

Esercizi sui file di testo — avanzati

File legati a vettori, matrici e struct in C++

Prima di iniziare. Qui i file (.txt, .dat, .csv) sono collegati a vettori, matrici e struct. In un CSV i campi sono separati da virgole. Dichiarare i vettori con dimensione massima (`const int MAX = 100;`) e lavorare con il numero di record letti.

Esercizio 1 — Vettore su dat

Salva un vettore di interi su un file .dat e poi rileggilo in un nuovo vettore stampandolo.

file dat + vettore

Prototipi suggeriti

```
void salvaVettore(string nomeFile, int v[], const int N);  
int caricaVettore(string nomeFile, int v[]);
```

Esercizio 2 — Matrice su file

Salva una matrice su file e poi rileggila, stampandola.

file + matrice

Prototipi suggeriti

```
void salvaMatrice(string nomeFile, int m[][C], const int R);  
void caricaMatrice(string nomeFile, int m[][C], const int R);
```

Esercizio 3 — Struct su CSV

Salva un vettore di studenti (nome e voto) su un file CSV `nome,voto`.

csv + struct

Struct e prototipi suggeriti

```
struct Studente { string nome; float voto; };  
void salvaCsv(string nomeFile, Studente s[], const int N);
```

Esercizio 4 — Media da CSV

Carica da un CSV `nome,voto` i record in un vettore di struct e calcola la media dei voti.

csv + struct + float

Struct e prototipi suggeriti

```
struct Studente { string nome; float voto; };  
int caricaCsv(string nomeFile, Studente s[]);  
float media(Studente s[], const int N);
```

Esercizio 5 — Esporta i promossi

Leggi gli studenti da un CSV e scrivi su un nuovo CSV solo quelli con voto maggiore o uguale a 6.

csv + struct · filtro

Struct e prototipi suggeriti

```
struct Studente { string nome; float voto; };
int caricaCsv(string nomeFile, Studente s[]);
void salvaPromossi(string nomeFile, Studente s[], const int N);
```

Esercizio 6 — Classifica su file

Leggi i giocatori (nome e punteggio) da un CSV, ordinali per punteggio decrescente e scrivi la classifica su un nuovo CSV.

csv + struct · ordinamento

Struct e prototipi suggeriti

```
struct Giocatore { string nome; int punteggio; };
int caricaCsv(string nomeFile, Giocatore g[]);
void ordina(Giocatore g[], const int N);
void salvaCsv(string nomeFile, Giocatore g[], const int N);
```

Esercizio 7 — Report statistico

Leggi una serie di numeri reali da un file .dat in un vettore e scrivi su un file di report minimo, massimo e media.

file dat + vettore

Prototipi suggeriti

```
int caricaNumeri(string nomeFile, float v[]);
void salvaReport(string nomeFile, float v[], const int N);
```

Esercizio 8 — Conteggio per genere

Leggi i brani (titolo e codice di genere) da un CSV e scrivi su un nuovo file quanti brani ci sono per ciascun genere.

csv + struct · conteggio

Struct e prototipi suggeriti

```
struct Brano { string titolo; int genere; };
int caricaCsv(string nomeFile, Brano b[]);
void salvaConteggio(string nomeFile, Brano b[], const int N, const int G);
```

Esercizio 9 — Unisci due CSV

Unisci due file CSV (con la stessa struttura) in un unico file CSV.

csv

Prototipo suggerito

```
void unisciCsv(string file1, string file2, string destinazione);
```

Esercizio 10 — Aggiorna magazzino

Leggi i prodotti (nome, prezzo, scorta) da un CSV, aumenta la scorta di quelli sotto soglia e riscrivi un CSV aggiornato.

csv + struct · modifica

Struct e prototipi suggeriti

```
struct Prodotto { string nome; float prezzo; int scorta; };  
int caricaCsv(string nomeFile, Prodotto p[]);  
void rifornisci(Prodotto p[], const int N, int soglia, int aggiunta);  
void salvaCsv(string nomeFile, Prodotto p[], const int N);
```