

Esercitazione 8



UNIVERSITÀ
DEGLI STUDI
DI GENOVA

| Dibris

Gualtiero Volpe
gualtiero.volpe@unige.it

Soluzione Esercizi Settimana Esercitazione

Trovate le soluzioni di questi esercizi qui:

https://github.com/gualtierovolpe/fondamenti_di_informatica_2024-25

Le soluzioni delle esercitazioni vengono rilasciate il giorno prima dell'esercitazione successiva.

Un po' di teoria

Dato il seguente programma C++ e il contenuto del file Dati.txt, si scrivano il contenuto della matrice M passata alla funzione test nel momento della chiamata e i valori di r e c che il programma stamperà a video.

```
#include <iostream>
#include <fstream>
using namespace std;

int test(int A[3][3], int& n)
{   int s = 0; n = 1;
    for (int h = 0; h < 3; h++) {
        s += A[h][h];
        n *= A[h][h];
    }
    return s;
}

int main() {
    int M[3][3] = {0};
    fstream fin;
    fin.open("Dati.txt", ios::in);
    int i = 0; int j = 0; int c = 0;
    while (!fin.eof()) {
        fin >> M[i][j];
        c++; i = c % 3;
        j += ((c % 3) == 0);
    }
    int r = test(M, c);
    cout << r << c;
    fin.close(); return 0; }
```

Contenuto del file Dati.txt: 3 0 6 3 2 1 2 8 4

Un po' di teoria

Identificare e correggere gli errori nella seguente serie di istruzioni:

```
1      #include <fstream>
2      using namespace std;
3      int main() {
4          fstream f; int a[100] = {0}; int i = 0;
5          f.open("Test.txt", ios:in);
6          while (f.eof() && i < 100) {
7              f << a[i]; i++;
8          }
9          f.open();
10         return 0; }
```

Esercizi da Svolgere

Esercizio 8.1 – Lettura di un file di testo di lunghezza non nota

Si scriva un programma in linguaggio C++ che apra in lettura il file "ciao.txt" e generi come output due file chiamati "consonanti.txt" e "vocali.txt". Il file "ciao.txt" contiene un testo di lunghezza non nota al programmatore che il programma leggerà carattere per carattere. Il programma inserirà nel file "consonanti.txt" tutte e sole le consonanti presenti nel file "ciao.txt" e nel file "vocali.txt" tutte e sole le vocali presenti nel file "ciao.txt". Al termine della lettura del file "ciao.txt", il programma stamperà a video il numero complessivo di caratteri letti.

Esempio: se il file "ciao.txt" contiene il testo "prova a fare l'esercizio", il file "consonanti.txt" conterrà i caratteri "prvfrlsrcz" e il file "vocali.txt" i caratteri "oaaaeieiio".

Esercizi da Svolgere

Esercizio 8.2 – Lettura di un file di testo di lunghezza nota

Un'azienda di lavori stradali opera su diverse strade e desidera tenere traccia dei lavori eseguiti. A tal fine si scriva un programma C++ che operi come segue: dichiarare un array km di dieci numeri reali e lo inizializzi in modo che tutti i suoi elementi valgano zero; apra il file "Lavori.txt", tale file contiene per ciascuna riga il numero della strada oggetto del lavoro (un numero intero), il chilometro di inizio lavori (un numero reale) e di fine lavori (un numero reale), il costo in Euro dei lavori (un numero reale), il file contiene dieci righe; per ogni riga del file, inserisca nell'array km la lunghezza della tratta interessata dai lavori, calcoli il costo totale dei lavori e la lunghezza totale delle tratte interessate dai lavori; stampi a video il costo medio a chilometro, dividendo il costo totale per la lunghezza totale delle tratte interessate. Si supponga che i valori contenuti nel file siano validi.

Esercizi da Svolgere

Esercizio 8.3 – **Conteggio delle occorrenze dei caratteri in un file**

Si scriva un programma in linguaggio C++ che apra in lettura il file "input.txt" e generi come output un file chiamato "output.txt". Il file "input.txt" contiene un testo di lunghezza non nota al programmatore che il programma leggerà carattere per carattere. Il file "output.txt" è formato da tante righe quante sono le lettere dell'alfabeto anglosassone (quindi 26 righe) e riporta per ciascuna riga una lettera e quante volte tale lettera compare nel file "input.txt".

Suggerimento: si può definire un array di 26 numeri interi, inizializzandolo a zero. Ogni volta che si legge un carattere, si incrementa l'elemento dell'array nella posizione corrispondente (il primo per il carattere 'a', il secondo per 'b' e così via).

Esercizi da Svolgere

Esercizio 8.4 – **Lettura di un file contenente un numero variabile di dati per ciascuna riga**

Il file "dati.txt" contiene le età in anni dei membri di un certo numero di gruppi di lavoro operanti all'interno di un'azienda. Ciascuna riga del file contiene le seguenti informazioni: un codice identificativo del gruppo (una stringa consistente di una sola parola contenente al massimo 16 caratteri), il numero di membri del gruppo (un numero intero), le età in anni di ciascun membro del gruppo (un numero intero per ciascun membro). Il numero di righe di cui il file si compone non è noto al programmatore. Si scriva un programma in linguaggio C++ che apra in lettura il file "dati.txt", lo scandisca e stampi a video l'identificativo di ciascun gruppo seguito dall'età media dei suoi componenti. Si supponga, per semplicità, che i valori contenuti nel file siano validi.

