

### Traccia:

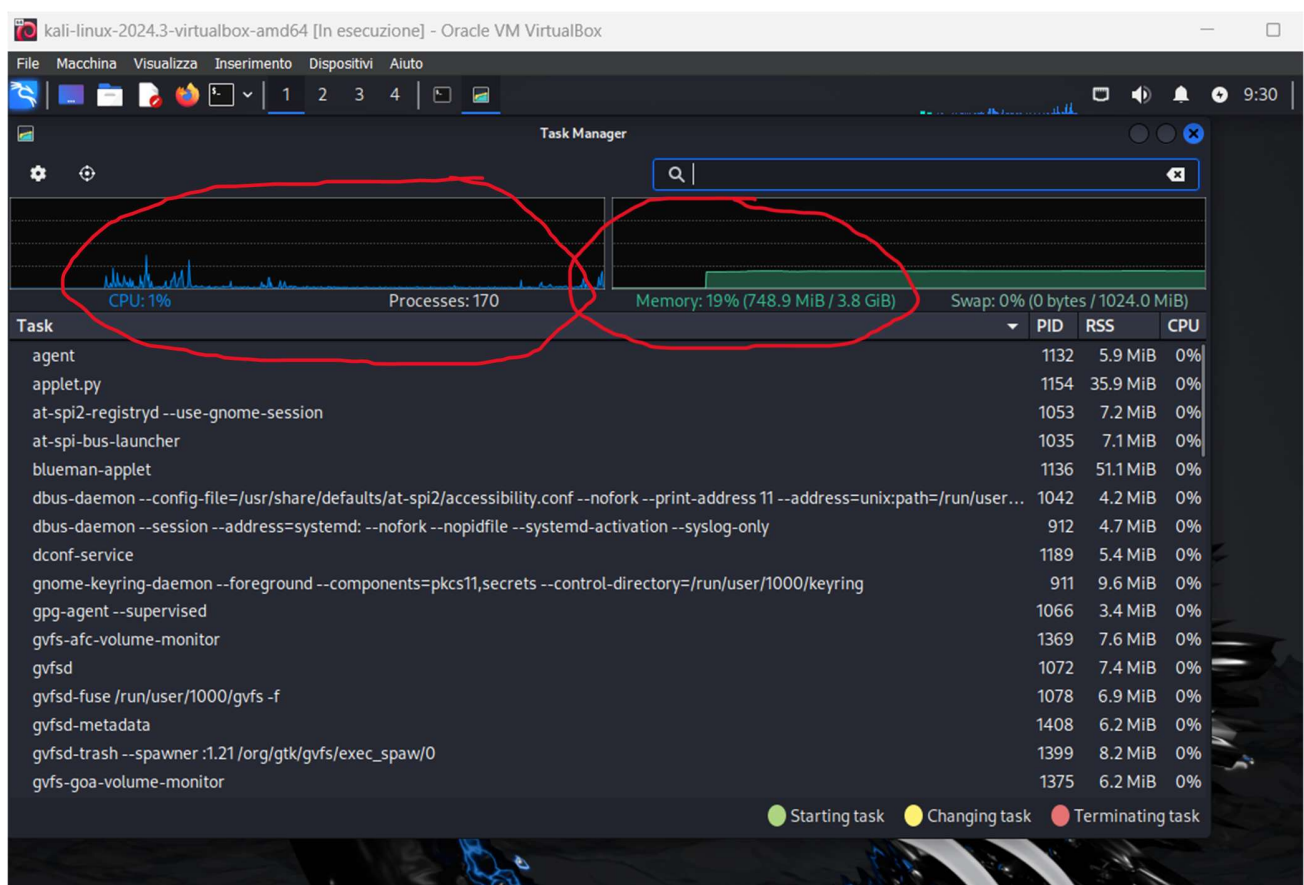
Esercizio Programmazione in C Lo scopo di oggi sarà realizzare due programmi in C

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

Opzionale:

2 - Si scriva un programma in linguaggio C che legga due valori interi e visualizzi la loro media aritmetica.

Apro Kali Linux da virtualbox, controllo che la ram e le cpu allocate siano sufficienti:



Le risorse sono sufficienti.

