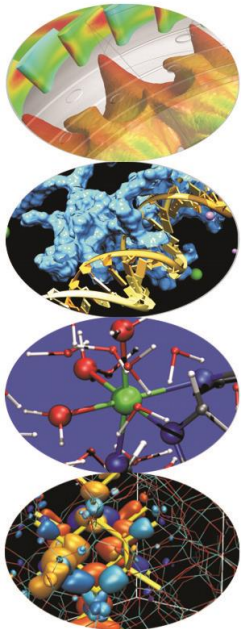


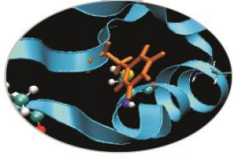


# Funzioni Intrinseche

***Introduction to Fortran 90***  
Paolo Ramieri, *CINECA*

*Aprile 2013*





# Funzioni intrinseche

Il Fortran mette a disposizione una libreria di funzioni intrinseche che possono essere invocate con la seguente sintassi:

```
nome_funzione ( {lista dei parametri} )
```

Alcune tra le funzioni intrinseche più utilizzate:

ABS, ACOS, COS, DOT\_PRODUCT, EXP, INT, LEN, LOG, LOG10, MATMUL,  
MAX, MIN, MOD, SIN, SQRT, TAN...

In genere le funzioni intrinseche hanno differenti versioni in base al tipo degli argomenti passati.

Già in FORTRAN 77, ma più propriamente a partire dal Fortran 90, si usa comunemente il nome generico della funzione e automaticamente verrà richiamata la versione più appropriata.



# Funzioni intrinseche

Le procedure intrinseche del Fortran si possono raggruppare in cinque categorie principali:

**Elementali:** sono definite per argomenti scalari, ma il risultato si conforma alla forma dell'entità passata in argomento e il valore calcolato elemento per elemento. Comprendono tra l'altro le funzioni numeriche (ABS, INT, ...) e matematiche (SIN, SQRT, ...).

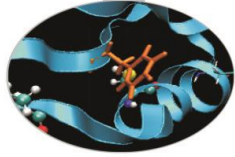
**Informative:** ritornano proprietà delle entità che non dipendono dal loro valore numerico.

**Trasformazione:** trasformano un'entità in un'altra. Per esempio generano una nuova entità copiando parti di un'altra.

**Non elementali:** si applicano ad argomenti scalari ma non rientrano in alcuna delle altre categorie.

**Procedure per matrici:** si applicano a vettori e matrici.

# Funzioni intrinseche



Funzioni che operano su stringhe di caratteri:

`ACHAR ( I )` : carattere di codice `I` nella tabella ASCII

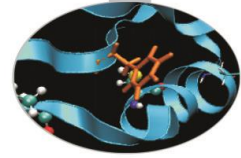
`IACHAR ( C )` : ritorna il codice del carattere `C` nella tabella ASCII

`LEN_TRIM ( STRING )` : lunghezza del testo, tolti gli spazi di coda

`ADJUSTL ( STRING )` : sposta gli spazi di testa mettendoli in coda

`ADJUSTR ( STRING )` : sposta gli spazi dalla coda alla testa

# Funzioni intrinseche



Funzioni che operano su stringhe di caratteri:

`INDEX (STRING, SUBSTRING [, BACK]) :`

posizione di partenza della prima (ultima se `BACK= .TRUE.`)

occorrenza della sottostringa `SUBSTRING`

`SCAN (STRING, SET [, BACK]) :`

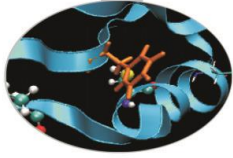
posizione della prima (ultima se `BACK= .TRUE.`) occorrenza di

uno dei caratteri di `SET`

`VERIFY (STRING, SET [, BACK]) :`

posizione della prima (ultima se `BACK= .TRUE.`) occorrenza di

un carattere che non è contenuto in `SET`



# Funzioni intrinseche

Funzioni che generano una trasformazione di stringhe di caratteri:

`REPEAT (STRING, NCOPIES) :`

genera una stringa di caratteri concatenando

`NCOPIES` copie della stringa `STRING`

`TRIM (STRING) :`

genera una stringa di caratteri eliminando gli spazi

sulla destra di `STRING`



# ESERCIZIO

Scrivere un programma per provare le funzioni:

ADJUSTL

ADJUSTR

INDEX

LEN\_TRIM

SCAN

VERIFY