



Mini-Corso di Informatica

CALCOLI DI PROCESSO DELL'INGEGNERIA CHIMICA

Ing. Sara Brambilla

Tel. 3299

sara.brambilla@polimi.it

Note sulle esercitazioni

- Durante le esercitazioni impareremo a implementare gli algoritmi discussi a lezione
- Ci avvaleremo di Matlab come strumento, ma si potrebbe usare un qualunque altro linguaggio di programmazione
- Matlab offre una serie di vantaggi che vi saranno chiari durante il corso

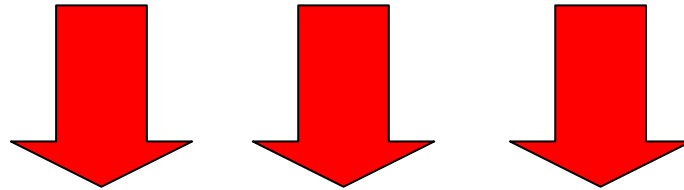
Note sull'esame

- Durante le esercitazioni è lecita qualsiasi domanda, anche se vi sembra banale
- Le domande stupide non influiscono sul voto dell'esame (a meno che non le facciate durante l'esame)

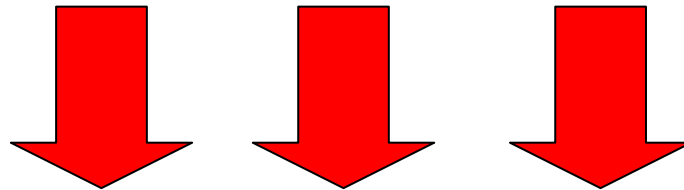
Il computer

- Il computer è una macchina e come tale ignorante
- Dovete spiegargli, attraverso un **linguaggio di programmazione**, che cosa deve fare, passo per passo, nella sequenza corretta

