

Laboratorio di programmazione per sistemi ciberfisici

11. Esercizi di riepilogo

Enrico Martini
January 13, 2026

Esercizio 1

Una piccola compagnia aerea vi ha chiesto di sviluppare un nuovo sistema per assegnare i posti su ogni volo dell'unico aereo a disposizione (capacità: 10 posti). Il programma dovrà visualizzare un menu in cui, se il cliente digita 1, prenoterà un posto nella prima classe dell'aereo (posti da 1 a 5); analogamente, se il cliente digita 2, prenoterà un posto nella classe economica (posti da 6 a 10). Il programma dovrà quindi stampare una carta d'imbarco che indichi il numero del posto assegnato e la relativa classe (prima o economica). Si utilizzi un array per rappresentare i posti dell'aereo: esso dovrà essere inizialmente azzerato (per indicare che tutti i posti sono vuoti). A mano a mano che vengono effettuate le prenotazioni dei posti, l'elemento corrispondente dell'array dovrà essere impostato a 1 (per indicare che il posto non è più disponibile).

Esercizio 2 - Esame 05/02/2015

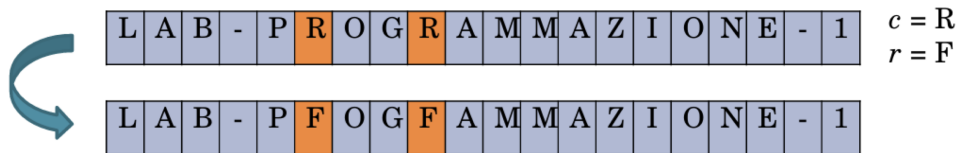
Si consideri un alfabeto di quattro simboli, dato dai caratteri A,C,G e T. Si scriva un programma C composto da:

- ▶ Una funzione `leggi` che richiede all'utente di inserire 10 caratteri e li memorizza in un array. I caratteri validi sono solo quelli dell'alfabeto descritto sopra.
- ▶ Una funzione `matrice_frequenze` che conta il numero di volte in cui ogni simbolo dell'alfabeto è seguito da ciascun altro simbolo e restituisce i risultati in una matrice 4x4.
- ▶ Una funzione `max` che prende in input la matrice e restituisce il valore massimo.
- ▶ Una funzione `min` che prende in input una matrice e restituisce il valore minimo.
- ▶ Una funzione `media` che prende in input la matrice e restituisce la media dei suoi elementi.
- ▶ Una funzione `vettore` che prende in input la matrice e restituisce un array monodimensionale contenente tutti i valori della matrice ordinati in ordine crescente.
- ▶ Una funzione `main` che richiama le altre funzioni e stampa i risultati.

Esercizio 3 - Esame 24/09/2012

Scrivere un programma C che:

- ▶ Richiede all'utente di inserire 20 caratteri che memorizza in un array, e altri due caratteri *c* e *r*.
- ▶ Restituisce una stringa costruita a partire dall'array iniziale in cui ogni occorrenza di *c* è sostituita con *r*.



Esercizio 4 - Esame 05/02/2010

Scrivere un programma C che:

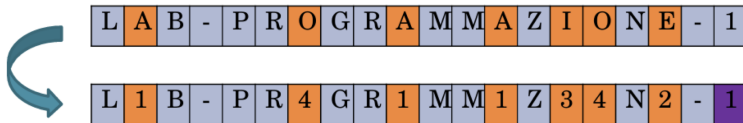
- ▶ Richiede all'utente di inserire 20 numeri interi a piacere e di salvarli in un array.
- ▶ Crea e stampa un array risultato in cui:
 - ▶ Ogni elemento in posizione i pari è uguale alla somma degli elementi dell'array originale in posizione pari da 0 a i .
 - ▶ Ogni elemento in posizione j dispari è uguale alla somma degli elementi dell'array originale in posizione dispari da 1 a j .



