

Bug Hunting

A close-up, low-angle shot of a person's hand typing on a laptop keyboard. The scene is dimly lit, with the primary light source being the backlit keys of the laptop, which cast a warm, yellowish glow. The hand is positioned over the keyboard, with fingers pressing down on the keys. The laptop is dark-colored, and the background is almost entirely black, emphasizing the hand and the keyboard.

Ernesto Robles

SCOPO DEL PROGRAMMA

Il programma esegue tre diverse operazioni a seconda della scelta dell'utente tramite un'interfaccia "user-friendly".

Moltiplicazione di due numeri:

Esegue la moltiplicazione e stampa il risultato se i dati inseriti sono validi.

Divisione di due numeri:

Esegue la divisione e stampa il risultato se i dati inseriti sono validi.

Inserimento di una stringa:

Chiede all'utente di inserire una stringa di massimo 9 caratteri.

ANALISI ERRORRI

IL CODICE QUI PRESENTE UNA STAMPA DEL MENU E LEGGE LA SCELTA DELL'UTENTE IN BASE ALLA SCELTA EFFETTUA UNA CHIAMATA DI FUNZIONE CORRISPONDETE ABBIAMO PERÒ DUE ERRORI SINTATTICI E LOGICI.

ERRORE SINTATTICO:
CHAR SCELTA = {'\0'};
LA DICHIARAZIONE PER CARATTERE NULLO È '\0'
SENZA LE GRAFFE

ERRORE LOGICO:
SCANF ("%D", &SCELTA);//AL POSTO DI %D CI VA %C IN QUANTO IL TIPO È CHAR E NON UN INT

```
#include <stdio.h>

void menu ();
void moltiplica ();
void dividi ();
void ins_string();

int main ()

{
    char scelta = {'\0'};
    menu ();
    scanf ( format: "%d", &scelta);

    switch (scelta)
    {
        case 'A':
            moltiplica();
            break;
        case 'B':
            dividi();
            break;
        case 'C':
            ins_string();
            break;
    }
}
```



```
void menu() {
    printf ( format: "Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
    printf ( format: "Come posso aiutarti?\n");
    printf ( format: "A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
}
```

```
void moltiplica ()
{
    short int a,b = 0;
    printf ( format: "Inserisci i due numeri da moltiplicare:");
    scanf ( format: "%f", &a);
    scanf ( format: "%d", &b);

    short int prodotto = a * b;

    printf ( format: "Il prodotto tra %d e %d e': %d", a,b,prodotto);
}
```

ANALISI ERRORRI

LA FUNZIONE MENU() VISUALIZZA A SCHERMO UN MESSAGGIO DI BENVENUTO E LE OPZIONI DI SCELTA.

**LA FUNZIONE MOLTIPLICA EFFETTUA UNA MOLTIPLICAZIONE FRA DUE NUMERI E NE VISUALIZZA IL RISULTATO AL SUO INTERNO
ABBIAMO ERRORI LOGICI**

SHORT INT A,B = 0; NON È NECESSARIO DICHIARARE B=0 IN QUANTO PRENDEREMO COME INPUT UN QUALSIASI NUMERO

SCANF ("%F", &A); A È UN SHORT INT PERCUI USEREMO IL %H AL POSTO DEL %F CHE SERVE PER I FLOAT

SCANF ("%D", &B); B È UN SHORT INT PERCUI USEREMO IL %H AL POSTO DEL %F CHE SERVE PER I FLOAT

ANALISI ERRORI

LA FUNZIONE DIVIDI() EFFETTUA LA DIVISIONE FRA DUE NUMERI E NE VISUALIZZA IL RISULTATO. HA AL SUO INTERNO DUE ERRORI LOGICI:

NT A,B = 0;
NON È NECESSARIO DICHIARARE B=0 IN QUANTO
PRENDEREMO COME INPUT UN QUALSIASI NUMERO

INT DIVISIONE = A % B;
LA DIVISIONE LA FACCIAMO CON / POICHÈ % TORNA
UN RESTO

LA FUNZIONE INS_STRING() SALVA L'INSERIMENTO DI UNA STRINGA DENTRO UNA VARIABILE. HA UN ERRORE DI SINTASSI:

SCANF ("%S", &STRINGA);
NON USIAMO "&" PRIMA DI UNA STRINGA PER
LEGGERE LA MEDESIMA

```
void dividi ()
{
    int a,b = 0;
    printf (format: "Inserisci il numeratore:");
    scanf (format: "%d", &a);
    printf (format: "Inserisci il denumeratore:");
    scanf (format: "%d", &b);

    int divisione = a % b;

    printf (format: "La divisione tra %d e %d e': %d", a,b,divisione)
}

void ins_string ()
{
    char stringa[10];
    printf (format: "Inserisci la stringa:");
    scanf (format: "%s", &stringa);
}
```

SOLUZIONE

La proposta di soluzione seguente corregge gli errori visti a livello codice e inoltre integra soluzioni per la gestione di eventuali errori d'input, il programma fornisce un messaggio di errore e/o termina l'esecuzione.

