentare un numero reale con d cifre dopo il punto decimale; la forma a punto fisso o esponenziale viene decisa dal programma in dipendenza del valore del numero; es: G10.2)
- nX (lascia in uscita n spazi bianchi; es.: 10X)
- Aw (w caratteri di tipo stringa; es: A5; se non si indica il numero w, viene trasferita in uscita la stringa così com'è stata definita in fase di dichiarazione di tipo CHARACTER; es: A)

L'esempio precedente:

```
PRINT '(A, I5, F10.5)', titolo, L, Z           oppure WRITE (*,'(A, I5, F10.5)') titolo, L, Z
PRINT '(I5, E15.5, I4, F10.2)', N, X, M, Y     oppure WRITE (*,'(I5, E15.5, I4, F10.2)') N, X, M, Y
```

produce la seguente uscita

```
PROVA_programma      -10   1.23457
 52   0.13457E+04   -1     0.00
```

Si noti che, con il descrittore di formato esponenziale, la prima cifra significativa è alla destra del punto di radice; che il valore $1.564567e-5$ viene stampato come 0.0 visto che viene richiesto il formato a punto fisso; che avendo utilizzato il descrittore A semplice, vengono stampati 20 caratteri (come dichiarato per la variabile titolo).

Attenzione: mentre con il formato libero i valori vengono sempre visualizzati, un utilizzo dei descrittori di formato non consono al dato che si vuole stampare, può provocare in uscita la scrittura di un certo numero di asterischi. In tal caso si deve variare il descrittore di formato relativo ai valori che non sono stati visualizzati. Infatti, nel caso del secondo gruppo di dati, si ottiene:

```
Questo e' un titolo   10*****
 1   0.15765E-05 -50*****
```

Ciò accade perchè i valori numerici che sono contenuti nelle variabili Z ed Y non possono essere stampati con il formato indicato (rispettivamente F10.5 e F10.2). Basta variare tali formati. Ad esempio indicando

```
PRINT '(A, I5, E15.5)', titolo, L, Z           oppure WRITE (*,'(A, I5, E15.5)') titolo, L, Z
PRINT '(I5, E15.5, I4, F20.2)', N, X, M, Y     oppure WRITE (*,'(I5, E15.5, I4, F20.2)') N, X, M, Y
```

si ottiene

```
Questo e' un titolo   10   0.12500E+11
 1   0.15765E-05 -50 1234567891234567.00
```

Si noti comunque che la parte dopo il punto decimale del valore di Y non corrisponde alle cifre inserite. Questo perchè in doppia precisione si riescono a memorizzare solo 15-16 cifre decimali. Quindi se si chiede di stampare dei numeri reali indicando globalmente più di 15-16 cifre significative (contando sia quelle prima che dopo il punto decimale), le cifre eccedenti non hanno alcun significato!

Ripetizione di formato

Quando si devono stampare (o leggere) valori di variabili **dello stesso tipo e consecutive**, con lo stesso formato, è possibile utilizzare un fattore moltiplicativo per non dover ripetere il descrittore.

Ad esempio con le istruzioni

```
PRINT '(A, 2I5, E15.5)', titolo, L, N, Z       oppure WRITE (*,'(A, 2I5, E15.5)') titolo, L, Z
PRINT '(I5, 2E15.5)', M, X, Y                 oppure WRITE (*,'(I5, 2E15.5)') N, X, M, Y
```

si ottengono le seguenti stampe (rispettivamente per il primo ed il secondo gruppo di dati)

```
PROVA_programma      -10  52   0.12346E+01
  -1   0.13457E+04   0.15646E-04
```

```
Questo e' un titolo   10   1   0.12500E+11
 -50   0.15765E-05   0.12346E+16
```