

Esercitazioni di Informatica B

Funzioni in MATLAB

Stefano Cereda

stefano.cereda@polimi.it

3/12/2018

Politecnico di Milano



Funzione split

Scrivere una funzione (split) che data una variabile di ingresso m (scalare, vettore o matrice di numeri) ed un numero x restituisce due vettori $vMin$ e $vMax$. $vMin$ contiene tutti gli elementi di m minori di x , mentre $vMax$ gli elementi maggiori di x .

Struct e funzioni: codice IBAN i

Un codice IBAN è una sequenza di 27 caratteri alfanumerici così composta:

- 2 caratteri maiuscoli rappresentanti la sigla internazionale della Nazione
- 2 cifre (CIN Europeo)
- 1 carattere maiuscolo (CIN Italiano)
- 5 cifre (ABI)
- 5 cifre (CAB)
- 12 cifre (Numero di Conto)

naz	cin_e	cin_i	abi	cab	conto
IT	02	L	12345	12345	123456789012

Struct e funzioni: codice IBAN ii

Si scrivano le seguenti funzioni:

- *removeSpaces*, che prenda in ingresso una stringa *strIn* e restituisca una stringa *strOut* contenente tutti i caratteri di *strIn* ad esclusione degli spazi.
- *isAlpha*, che prenda in ingresso una stringa *str* e restituisca 1 quando *str* contiene esclusivamente lettere maiuscole, o 0 altrimenti.
- *isDigit*, che prenda in ingresso una stringa *str* e restituisca 1 quando *str* contiene esclusivamente caratteri corrispondenti a cifre, o 0 altrimenti.

Si utilizzino tali funzioni per scrivere un'ulteriore funzione *checkIBAN* che riceve come parametro una stringa e restituisce 1 se la stringa inserita, una volta rimossi gli spazi, corrisponde ad un codice IBAN che rispetta lo schema sopra illustrato, o 0 altrimenti.

Struct e funzioni: biblioteca i

Si vuole automatizzare la gestione di una biblioteca di un piccolo paese.

Come primo passo si vuole gestire una matrice di dimensioni $N \times 2$ contenente nella prima colonna un identificativo numerico e nella seconda colonna un numero di tessera.

Immaginate l'identificativo numerico come un codice interno alla biblioteca mentre il numero di tessera come il numero di una tessera esistente appartenente (ad esempio il vostro codice persona del politecnico).

ID	Codice Persona
1	123456789
2	987654321
...	...

Si forniscano le funzioni necessarie all'inserimento di una nuova tessera e alla ricerca dell'identificativo numerico dato un numero di tessera (l'identificativo numerico è un numero crescente che parte da 1).

Funzione *inserisci*:

input: matrice Soci e nuovo numero di tessera

output: la matrice Soci

Funzione *cerca*:

input: matrice Soci e numero di tessera da cercare

output: l'identificativo numerico associato al numero tessera

Un'azienda ha installato un sistema che registra gli ingressi e le uscite dei propri dipendenti tramite l'uso di badge. All'entrata e all'uscita, ogni dipendente deve registrarsi passando il badge nel lettore posto all'ingresso. Siete stati incaricati di scrivere in Matlab parte del software per la gestione dei dati acquisiti.

Il programma Matlab è costituito da una serie di funzioni che lavorano sui seguenti dati:

1. Informazioni relative a ciascun dipendente. Queste sono organizzate in una struct con i seguenti campi:
 - ID (un numero intero che identifica univocamente ogni dipendente),
 - nome (stringa)
 - cognome (stringa)
2. Il registro dei dipendenti. Questo è un array di struct che contiene in ogni suo elemento:
 - I campi di un dipendente (ID, nome, cognome).
 - La data che corrisponde a un ingresso e a un'uscita del dipendente (si suppone che avvengano nello stesso giorno). Questa data è rappresentata con un numero intero di 8 cifre. Per esempio, il numero 20171223 corrisponde alla data 23 Dicembre 2017.

- oraIngresso e oraUscita, che vengono rappresentati con un numero intero che indica l'ora come il numero di minuti trascorsi dalla mezzanotte precedente (ad esempio, le ore 7:30 corrispondono a 450 minuti, infatti $7 * 60 + 30 = 450$).

1. Per verificare se il dipendente può uscire oppure no, scrivere la funzione *verificaUscita* che prende in ingresso il registro, una data nel formato indicato al punto 2, e un ID e controlla se il dipendente corrispondente ha lavorato nella data passata come parametro per almeno 7 ore e 30 minuti (i.e., 450 minuti). La funzione restituisce vero se questa condizione risulta verificata, falso in caso contrario.
2. Scrivere la funzione *badge* che prende in ingresso un dipendente, la struttura registro, la data odierna (rappresentata come indicato al punto 2), un intero contenente l'ora corrente (rappresentata come indicato al punto 2). La funzione *badge* aggiorna registro nel seguente modo: se nella data odierna il dipendente non compare nella struttura registro (ossia l'utente sta entrando in azienda), la funzione ne aggiunge le

informazioni indicando come data e oraIngresso la data e l'ora passate come parametro. Altrimenti, se il dipendente è già presente nel registro alla data odierna (ossia l'utente sta uscendo dall'azienda), la funzione *badge* aggiorna il campo *oraUscita* relativo al dipendente con il valore del parametro *ora*. La funzione *badge* quindi richiama la funzione *verificaUscita* descritta al punto 1 e, se il dipendente può uscire stampa a video 'Arrivederci!', altrimenti 'Errore uscita!'

3. Si supponga che il registro sia stato precedentemente popolato con i dati relativi ai dipendenti di un'azienda. Si scriva un frammento di codice per definire una variabile *dip1* corrispondente al dipendente "Giacomo Puccini" avente ID = 1858. Si invochi inoltre la funzione *badge* al fine di inserire nel registro il suo ingresso il 3 Febbraio 2017 alle 8:15.

Un comitato sportivo di sci vuole memorizzare le performance degli sportivi che percorrono la più ripida delle piste del comprensorio. Per ogni atleta, la memorizzazione parte da quota $h_0 = 3000m$ e termina a quota $h_{end} = 1000m$, e viene effettuata in 100 punti diversi del tracciato equidistanti tra loro. I dati vengono salvati in:

- la matrice *atleti*, le cui righe rappresentano i diversi atleti monitorati, e le cui colonne rappresentano i vari tempi collezionati sul tracciato (quindi l'elemento (i, j) della matrice contiene il tempo in cui l'atleta i -esimo è transitato per il j -esimo punto della pista);
- il vettore colonna *velocitaMedia*, con stesso numero di righe della matrice *atleti*, che contiene un valore numerico per ogni atleta, indicativo della sua velocità media lungo tutta la pista.

Scrivere in linguaggio Matlab una funzione *dividiMatrice* che:

1. riceva in input una matrice *atletiTot* (con la stessa struttura di *atleti*), un vettore *velocitaM* (con stessa struttura di vettore *velocitaMedia*) e uno scalare *soglia*;
2. fornisca in output due matrici *atletiOver* e *atletiUnder* (ognuna con la stessa struttura di *atletiTot*). *atletiOver* include solo le righe di *atletiTot* corrispondenti agli elementi di *velocitaM* con valore maggiore o uguale di *soglia*. *atletiUnder*, invece, include solo le righe di *atletiTot* corrispondenti agli elementi di *velocitaM* con valore minore di *soglia*.