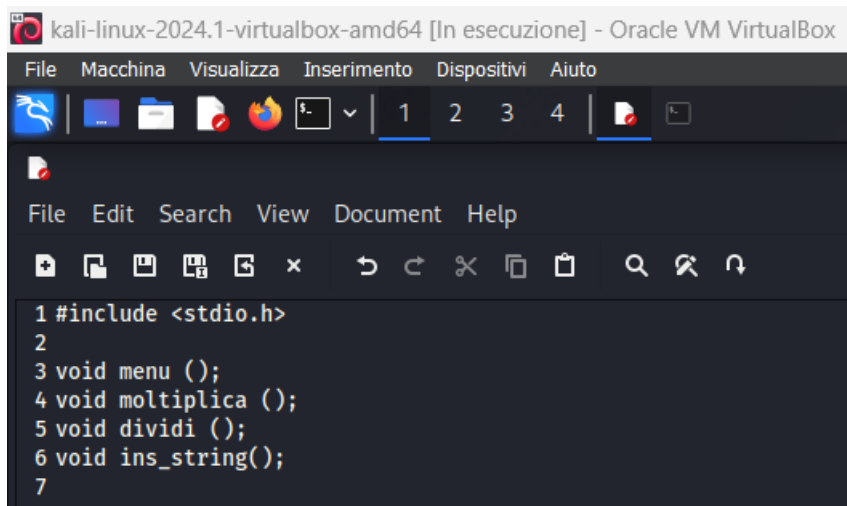


REPORT CORREZIONE CODICE S2L5

Descrizione del codice : un “assistente” capace di eseguire 3 tipi di operazioni

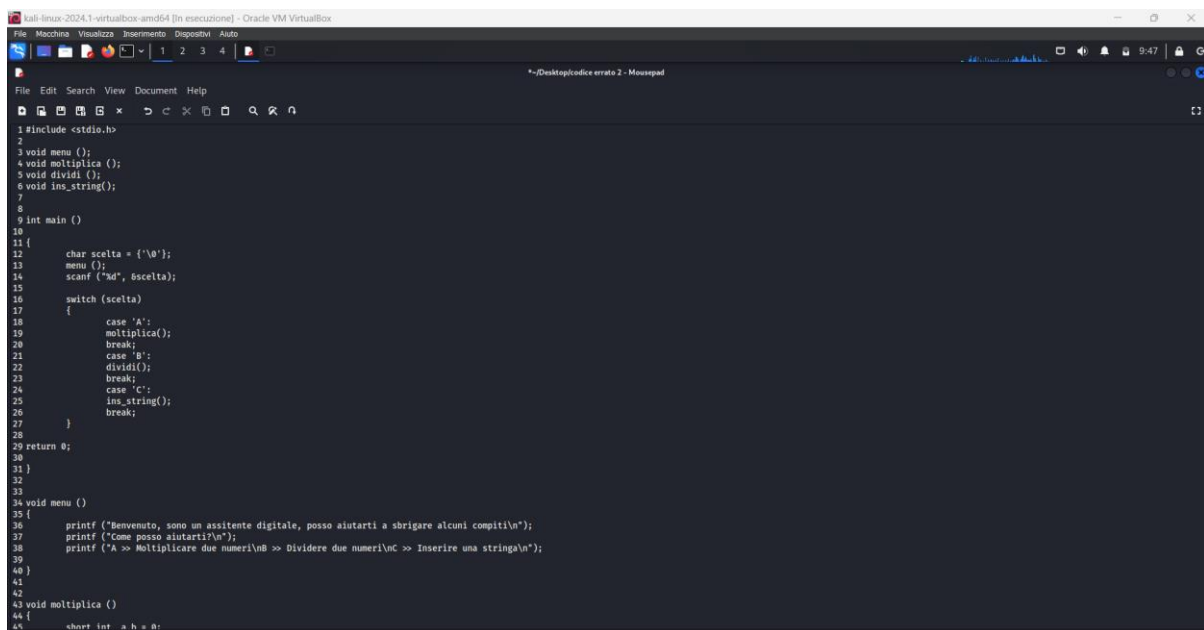
- 1) Moltiplicazione
- 2) Divisione
- 3) Scrivere una stringa

L’utente può inserire i fattori di cui necessita e il programma eseguirà queste 3 operazioni



```
1 #include <stdio.h>
2
3 void menu ();
4 void moltiplica ();
5 void dividi ();
6 void ins_string();
7
8
```

ERRORE 1) mancanza della libreria “#include <string.h>” per far funzionare la variabile “stringa” nella riga 2



```
1 #include <stdio.h>
2 #include <string.h>
3
4 void menu ();
5 void moltiplica ();
6 void dividi ();
7 void ins_string();
8
9 int main ()
10 {
11     char scelta = {'\0'};
12     menu ();
13     scanf ("%d", &scelta);
14
15     switch (scelta)
16     {
17         case 'A':
18             moltiplica();
19             break;
20         case 'B':
21             dividi();
22             break;
23         case 'C':
24             ins_string();
25             break;
26     }
27
28     return 0;
29 }
30
31 void menu ()
32 {
33     printf ("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti\n");
34     printf ("Come posso aiutarti?\n");
35     printf ("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
36 }
37
38 void moltiplica ()
39 {
40     char int_a, b = 0;
41 }
```

