

Esercitazione 7



UNIVERSITÀ
DEGLI STUDI
DI GENOVA

Dibris

Gualtiero Volpe
gualtiero.volpe@unige.it

Soluzione Esercizi Sesta Esercitazione

Trovate le soluzioni di questi esercizi qui:

[https://github.com/gualtierovolpe/fondamenti di informatica 2023-24](https://github.com/gualtierovolpe/fondamenti_di_informatica_2023-24)

Le soluzioni delle esercitazioni vengono rilasciate il giorno prima dell'esercitazione successiva.



Un po' di teoria

Si scriva che cosa stamperà in uscita il seguente programma C++, spiegando brevemente il funzionamento delle funzioni test1 e test2:

```
#include <iostream>
using namespace std;

const int r = 3, c = 2;

void test1(int M[r][c], int a[r*c]) {
    for (int i = 0; i < r; i++)
        for (int j = 0; j < c; j++)
            a[r * j + i] = (M[i][j] % 2);
}

void test2(int M[r][c], int a[r*c]) {
    for (int k = 0; k < r * c; k++)
        if (a[k] == 1)
            M[k % r][k / r]++;
}

// stampa a video M e a
void stampa(int M[r][c], int a[r*c]);

int main() {
    int S[r][c] = {1, 6, 5, 4, 2, 3};
    int v[r*c] = {0};

    test1(S, v);
    stampa(S, v);

    test2(S, v);
    stampa(S, v);

    return 0;
}
```

Un po' di teoria

Identificare e correggere gli errori nella seguente serie di istruzioni:

```
#include <iostream>
using namespace std;

int r = 10; int c = 20;
int elabora(int M[r][c]);

int main() {
    int P[r][c] = {0};
    for (int i = 0; i <= r; i++)
        for (int j = 0; j <= c; j++)
            cin >> P[i, j];
            cout << elabora(P[r][c]) << endl;
    return 0;
}
```

Esercizi da Svolgere

Esercizio 7.1 – Prodotto di due matrici

Si scriva un programma in linguaggio C++ che chieda all'utente di immettere da tastiera gli elementi di due matrici quadrate di dimensione 3 e ne calcoli e stampi a video il prodotto righe per colonne. Implementare l'inserimento dei valori, il calcolo del prodotto righe per colonne e la stampa a video mediante tre opportune funzioni.

$$\begin{matrix} \begin{bmatrix} \mathbf{3} & \mathbf{5} & \mathbf{1} \\ 0 & 4 & 7 \\ 0 & 0 & 1 \end{bmatrix} & \cdot & \begin{bmatrix} \mathbf{1} & 4 & 7 \\ \mathbf{1} & 0 & 0 \\ \mathbf{2} & 6 & 9 \end{bmatrix} & = & \begin{bmatrix} \mathbf{10} & 18 & 30 \\ 18 & 42 & 63 \\ 2 & 6 & 9 \end{bmatrix} \\ \mathbf{A} & & \mathbf{B} & & \mathbf{C} \end{matrix}$$

Esercizi da Svolgere

Esercizio 7.1 – Prodotto di due matrici (continua)

Nota:

si definisce prodotto righe per colonne di una matrice $A = [a_{ik}]$ di ordine $n \times m$ per una matrice $B = [b_{kj}]$ di ordine $m \times q$, la matrice $C = [c_{ij}]$ di ordine $n \times q$ i cui elementi sono:

$$c_{ij} = \sum_{k=1}^m a_{ik} \cdot b_{kj}$$

Ad esempio, nel caso dell'esempio precedente si ha che:

$n = 3, k = 3, m = 3$ e $c_{11} = a_{11}b_{11} + a_{12}b_{21} + a_{13}b_{31}$ e così via.

Esercizi da Svolgere

Esercizio 7.2 – Traccia di una matrice

Si scriva una funzione in linguaggio C++ che ricevuta come parametro una matrice quadrata di numeri reali, calcoli e restituisca come valore di ritorno la sua traccia, ovvero la somma di tutti gli elementi della diagonale principale (un numero reale). Si scriva quindi un programma C++ per verificare il corretto funzionamento della funzione. E' responsabilità del programma chiamante passare alla funzione una matrice delle dimensioni corrette.

Esercizi da Svolgere

Esercizio 7.3 – Individuare la capolista di un campionato di calcio

Una matrice M è formata da $n \times m$ numeri interi, ognuno dei quali può valere soltanto 0, 1 o 3. La riga i -esima della matrice rappresenta i punti acquisiti dalla i -esima squadra di calcio nelle partite disputate nelle diverse giornate di un campionato. Il campionato comprende, quindi, n squadre e m giornate. In particolare, $M[i][k]$ denota il numero di punti conquistati dalla squadra i nella k -esima giornata. Si scriva in linguaggio C++ la funzione `capolista` che riceva come parametro la matrice M e un numero intero $0 \leq k < m$ e restituisca come valore di ritorno l'indice della squadra prima classificata alla k -esima giornata o -1 nel caso in cui sia $k \geq n$ (un numero intero). Per semplicità si assuma che i valori contenuti nella matrice siano validi. Si scriva quindi un programma per verificare il corretto funzionamento della funzione.

