

CORSO DI FONDAMENTI DI INFORMATICA

Prof. Salvatore Andolina - A.A. 2023/2024

Laboratorio di Programmazione in C - Laboratorio 4

Esercizio 1 - Esami 3.0

Modificare il programma del laboratorio precedente (**Esercizio 2 - Esami 2.0**).

Presentare a schermo il seguente menù:

- 1 - Calcolare media pesata dei voti
- 2 - Stampare i dati
- 3 - Ordinare i dati in ordine alfabetico per nome del professore e stampare
- 4 - Ordinare i dati per voto crescente e stampare
- 5 - Uscire

Completare il programma implementando le funzioni richieste dal menù.

NB: le opzioni 3 e 4 possono utilizzare bubble sort;

l'opzione 3 può utilizzare la funzione *strcmp* della libreria *string.h*

Esercizio 2 - String to int

Si scriva un programma che prenda in input da tastiera una stringa numerica e la converta nel corrispettivo valore intero. **Usare le funzioni!**

Controllare che la stringa inserita sia un numero valido.

Bonus: Considerare anche il caso di numeri negativi.

Esempio:

s = "1984"

output: 1984

s = "-1245"

output: -1245

Esercizio 3 - Matrice trasposta

Si scriva un programma che prenda in input una matrice di dimensione fissata (es. 3x3) e ne calcoli la matrice trasposta.

Esempio:

1 2 3		1 4 7
4 5 6	-->	2 5 8
7 8 9		3 6 9

Esercizio 4 - Numeri complessi

Si scriva un programma che prenda in input un numero complesso nelle parti Re e Im e se ne calcoli e stampi il modulo;

Estendere il programma per leggere due numeri complessi, e se ne calcoli la somma (anch'essa un numero complesso)

Estendere ancora il programma per leggere un array di MAX numeri complessi e si stampi il numero con modulo piu' grande.

Esercizio 5 - Sudoku

Si realizzi un programma che verifichi la corretta soluzione della griglia di un Sudoku. Un Sudoku è una griglia 9x9 che rispetta queste caratteristiche:

- tutti i valori devono essere singoli numeri tra 1 e 9
- su ciascuna delle 9 **righe** non devono esserci valori ripetuti
- su ciascuna delle 9 **colonne** non devono esserci valori ripetuti
- in ciascuno dei 9 **blocchi 3x3** che compongono la griglia non devono esserci valori ripetuti

L'esercizio è da risolvere svolgendo un livello dopo l'altro, secondo questa sequenza:

Livello 1 – Utilizzare il file *sudoku_template.c* fornito assieme a questo testo come punto di partenza: il file contiene la dichiarazione di due matrici Sudoku da verificare (una corretta e l'altra sbagliata). Il programma deve verificare la validità di queste due matrici e stampare a video il risultato del controllo. In questo livello il programma va scritto **usando le funzioni** (quindi completando solo il *main()* già esistente) e dividendo il problema assegnato in sottoproblemi.

Livello 2 – Modificare il programma già scritto in modo che la matrice da verificare venga acquisita **numero per numero**, e che venga eseguito il controllo di validità **già in fase di inserimento**.

(Bonus) Livello 3 – Modificare il programma già scritto in modo che la matrice da verificare venga letta dal file di testo *sudoku.txt* fornito assieme a questo testo.