

Esercizi 5

CICLI



Esercizio 1

Realizzare una funzione che, chiesto all'utente un numero intero pari a N, ricavi e stampi a video tutte le terne pitagoriche con i cateti minori o uguali ad N.

Una terna pitagorica è formata da tre numeri naturali (a, b, c) tali che $a^2 + b^2 = c^2$.

Esempio 3,4,5 ($9+16=25$).

N.B. : Calcolare solo le terne in cui $a < b$.

Esercizio 2

Codificare in matlab la funzione **p=primo(x)** che restituisce:

- 1 se x è un numero primo
- 0 altrimenti.

Si utilizzi a tal proposito la funzione rem.

Si progetti uno script che legge da tastiera un numero N, e stampa a video tutti i numeri primi compresi tra 0 e N.

Esercizio 3

- Scrivere una funzione matlab che richiede a terminale dei numeri, inserendoli man mano in un vettore. L'inserimento termina quando si preme invio senza inserire alcun numero.
- La funzione calcoli e visualizzi il massimo numero inserito (senza usare la max).
- La funzione restituisca in output il vettore completo.

Esercizio 4 (esami)

Un programma MATLAB deve analizzare i voti degli studenti iscritti al primo anno di università. I dati sono contenuti in un file di testo esami.txt . Per ogni studente, il file esami.txt contiene le seguenti informazioni, separate da punto e virgola:

- cognome dello studente (stringa)
- una sequenza di 10 numeri (uno per ogni esame). Per ogni esame viene riportato il voto dello studente (0 se non ha sostenuto tale esame).

(segue)

Esami (continua)

Il programma deve visualizzare sullo schermo la media dei voti per ogni studente. Per fare questo, si usi il seguente algoritmo:

1. Si importi il file `esami.txt` usando l'import wizard separando il vettore dei nomi dalla matrice dei voti. La funzione principale dovrà prendere in ingresso queste variabili.
2. Si scriva una sottofunzione "nonNulli" che prende in ingresso un vettore e fornisce in uscita il numero dei valori diversi da zero contenuti in tale vettore.

Esami (continua)

3. Si scriva una sottofunzione "mediaNonNulli" che prende in ingresso un vettore e fornisce in uscita la media dei valori diversi da zero contenuti in tale vettore. N.B. Per calcolare la media non bisogna considerare i valori a zero (perché rappresentano esami non sostenuti).
4. Nella funzione principale si analizzi ogni riga della matrice dei voti (costruita al punto 1) e si stampi a video il nome dello studente, il numero di esami sostenuti e la sua media. (Per far questo, bisogna utilizzare le funzioni definite al passo 2 e 3).

Esami (continua)

5. Si stampi la lista visualizzata al punto 4 anche in un file di testo resoconto.txt.

Ad esempio, se il file esami.txt contiene i seguenti dati:

Rossi	30	27	24	0	0	27	0	0	27	0
Bianchi	0	18	24	0	24	0	0	0	0	30

il programma dovrà visualizzare:

Rossi: 5 esami, media = 27

Bianchi: 4 esami, media = 24

Esercizio 5 (orario lezioni)

Un file di testo ORARIO.TXT contiene l'orario delle lezioni di un giorno della settimana per il primo anno di Ingegneria dell'Informazione. L'orario è strutturato per lezioni; ogni lezione contiene le seguenti informazioni

- nome del corso (stringa di 20 caratteri)
- ora di inizio (int)
- durata della lezione (int)

Si vuole inserire una nuova lezione nell'orario; per questo è necessario sapere se la nuova lezione si sovrappone con le lezioni attuali. Si dovrà scrivere una funzione MATLAB che visualizza le eventuali sovrapposizioni, seguendo il seguente algoritmo.

1. Si legga, usando l'import wizard, il file ORARIO.TXT e lo si divida in vettori opportuni. Si crei la funzione principale, che prende in ingresso i vettori appena creati. Si visualizzino, tramite una sottofunzione, i tre vettori in forma di tabella.

Orario lezioni (continua)

2. Si scriva una sottofunzione (che verrà utilizzata al punto 3) che dati due orari di inizio e due durate di lezioni fornisce:
 - 0 se le lezioni non si sovrappongono
 - 1 se le lezioni si sovrappongono.

Due lezioni L1 ed L2 si sovrappongono se l'inizio di L1 precede la fine di L2 e contemporaneamente l'inizio di L2 precede la fine di L1.

3. All'interno della funzione principale, si leggano da tastiera le informazioni sulla nuova lezione da inserire. Si visualizzino a video tutte le lezioni che si sovrappongono con la nuova lezione.