Esercizio 5 - Esame 05/02/2010

Scrivere un programma C che:

- ▶ Richiede all'utente di inserire 20 caratteri e di salvarli in un array.
- ▶ Produce due risultati:
 - ▶ Un array costruito a partire dai caratteri inseriti, in cui ogni vocale (maiuscola o minuscola) è sostituita da un numero corrispondente (A=1, E=2, I=3, O=4, U=5).
 - ▶ Un numero che rappresenta la somma delle cifre (caratteri da 0 a 9) presenti nella stringa prodotta.



Somma = **17**

Esercizio 6

Scrivere un programma C che:

- ▶ Richiede all'utente di inserire 10 double
- ▶ Ne calcola la media
- ▶ Trova il più grande dei valori che stanno sotto la media e il più piccolo di quelli che stanno sopra la media

Esempio: se l'utente immette: 1, 7, 9, 2, 5, 8, 3, 8, 8, 7, la media è 5,8. Il più grande dei valori che ci stanno sotto è 5; il più piccolo dei valori che ci stanno sopra è 7.

Esercizio 7

Scrivere un programma C che:

- ▶ Richiede all'utente di inserire 10 caratteri.
- ▶ Stampa a video i caratteri immessi, nello stesso ordine di immissione ma senza duplicazioni.

Esempio: se l'utente immette: *b, d, a, b, g, a, f, a, a, t*, l'output deve essere: *b, d, a, g, f, t*.

Esercizio 8

Scrivere un programma che, data una matrice `int M[D1][D2]` e un valore intero `x` inserito da tastiera, conti quante volte `x` appare in `M`.

Esercizio 9

Scrivere un programma che conti quanti elementi in una matrice compaiono almeno due volte.

Esercizio 10

In un concorso di bruttezza, n giudici esprimono il loro giudizio su m candidati. Il giudizio è un valore numerico compreso tra 0 e 5 (più alto è il voto, più brutto è il candidato). I risultati sono organizzati in una matrice `int R[N][M]`. Scrivere un programma in linguaggio C per determinare il candidato più brutto ed il giudice più cattivo (ovvero quello che attribuisce i più alti voti di bruttezza)

Esercizio 11

Data una matrice quadrata di interi $M[N][N]$, copiare tutti i suoi elementi in un vettore di lunghezza N^2 secondo l'ordine mostrato in figura.

A 6x6 grid containing numbers from 1 to 36. The numbers are arranged as follows:

1	2	6	7	15	16
3	5	8	14	17	26
4	9	13	18	25	27
10	12	19	24	28	33
11	20	23	29	32	34
21	22	30	31	35	36

Blue arrows indicate the following connections:

- 1 → 2 (horizontal)
- 2 → 6 (diagonal down-right)
- 6 → 7 (horizontal)
- 7 → 15 (diagonal up-right)
- 15 → 16 (horizontal)
- 16 → 26 (diagonal down-right)
- 26 → 27 (vertical down)
- 27 → 33 (diagonal down-left)
- 33 → 34 (vertical down)
- 34 → 36 (diagonal down-left)
- 36 → 35 (diagonal up-left)
- 35 → 31 (diagonal up-left)
- 31 → 30 (horizontal)
- 30 → 22 (diagonal up-left)
- 22 → 21 (horizontal)
- 21 → 11 (vertical down)
- 11 → 10 (vertical down)
- 10 → 4 (diagonal down-left)
- 4 → 3 (vertical up)
- 3 → 5 (diagonal up-right)
- 5 → 8 (diagonal up-right)
- 8 → 14 (diagonal up-right)
- 14 → 17 (diagonal up-right)
- 17 → 25 (diagonal up-right)
- 25 → 28 (diagonal up-right)
- 28 → 32 (diagonal up-right)
- 32 → 29 (diagonal up-left)
- 29 → 23 (diagonal up-left)
- 23 → 19 (diagonal up-left)
- 19 → 12 (diagonal up-left)
- 12 → 10 (diagonal up-left)