



Avvertenze

- Usare **ESCLUSIVAMENTE** penne ad inchiostro nero o blu (**NO MATITE**).
- Consegnare solo fogli formato A4 scritti da **un solo lato**.
- In testa a ciascun foglio scrivere: **cognome, nome, numero progressivo di pagina rispetto al totale**; esempio per il secondo foglio di 3 consegnati: Giuseppe Russo 2/3
- Mantenere sul banco il libretto o altro documento di riconoscimento fino a controllo avvenuto
- Nient'altro deve trovarsi sul banco: non è consentito consultare libri, dispense, appunti, ecc.
- La correzione di riferimento per l'autovalutazione verrà fornita sul sito internet del Corso

Specifiche

Sono dati tre file di testo di nome “elenco1.txt”, “elenco2.txt” ed “elenco3.txt”, contenenti numeri interi. Ogni numero intero è seguito da un carattere newline. Il numero di interi presente in ogni file non è noto a priori.

Si chiede allo studente di realizzare un programma che prenda in ingresso i tre file e inserisca i numeri interi in essi contenuti in vettori di interi, uno per ogni file.

I vettori relativi ai file “elenco1.txt” ed “elenco2.txt” dovranno quindi essere uniti in un unico vettore, che comprenda tutti gli interi contenuti nei vettori originari. Da questo vettore si dovranno eliminare in seguito i valori contenuti nel vettore relativo al file “elenco3.txt”.

Il vettore risultante in seguito a quest'ultimo passaggio dovrà essere ordinato tramite algoritmo bubblesort e memorizzato in un file di nome “risultato.txt”. Al termine del processo, dunque, questo file conterrà tutti i valori interi elencati nei file “elenco1.txt” ed “elenco2.txt”, ad eccezione di quelli elencati in “elenco3.txt”.

Prototipi delle funzioni da implementare

```
// PUNTI 2 - Questa funzione conta le righe del file.
int contaRighe ( char nomeFile[] );

// PUNTI 3 - Questa procedura carica i dati contenuti nel file nel vettore.
void caricaVettore ( char nomeFile[], int vettore[] );

// PUNTI 3 - Questa procedura ordina il vettore.
void ordina ( int vettore[], int dimensione );

// PUNTI 3 - Questa procedura prende i dati contenuti in vettoreIn1 e in vettoreIn2 e li inserisce
in vettoreSum.
void fondi ( int vettoreSum[], int vettoreIn1[], int vettoreIn2[], int dimVett1, int dimVett2 );

// PUNTI 3 - Questa procedura elimina i dati contenuti in vettDel da vettSum e mette i dati
rimanenti in vettOut.
void pulisci ( int vettOut[], int vettSum[], int vettDel[], int dimVettSum, int dimVettDel );

// PUNTI 2 - Questa procedura salva il vettore risultante in un file.
void salvaFile ( char nomeFile[], int vettore[], int dim );

// PUNTI 1 - Questa procedura scambia gli elementi a e b di un vettore.
void scambiaElementi( int &a, int &b );

// PUNTI 1 - Il main.
int main ( )
```



Esempio di esecuzione

File “elenco1.txt”

4471
7824
4390
4823
507
4897
5900
5827
2829
1899

File “elenco2.txt”

4468
10245
30001
3432
123
3
4481
23666

File “elenco3.txt”

7824
4481
4897
1899

File “risultato.txt”

3
123
507
2829
3432
4390
4468
4471
4823
5827
5900
10245
23666
30001