

APPELLO SCRITTO DI PROGRAMMAZIONE 1
CORSO DI LAUREA IN MATEMATICA
UNIVERSITÀ DEGLI STUDI DI MILANO
2013–2014
20.XI.2014

VINCENZO MARRA

INDICE

Esercizio 1	2
Leggere i dati dal file e memorizzarli nella matrice.	2
Punti: 3.	2
Esercizio 2	2
Visualizzare la matrice.	2
Punti: 2.	2
Esercizio 3	2
Prima versione del <code>main</code> .	2
Punti: 5 punti.	2
Esercizio 4	3
Calcolo del massimo e del minimo valore.	3
Punti: 4 punti.	3
Esercizio 5	3
Calcolo del valor medio.	3
Punti: 4 punti.	3
Esercizio 6	3
Calcolo dell'indice di una colonna di norma massimale.	3
Punti: 6 punti.	3
Esercizio 7	3
Completare il <code>main</code> .	3
Punti: 6 punti.	4
<i>Istruzioni per la consegna</i>	4

Avvertenza. Il tema d'esame richiede lo sviluppo di un singolo programma che dia all'utente la possibilità di eseguire alcune semplici operazioni su una matrice di numeri in virgola mobile. Gli elementi della matrice sono acquisiti dal file `dati` allegato al tema d'esame. Svilupperete alcune funzioni ausiliarie nei primi esercizi. Integrerete poi le funzioni ausiliarie in un unico programma finale,

che consisterà di un unico file. Si raccomanda di leggere interamente il tema d'esame prima di cominciare a scrivere le soluzioni dei singoli esercizi, in modo da avere chiaro l'obiettivo finale.

Preliminare. Dichiarate all'inizio del file sorgente una costante di nome `DIM` e di valore 5. Usate la direttiva per il preprocessore `#define DIM 5`. Questo valore rappresenterà per tutto il programma la dimensione della matrice quadrata `DIMxDIM` su cui lavorare. Dichiarate poi come variabile globale accessibile a tutto il programma un array bidimensionale di dimensione `DIMxDIM`, di tipo `float` e di nome `M`. Tutte le funzioni del vostro programma avranno accesso alla costante `DIM` e alla variabile `M`.

ESERCIZIO 1

Leggere i dati dal file e memorizzarli nella matrice.

Punti: 3.

Scrivete una funzione di prototipo `int leggi dati(FILE* f)` che legga dal file ricevuto come parametro in ingresso i valori `float` da inserire nella matrice `M`, uno alla volta. La matrice va riempita riga per riga. Assumete che il file sia già aperto in lettura. Assumete che il file contenga valori `float` separati da spazi. Se la funzione riesce a leggere senza errori il numero corretto di valori `float`, restituisce 1. Altrimenti restituisce 0 per segnalare un errore. (*Suggerimento.* Usate `fscanf` nel modo opportuno.)

ESERCIZIO 2

Visualizzare la matrice.

Punti: 2.

Scrivete una funzione di nome `stampa` e di prototipo appropriato che visualizzi a video la matrice `M`. Visualizzate i valori `float` con una precisione di 2 cifre decimali. (*Suggerimento.* Usate la sequenza di controllo `%.2f`, aggiungendo eventualmente una tabulazione.)

ESERCIZIO 3

Prima versione del main.

Punti: 5 punti.

Scrivete una prima versione della funzione `main` seguendo le specifiche seguenti. In esercizi successivi completerete quanto scriverete qui. Usate quanto svolto negli esercizi precedenti.

All'avvio, il programma tenta di aprire in lettura il file il cui nome è passato al `main` come argomento dalla riga di comando. Se non vi sono argomenti sulla riga di comando, il programma termina con messaggio d'errore appropriato. Se vi sono più argomenti sulla riga di comando, il programma utilizza solo il primo. Il programma tenta di aprire in modalità di lettura il file indicato dall'utente come

argomento. Se l'apertura non va a buon fine, il programma termina con messaggio d'errore appropriato.

