

# Esercitazione 9



UNIVERSITÀ  
DEGLI STUDI  
DI GENOVA

Dibris

Gualtiero Volpe  
[gualtiero.volpe@unige.it](mailto:gualtiero.volpe@unige.it)

# Soluzione Esercizi Ottava Esercitazione

Trovate le soluzioni di questi esercizi qui:

[https://github.com/gualtierovolpe/fondamenti\\_di\\_informatica\\_2023-24](https://github.com/gualtierovolpe/fondamenti_di_informatica_2023-24)

# Simulazione della Prima Prova Intermedia

Si terrà **mercoledì 20 dicembre 2023**.

La prova sarà disponibile **a partire dalle 8:00 e fino alle 23:00**.

La prova si potrà sostenere accedendo alla **piattaforma AulaWeb degli esami** a questo link:

<https://esami2.aulaweb.unige.it/>

Il docente sarà disponibile presso **l'aula E3 dalle 15:00 alle 16:30 e su MS Teams**.

Chi volesse effettuare la simulazione in aula, deve portare con sé un computer portatile.

La prova non viene valutata, ma gli studenti potranno autovalutare il loro elaborato confrontando le loro risposte con le soluzioni fornite.

# Un po' di teoria

## Prova intermedia del 9 gennaio 2017, esercizio 2

*Si scriva la dichiarazione delle seguenti variabili e costanti C++:*

1. Una variabile di tipo intero inizializzata al valore 5: \_\_\_\_\_
2. Una stringa di 63 caratteri di nome str: \_\_\_\_\_
3. Una matrice di 50x40 numeri reali: \_\_\_\_\_
4. Un array di 32 numeri interi di nome a: \_\_\_\_\_
5. Una costante reale inizializzata al valore 6.28: \_\_\_\_\_
6. Una variabile di tipo reale inizializzata al valore 3.5: \_\_\_\_\_
7. Una variabile di tipo carattere non inizializzata: \_\_\_\_\_
8. Una struttura i cui campi sono un carattere, un numero reale e una matrice di 20x5 numeri interi: \_\_\_\_\_

# Un po' di teoria

*Si scriva la dichiarazione (ovvero soltanto il prototipo) delle seguenti funzioni C++:*

1. La funzione *test1* che riceve come parametri per valore un numero intero e un carattere alfanumerico e restituisce come valore di ritorno un numero reale:
2. La funzione *test2* che riceve come parametro per riferimento un numero intero e come parametro per valore un numero reale e restituisce come valore di ritorno un numero intero:

# Un po' di teoria

3. La funzione *test3* che riceve come parametri una matrice di 15x12 numeri interi, un array di 12 numeri reali e, per valore, un carattere alfanumerico e restituisce come valore di ritorno un numero reale:
4. La funzione *test4* che riceve come parametri due stringhe di caratteri s e t e, per riferimento, un carattere alfanumerico e non restituisce alcun valore di ritorno
5. La funzione *test5* che riceve come parametri un array di strutture di tipo libro, la sua dimensione n (per valore) e una stringa di caratteri t e restituisce come valore di ritorno una struttura di tipo libro:

# Un po' di teoria

*Si completino i seguenti frammenti di codice C++:*

```
1 // prodotto degli elementi sulla
  // diagonale di una matrice
  _____ int r = 6, c = 6;
double prodotto(double M[r][c])
{ double prodotto = 1.0;
  for (int k = 0; _____; k++)
    prodotto _____ M[k][k];
  return prodotto; }
```

```
3 // somma dei numeri interi
  // contenuti in un file
double somma = _____;
fstream fin; int x;
fin._____("Test.txt", ios::in);
while (_____) {
  fin >> x; somma += x;}
```

```
2 // numero di caratteri numerici
  // e alfabetici minuscoli in una stringa
void analizza(char s[], _____ n, _____ a)
{ for (int i = 0; _____; i++) {
  if (s[i] >= 'a' _____ s[i] <= 'z')
    a++;
  if (s[i] >= '0' _____ s[i] <= '9')
    n++; } }
```

```
4 // verifica se un punto inserito da
  // tastiera appartiene al primo quadrante
struct Punto2D {double x; double y;};
Punto2D a; cin >>_____ >> _____;
if (_____ >= 0 && _____ >= 0)
  cout << "Appartiene" << endl;
  _____ cout << "Non appartiene" << endl;
```