ERRORE 2) mancanza di una condizione che impedisca un input errato, per evitare immissioni di codici sorgente malevoli

ERRORE 3) tipo di variabile errato all'interno dello scanf alla riga 14 (%d invece di %c)

```
36 printf ("Benvenuto, sono un assistente digitale, posso aiutarti a sbrigare alcuni compiti!\n");
37 printf ("Come posso aiutarti?\n");
38 printf ("A >> Moltiplicare due numeri\nB >> Dividere due numeri\nC >> Inserire una stringa\n");
39
40 }
41
42
43 void moltiplica ()
44 {
45     short int a,b = 0;
46     printf ("Inserisci i due numeri da moltiplicare:");
47     scanf ("%f", &a);
48     scanf ("%d", &b);
49
50     short int prodotto = a * b;
51
52     printf ("Il prodotto tra %d e %d e': %d", a,b,prodotto);
53 }
54
55 void dividi ()
56 {
57     int a,b = 0;
58     printf ("Inserisci il numeratore:");
59     scanf ("%d", &a);
60     printf ("Inserisci il denominatore:");
61     scanf ("%d", &b);
62
63     int divisione = a % b;
64
65     printf ("La divisione tra %d e %d e': %d", a,b,divisione);
66 }
67
68
69 void ins_string ()
70 {
71     char stringa[10];
72     printf ("Inserisci la stringa:");
73     scanf ("%s", &stringa);
74 }
75
76
77 }
```

ERRORE 4) è preferibile il tipo di variabile INT invece di SHORT INT inquanto, SHORT INT possiede meno BYTE (2) e quindi essere facilmente più attaccabile causando un buffer overflow specialmente in una operazione come la moltiplicazione che accresce il numero dei moltiplicatori(errone a riga 45)

ERRORE 5) tipo di variabile errato all'interno dello scanf alla riga 47 (%f invece di %d)

ERRORE 6) è preferibile il tipo di variabile FLOAT invece di INT inquanto, FLOAT possiede più BYTE (4) e comprende anche numeri reali, che spesso sono risultanti dalle operazioni di divisione, con apposita correzione alle tipologie di variabili dei printf/scanf (da "%d" a "%f") (righe 58-66)

ERRORE 7) probabile errore alla riga 61, c'è scritto "denumeratore" invece di "denominatore"

ERRORE 8) errore alla riga 64, operazione di resto intero (a%b) invece della normale operazione di divisione (a/b)

ERRORE 9) alla riga 75 il numero di celle riservato all' Array è troppo basso con il rischio di un buffer overflow

ERRORE 10) dalla riga 75 alla riga 77 il numero di celle non è provvisto di eventuale codice di controllo variabili che impedisca un eccessivo input di caratteri, risultanti in problemi di sicurezza

```
kali-linux-2024.1-virtualbox-amd64 [in esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto
Shell No. 1
File Actions Edit View Help
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:\n");
    scanf ("%d", &a);
    scanf ("%d", &b);

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

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

    float divisione = a / b;
    printf ("La divisione tra %f e %f e' %f", a,b,divisione);
}

void ins_string ()
{
    char stringa[10];
    printf ("Inserisci la stringa:\n");
    scanf ("%s", &stringa);
    if (strlen(stringa) > 10)
    {
        printf ("I caratteri superano le 10 unita'\n");
    }
    printf ("La stringa che hai scritto e' %s", stringa);
}

#include <stdio.h>
#include <string.h>

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

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

    if (scelta == 'A' || scelta == 'B' || scelta == 'C') {
        switch (scelta)
        {
            case 'A':
                moltiplica();
                break;
            case 'B':
                dividi();
                break;
            case 'C':
                ins_string();
                break;
        }
    } else {
        printf("Il carattere inserito non e' valido\n");
    }
    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;
}
```

RISULTATO :

Una volta eseguite tutte le correzioni otteniamo un codice più sicuro e privo di errori capace di eseguire le operazioni citate nella descrizione del programma

Correzione 1) aggiunta la libreria “#include <string.h>”

Correzione 2) aggiunta di una condizione che impedisca un input errato

Correzione 3) tipo di variabile corretto all’interno dello scanf alla riga 14 (%c invece di %d

)Correzione 4) SHORT INT (2 BYTE) è stato cambiato in INT (4 BYTE)

Correzione 5) tipo di variabile corretto all'interno dello scanf alla riga 47 (%d invece di %f)

Correzione 6) INT (4 BYTE numeri interi) FLOAT (4 BYTE numeri reali)

Correzione 7) denominatore riscritto come denominatore

Correzione 8) a%b corretto con la operazione di divisione a/b

Correzione 9) aumentato il numero di celle riservate all' Array

Correzione 10) aggiunto il codice che impedisce un buffer overflow tramite le celle riservate all' Array limitando il numero di caratteri inseribili nella stringa

