

# **PROGETTO S2/L5 REPORT**

BUG HUNTING IN C

**7 GIUGNO ————— 2024**

# ERRORI RISCONTRATI

## FILE: ESERCIZIO\_5\_EPICODE\_W2.TXT

Nel repository GitHub allegato, è disponibile il progetto che ho **riscritto** seguendo le mie personali modifiche e miglioramenti. Ho organizzato le funzioni in un file header, suddividendole e sviluppandole ulteriormente in un file .c separato, prima di integrarle nel main.c.

- `char scelta = {'\0'};` - dovrebbe essere: `char scelta = '\0';` L'intenzione di questa parte di codice dovrebbe essere quella di inizializzare la variabile di tipo char con un valore nullo. Scritta in questo modo, il compilatore segnalerà un errore. La scelta delle parentesi graffe in questo caso è consentita solo per gli array e altre strutture dati.
- `scanf ("%d", &scelta);` - dovrebbe essere: `scanf ("%c", &scelta);` poichè la variabile scelta è di tipo char, non int.
- Il costrutto switch non crea problemi al compilatore, ma poteva essere migliorato e definito meglio. Il codice in generale, oltre agli errori di sintassi presenta molte imprecisioni non convenzionali e molti spazi superflui. Potevano essere inseriti controlli sul case sensitive della scelta utente, o nel caso inserisca altri valori. Vedi progetto riscritto su github con l'aggiunta anche di default:.
- la funzione void `moltiplica()` presenta svariati errori. - `scanf ("%f", &a);` `scanf ("%d", &b);` - Queste istruzioni sono errate. Si sta riferendo ad un float (%f), e un intero %d per leggere i valori con scanf. Ma nella dichiarazione delle variabili, inizializza una short int, e quindi: %hd.
- `short int prodotto = a * b; printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto);` - ovviamente anche queste due istruzioni appartenenti alla funzione moltiplica, presentano gli errori descritti sopra.

- `int divisione = a % b;` La funzione `dividi()` presenta errori. il carattere `%` (modulo), non si occupa di dividere ma calcolare e restituire come risultato il resto della divisione. Inoltre, è presente un errore grammaticale: "denumeratore". A proposito di "denominatore", non vi è un controllo nel caso in cui si dividesse per zero. (Anche questo inserito nel progetto revisionato).
- `scanf ("%s", &stringa);` - Nella funzione `ins_string()`, vi è questa imprecisione: `&stringa`. Essendoci `%s`, ci si aspetta già un puntatore a una stringa di caratteri. Quindi per evitare errori del compilatore va omessa la `&`. Inoltre, si sarebbero potuti aggiungere dei controlli maggiori a questa funzione per evitare `bufferoverflow`, come un controllo sul numero massimo di caratteri inseribili dall'utente: `char stringa[11]; scanf ("%10s", stringa);` In questo caso, leggerà esattamente 10 caratteri.
- Errori generici di grammatica italiana: "assistente", "denumeratore" ecc...