Ora apro il terminale e creo una cartella sul desktop in cui inserire i miei programmini in C e poi creo i programmi stessi

Per fare questo uso il terminale. Io uso windows quindi premo CTRL + ALT + T per aprire la shell

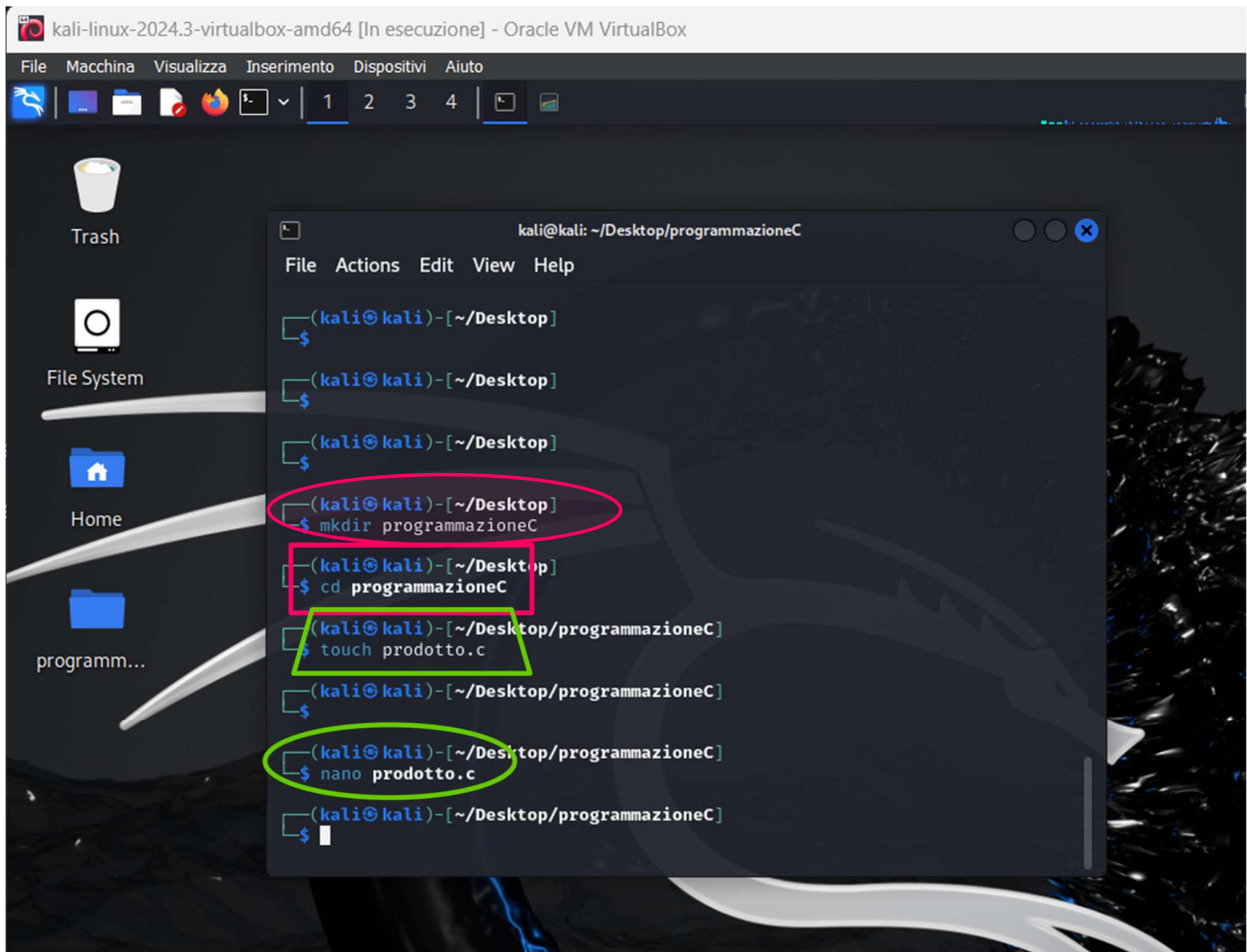
Uso i seguenti comandi:

**mkdir** per creare una cartella

**cd** per navigare all'interno di una cartella

**touch** per creare un file, può essere aggiunta l'estensione.c

**nano** per modificare il file con un editor di testo integrato nelle distribuzioni debian based



```
kali@kali: ~/Desktop/programmazioneC
File Actions Edit View Help

(kali@kali)-[~/Desktop]
$
(kali@kali)-[~/Desktop]
$
(kali@kali)-[~/Desktop]
$
(kali@kali)-[~/Desktop]
$ mkdir programmazioneC
(kali@kali)-[~/Desktop]
$ cd programmazioneC
(kali@kali)-[~/Desktop/programmazioneC]
$ touch prodotto.c
(kali@kali)-[~/Desktop/programmazioneC]
$
(kali@kali)-[~/Desktop/programmazioneC]
$ nano prodotto.c
(kali@kali)-[~/Desktop/programmazioneC]
$
```

A seguire il codice relativo al prodotto tra due numeri

Inizio importando la libreria di input - output con **#include <stdio.h>**

Aggiungo tre variabili float all'interno della funzione:

num1, num2, prodotto. Acquisisco le prime due tramite input da tastiera. Dopodiché assegno alla terza variabile il prodotto delle prime due, moltiplicandole e le mostro come output.

```
kali@kali: ~/Desktop/programmazioneC
File Actions Edit View Help
GNU nano 8.1 prodotto.c *
#include <stdio.h>

int main() {
    float num1, num2, prodotto;

    // Richiesta di input da parte dell'utente
    printf("Inserisci il primo numero: ");
    scanf("%f", &num1);
    printf("Inserisci il secondo numero: ");
    scanf("%f", &num2);

    // Calcolo del prodotto
    prodotto = num1 * num2;

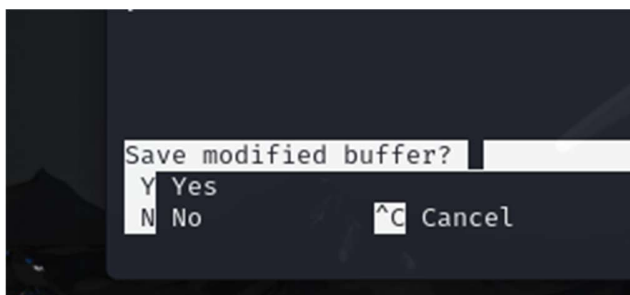
    // Output del risultato
    printf("Il prodotto di %.2f e %.2f è %.2f\n", num1, num2, prodotto);

    return 0;
}
```

[ Cancelled ]

<sup>^</sup>G Help    <sup>^</sup>O Write Out    <sup>^</sup>F Where Is    <sup>^</sup>K Cut    <sup>^</sup>T Execute  
<sup>^</sup>X Exit    <sup>^</sup>R Read File    <sup>^</sup>\ Replace    <sup>^</sup>U Paste    <sup>^</sup>J Justify

Con CTRL + X chiudo e salvo nel buffer



A seguire il codice del prossimo esercizio.

Importo la libreria di input - output.

