

9 Funzioni MATLAB

Le funzioni in un programma sono utilizzate per strutturare il codice in sottoparti e per evitare di replicare inutilmente il codice. In MATLAB le funzioni vengono identificate con la parola chiave `function` ed è buona regola che abbiano lo stesso nome dello script che le contiene. In generale sono strutturate nel seguente modo:

```
1 function [output1, output2, ..] = nome(input1, input2, ..)
2 %istruzioni
3 output1 = ..
4 output2 = ..
5 ...
```

dove:

- `output1, output2, ..` sono gli output (opzionali), che, se dichiarati, devono essere inizializzati dalla funzione
- `input1, input2, ..` sono gli input (opzionali) che servono per il calcolo degli output

Prima di iniziare a scrivere il corpo di una funzione, la prima cosa da stabilire è quali siano input (argomenti) e output (valori restituiti) necessari.

MATLAB vede solo le funzioni delle proprie librerie (come `mean()`, `min()`) o le funzioni dichiarate nella stessa cartella di dove viene eseguito lo script in cui si richiamano le funzioni.

Un elenco (non esaustivo) delle funzioni di MATLAB è il seguente:

```
1 zeros(m,n) %crea una matrice di zeri di dimensioni m*n
2 ones(m,n) %crea una matrice di uni di dimensioni m*n
3 eye(n) %crea una matrice identita' di ordine n
4 rand(m,n) %crea una matrice di numeri casuali in [0,1] di
   dimensioni m*n
5 randi(p,m,n) %crea una matrice di numeri interi casuali in [1,p
   ] di dimensioni m*n
6
7 length(v) %restituisce la lunghezza del vettore v
8 size(M) %restituisce le dimensioni della matrice M
9
10 ceil(x) %arrotonda x all'intero superiore
11 floor(x) %arrotonda x all'intero inferiore
12 fix(x) %arrotonda x all'intero piu' vicino a 0
13
14 [M, idxM] = max(v) %restituisce il massimo M del vettore v e il
   suo indice idxM
15 [m, idxm] = min(v) %restituisce il minimo m del vettore v e il
   suo indice idxm
16
17 mean(v) %restituisce la media del vettore v
18 median(v) %restituisce la mediana del vettore v
19 mod(m,n) %restituisce il modulo n di m
20
21 find(p) %restituisce gli indici degli elementi che soddisfano p
```

9.1 Esercizi

Esercizio 9.1

La *radice numerica* di un numero intero è ottenuta sommando le sue cifre, poi sommando le cifre del risultato, eccetera, fino a quando non si ottiene una singola cifra.

Esempio: $65536 \rightarrow 6 + 5 + 5 + 3 + 6 = 25 \rightarrow 2 + 5 = 7$.

Scrivere una **funzione** che calcoli la radice numerica di un numero intero. Quindi, scrivere uno **script** che, dato un array di interi, calcola la radice numerica di ogni suo elemento.

Esercizio 9.2

Scrivere una funzione `closestVal` che prende in ingresso:

- un vettore v
- un valore n

e restituisce un valore dell'elemento di v più vicino a n , ossia a distanza minima.

Ad esempio:

```
1 v = [1 4 40];  
2 n = 20;  
3 closestVal(v, n)
```

restituisce 4, mentre

```
1 v = [1 4 40];  
2 n = 32;  
3 closestVal(v, n)
```

restituisce 40.

Utilizzare in uno script la suddetta funzione e commentare il caso in cui si debbano restituire tutti i valori a distanza minima (nel caso ce ne sia più di uno).

Esercizio 9.3

Scrivere uno script che chiede all'utente di inserire un numero positivo a (nel caso in cui il numero non sia positivo ripetere l'inserimento) e verifichi se il numero è perfetto, in caso contrario dice se è abbondante o difettivo.

