

Laboratorio di programmazione e Informatica 1

- A.A. 2020-2021 -

Prova di laboratorio: Primo Appello - 22 giugno 2021

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. Gli esercizi vanno consegnati come file singoli denominati rispettivamente `CognomeNomeEse1.c`, `CognomeNomeEse2.c`, `CognomeNomeEse3.c`.

L'analisi, ove richiesta, dell'algoritmo utilizzato va scritta alla fine del file `.C` come commento.

L'esercizio 3 consiste nello sviluppo di una funzione da inserire poi nel file `EseListaBaseApp1.c` fornito a parte.

Ogni esercizio vale un max di 10 punti. Si è ammessi all'orale se si raggiunge un punteggio totale ≥ 18 con un punteggio ≥ 5 sui singoli esercizi.

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. I file eseguibili non vanno consegnati.

Tempo a disposizione: 3 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
1	3	3	5	2	4	3
2	1	4	2	2	4	3
5	5	3	5	2	4	1
1	1	4	2	2	4	1

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 $[11, 44]$ 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 $[11, 44]$ 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 la stringa `TTT`. Nel caso dell'esempio sarebbe:

```

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

```

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

Esercizio 2

Si scriva un programma C che richiede all'utente di inserire una sequenza di nomi di persona. La sequenza termina quando l'utente inserisce tre volte consecutive lo stesso nome. Al termine dell'inserimento viene scritto sullo schermo il nome più lungo che è stato inserito.

Non è consentito utilizzare funzioni predefinite sulle stringhe.

Variante (+2 punti): Si aggiunge anche la possibilità di leggere i nomi da un file (in questo caso la terminazione avviene naturalmente quando finisce il file). Fornire anche il file di testo per provare il programma.

Esercizio 3

Sia L una lista di interi ordinata in cui sono eventualmente presenti elementi duplicati.

1. Si scriva una funzione `Popolare` che prende in input una lista L e restituisce l'elemento che compare il maggior numero di volte. In caso di parità si restituisca il maggiore tra tali numeri.
Ad esempio se $L = [0, 1, 1, 1, 2, 2, 3, 3, 3, 4]$ allora dovrà restituire il valore 3.
2. Si scriva una funzione `Cancella` che prende in input una lista L ed un valore x e cancella tutta sequenza di valori x presente nella lista. L'algoritmo implementato deve essere lineare nella lunghezza della lista.