

S2/L2 Iqra Razzaq

Programma in C:

Si scriva un programma che esegua l'operazione di moltiplicazione tra due numeri inseriti dall'utente.

```
Terminal Emulator
root@kali: /home/kali

GNU nano 8.1 es1.c
#include <stdio.h>

int main() {
    // Dichiarazione delle variabili
    float numero1, numero2, risultato;

    // Richiesta di input all'utente
    printf("Inserisci il primo numero: ");
    scanf("%f", &numero1);

    printf("Inserisci il secondo numero: ");
    scanf("%f", &numero2);

    // Calcolo del prodotto
    risultato = numero1 * numero2;

    // Stampa del risultato
    printf("Il risultato della moltiplicazione tra %.2f e %.2f è: %.2f\n", numero1, numero2, risultato); //Stringa di formato: La stringa "Il risultato della moltiplicazione tra %.2f e %.2f è: %.2f\n" contiene testo e specificat

    //%.2f: Questo è un especificatore di formato per numeri in virgola mobile (float). Il .2 indica che vogliamo visualizzare il numero con due cifre decimali.
    //\n: Questo è un carattere di nuova linea, che serve per andare a capo dopo che il messaggio è stato stampato.

    return 0;
}
```

GNU nano 8.1

es1.c *

#include <stdio.h>

int main() {

// Dichiarazione delle variabili

float numero1, numero2, risultato;

// Richiesta di input all'utente

printf("Inserisci il primo numero: ");

scanf("%f", &numero1);

printf("Inserisci il secondo numero: ");

scanf("%f", &numero2);

// Calcolo del prodotto

risultato = numero1 * numero2;

// Stampa del risultato

printf("Il risultato della moltiplicazione tra %.2f e %.2f è: %.2f\n", numero1, numero2, risultato); //Stringa di formato: La stringa "Il risultato della moltiplicazione tra %.2f e %.2f è: %.2f\n" contiene testo e specificat>

//%.2f: Questo è un especificatore di formato per numeri in virgola mobile (float). Il .2 indica che vogliamo visualizzare il numero con due cifre decimali.

//\n: Questo è un carattere di nuova linea, che serve per andare a capo dopo che il messaggio è stato stampato.

```
return 0;  
}
```

```
(root@kali)-[/home/kali]  
# nano es1.c  
  
(root@kali)-[/home/kali]  
# ./es1  
Inserisci il primo numero: 1  
zsh: segmentation fault ./es1  
  
(root@kali)-[/home/kali]  
# gcc es1.c -o es1  
  
(root@kali)-[/home/kali]  
# ./es1  
Inserisci il primo numero: 3  
Inserisci il secondo numero: 3  
Il risultato della moltiplicazione tra 3.00 e 3.00 è: 9.00
```

Per finire **compilo** (serve per controllare errori) con # **gcc** es1.c -o es1, poi **eseguo** con comando **./ es1**