

## INDICE

|   |     |
|---|-----|
| Indice.....   | 1   |
| Introduzione ed obiettivo.....                            | 2   |
| Introduzione della variabile globale.....                 | 3   |
| Somma del punteggio della partita al totale generale..... | 4   |
| Visualizzazione punteggio tot al termine gioco.....       | 5   |
| Verifica e risultati.....                                 | 6-7 |
| Conclusione.....  | 8   |

## INTRODUZIONE ED OBIETTIVO

INTRODUZIONE: Questo esercizio facoltativo è un'estensione del progetto principale "Gioco a domande e risposte in C".

L'obiettivo era modificare il programma originale in modo che il **punteggio ottenuto dal giocatore non si azzeri a ogni nuova partita** ma venga **mantenuto in memoria** e sommato partita dopo partita durante la stessa esecuzione del programma.

OBIETTIVO: L'obiettivo specifico era quello di:

- creare una **variabile globale** in grado di memorizzare il punteggio totale del giocatore;
- aggiornare tale variabile ad ogni nuova partita;
- mostrare il punteggio complessivo sia durante l'esecuzione del gioco sia al momento dell'uscita.

## VARIABILE GLOBALE

```
5 #include <string.h>
6
7 int punteggio_totale = 0; // Nuova variabile per sommare i punti di tutte le partite
8
```

Questa riga di codice introduce una variabile globale dichiarata al di fuori di qualsiasi funzione ma rimane in memoria per tutta la durata dell'esecuzione del programma. È stata scelta di tipo **int** perché deve contenere un numero intero (il totale dei punti), e inizializzata a **0** per partire da un valore neutro.

La sua collocazione strategica all'inizio del file permette a ogni funzione, come **gioco()** o **main()**, di poter leggere e modificare il suo valore senza doverla passare come parametro.

L'introduzione di questa variabile è il cuore logico dell'estensione: essa funge da "contenitore permanente" del punteggio, evitando che si azzeri quando la funzione **gioco()** termina.

## SOMMA DEL PUNTEGGIO PARTITA AL TOT GENERALE

```
punteggio_totale += punteggio; // somma il punteggio della partita al punteggio generale

printf("\nHai totalizzato %d punti in questa partita.\n", punteggio); // mostra punti partita
printf("Il tuo punteggio complessivo è ora: %d punti.\n\n", punteggio_totale); // mostra tot aggiornato
```

La seconda modifica è avvenuta all'interno della funzione **gioco()**.

Al termine delle tre domande, dopo aver calcolato il punteggio della singola partita, è stata aggiunta l'istruzione: **punteggio\_totale += punteggio;**

Questa riga indica che al valore già esistente di `punteggio_totale` va sommato (+=) il nuovo punteggio appena ottenuto.

In questo modo, il punteggio della partita corrente non sostituisce quello precedente ma lo incrementa, mantenendo una memoria cumulativa.

Sono poi stati aggiunti due comandi **printf()** per comunicare chiaramente all'utente sia il risultato della partita appena conclusa, sia l'aggiornamento del punteggio totale: il primo messaggio fornisce un feedback immediato sul punteggio del singolo quiz, mentre il secondo mostra il valore globale aggiornato, rendendo il programma più interattivo e informativo.

Queste stampe servono anche come *debug visivo* per confermare che la somma venga effettivamente eseguita.

## VISUALIZZAZIONE PUNTEGGIO TOT AL TERMINE GIOCO

```
case 'B':  
    printf("\nArrivederci! Il tuo punteggio totale finale è: %d punti.\n",  
           punteggio_totale); // stampa il punteggio totale all'uscita dal gioco  
    break;
```

L'ultima modifica riguarda la funzione `main()` e in particolare il blocco `case 'B'` dello `switch`, cioè la parte di codice eseguita quando l'utente sceglie di uscire.

È stata aggiunta la seguente riga:

**`printf("\nArrivederci! Il tuo punteggio totale finale è: %d punti.\n",  
punteggio_totale);`** Questa aggiunta serve a dare una chiusura più completa e informativa al programma.

Invece di mostrare solo un messaggio di fine ("Arrivederci!"), il giocatore riceve anche il riepilogo del proprio punteggio complessivo.

È un piccolo miglioramento dal punto di vista funzionale e dell'esperienza utente, che consolida il senso di continuità tra una partita e l'altra.

## VERIFICA E RISULTATI

```
Hai totalizzato 3 punti in questa partita.  
Il tuo punteggio complessivo è ora: 3 punti.  
  
Scegli una delle due opzioni:  
  A) Inizia una nuova partita  
  B) Esci dal gioco  
Selezione: A  
  
Inserisci il tuo nome: C  
Ciao C!  
  
Domanda 1: Chi ha vinto lo scudetto quest'anno?  
A) MILAN  
B) CATANZARO  
C) NAPOLI  
Risposta: C  
Risposta esatta!  
  
Domanda 2: Quante dita ha un umano (mani + piedi)?  
A) 10  
B) 20  
C) 47  
Risposta: B  
Risposta esatta!  
  
Domanda 3: Chi ha scritto la Divina Commedia?  
A) Socrate  
B) Dante  
C) Ariosto  
Risposta: B  
Risposta esatta!  
  
Hai totalizzato 3 punti in questa partita.  
Il tuo punteggio complessivo è ora: 6 punti.  
  
Scegli una delle due opzioni:  
  A) Inizia una nuova partita  
  B) Esci dal gioco  
Selezione: B  
  
Arrivederci! Il tuo punteggio totale finale è: 6 punti.
```

Per verificare il corretto funzionamento dell'estensione, il programma è stato eseguito in terminale e sono state giocate due partite consecutive all'interno della stessa sessione: al termine della prima partita il sistema stampa il riepilogo "Hai totalizzato 3 punti in questa partita. Il tuo punteggio complessivo è ora: 3 punti.", confermando che il punteggio parziale viene calcolato e comunicato all'utente;

avviando subito una seconda partita, il flusso si ripete e, al termine, il messaggio diventa “Hai totalizzato 3 punti in questa partita. Il tuo punteggio complessivo è ora: 6 punti.”, evidenziando che la **variabile globale punteggio\_totale** è stata aggiornata tramite l’istruzione **punteggio\_totale += punteggio** e che i punti della seconda partita si sono correttamente sommati a quelli precedenti senza azzerarsi tra un quiz e l’altro.

Infine, selezionando l’opzione B) Esci dal gioco, il programma chiude il loop e stampa il riepilogo finale “Arrivederci! Il tuo punteggio totale finale è: 6 punti.”, a conferma che il valore cumulativo è mantenuto per tutta la durata dell’esecuzione e viene presentato chiaramente all’utente al momento dell’uscita.

## CONCLUSIONE

L'esercizio facoltativo ha rappresentato un'estensione logica e funzionale del gioco principale, permettendo di mettere in pratica il concetto di **persistenza delle variabili globali**.

Grazie a poche ma mirate modifiche, il programma è diventato più realistico e coinvolgente, mostrando come una singola variabile globale possa cambiare radicalmente la dinamica di esecuzione.

Questo lavoro ha inoltre rinforzato la comprensione della gestione della memoria in C, dell'ambito di visibilità delle variabili e della comunicazione tra funzioni.