

Rappresentazione di segnali

- L'asse continuo dei tempi è approssimato da un vettore numerico equispaziato (es: $t=-10:1:10$)
- Un segnale è rappresentato da 2 vettori: il vettore dei tempi (t) e un vettore con i valori del segnale negli istanti di t (es. x)

t	-10	-9.9	-9.8	...	-0.1	0	0.1	...	9.8	9.9	10
i	1	2	3	...	100	101	102	...	199	200	201
x	100	98.01	96.04	...	0.01	0	0.01	...	96.04	98.01	100

- Accesso tramite indice: $x(1)$ non è x al tempo $t=1s$, ma è il primo elemento di x ($t(1)=-10s$)

Rappresentazione di segnali

- Accesso logico: l'indice è un vettore logico
- Per accedere al valore di x al tempo $t=1s$, si usa $x(t==1)$

t	-10	-9.9	-9.8	...	-0.1	0	0.1	...	0.9	1	1.1	...	9.8	9.9	10
i	1	2	3	...	100	101	102	...	110	111	112	...	199	200	201
$t==1$	0	0	0	...	0	0	0	...	0	1	0	...	0	0	0
$t \geq 0 \ \&\& \ t \leq 1$															
	0	0	0	...	0	1	1	...	1	1	0	...	0	0	0

Comando `function`

- Deve stare nella *prima riga* di un m-file che porta lo *stesso nome* della funzione
- Sintassi: `function [out1,out2,...] = nome_fun (param1,param2,...)`

Esempi

sum_pos.m

```
function S = sum_pos(x)

S = 0;
for i = 1:length(x)
    if x(i)>0
        S = S+x(i);
    end
end
```

elimina_val.m

```
function [y,quanti] = elimina_val(x,val)

y = x;
quanti = 0;
for i = length(x):-1:1
    if x(i)==val
        y(i) = [];
        quanti = quanti+1;
    end
end
```

Shell (oppure lab2.m)

```
t = -10:0.1:10;
somma = sum_pos(t);

somma = sum(t(t>0));
```

```
x = [1:10 1:10];
[v,q] = elimina_val(x,5)
```

Comando function

- Il comando `return` esce dalla funzione

Esempio

scomp.m

```
function divisori = scomp(x)

divisori = [];
x = int32(x);
fprintf('Searching the decomposition of %d...\n',x);
if x<=1
    fprintf('Warning! The argument is <=1, returning empty vector...\n');
    return;
end

primi = primes(double(x)/2);
for i = 1:length(primi)
    while ~mod(x,primi(i))
        divisori = [divisori primi(i)];
        x = x/primi(i);
    end
    if x==1
        return;
    end
end
end
```

Shell (oppure lab2.m)

```
vett_scomp = scomp(831096)
```