Altrimenti, il programma tenta di memorizzare nella matrice `M` i valori `float` letti dal file. Se la cosa non va a buon fine, il programma termina con messaggio d'errore appropriato. Se invece la lettura ha buon esito, il programma chiude il file, visualizza la matrice e termina. Per testare il programma usate il file `dati` allegato al tema d'esame.

ESERCIZIO 4

Calcolo del massimo e del minimo valore.

Punti: 4 punti.

Si scriva una funzione di nome `max` e di prototipo appropriato che calcoli il valore massimo fra gli elementi della matrice `M`, e lo restituisca al chiamante. Scrivete anche una funzione di nome `min` che faccia lo stesso per il valore minimo.

ESERCIZIO 5

Calcolo del valor medio.

Punti: 4 punti.

Si scriva una funzione di nome `media` e di prototipo appropriato che calcoli la media aritmetica degli elementi della matrice `M`, e la restituisca al chiamante.

ESERCIZIO 6

Calcolo dell'indice di una colonna di norma massimale.

Punti: 6 punti.

Si scriva una funzione di nome `maxcol` e di prototipo appropriato che restituisca l'indice (compreso fra 1 e `DIM` come nella pratica matematica, non fra 0 e `DIM - 1` come nella pratica informatica) di una colonna della matrice `M` che abbia norma massimale fra tutte le colonne. Si usi la norma euclidea: se $\vec{v} := (v_1, \dots, v_n)$ è un vettore in \mathbb{R}^n , con $n \geq 1$ un intero, la sua norma euclidea è

$$\|\vec{v}\| := \sqrt{\sum_{i=1}^n v_i^2}.$$

Per implementare `maxcol` si scriva una funzione ausiliaria `float normcol(int j)` che restituisce la norma della colonna j -esima della matrice `M`, dove, di nuovo, si intende $j \in \{1, \dots, \text{DIM}\}$. (*Suggerimento.* Per estrarre la radice quadrata si usi la funzione `float sqrtf(float)` definita nel file di intestazione `math.h`. Si ricordi inoltre che alcune implementazioni del compilatore C richiedono la compilazione con l'opzione `-lm` quando si usano funzioni matematiche come `sqrtf`.)

ESERCIZIO 7

Completare il main.

Punti: 6 punti.

Completate la funzione `main` che avete cominciato a scrivere nell'Esercizio 3, come specificato di seguito.

Dopo aver acquisito i valori dal file in ingresso e averli memorizzati nella matrice `M`, il programma mostra all'utente il menu seguente:

- a) Visualizza matrice.
- b) Calcola max e min.
- c) Calcola media.
- d) Calcola indice di una colonna di norma max.
- e) Esci.

Il programma legge quindi la scelta dell'utente, che deve essere un singolo carattere fra `a` ed `e`. (*Suggerimento.* Usate la funzione `getchar`, per semplicità. Ricordate di rimuovere dallo stream `stdin` con una seconda chiamata a `getchar` il carattere di nuova riga che l'utente digiterà dopo la sua scelta, affinché il programma funzioni correttamente.) Se l'utente specifica una scelta inesistente, il programma informa l'utente con un messaggio opportuno e torna a visualizzare il menu. Se l'utente sceglie una delle opzioni fra `a` e `d`, il programma visualizza le informazioni richieste, calcolandole tramite le funzioni sviluppate agli esercizi precedenti, e torna a visualizzare il menu. Se l'utente sceglie l'opzione `e` il programma termina.

Istruzioni per la consegna

- Funzioni da consegnare (in un unico file sorgente).

```
main()
leggidati()
stampa()
max()
min()
media()
normcol()
maxcol()
```

Tutte le eventuali altre funzioni ausiliarie
che abbiate scritto.

- Sito per la consegna:

<https://upload.mat.unimi.it>

- Autenticarsi con le proprie credenziali di posta elettronica d'ateneo.
- Consegnare solo il codice sorgente (file `*.c`), non l'eseguibile.

(V. Marra) DIPARTIMENTO DI MATEMATICA *Federigo Enriques*, UNIVERSITÀ DEGLI STUDI DI MILANO, VIA CESARE SALDINI, 50, I-20133 MILANO
E-mail address: vincenzo.marra@unimi.it