

Laboratorio di programmazione e Informatica 1

- A.A. 2023-2024 -

Prova di laboratorio: Quarto appello - 19 luglio 2024

ISTRUZIONI: La prima riga di ogni programma C deve contenere il proprio nome e cognome. I programmi devono essere strutturati in funzioni e completi di commenti che spieghino il procedimento.

ATTENZIONE! Non saranno valutati programmi che non passano la fase di compilazione. Si consiglia pertanto di “mettere sotto commento” le parti di programma che danno errore in compilazione.

Ogni esercizio vale 15 punti. Per passare all'orale bisogna avere un punteggio ≥ 18 con almeno 8 punti in ogni esercizio.

Tempo a disposizione: 2:30 ore.

ESERCIZI

Esercizio1

Data una matrice M a valori interi, tre posizioni $(i, j), (i, h), (k, j)$ con $j < h$ e $i < k$ definiscono un triangolo se contengono lo stesso valore. A esempio, nella matrice qui sotto le posizioni segnate in grassetto $(1, 2), (1, 6), (2, 4)$ costituiscono un triangolo di valore 3.

1	3	2	4	4	4	5
3	2	3	5	2	2	3
2	1	4	2	4	4	3
5	5	3	5	2	2	1
1	1	4	2	4	4	2

1. Scrivere una funzione `int Triangolo(int A[ROW][COL], int *i, int *j, int *h, int*k)` che calcola se esiste un triangolo nella matrice A mettendo le sue coordinate nei corrispondenti parametri i, j, h, k e restituisce il valore che definisce il triangolo (in presenza di più triangoli, sceglierne uno a piacere). Se non esiste alcun triangolo restituisce 0.
2. Scrivere una funzione che genera una matrice di interi a valori nell'intervallo $[1, 9]$ in modo che non ci siano elementi uguali consecutivi nella stessa colonna (ogni elemento sia diverso da quello che gli sta sotto e che gli sta sopra)
3. Scrivere un programma in C (funzione main) che genera una matrice di interi random in $[1, 9]$ di dimensione 25×15 e la stampa ben formattata sullo schermo. Poi, richiamando la funzione `Triangolo`, calcola se esiste un triangolo e, in caso affermativo, ne stampa sullo schermo le posizioni e il valore.

4. Stampa nuovamente la matrice ben formattata sullo schermo scrivendo nelle posizioni calcolate del triangolo il valore 999. Nel caso dell'esempio sarebbe:

1	3	2	4	4	4	5
3	2	999	5	2	2	999
2	1	4	2	4	4	3
5	5	999	5	2	2	1
1	1	4	2	4	4	2

5. Analizzare la complessità dell'algoritmo utilizzato nella funzione **Triangolo**.

Esercizio 2

Sia L una lista di interi.

1. Si scriva una funzione **Raddoppia** che prende in input una lista L e restituisce una lista R di lunghezza doppia rispetto ad L in cui gli elementi della seconda metà sono esattamente uguali a quelli della prima metà.
Ad esempio se $L = [0, 1, 2, 3, 4]$ allora $R = [0, 1, 2, 3, 4, 0, 1, 2, 3, 4]$.
La funzione dovrà essere di complessità lineare nella lunghezza della lista.
2. Si scriva una funzione **Palindroma** che prende in input una lista L ed restituisce una lista P di lunghezza doppia rispetto ad L in cui gli elementi della seconda metà sono esattamente uguali a quelli della prima metà ma nell'ordine contrario.
Ad esempio se $L = [0, 1, 2, 3, 4]$ allora $P = [0, 1, 2, 3, 4, 4, 3, 2, 1, 0]$.
La funzione dovrà essere di complessità lineare nella lunghezza della lista. .

Inserire tali funzioni nel file **ListaBase.c**.