Un numero è perfetto se corrisponde alla somma dei suoi divisori, escluso se stesso. Ad esempio 6 è perfetto perchè $1 + 2 + 3 = 6$. Un numero è abbondante se è minore della somma dei suoi divisori e altrimenti è difettivo. Ad esempio 20 è abbondante visto che $1 + 2 + 4 + 5 + 10 > 20$, mentre 19 (come tutti i numeri primi) è difettivo.

Richiede quindi un secondo numero b e controllare se a e b sono due numeri amici.

Due numeri a e b sono amici se la somma dei divisori di a è uguale a b e viceversa. Ad esempio 284 e 220 sono amici.

Strutturare lo script con delle funzioni:

- `inserisciPositivo` che legge un numero intero positivo;
- `sommaDivisori` che dato un numero restituisce la somma dei suoi divisori;
- `controllaSePerfetto` che restituisce se un numero è perfetto, se è abbondante o difettivo:
- `controllaSeAmici` che restituisce se due numeri sono amici.

Esercizio 9.4

Scrivere uno script che chiede all'utente di inserire due parole e che controlli se una è l'anagramma dell'altra.

Strutturare lo script scrivendo una funzione `istogramma` che legge una parola e ne restituisce l'istogramma delle frequenze delle lettere presenti nella parola.

Esercizio 9.5

(TdE 27 Giugno 2016) Si vogliono costruire in MATLAB opportune funzioni per gestire un vettore numerico come se fosse una coda (si pensi alla coda di persone davanti allo sportello di un ufficio pubblico). Assumendo che l'elemento 1 del vettore corrisponda al primo della coda, sviluppare le seguenti funzioni:

- `function [nuovaCoda] = accoda(coda, valore)`: restituisce una nuova coda contenente tutti gli elementi di coda più il nuovo elemento `valore`, che viene inserito alla fine della nuova coda;
- `function [nuovaCoda, valore] = estraiPrimo(coda)`: estrae da coda il primo valore. Restituisce la nuova coda che non contiene più il primo valore, e il valore stesso. Per esempio, se `coda = [10, 3, 4, 12, 17]`, dopo l'esecuzione della funzione si avrà che: `nuovaCoda = [3, 4, 12, 17]` e `valore = 10`;
- `function [lunghezza] = calcolaLunghezza(coda)`: restituisce lunghez-

za, che rappresenta il numero di elementi in coda.

Si considerino due vettori numerici che rappresentino due code, normale e prioritaria, definite come:

```
1 codaNormale = [10, 3, 4, 12, 17, 13, 68, 45, 32, 22];
2 codaPrioritaria = [11, 5, 6, 8, 9, 15];
```

Si ipotizzi che gli elementi di una coda siano estratti partendo dal primo valore del vettore che rappresenta la coda, e che la variabile `ultimoEstratto` indichi se l'ultimo valore estratto sia stato estratto da `codaNormale` (in questo caso `ultimoEstratto` assume il valore 0) oppure da `codaPrioritaria` (in questo caso `ultimoEstratto` assume il valore 1).

Si implementi uno script MATLAB in modo che richieda all'utente di inserire le due code e il valore di `ultimoEstratto`, quindi estragga un singolo numero da una delle due code e ne stampi a video il valore, seguendo il seguente criterio per decidere da che coda estrarre il numero:

- se l'ultimo valore è stato estratto da `codaNormale`, allora il nuovo valore sarà estratto da `codaPrioritaria`;
- se l'ultimo valore è stato estratto da `codaPrioritaria` e se la lunghezza di `codaPrioritaria` è maggiore della lunghezza di `codaNormale/3` (opportunamente arrotondato), allora estrae il valore da `codaPrioritaria`; altrimenti, estrae il valore da `codaNormale`.

Ad esempio, se venissero inseriti i due vettori precedentemente definiti e la variabile `ultimoEstratto` avesse valore 0, il programma dovrebbe stampare a video il valore 11 e modificare il secondo vettore nel seguente modo: `codaPrioritaria = [5, 6, 8, 9, 15]`.