# Un po' di teoria

*Identificare e correggere gli errori nella seguente serie di istruzioni:*

```
1      struct test (  
2          int x, y;  
3      );  
4  
5      test s;  
6      x.s = 2;  
7      y.s = 3;  
8  
9      if (x.s = 0 && y.s = 0)  
10         cout << "Entrambi nulli";  
11      else (x.s > 0 && y.s > 0)  
12         cout << "Entrambi positivi";
```



# Esercizi da Svolgere

## Esercizio 9.1 – Risolutore di equazioni di secondo grado (rivisitato)

*Si rielabori il programma già sviluppato per la risoluzione di una equazione di secondo grado (Esercizio 3.1) in questo modo:*

- Si definisca una struttura equazione che abbia come campi i tre coefficienti  $a$ ,  $b$  e  $c$  (tre numeri reali) di una equazione di secondo grado nella forma  $ax^2 + bx + c = 0$ .*
- Si definisca una struttura soluzione che abbia come campi le due soluzioni  $x_1$  e  $x_2$  (due numeri reali).*
- Si definisca la funzione risolvi che riceva come parametro in ingresso una struttura di tipo equazione, come parametro di uscita una struttura di tipo soluzione e restituisca come valore di ritorno 1 se l'equazione ha soluzioni reali e 0 altrimenti.*
- Si rielabori la funzione main in modo da utilizzare le strutture e la funzione definite sopra.*

# Esercizi da Svolgere

## Esercizio 9.2 – Calcolo del centro di una circonferenza

*Al fine di sviluppare un programma C++ per elaborare dati relativi a circonferenze:*

- Si definisca la struttura punto2d: la struttura avrà come campi due numeri reali  $x$  e  $y$ .*
- Si definisca la struttura circonferenza: la struttura avrà come campi tre numeri reali  $a$ ,  $b$ ,  $c$ , che costituiscono i coefficienti dell'equazione  $x^2 + y^2 + ax + by + c = 0$ .*
- Si scriva la funzione centro che riceva come parametro una struttura di tipo circonferenza e restituisca come valore di ritorno una struttura di tipo punto2d. La funzione calcolerà e restituirà il centro  $C$  della circonferenza calcolato come  $x_C = -a/2$ ,  $y_C = -b/2$ .*
- Si scriva, infine, la funzione main che chieda all'utente di inserire da tastiera, almeno una volta e finché lo desidera, i coefficienti  $a$ ,  $b$  e  $c$  dell'equazione di una circonferenza, chiami la funzione elabora e stampi a video le coordinate  $(x_C, y_C)$  del centro della circonferenza.*

# Esercizi da Svolgere

## Esercizio 9.3 – Calcolo della distanza punto-retta

*Al fine di sviluppare un programma C++ per calcolare la distanza tra un punto e una retta:*

- Si utilizzi la struttura punto2d definita nell'esercizio 9.2.*
- Si definisca la struttura retta: la struttura avrà come campi tre numeri reali  $a$ ,  $b$ ,  $c$ , che costituiscono i coefficienti dell'equazione  $ax + by + c = 0$ .*
- Si scriva la funzione distanza che riceva come parametri una struttura di tipo punto e una di tipo retta, calcoli e restituisca la distanza tra il punto e la retta (un numero reale).*
- Si scriva, infine, la funzione main che chieda all'utente di inserire da tastiera, almeno una volta e finché lo desidera, i coefficienti  $a_R$ ,  $b_R$  e  $c_R$  di una retta (verificandone la validità, cioè che non si abbia  $a_R = b_R = 0$ ) e le coordinate  $x_p$  e  $y_p$  di un punto, chiami la funzione elabora e stampi a video la distanza del punto dalla retta.*

# Esercizi da Svolgere

## Esercizio 9.4 – Cercare informazioni sui libri presenti in una libreria

*Si definisca la struttura libro, composta dai seguenti campi un codice numerico (un numero intero) per identificare il libro, l'autore (una stringa di 31 caratteri), il titolo (una stringa di 63 caratteri), il numero di pagine (un numero intero) e il costo (un numero reale). Si scriva, quindi, un programma in linguaggio C++ che operi come segue:*

- Definisca un array libreria di 100 strutture di tipo libro.*
- Apra in lettura il file inventario.txt che descrive l'inventario di una libreria: ciascuna riga del file contiene il codice identificativo di un libro, il titolo del libro, il suo autore, il numero di pagine e il costo. Il file non è vuoto, contiene meno di 100 righe, ma il numero esatto di righe non è noto. Si supponga, inoltre, che i dati contenuti nel file siano validi.*