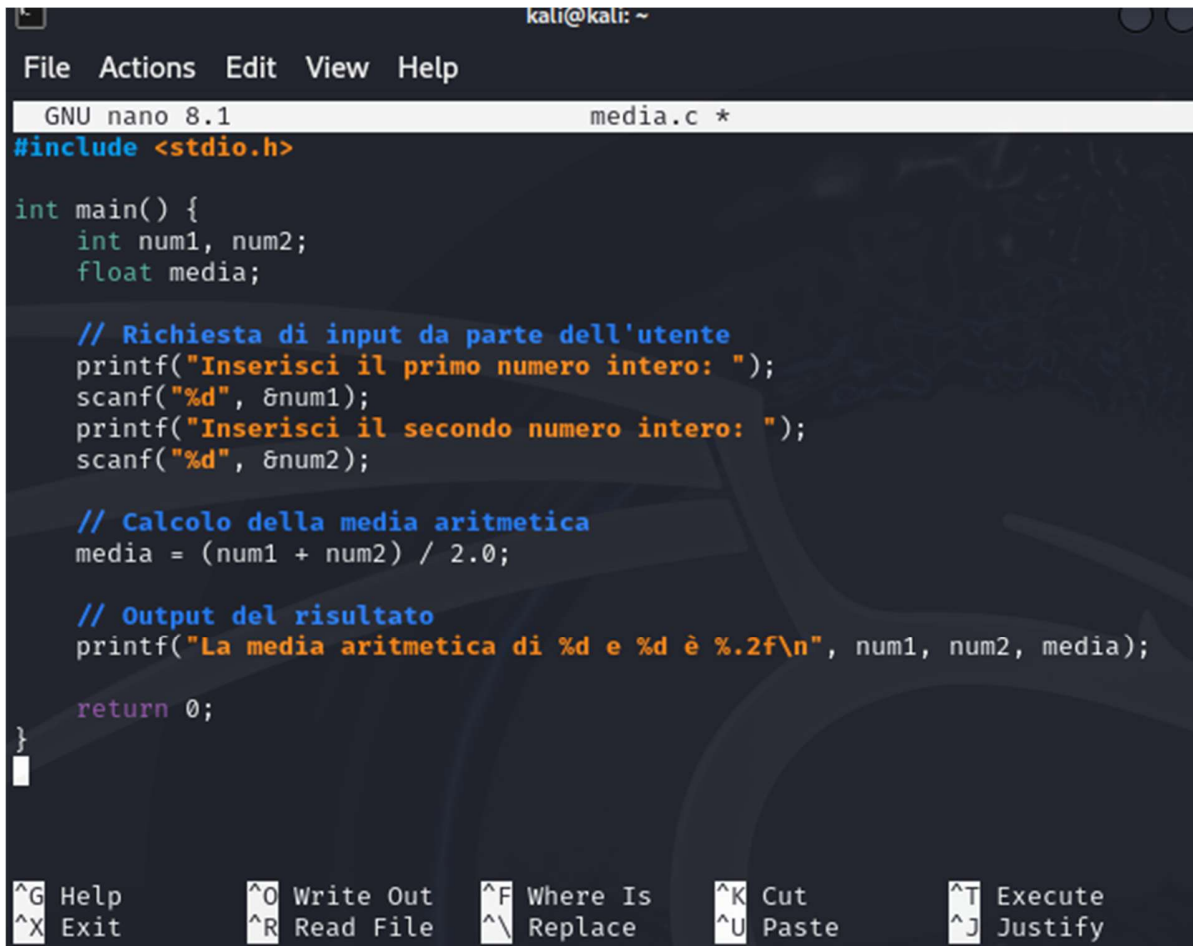
Aggiungo come variabili due variabili intere (**num1** e **num2**) e la **media** come **float**.

La variabile **float** è necessaria perché la media tra due numeri interi non è un numero intero.

Quando calcolo da media il risultato viene salvato nella variabile "**media**"

Dividendo per **2.0** e non per 2, in questo modo forzo il passaggio a float ed evito il troncamento.

Stampo quindi il risultato della **media** aritmetica.



```
File Actions Edit View Help
GNU nano 8.1 media.c *
#include <stdio.h>

int main() {
    int num1, num2;
    float media;

    // Richiesta di input da parte dell'utente
    printf("Inserisci il primo numero intero: ");
    scanf("%d", &num1);
    printf("Inserisci il secondo numero intero: ");
    scanf("%d", &num2);

    // Calcolo della media aritmetica
    media = (num1 + num2) / 2.0;

    // Output del risultato
    printf("La media aritmetica di %d e %d è %.2f\n", num1, num2, media);

    return 0;
}
```

Help Write Out Where Is Cut Execute  
Exit Read File Replace Paste Justify

Ora compilo con gcc. La sintassi prevede gcc fileinput.c -o fileoutput

Per eseguirlo uso ./fileoutput

```
/usr/bin/ld: cannot find prodotto: No such file or di  
collect2: error: ld returned 1 exit status
```

```
(kali㉿kali)-[~/Desktop/programmazioneC]  
$ gcc prodotto.c -o prodotto
```

```
(kali㉿kali)-[~/Desktop/programmazioneC]  
$ ./prodotto
```

```
Inserisci il primo numero: 1
```

```
Inserisci il secondo numero: 2
```

```
Il prodotto di 1.00 e 2.00 è 2.00
```

```
(kali㉿kali)-[~/Desktop/programmazioneC]  
$
```

Il programma è ok e funziona

Analogamente, guardiamo la media.

```
(kali㉿kali)-[~/Desktop/programmazioneC]  
$ gcc media.c -o media
```

```
(kali㉿kali)-[~/Desktop/programmazioneC]  
$ ./media
```

```
Inserisci il primo numero intero: 4
```

```
Inserisci il secondo numero intero: 8
```

```
La media aritmetica di 4 e 8 è 6.00
```

```
(kali㉿kali)-[~/Desktop/programmazioneC]  
$
```