Per lavorare con le due code si utilizzino le funzioni definite al punto precedente. Inoltre, lo script dovrà aggiornare anche il valore delle variabili `codaNormale`, `codaPrioritaria` e `ultimoEstratto` in modo coerente con quanto definito.

Esercizio 9.6

(TdE 04 Febbraio 2016) Si implementi in linguaggio MATLAB una funzione `partition` che prende in ingresso un vettore V di valori numerici non ordinati, e restituisce in uscita un nuovo vettore T (composto dagli stessi elementi di V) e un valore numerico m , definiti nel modo seguente:

1. Se V ha lunghezza dispari, m è l'elemento in posizione centrale di V e il vettore T è composto dagli stessi elementi di V disposti nel seguente modo: a sinistra di m ci sono tutti gli elementi di V minori di m , poi compare m (in tutte le sue occorrenze),

quindi a destra tutti gli elementi maggiori di m . Ad esempio, dato il vettore $V = [6 \ 5 \ 3 \ 4 \ 3 \ 1 \ 2]$, la funzione restituisce il vettore $T = [3 \ 3 \ 1 \ 2 \ 4 \ 6 \ 5]$ e il valore centrale $m = 4$:

2. Se V ha lunghezza pari, il valore m è definito come la media dei due elementi centrali di V , e tutti gli elementi di V vengono disposti in T in modo tale che tutti quelli minori o uguali di m precedono quelli maggiori di m . Ad esempio, dato il vettore $V = [5 \ 6 \ 1 \ 2 \ 3 \ 4]$, la funzione restituisce il vettore $T = [1 \ 5 \ 6 \ 2 \ 3 \ 4]$ e il valore centrale $m = 1.5$ (cioè la media dei due valori in posizione centrale 1 e 2).

Si scriva uno script MATLAB che svolge le seguenti operazioni:

- Acquisisce da tastiera un vettore numerico `vett` e un intero positivo n ;
- Crea un vettore `med` vuoto;
- Svolge n volte le seguenti operazioni:
 - Invoca la funzione `partition` passando come parametro `vett`, sovrascrivendo `vett` con il vettore restituito da `partition` e memorizza nella prima posizione libera di `med` il secondo valore restituito;
 - Stampa il nuovo valore di `vett`;
- Al termine dell'esecuzione del ciclo, stampa il vettore `med`.

Esercizio 9.7

Quest'anno Babbo Natale ha deciso di farsi aiutare per la consegna dei regali da KwanzaBot¹ e Superman. Babbo natale ha un elenco di bambini (`elenco` matrice di caratteri) e deve consegnare un regalo particolare ad ognuno di essi tra piccolo (1), medio (2) e grande (3) a seconda di quanto sono stati buoni (`buono` vettore di interi). Per assegnare il regalo ci sono delle soglie di punteggi: fino a 700 punti il bambino ha un regalo piccolo, fino a 900 medio, altrimenti grande. Ad ogni consegna si deve stampare a video il nome del bambino, il suo regalo e chi l'ha portato.

Per dividersi il lavoro i tre hanno deciso che Babbo Natale avrebbe consegnato i regali partendo dal primo dell'elenco, KwanzaBot dal fondo e Superman scegliendo uno dei bambini a caso (`randi()`). I bambini che hanno il regalo grande lasciano anche dei biscotti a chi consegna il regalo, quindi ringrazia riportando il suo apprezzamento per i biscotti (stampandolo a video).

Strutturare lo script che consegna i regali a tutti i bambini in funzioni che dividano in maniera logica l'azione di selezione del regalo (`selezionaRegalo`), la consegna del

¹<http://futurama.wikia.com/wiki/Kwanzaabot>

regalo (consegnaRegalo) e l'eliminazione dall'elenco di chi ha già ricevuto il regalo (cancellaBambino).