Orario Lezioni (continua)

Ad esempio, supponiamo che il file ORARIO.TXT contenga le seguenti informazioni

Corso	ora inizio	durata
Analisi2	8	3
Fisica1	11	2
Fisica2	14	3

e che la nuova lezione sia *Inglese*, con inizio alle 10 e durata 2 ore. Il programma dovrà indicare che il nuovo corso si sovrappone ad Analisi2 e Fisica1

Esercizio 6 (Robot)

Un robot è in grado di muoversi nel piano nelle quattro direzioni est, nord, ovest e sud (denotate rispettivamente dagli angoli 0, 90, 180, 270). In particolare, il robot accetta comandi che consistono in una direzione e un numero, che rappresenta la distanza (in metri) da percorrere in quella direzione. I comandi sono immagazzinati in un file di testo `comandi.txt`, in cui ogni riga contiene un comando, e vengono eseguiti dal robot in sequenza. Ad esempio, una possibile sequenza di comandi è la seguente:

90	5
0	14
270	7
0	2

Robot (continua)

Si importi in Matlab il contenuto del file comandi.txt e si scriva una funzione principale.m che prende in ingresso tale vettore e visualizza su schermo di quanti metri si è spostato il robot ed in quale direzione, secondo il seguente algoritmo:

1. Il programma calcola di quanti metri si è spostato il robot ed in quale direzione rispetto al punto di partenza. Tale calcolo deve essere effettuato tramite una opportuna sottofunzione che prende in ingresso il vettore dei comandi.

Robot (continua)

2. Il programma invoca dalla funzione principale una procedura o funzione che visualizza sullo schermo di quanti metri si è spostato complessivamente il robot rispetto alla posizione originale ed in quale direzione (cioè le informazioni calcolate al punto 1).
Nell'esempio visto sopra, il programma dovrà visualizzare

2 METRI VERSO SUD

16 METRI VERSO EST

Robot (continua)

3. il programma riscrive il file comandi.txt, includendo alla fine una sequenza di istruzioni che riportano il robot al punto di partenza. Nell'esempio, verranno aggiunte due righe:

90	2
180	16

4. Si disegni il grafico del percorso del robot (facoltativo).

Esercizio 7 (la clinica)

Un programma MATLAB deve verificare i dati clinici di alcuni pazienti per determinare il dosaggio di un medicinale. Tale medicinale è dosato in base alla misurazione della pressione sistolica e diastolica.

Il file **misurazioni.txt** contiene un elenco di codici paziente, seguiti dalle misurazioni di pressione e dal dosaggio del medicinale (in grammi). Il contenuto del file e' separato da punto e virgola ed e' il seguente:

- codice paziente (stringa),
- pressione sistolica (numero),
- pressione diastolica (numero),
- dosaggio medicinale (numero).

La clinica (continua)

Il programma deve valutare per ogni paziente le misurazioni di pressione e calcolare la variazione nel dosaggio del medicinale.

In particolare se entrambe le misurazioni sono maggiori di 110 il dosaggio del medicinale deve essere aumentato, se invece la pressione diastolica è minore di 68 e la differenza tra le due è minore di 30 allora il medicinale deve essere diminuito.

Diversamente il dosaggio rimane invariato.

La clinica (continua)

Per fare questo, si usi il seguente algoritmo:

1. Si leggano i dati dal file (usando l'import wizard) e si dividano in opportune variabili (vettore dei codici paziente, matrice delle misurazioni e dosaggio medicinale). La funzione principale dovrà prendere in ingresso queste variabili.
(Si salvi il workspace in un file cartellecliniche.mat da consegnare).
2. Si scriva una sottofunzione "test_pressione_alta" che prende in ingresso due numeri (corrispondenti alle misurazioni di pressione sistolica e diastolica) e fornisce in uscita:
 - 1 nel caso in cui entrambi i numeri siano superiori a 110;
 - 0 altrimenti.

La clinica (continua)

3. Si scriva una sottofunzione **"test_pressione_bassa"** che prende in ingresso due numeri (corrispondenti alle misurazioni di pressione sistolica e diastolica) e fornisce in uscita:
 - 1 nel caso in cui la pressione diastolica sia minore di 68 e la differenza tra i due numeri sia inferiore a 30 (sistolica - diastolica < 30);
 - 0 altrimenti.

4. Nella funzione principale si esegua un ciclo e, per ogni paziente si calcoli la variazione del dosaggio usando le funzioni al punto 2 e 3. Se la sottofunzione **"test_pressione_alta"** restituisce 1 allora il dosaggio deve essere aumentato di 10 unità, se la sottofunzione **"test_pressione_bassa"** restituisce 1 allora il dosaggio deve essere diminuito di 20 unità, diversamente il dosaggio non deve essere modificato. Il nuovo valore del dosaggio deve sostituire quello già presente nella matrice delle misurazioni.

La clinica (continua)

5. Si modifichi il programma precedente perché visualizzi a video un messaggio rivolto al medico che notifichi i dettagli della modifica del dosaggio (solo nel caso di modifica).