

## Esercizio di Programmazione in Linguaggio C

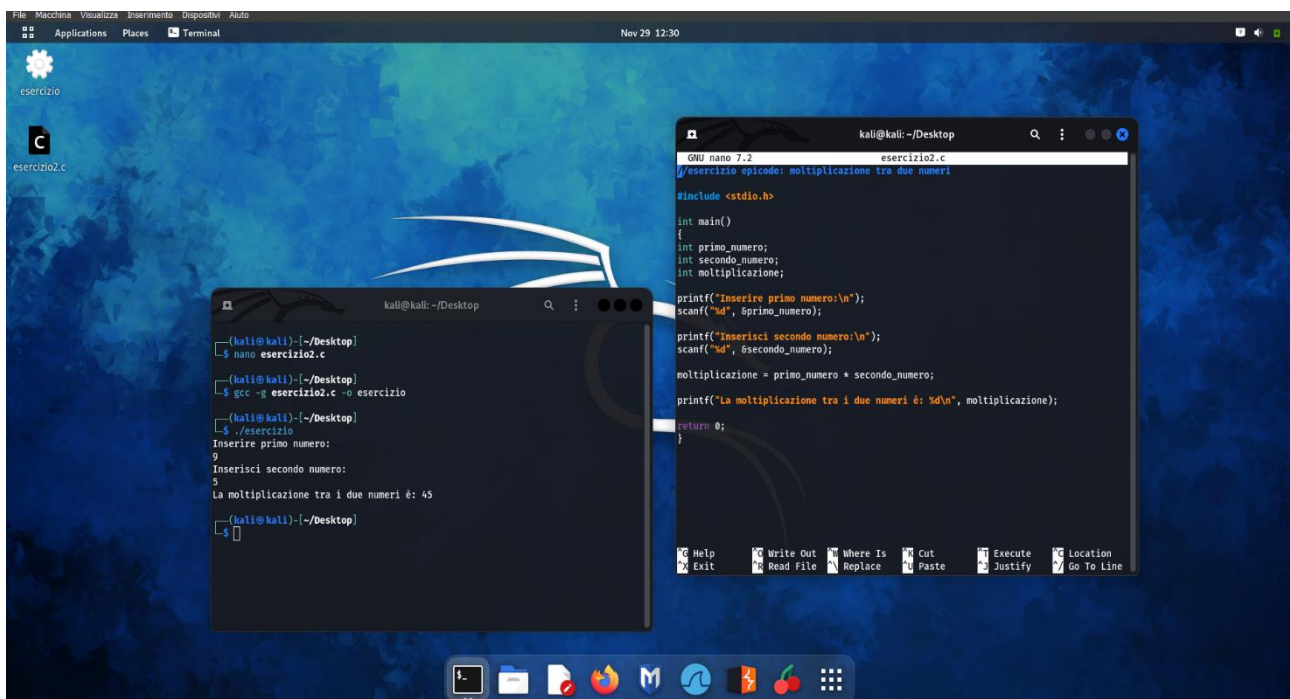
Data 29/11/23

Guglielmo Carratello

L'obiettivo dell'esercizio odierno è la realizzazione di due programmi in linguaggio C che eseguano operazioni specifiche:

### 1. Moltiplicazione tra due numeri:

Abbiamo scritto un programma scritto in C in Kali Linux che chiede di inserire due numeri e successivamente esegue l'operazione di moltiplicazione tra di essi. Abbiamo creato il file chiamato **esercizio2.c** con il comando " nano esercizio.c " e scritto il programma. La funzione int che abbiamo inserito davanti alla funzione main e davanti a nomi delle variabili sta a identificare il tipo di dato che la funzione main deve restituire, nonché il tipo di dato rappresentato dalle variabili. Infine il programma fornisce come output il risultato della moltiplicazione. Per questo tipo di programma richiamo la libreria `<stdio.h>` che contiene le definizioni delle funzioni per l'input/output standard di un programma C.