Esercizi da Svolgere

Esercizio 7.4 – Verifica dell'appartenenza ad un intervallo

Si scriva una funzione in linguaggio C++ che riceva come parametro una matrice di numeri reali M di 4 righe e 5 colonne, due numeri reali a e b , e un'ulteriore matrice di numeri interi P di due righe e lo stesso numero di colonne di M . La funzione calcola i valori degli elementi della matrice P nel modo seguente: gli elementi della prima riga di P contano quanti elementi di ciascuna colonna di M appartengono all'intervallo $[a, b]$ mentre gli elementi della seconda riga di P contano quanti elementi di ciascuna colonna di M non appartengono all'intervallo $[a, b]$. La funzione non restituisce alcun valore di ritorno. Si scriva quindi un programma per verificare il corretto funzionamento della funzione.

Esercizi da Svolgere

Esercizio 7.4 – Verifica dell'appartenenza ad un intervallo (continua)

Esempio:

$$\text{Data } M = \begin{bmatrix} 1.0 & -5.5 & 0.0 & 6.9 & 1.0 \\ 0.0 & 1.0 & 3.2 & 2.3 & 0.0 \\ 1.0 & -1.8 & 0.0 & 2.8 & 0.0 \\ 4.0 & 3.0 & -1.0 & 3.1 & 1.0 \end{bmatrix},$$

e $a = 0.0$ e $b = 1.0$, la funzione calcolerà

$$P = \begin{bmatrix} 3 & 1 & 2 & 0 & 4 \\ 1 & 3 & 2 & 4 & 0 \end{bmatrix}.$$

Esercizi da Svolgere

Esercizio 7.5 – Triangolo superiore e inferiore

Si scriva in linguaggio C++ la funzione `differenza_medie_triangoli` che riceva come parametro un matrice quadrata M di numeri interi di dimensione 5 (la matrice è cioè composta di 5 righe e 5 colonne) e calcoli e restituisca come valore di ritorno un numero reale che rappresenta il valore assoluto della differenza tra la media degli elementi appartenenti al triangolo superiore della matrice (diagonale compresa) e la media degli elementi appartenenti al triangolo inferiore della matrice (diagonale compresa).

Esercizi da Svolgere

Esercizio 7.5 – Triangolo superiore e inferiore (continua)

Esempio:

$$\text{Data } M = \begin{bmatrix} 2 & 8 & 0 & 1 & 3 \\ 1 & 1 & 6 & 4 & 0 \\ 0 & 5 & 2 & 9 & 1 \\ 1 & 2 & 3 & 0 & 3 \\ 0 & 0 & 1 & 1 & 2 \end{bmatrix},$$

la funzione calcolerà: media elementi triangolo superiore = 2.8
media elementi triangolo inferiore = 1.4

quindi il valore di ritorno sarà: differenza assoluta = $|2.8 - 1.4| = 1.4$.

Esercizi da Svolgere

Esercizio 7.5 – Triangolo superiore e inferiore (continua)

Si scriva quindi un programma che definisca una matrice quadrata Q di numeri interi di dimensione 5. Per 1000 volte, il programma assegna a ciascuno degli elementi della matrice Q un valore casuale compreso tra 0 e 9 (si può sviluppare una funzione per implementare tale operazione) e chiama la funzione `differenza_medie_triangoli`, ottenendone il valore di ritorno. Infine, il programma calcola e stampa a video la media di tali 1000 valori di ritorno.

Esercizi da Svolgere

Esercizio 7.6 – Prova intermedia del 8 gennaio 2018, esercizio 6

Si scriva un programma C++ che dichiari una matrice M di numeri reali di 4 righe e 3 colonne e due matrici X e Y di numeri reali, ciascuna di 2 righe e 3 colonne. Dopo avere chiesto all'utente di inserire da tastiera gli elementi della matrice M , il programma copierà nella matrice X le righe pari di M e nella matrice Y le righe dispari di M . Infine, il programma calcolerà e stamperà a video la somma degli elementi contenuti in ciascuna delle matrici X e Y .

Istruzioni Finali

Cercate di svolgere gli esercizi in autonomia/a piccoli gruppi
(è il miglior modo per vedere se avete capito)

Per ogni domanda, siamo a disposizione

Troverete le soluzioni di questi esercizi qui:

https://github.com/gualtierovolpe/fondamenti_di_informatica_2023-24

Le soluzioni delle esercitazioni vengono rilasciate il giorno prima dell'esercitazione successiva.