DATI DI INPUT



PROGRAMMA



DATI DI OUTPUT



Concetti fondamentali

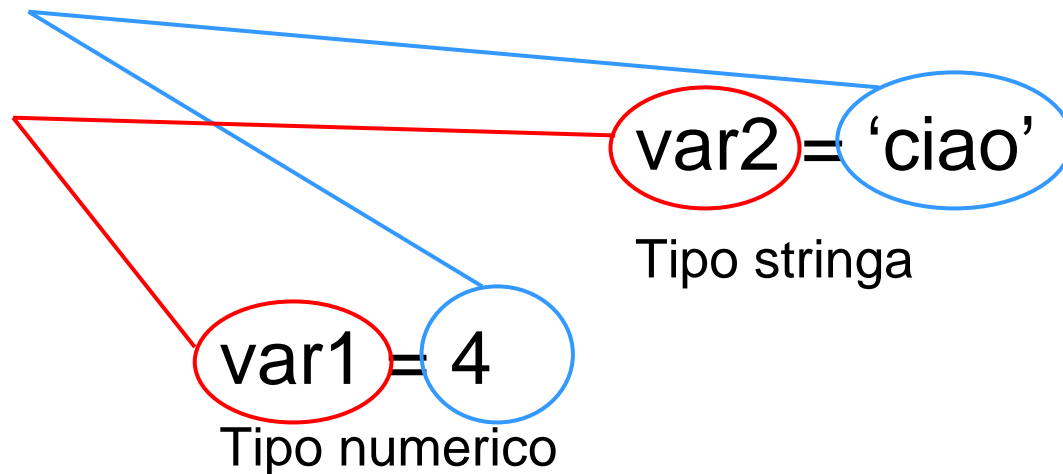
- **VARIABILE**

- **ISTRUZIONI FONDAMENTALI**

- **FUNZIONE**

Variabili

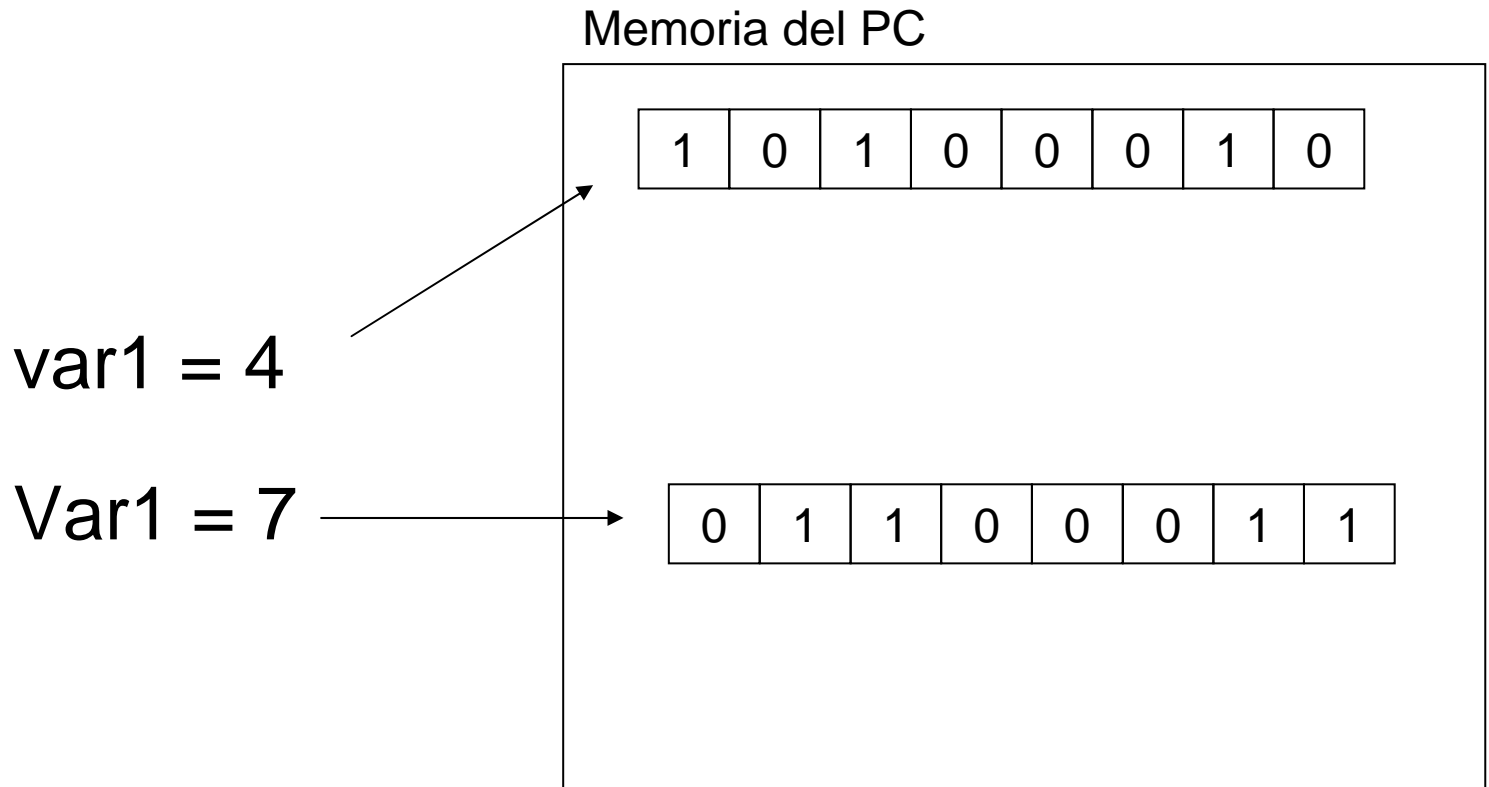
- Le variabili sono caratterizzate da:
 - Il tipo (variabili numeriche, stringhe di caratteri,...)
 - Il valore
 - Il nome



Le variabili

- Matlab è **case-sensitive**: la variabile “pippo” è diversa dalla variabile “PiPpo”
- È bene usare la *camel notation*:
 laMiaNuovaVariabile
- È bene evitare variabili chiamate “l”, perché in Matlab la elle minuscola sembra un uno, il che rende difficile rileggere il codice e scovare gli errori

