

Laboratorio di programmazione e Informatica 1

- A.A. 2022-23 -

Prova di laboratorio: Secondo Appello - 11 luglio 2023

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 `ListeBase2.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.

Tempo a disposizione: 3 ore.

ESERCIZI

Esercizio 1

Data una matrice $n \times n$ a valori interi, la posizione (i_b, j_b) è un *baricentro* per la matrice se la somma di tutti i valori nelle posizioni dalla colonna 0 alla colonna j_b differisce al più di 10 dalla somma di tutti i valori nelle posizioni dalla colonna $j_b + 1$ alla colonna $n - 1$ e anche la somma dei valori nelle posizioni dalla riga 0 alla riga i_b differisce al più di 10 dalla somma dei valori nelle posizioni dalla riga $i_b + 1$ alla riga $n - 1$.

1. Scrivere una funzione in C che genera una matrice $n \times n$ a valori interi random tra 1 e 9 tali che non ci siano elementi consecutivi uguali sulla stessa riga e/o colonna.
2. Scrivere una funzione

```
int Baricentro(int a[][SIZE], int * x, int * y)
```

che, data una matrice restituisce 1 se esiste un baricentro e in quel caso ne mette le coordinate nelle variabili x e y passate per riferimento. In caso contrario, restituisce 0. In caso di presenza di più baricentri sceglierne uno.

3. Scrivere la funzione `main()` che prima richiama la funzione che genera la matrice per definire una matrice di interi di 20 righe e 20 colonne. Poi stampa la matrice ben formattata. Utilizzando la funzione `Baricentro` calcola se esiste un baricentro e, in caso affermativo, stampa sullo schermo le sue coordinate. Ristampa poi la matrice mettendo nella posizione del baricentro la stringa BB.
4. Analizzare la complessità dell'algoritmo per il calcolo del baricentro in funzione di n . Esiste un algoritmo $O(n^2)$? Giustificare la risposta.

Esercizio 2

Definisco una stringa $s1$ come *multianagramma* della stringa $s2$ se $s1$ contiene esattamente le stesse lettere di $s2$ eventualmente in numero diverso. Per esempio **morra** è un multianagramma di **roma**.

Scrivere un programma C che legge in input due stringhe $s1$ ed $s2$ e verifica se $s1$ è un multianagramma di $s2$. In caso negativo il programma dovrà riportare una lettera in cui differiscono. (Le stringhe in input possono contenere lettere, cifre, spazi bianchi e punteggiatura).

Esercizio 3

1. Si scriva una funzione in C che, prende in input (come parametri) una lista di interi $A = a_1, a_2, \dots, a_n$, e un intero x . Verifica se x è un elemento della lista. Supponiamo che $a_i = x$. In questo caso la funzione **modifica** la lista ruotandola in modo che a_i sia il primo elemento. La lista modificata diventa $a_i, a_{i+1}, \dots, a_n, a_1, a_2, \dots, a_{i-1}$. Se x non compare nella lista, invece sarà aggiunto in testa alla lista.

La funzione deve essere di complessità lineare rispetto alla dimensione della lista in input.

2. Inserire la funzione nel file `ListeBase2.c` e stampare i risultati.