Esercizi da Svolgere

Esercizio 8.5 – **Lettura di un file indicato dall'utente**

Alcuni sensori di temperatura scrivono le loro misure (dei numeri reali) su file il cui nome è la località in cui sono installati (ad esempio "genova.txt"). Il numero di misure trascritte è diverso per ciascun file e non è noto al programmatore. Si scriva un programma in linguaggio C++ che chieda all'utente di immettere da tastiera il nome del file che desidera analizzare, apra il file indicato, lo scandisca e stampi a video la temperatura massima e minima registrata dal sensore corrispondente. Se il file non esiste, il programma chiede all'utente di immettere di nuovo il nome del file. L'utente può ripetere le operazioni quante volte desidera e può uscire dal programma digitando la parola "exit", anziché il nome di un file. Prima di terminare, il programma stampa a video il numero di file non vuoti elaborati. Si supponga che i valori contenuti nei file siano validi.

Esercizi da Svolgere

Esercizio 8.6 – Prova intermedia del 5 febbraio 2019, esercizio 5

Un'agenzia di spedizioni organizza quasi quotidianamente la movimentazione di merci tra due città collegate da una ferrovia. La merce viene collocata su di un treno che può essere composto da un massimo di 10 vagoni. Per rappresentare i dati relativi a tale attività nel corso di un mese, si può utilizzare una matrice T di numeri reali di 31 righe e 10 colonne, dove le righe rappresentano i giorni di un mese e le colonne i vagoni del treno. L'elemento T_{ij} della matrice T rappresenta il numero di tonnellate di merce caricata su di uno specifico vagone (il j -esimo vagone) in un giorno specifico (l' i -esimo giorno). Nel caso in cui il treno in un certo giorno fosse composto da meno di 10 vagoni, gli elementi restanti della riga corrispondente della matrice T sono impostati a zero. Se in un certo giorno il treno non ha viaggiato, tutti gli elementi della riga corrispondente della matrice T sono impostati a zero (nello stesso modo si gestiscono i mesi che hanno meno di 31 giorni).

Esercizi da Svolgere

Esercizio 8.6 – **Prova intermedia del 5 febbraio 2019, esercizio 5** (continua)

Per elaborare tali dati, si sviluppi quanto segue in linguaggio C++:

1. La funzione `totaleMerce` che riceva come parametro la matrice T sopra descritta e restituisca come valore di ritorno il numero totale di tonnellate di merce movimentata nel corso del mese (un numero reale). La funzione calcolerà e restituirà, quindi, la somma dei valori degli elementi della matrice T . Per semplicità si assuma che i valori assunti dagli elementi della matrice T siano sempre validi.

Esercizi da Svolgere

Esercizio 8.6 – Prova intermedia del 5 febbraio 2019, esercizio 5 (continua)

2. La funzione `numTreni` che riceva come parametro la matrice T sopra descritta e restituisca come valore di ritorno il numero totale di treni che hanno viaggiato nel corso del mese (un numero intero). La funzione calcolerà e restituirà, quindi, il numero di righe della matrice T che contengono almeno un elemento diverso da zero. Per semplicità si assuma che i valori assunti dagli elementi della matrice T siano sempre validi.

Esercizi da Svolgere

Esercizio 8.6 – Prova intermedia del 5 febbraio 2019, esercizio 5 (continua)

3. La funzione maxTreno che riceva come parametro la matrice T sopra descritta, abbia come parametri di uscita (ovvero per riferimento) due numeri interi v e g e restituisca come valore di ritorno un numero reale. La funzione assegna a v il numero di vagoni del treno composto dal maggior numero di vagoni, assegna a g l'indice del giorno in cui tale treno ha viaggiato e restituisce come valore di ritorno il numero di tonnellate di merce che tale treno ha trasportato. Se vi è più di un treno con lo stesso numero massimo di vagoni, la funzione restituisce i dati del treno che ha viaggiato per primo (ovvero il primo che viene trovato scandendo le righe della matrice T dalla prima all'ultima). Per semplicità si assuma che i valori assunti dagli elementi della matrice T siano sempre validi.

Esercizi da Svolgere

Esercizio 8.6 – Prova intermedia del 5 febbraio 2019, esercizio 5 (continua)

4. Un programma che operi come segue: dichiarare e inizializzare a zero una matrice `Treni` di numeri reali di 31 righe e 10 colonne organizzata come sopra descritto; aprire in lettura il file `"Dati.txt"`: il file è composto di 31 linee, ciascuna contenente 10 numeri reali (per semplicità si supponga che il file esista e che il suo contenuto sia sempre valido); scandire il file e assegnare a ciascun elemento della matrice `Treni` il corrispondente valore contenuto nel file; chiamare la funzione `totaleMerce` e stampare a video il suo valore di ritorno; chiamare la funzione `numTreni` e stampare a video il suo valore di ritorno; chiamare la funzione `maxTreno` e stampare a video il suo valore di ritorno e i valori assunti dai parametri passati per riferimento dopo la chiamata della funzione.

Istruzioni Finali

Cercate di svolgere gli esercizi in autonomia/a piccoli gruppi
(è il miglior modo per vedere se avete capito)

Per ogni domanda, siamo a disposizione

Troverete le soluzioni di questi esercizi qui:

https://github.com/gualtierovolpe/fondamenti_di_informatica_2024-25

Le soluzioni delle esercitazioni vengono rilasciate il giorno prima dell'esercitazione successiva.