Variabili

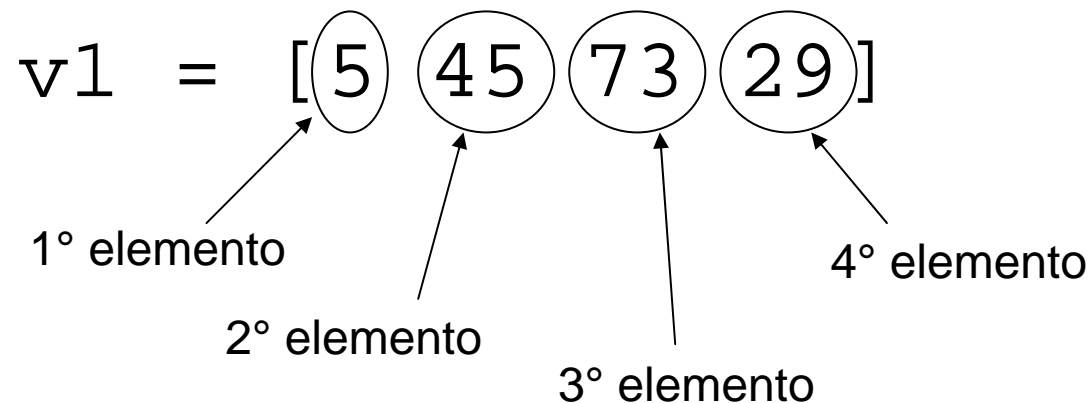


Vettori

- In Matlab un vettore si rappresenta come

$$v1 = [5 \ 45 \ 73 \ 29]$$

- Gli elementi del vettore si contano a partire da 1:



Vettori

- Per accedere all'i-esimo elemento del vettore `v1` si utilizza l'espressione

`v1(i)`

Quindi per accedere al 3° elemento:

`v1(3)`

Vettori

- Creazione di un vettore con tutti gli elementi = 0

```
v1 = zeros(3)
```

```
⇒ v1 [0 0 0]
```

- Creazione di un vettore con tutti gli elementi = 1

```
v1 = ones(1,3)
```

```
⇒ v1 [1 1 1]
```

Vettori

■ Vettore con elementi equispaziati

$v1 = [1 : 2 : 11]$

Diagram illustrating the components of the colon operator: '1' is labeled 'inizio', '2' is labeled 'incremento', and '11' is labeled 'fine'.

$\Rightarrow [1 \ 3 \ 5 \ 7 \ 9 \ 11]$

■ Meglio usare l'istruzione

$v1 = \text{linspace}(1, 11, 6);$

Diagram illustrating the components of the linspace function: '1' is labeled 'partenza', '11' is labeled 'fine', and '6' is labeled 'numero elementi'.

Matrici

- In Matlab una matrice si rappresenta come:

$$A = [17 \ 15 ; 4 \ 32]$$

Si ottiene la matrice:

$$A = \begin{bmatrix} 17 & 15 \\ 4 & 32 \end{bmatrix}$$

Segnala l'inizio di una nuova riga



Matrici

- Per accedere al j -esimo elemento della riga k :

$$A(k, j)$$

Ad esempio il 2° elemento della 1° riga è:

$$A(1, 2)$$

E si ottiene

$$A = \begin{bmatrix} 17 & 15 \\ 4 & 32 \end{bmatrix} \Rightarrow 15$$

Dimensioni

- Per conoscere le dimensioni di un vettore si utilizza il comando:

```
dim = length(v1)
```

- Per conoscere le dimensioni di una matrice si utilizza il comando:

```
[nRig, nCol] = size(A)
```


Copia di vettori e matrici

- Un vettore (o matrice) a può essere duplicato, copiato in un altro vettore (o matrice) b con l'istruzione: $b = a$

Non serve, cioè, copiare tutti gli elementi uno ad uno

Costrutti

- Ciclo **FOR**
- Ciclo **WHILE**
- Costrutto **IF – ELSE – ELSE IF**

Ciclo FOR

- Il ciclo FOR è un'istruzione che permette di eseguire un certo numero di volte una serie di comandi.

Diagramma di un ciclo FOR con etichette:

```
for i = 1:1:10  
    ... istruzioni  
end
```

Le etichette e le frecce puntano ai seguenti elementi:

- Variable contatore**: punta a **i**
- Valore iniziale**: punta al primo **1**
- Incremento**: punta al secondo **1**
- Valore finale**: punta al **10**

Ciclo FOR - Esempio

- Sommare i numeri da 1 a 100

```
somma = 0;  
for i = 1:100  
    somma = somma + i;  
end
```

Ciclo WHILE

- Il ciclo while è un ciclo che ripete le istruzioni al suo interno fino a che la condizione è vera

```
while ( condizione )  
    ..... istruzioni  
end
```

Ciclo WHILE - Esempio

- Sommare i numeri interi a partire da 1 fino a che la loro somma non sia maggiore o uguale a 325

```
somma = 0;
cont = 0;
while(somma < 325)
    cont = cont + 1;
    somma = somma + cont;
end
disp(['Numero iter : ', num2str(cont)]);
```

