

Report - Bug hunting - 12/05/2023

Rossella Amore

1- Cosa fa il programma?

- Il programma è stato creato per essere un assistente digitale con tre funzioni disponibili all'utente: moltiplicazione di due numeri, divisione di due numeri e inserimento di una stringa da 9+1 caratteri. Queste funzioni possono essere scelte rispettivamente inserendo A, B o C da parte dell'utente.

2- Casistiche non standard

- La variabile short int non corrisponde agli identificatori di formato nella funzione moltiplica().
- La funzione dividi() presenta l'operatore di modulo invece che quello di divisione intera, come dichiara invece il programma.
- La funzione dividi() consente di inserire qualsiasi numero come divisore, incluso 0, che renderebbe la divisione impossibile. Questo potrebbe generare dei problemi durante il calcolo.

3- Errori di sintassi e logici (Vedi codice in fondo)

- La sintassi del "char scelta" presenta un errore. {'\0'}
- L'indicatore di formato nella scansione di scelta ha un errore logico. "%d"
- La variabile short int nella funzione moltiplica() ha un errore logico.
- Gli identificatori di formato "%f" e "%d" nella funzione moltiplica() non corrispondono alla variabile selezionata.
- L'operatore di modulo nella funzione dividi() potrebbe essere considerato un errore di sintassi.

4- Soluzione

- Correggere la variabile "char scelta" da {'\0'} a '\0'. Non è un blocco di codice e non ha bisogno di parentesi graffe.
- Correggere l'indicatore di formato nella scansione di scelta da "%d" a "%c" per farlo corrispondere alla variabile selezionata.

- La variabile short int nella funzione moltiplica() andrebbe sostituita da int e gli indicatori di formato andrebbero cambiati entrambi in “%d” per corrispondere alla variabile e non creare comportamenti imprevisti.
- L’operatore di modulo nella funzione dividi() andrebbe sostituito con un operatore di divisione intera.
- Nella funzione dividi() andrebbe inserita una condizione per bloccare l’utente nel caso in cui inserisca il numero 0 come divisore.
- Nello switch di scelta andrebbe inserita una condizione per bloccare l’utente nel caso in cui inserisca qualsiasi carattere diverso da ‘A’, ‘B’ o ‘C’, come un default.
- *Non è da considerarsi errore logico, ma è consigliabile aumentare il limite di caratteri inseribili nella funzione ins_string().*
- *Si consiglia di migliorare l’indentazione di alcune parti del codice e di aggiungere dei \n dove necessario per rendere il codice più gradevole e comprensibile all’occhio umano.*

Codice sorgente originale (con commenti)

```
#include <stdio.h>

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

int main ()

{
    char scelta = {"\0"}; //errore di sintassi
    menu ();
    scanf ("%d", &scelta); //errore logico

    switch (scelta) //si potrebbe inserire una condizione di errore in caso di inserimento diverso da 'A', 'B' e 'C'
    {
        case 'A':
            moltiplica();
            break;
        case 'B':
```

```

        dividi();
        break;
        case 'C':
            ins_string();
            break;
    }

return 0;

}

```

```

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

```

```

void moltiplica ()
{
    short int a,b = 0; //errore logico, potrebbe creare problemi
    printf ("Inserisci i due numeri da moltiplicare:");
    scanf ("%f", &a); //gli identificatori float e int non corrispondono.
    scanf ("%d", &b);

    short int prodotto = a * b;

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

```

```

void dividi ()
{
    int a,b = 0;
    printf ("Inserisci il numeratore:");
}

```

```
scanf ("%d", &a);
printf ("Inserisci il denominatore:");
scanf ("%d", &b); //si dovrebbe aggiungere una condizione di errore
in caso di inserimento del numero 0
```

```
int divisione = a % b; //l'operatore dovrebbe essere di divisione intera,
non di modulo
```

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

```
void ins_string ()
{
    char stringa[10]; //è consigliabile avere a disposizione più caratteri,
magari 20 o 30.
    printf ("Inserisci la stringa:");
    scanf ("%s", &stringa);
}
```

Codice sorgente corretto:

```
#include <stdio.h>
```

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

```
int main ()
```

```
{
    char scelta = '\0';
    menu ();
    scanf ("%c", &scelta);
```

```

switch (scelta)
{
    case 'A':
        moltiplica();
        break;
    case 'B':
        dividi();
        break;
    case 'C':
        ins_string();
        break;
    default:
        printf ("Riprova.\n");
        break;
}

return 0;

}

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

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

```

```

    int prodotto = a * b;

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

void dividi ()
{
    int a,b = 0;
    printf ("Inserisci il numeratore:\n");
    scanf ("%d", &a);
    printf ("Inserisci il denominatore:\n");
    scanf ("%d", &b);

    if (b ==0) {
        printf ("L'operazione è impossibile. Riprova con un altro
numero.\n");
        scanf ("%d", &b)
    }
    else {

        int divisione = a / b;

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

```

```

void ins_string ()
{
    char stringa[30];
    printf ("Inserisci la stringa:\n");
    scanf ("%s", &stringa);
    printf ("Ecco a te la stringa: %s\n", stringa);
}

```