# Esercizi da Svolgere

## Esercizio 9.4 – Cercare informazioni sui libri presenti in una libreria (continua)

- *Assegni il contenuto di ciascuna riga del file all'elemento corrispondente dell'array libreria*
- *Chieda all'utente (finché lo desidera) di immettere da tastiera il titolo di un libro, cerchi nell'array libreria il libro corrispondente e, se esiste, stampi a video le altre informazioni. Nel caso in cui il libro non sia nell'array, viene visualizzato un opportuno messaggio.*

*Esempio di contenuto del file inventario.txt:*

*1, Il codice da Vinci, Dan Brown, 500, 20*

*2, Lo Hobbit, J.R.R. Tolkien, 700, 35*

*3, L'alchimista, Paulo Coelho, 350, 25*

*4, Il giovane Holden, J.D. Salinger, 225, 15*

# Esercizi da Svolgere

## Esercizio 9.5 – **Elaborare i dati degli esami di un insegnamento universitario**

*Si scriva un programma in linguaggio C++ che elabori le informazioni relative ad una sessione di esami di un insegnamento universitario. Per ogni esame superato si dovranno memorizzare le seguenti informazioni: nome e cognome dello studente (due stringhe lunghe al massimo 31 caratteri), numero di matricola (un numero intero), voto conseguito in trentesimi (un numero intero). Si supponga che la sessione consista al massimo di 100 esami. La sessione può essere realizzata come un array di esami opportunamente inizializzato.*

# Esercizi da Svolgere

Esercizio 9.5 – **Elaborare i dati degli esami di un insegnamento universitario** (continua)

*Il programma fornisce le seguenti funzionalità, implementate mediante funzioni:*

- 1. Inserimento dei dati da parte dell'utente da tastiera; occorrerà verificare che il voto assuma valori validi.*
- 2. Calcolo e stampa a video della media dei voti conseguiti.*
- 3. Calcolo e stampa a video del numero di studenti che hanno conseguito un voto superiore a  $n$ , con  $n$  inserito dall'utente.*
- 4. Calcolo e stampa a video dell'istogramma dei voti conseguiti, ovvero per ogni voto da 18 a 30 si dovranno contare le occorrenze di ciascun voto.*

*La funzione main permetterà all'utente di selezionare una delle funzionalità fornite dal programma ed eseguire la funzione corrispondente.*

# Esercizi da Svolgere

## Esercizio 9.6 – **Elaborare i dati di appartamenti in vendita**

*Si scriva un programma C++ che elabori le informazioni relative agli appartamenti in vendita presso un'agenzia immobiliare. Per ogni appartamento si dovranno memorizzare le seguenti informazioni: codice identificativo (un numero intero), indirizzo (una stringa di caratteri per la via e una per il comune), numero di vani, superficie in mq, prezzo. Si supponga che il numero di appartamenti in vendita non superi le 100 unità. La collezione di appartamenti in vendita può essere realizzata come un array di appartamenti opportunamente inizializzato. Il programma dovrà fornire le seguenti funzionalità:*

- 1. Inserimento dei dati da parte dell'utente da tastiera, verificando la validità degli stessi (ovvero numero di vani, superficie e prezzo dovranno essere numeri positivi).*
- 2. Calcolo e stampa a video del prezzo medio, minimo e massimo degli appartamenti.*



# Esercizi da Svolgere

## Esercizio 9.6 – **Elaborare i dati di appartamenti in vendita** (continua)

- 3. Calcolo e stampa a video del prezzo medio degli appartamenti la cui superficie è compresa tra un valore minimo e un valore massimo inseriti dall'utente.*
- 4. Calcolo e stampa a video del numero di monolocali ( $n = 1$ ), di bilocali ( $n = 2$ ), di trilocali ( $n = 3$ ), di appartamenti di 4, 5, 6, 7 e 8 vani e di quelli con più di 8 vani.*
- 5. Stampa a video delle informazioni relative a tutti gli appartamenti di  $n$  vani, con  $n$  inserito dall'utente.*
- 6. Stampa a video delle informazioni relative a tutti gli appartamenti il cui prezzo è compreso tra un valore minimo ed un valore massimo inseriti dall'utente.*

*La funzione main permetterà all'utente di selezionare una delle funzionalità fornite dal programma ed eseguire la funzione corrispondente.*

# 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\\_2023-24](https://github.com/gualtierovolpe/fondamenti_di_informatica_2023-24)

Le soluzioni di questa esercitazione saranno rilasciate in occasione della simulazione della prima prova intermedia.