The screenshot displays a Kali Linux desktop with a blue background. In the top-left corner, there is a desktop icon labeled 'esercizio' with a C logo. A terminal window in the foreground shows the execution of the program. The user enters two numbers, 9 and 5, and the program outputs 'La moltiplicazione tra i due numeri è: 45'. In the background, another terminal window shows the source code of 'esercizio2.c' in nano editor. The code includes `<stdio.h>`, declares `int` variables for the numbers and the result, and uses `printf` and `scanf` for input/output. The desktop taskbar at the bottom contains icons for a terminal, file manager, web browser, and other applications.

```
(kali@kali)-[~/Desktop]
└─$ nano esercizio2.c
└─(kali@kali)-[~/Desktop]
   └─$ gcc -g esercizio2.c -o esercizio
└─(kali@kali)-[~/Desktop]
   └─$ ./esercizio
Inserire primo numero:
9
Inserisci secondo numero:
5
La moltiplicazione tra i due numeri è: 45
└─(kali@kali)-[~/Desktop]
   └─$
```

```
GNU nano 7.2 esercizio2.c
#esercizio episodio: moltiplicazione tra due numeri
#include <stdio.h>

int main()
{
    int primo_numero;
    int secondo_numero;
    int moltiplicazione;

    printf("Inserire primo numero:\n");
    scanf("%d", &primo_numero);

    printf("Inserisci secondo numero:\n");
    scanf("%d", &secondo_numero);

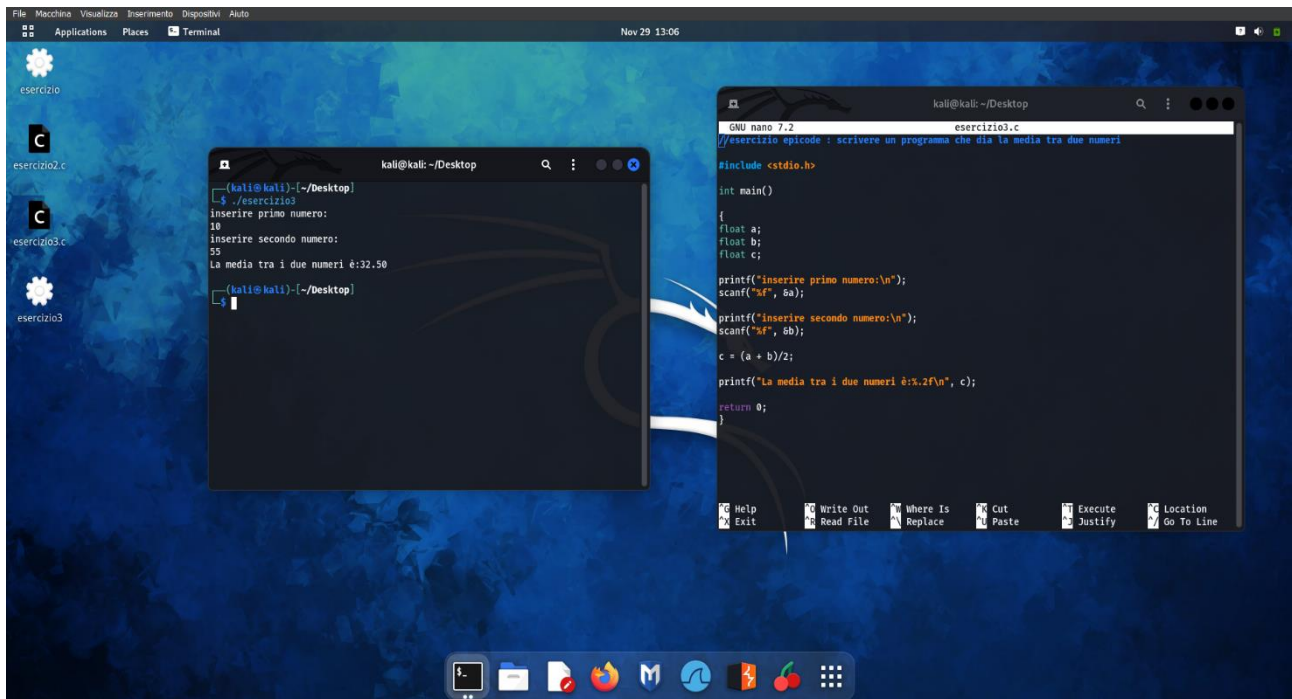
    moltiplicazione = primo_numero * secondo_numero;
    printf("La moltiplicazione tra i due numeri è: %d\n", moltiplicazione);

    return 0;
}
```

### 2. Media Aritmetica di due numeri:

Il secondo programma richiede all'utente di inserire due valori interi. Successivamente, calcola la media aritmetica di questi due numeri e restituisce il risultato come output. Abbiamo lavorato come nel programma precedente, richiamato la libreria `<stdio.h>` e la funzione main. A differenza del programma precedente abbiamo utilizzato la funzione float per le variabili a, b e c che corrispondono al primo numero, secondo numero e c la media. Questo perché l'operazione di media darà con molta probabilità un numero

con la virgola di conseguenza dobbiamo specificare al computer che il risultato considerato sia di questo tipo di dato. Di seguito è riportato lo screen del lavoro che abbiamo effettuato.



```
(kali@kali)-[~/Desktop]
└─$ ./esercizio3
Inserire primo numero:
10
Inserire secondo numero:
55
La media tra i due numeri è:32.50
(kali@kali)-[~/Desktop]
└─$
```

```
GNU nano 7.2 esercizio3.c
//esercizio epicode : scrivere un programma che dia la media tra due numeri
#include <stdio.h>

int main()
{
    float a;
    float b;
    float c;

    printf("Inserire primo numero:\n");
    scanf("%f", &a);

    printf("Inserire secondo numero:\n");
    scanf("%f", &b);

    c = (a + b)/2;

    printf("La media tra i due numeri è:%.2f\n", c);

    return 0;
}
```

## Conclusioni:

La verifica dell'input contribuisce a prevenire errori e comportamenti inaspettati.

L'esercizio ha fornito l'opportunità di applicare concetti fondamentali di programmazione in linguaggio C, come l'acquisizione dell'input utente, la gestione dei tipi di dati e la realizzazione di operazioni aritmetiche.

Questo esercizio ha contribuito a consolidare le competenze di base necessarie per scrivere programmi C funzionali.