IF – ELSE – ELSE IF

- Le istruzioni vengono svolte solo se la condizione è vera

```
if(condizione)
    ..... istruzioni
elseif(condizione)
    ..... istruzioni
elseif(condizione)
    ..... istruzioni
.....
else
    ..... istruzioni
end
```

IF – ELSE – ELSE IF - Esempio

- Calcolare il valore del modulo di un numero

```
x = 45;
```

```
if(x >= 0.)
```

```
    valoreAssoluto = x;
```

```
else
```

```
    valoreAssoluto = -x;
```

```
end
```




Le Funzioni

- Le funzioni sono blocchi di codice che svolgono un particolare compito.

```
a = 4.2;
```

```
b = 7.3;
```

```
c = MySum(a, b);
```

Copia dei valori di a
e b nella nuove
variabili x1 e x2

```
function y = MySum(x1, x2)
```

```
y = x1 + x2;
```

Memoria del PC

a

b

x1

x2

In y viene salvato il risultato dell'operazione eseguita dalla funzione `MySum`. La variabile y viene restituita al codice che ha chiamato la funzione `MySum`

Note

- Quando Matlab entra nella funzione MySum, le variabili note sono SOLO quelle presenti nella funzione o passate alla funzione



`c = MySum(a, b) ;`

`a, b` = variabili passate alla funzione `MySum`

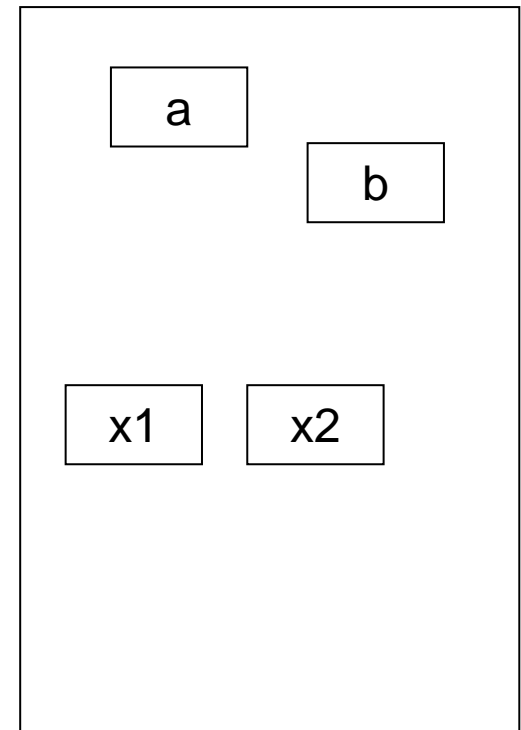
```
function y = MySum(x1, x2)
```

`x1, x2` = nome delle variabili passate a `MySum` e utilizzabili all'interno della funzione

`y` = valore restituito da `MySum` alla funzione che l'ha chiamata

- Se in MySum eseguiamo delle operazioni su $x1$ e $x2$ alterandone il valore, il valore di a e b rimane sempre lo stesso. Questo perché $x1$ e $x2$ non occupano lo stesso spazio di memoria di a e b

Memoria del PC





Le funzioni

- Le funzioni vanno salvate su file separati dal codice principale
- Il file deve avere lo stesso nome della funzione che è in esso contenuta

Problemi esempio

1. Sommare gli inversi dei numeri da 1 a 1E6 (un milione)
2. Creare un vettore con inizio da 1.25 e fine 2.55 con spaziatura 0.05 tra gli elementi
3. Individuare quanti inversi dei primi N numeri interi occorre sommare per ottenere un risultato maggiore di 3

4. Creare una funzione per la somma di due vettori

5. **COMPITO A CASA:**


■ Dato l'elenco dei numeri di matricola, determinare se nell'elenco ci sono ripetizioni e quante sono. Con carta e penna determinare il numero minimo di confronti da effettuare.

■ Sommare i due vettori

$$a = [3 \ 4 \ 5 \ 6];$$

$$b = [7 \ 8 \ 9 \ 1];$$

elemento per elemento.

- 
- Determinare quanti anni sono necessari per accumulare almeno la somma di 30 k€ avendo depositato in banca 10 k€ e potendo contare su un interesse al netto di tutte le spese pari al 3.51%. Diagrammare l'andamento del deposito bancario nel tempo.