

Compito di Ciaschini Giorgio del 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.

Soluzione:

- 1) Il seguente programma chiede all'utente quale operazione vuole eseguire e poi gli chiede di inserire i dati per svolgere le funzioni.
- 2) Ha dichiarato per prima cosa le funzioni "void" per farle riconoscere successivamente nell'int main.
- 3) Sono stati dichiarati successivamente in rosso.
- 4) Come sopra.

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

```
void menu ();
```

```
void moltiplica ();
```

```
void dividi ();
```

```
void ins_string();
```

```
int main ()
```

```
{
```

```
char scelta = {'\0'}; // errore sintassi, senza parentesi graffe.
```

```
menu ();
```

```
scanf ("%d", &scelta); //errore logico. Scelta è un char e quindi bisogna usare %c.
```

```
    switch (scelta)
```

```
{
```

```
case 'A': // bisogna inserire anche la possibilità che l'utente inserisca una lettera minuscola. Lo stesso per i "case B" e "case C".
```

```
moltiplica();
```

```

break;

case 'B':
    dividi();
    break;

case 'C':
    ins_string();
    break; //sarebbe più opportuno inserire un default in caso l'utente non inserisse nessuno
    dei vari casi precedenti. Anche un return 1 (ad esempio) che far tornare un errore e un printf per la
    persona che lo utilizza.
}

```

```

return 0;

```

```

}

```

```

void menu ()

```

```

{

```

```

printf ("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n"); //Errore
di sintassi. Assistente con doppia Esse. Non darà errori al programma ma è comunque un errore.

```

```

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. Sarebbe più opportuno utilizzare solo int senza short perché
l'utente potrebbe inserire molti numeri.

```

```

printf ("Inserisci i due numeri da moltiplicare:");

```

```

scanf ("%f", &a); //errore sintassi. Bisogna usare %d per gli int.

```

```

scanf ("%d", &b);

    short int prodotto = a * b; // errore logico. Sarebbe più opportuno utilizzare solo int senza
    short perché l'utente potrebbe inserire molti numeri.

    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; //errore di sintassi perché la divisione si scrive con lo slash "/". Anche un
    piccolo errore logico è che siccome la divisione può dare come risultato numeri non interi sarebbe
    opportuno definire la "b" come diverso da zero e "divisione" come un float.

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

```

```

void ins_string ()
{
    char stringa[10]; //errore logico. c'è la possibilità di una digitazione più lunga con un rischio di
    overflow.

    printf ("Inserisci la stringa:");

    scanf ("%s", &stringa); // non serve "&" perché prenderebbe altrimenti tutti i dati anche non
    digitati.
}

```