

BENVENUTI LUIGI PROGETTO S2/L5 02/02/2024

Traccia:

Per agire come un Hacker bisogna capire come pensare fuori dagli schemi. L'esercizio di oggi ha lo scopo di allenare l'osservazione critica. Dato il codice in allegato, si richiede allo studente di:

- 1 - Capire cosa fa il programma senza eseguirlo.
- 2 - Individuare dal codice sorgente le casistiche non standard che il programma non gestisce (esempio, comportamenti potenziali che non sono stati contemplati).
- 3 - Individuare eventuali errori di sintassi / logici.
- 4 - Proporre una soluzione per ognuno di essi.

Ecco il codice da analizzare:

```
#include <stdio.h>

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

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

    switch (scelta)
    {
        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;
    printf ("Inserisci i due numeri da moltiplicare:");
    scanf ("%f", &a);
    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);

    int divisione = a % b;

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

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

1 – Il programma simula un assistente digitale che permette di andare ad effettuare 3 operazioni, scegliendole attraverso un menù:

- Moltiplicazione;
- Divisione;
- Inserimento stringa;

In alto, la prima parte riguarda la **dichiarazione** delle funzioni.

**Int main()** serve per descrivere il **corpo** del programma; all'interno del main vengono **chiamate le funzioni** e si esplicita il codice di esecuzione **principale**. In questo caso, vi è anche uno **switch**, uno strumento che ci permette di selezionare fra più blocchi di codice, in base ad una **condizione**.

La funzione **menu()** ci permette di scegliere quale delle 3 operazioni disponibili andare a selezionare, la funzione **moltiplica()** ci permette di moltiplicare 2 numeri interi e stampare il risultato a schermo, la funzione **dividi()** ci permette invece di andare a dividere 2 interi stampando il risultato a schermo e la funzione **ins\_string()** permette di inserire una stringa.

2 – Fra i comportamenti potenziali e casistiche non standard non implementate dal programma, il primo a balzare all'occhio è il fatto che non esista una funzione che permetta di decidere all'utente **un'uscita programmata dall'esecuzione**. Per migliorare la comprensione da parte dell'utente, andrebbe inclusa la possibilità di digitare la scelta negli switch in minuscolo (questo perché il linguaggio C è un linguaggio "case sensitive", ovvero fa differenza tra caratteri minuscoli e maiuscoli). Un ultimo caso non standard riguarda la possibilità di **overflow** della stringa Char stringa [10]. Questo errore è molto pericoloso, in quanto un potenziale attaccante può andare ad inserire un numero indeterminato di caratteri rendendo possibile la sovrascrittura di altre celle di memoria. Inoltre, sarebbe opportuno dividere chiaramente le linee tramite l'utilizzo di "\n".

3a – Vediamo adesso eventuali errori logici:

- La funzione **menu()** non effettua assegnazioni né operazioni, può essere dunque omessa andando a scrivere il suo contenuto direttamente nel main.
- All'interno dello switch non è presente una voce **default**; manca dunque la gestione dei casi diversi dalle 3 opzioni selezionabili.
- Le variabili dichiarate come **short int** permettono di eseguire solamente un numero limitato di operazioni a causa della loro dimensione ridotta (2 byte). E' preferibile utilizzare delle variabili di tipo **int**.
- Nelle operazioni di divisione, è più comodo andare a dichiarare i valori come **float**. Questo permetterà di avere un quoziente con decimali. Inoltre, va considerato impossibile assegnare al denominatore il valore 0.
- I nomi delle variabili andrebbero assegnati in maniera diversificata.

3b – Vediamo adesso gli errori di sintassi:

- **Riga 14:** scanf("%d", &scelta) -> scanf("%s", &scelta) // %d è l'operatore per int, % di s per stringhe.
- **Riga 47:** scanf("%f", &a) -> scanf("%hi", &a); // %hi (o %hd) è l'operatore per gli short int.
- **Riga 48:** scanf("%d", &a) -> scanf("%hi", &a); // %hi (o %hd) è l'operatore per gli short int.
- **Riga 52:** printf("Il prodotto tra %d e %d è: %d, a, b, prodotto); -> printf("Il prodotto tra %hi e %hi è: %hi, a, b, prodotto); // %hi per short int.
- **Riga 61:** denominatore -> denominatore; // il dividendo della divisione si chiama denominatore.
- **Riga 64:** int divisione = a % b; -> int divisione a / b; // l'operatore per la divisione è /, % si usa per ottenere il resto di una divisione.
- **Riga 66:** printf("La divisione tra %d e %d è: %d, a, b, prodotto); -> printf("Il prodotto tra %hi e %hi è: %hi, a, b, prodotto); // %hi per short int.
- **Riga 71:** char stringa [10]; -> char stringa [10] = {'\0'}; // dichiarazione erronea array di char
- **Riga 73:** scanf("%s", &stringa); -> scanf("%s", stringa) // assegnazione valore non necessita & per array char

Ecco un possibile codice con tutte le dovute correzioni:

```
#include <stdio.h>
/*
*simulazione esercizio
*/

//dichiarazione funzioni
void moltiplicazione();
void divisione();
void ins_string();

int main(){

char scelta = {'\0'}; // inizializzazione variabile scelta
int avvio = 0; // intero per controllo avvio

    printf("Premi 0 per avviare il programma \n");
    scanf("%d", &avvio); //controllo avvio programma

    while (avvio == 0){

        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");
        scanf ("%s", &scelta); //selezione operazione

        switch (scelta) //menù di scelta
        {
            case 'A':
            case 'a': //implementazione case sensitive
            moltiplicazione();
            break;

            case 'B':
            case 'b':
            divisione();
            break;

            case 'C':
            case 'c':
            ins_string();
            break;

            default: //default gestione input indesiderati
            printf("\nInserire una delle lettere richieste");
            break;

        }

        printf("\nPremi 0 per riniziare, qualsiasi altro tasto per uscire\n");
        scanf("%d", &avvio); //controllo su riavvio partita
    }
}
```

```

    }

return 0;
}

void moltiplicazione(){

    int a,b = 0; //dichiarazione fattori

    printf("Inserisci i 2 numeri da moltiplicare:\n");
    scanf("%d", &a);
    scanf("%d", &b);

    int prodotto = a * b;
    printf("Il prodotto tra %d e %d e': %d", a,b,prodotto); //stampa a
    schermo risultato
}

void divisione(){

    float num,den = 1.00; //dichiarazione numeratore e denominatore

    printf("Inserisci i 2 numeri da dividere:\n");
    scanf("%f", &num);
    scanf("%f", &den);

    while( den == 0){

        printf("Il denominatore non può valere 0! Inserisci un denominatore
diverso da 0:\n");
        scanf("%f", &den);
    } // controllo den != 0

    float quoziente = num / den;
    printf ("Il quoziente tra %10.2f e %10.2f e': %10.2f", num, den,
quoziente); //stampa a schermo risultato

}

void ins_string(){

    char stringa[10] = {'\0'}; //inizializzazione stringa
    printf("Inserisci la stringa:\n");
    scanf("%s", stringa);
    printf("La tua stringa e': %.9s", stringa); //controllo su possibile
    overflow

}

```

Ed ecco una possibile esecuzione:

```
Premi 0 per avviare il programma
0
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
C
Inserisci la stringa:
AFARGFGBARGFCVD
La tua stringa e': AFARGFGBA
Premi 0 per riniziare, qualsiasi altro tasto per uscire
0
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
b
Inserisci i 2 numeri da dividere:
20
0
Il denominatore non pu_ valere 0! Inserisci un denominatore diverso da 0:
4
Il quoziente tra      20.00 e      4.00 e':      5.00
Premi 0 per riniziare, qualsiasi altro tasto per uscire
0
Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti
Come posso aiutarti?
A >> Moltiplicare due numeri
B >> Dividere due numeri
C >> Inserire una stringa
a
Inserisci i 2 numeri da moltiplicare:
34
56
Il prodotto tra 34 e 56 e': 1904
Premi 0 per riniziare, qualsiasi altro tasto per uscire
1
-----
Process exited after 53.29 seconds with return value 0
Premere un tasto per continuare . . . _
```