SOLUZIONE

ALL'INTERNO DEL MAIN() È PRESENTE UN CICLO DO-WHILE CHE ESEGUE IL MENU E LEGGE LA SCELTA DELL'UTENTE FINCHÉ LA SCELTA NON È 'D' O 'd' (PER USCIRE DAL PROGRAMMA).

ALL'INTERNO DEL CICLO, IL PROGRAMMA STAMPA IL MENU E LEGGE LA SCELTA DELL'UTENTE. IN BASE ALLA SCELTA, EFFETTUA UNA CHIAMATA DI FUNZIONE CORRISPONDETE .

```
int main() {  
  
    char scelta = '\0';  
  
    do {  
  
        menu();  
        scanf( format: " %c", &scelta);  
  
        switch (scelta) {  
            case 'A':  
            case 'a':  
                moltiplica();  
                break;  
  
            case 'B':  
            case 'b':  
                dividi();  
                break;  
  
            case 'C':  
            case 'c':  
                ins_string();  
                break;  
  
            case 'D':  
            case 'd':  
                printf( format: "Hai scelto di uscire. Arrivederci.\n");
```

```
                default:  
                    printf( format: "Input non valido. Uscita.\n");  
            }  
        } while (scelta!='D' && scelta!='d');  
  
    return 0;
```

SOLUZIONE

LA FUNZIONE MENU() VISUALIZZA A SCHERMO UN MESSAGGIO DI BENVENUTO E LE OPZIONI DI SCELTA.

LA FUNZIONE DIVIDI() EFFETTUA LA DIVISIONE FRA DUE NUMERI INTERI E NE VISUALIZZA IL RISULTATO, GESTISCE ANCHE IL CASO IN CUI L'UTENTE INSERISCA UN DENOMINATORE UGUALE A ZERO. SE L'INPUT NON È VALIDO, VIENE VISUALIZZATO UN MESSAGGIO DI ERRORE.

```
#include <stdio.h>

//soluzione giusta
void menu() {
    printf( format: "Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
    printf( format: "Come posso aiutarti?\n");
    printf( format: "A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\nD >> Exit\n");
}

void dividi() {
    int a, b;
    printf( format: "Inserisci il numeratore:");
    if (scanf( format: "%d", &a) != 1) {
        printf( format: "Input non valido. Uscita.\n");
        return;
    }

    printf( format: "Inserisci il denominatore:");
    if (scanf( format: "%d", &b) != 1 || b == 0) {
        printf( format: "Input non valido. Uscita.\n");
        return;
    }

    int divisione = a / b;

    printf( format: "La divisione tra %d e %d e': %d\n", a, b, divisione);
}
```


SOLUZIONE

LA FUNZIONE INS_STRING() SALVA L'INSERIMENTO DI UNA STRINGA DI MASSIMO 9 CARATTERI. SE L'INPUT SUPERA I 9 CARATTERI, VIENE VISUALIZZATO UN MESSAGGIO DI ERRORE.

LA FUNZIONE MOLTIPLICA() EFFETTUA UNA MOLTIPLICAZIONE FRA DUE NUMERI E NE VISUALIZZA IL RISULTATO. SE L'INPUT NON È VALIDO (SE L'UTENTE INSERISCE QUALCOSA DI DIVERSO DA DUE NUMERI), VIENE VISUALIZZATO UN MESSAGGIO DI ERRORE.

```
void ins_string() {
    char stringa[10];
    printf( format: "Inserisci la stringa (massimo 9 caratteri): ");
    if (scanf( format: "%9s", stringa) != 1) {
        printf( format: "Input non valido. Uscita.\n");
        return;
    }

    printf( format: "La stringa inserita e': %s\n", stringa);
}

void moltiplica() {
    short int a, b;
    printf( format: "Inserisci i due numeri da moltiplicare:");
    if (scanf( format: "%hd %hd", &a, &b) != 2) {

        printf( format: "Input non valido. Uscita.\n");
        return;
    }

    short int prodotto = a * b;

    printf( format: "Il prodotto tra %hd e %hd e': %hd\n", a, b, prodotto);
}
```