

Consegna S2/L3
di
Giuseppe Lupoi



Traccia

La traccia di oggi richiede:

- Si scriva un programma che esegua l'operazione moltiplicazione tra i due numeri inseriti dall'utente
- Si scriva un programma in linguaggio C che legga due valori interi e visualizzi la loro media aritmetica

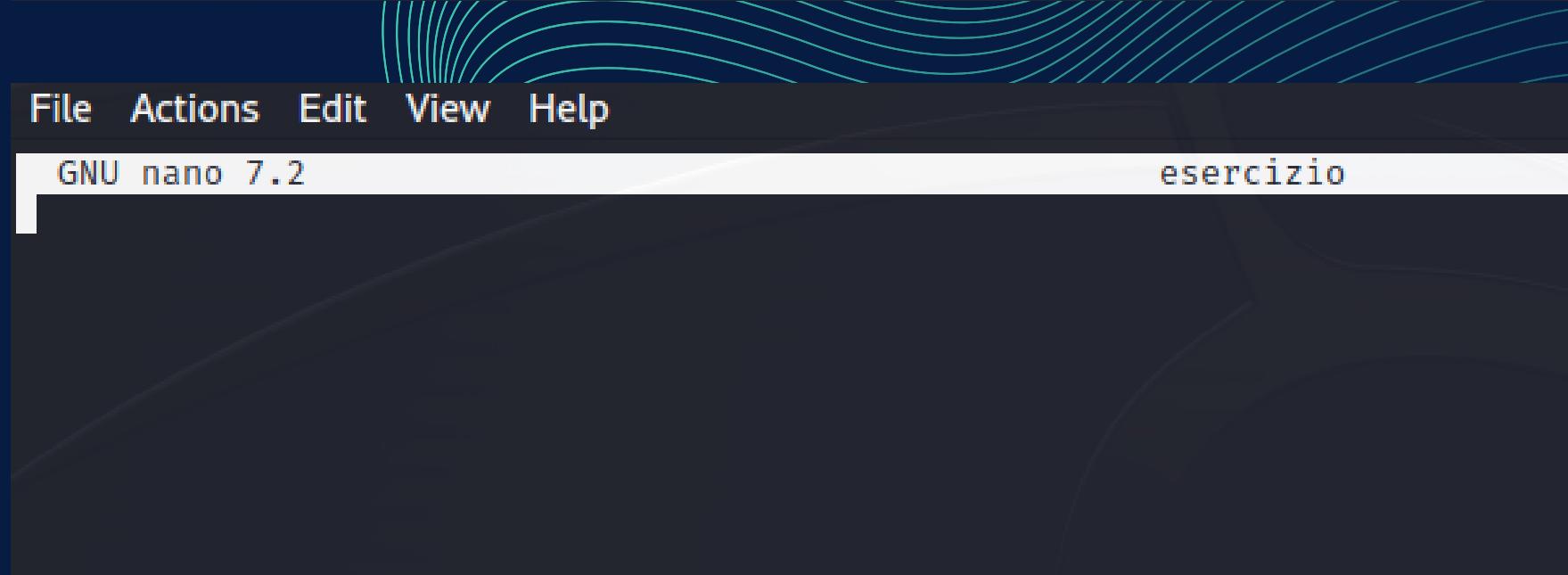
Step 1 - Procederemo dunque creando una cartella dove inserire il file, continueremo creando il file editabile

Subito sotto vediamo appunto che il file è stato creato correttamente e possiamo dunque modificarlo

```
(kali㉿kali)-[~]
└─$ mkdir /home/kali/Desktop/esercizio.d

(kali㉿kali)-[~]
└─$ cd /home/kali/Desktop/esercizio.d

(kali㉿kali)-[~/Desktop/esercizio.d]
└─$ nano esercizio
```



Step 2 - Andremo dunque a creare l' "executive" di questi due file per poi poterli utilizzare, io li avevo già rinominati in precedenza, mi sono poi accorto che ciò non era necessario perchè come vediamo dal comando dei due screen dopo aver specificato il file ci chiederà di dargli un nome.

Ho scelto dunque:

- 1, per il codice della moltiplicazione
- 2, per il codice della media

```
(kali㉿kali)-[~/esempio]
$ gcc -g moltiplicazione.c -o 1
```

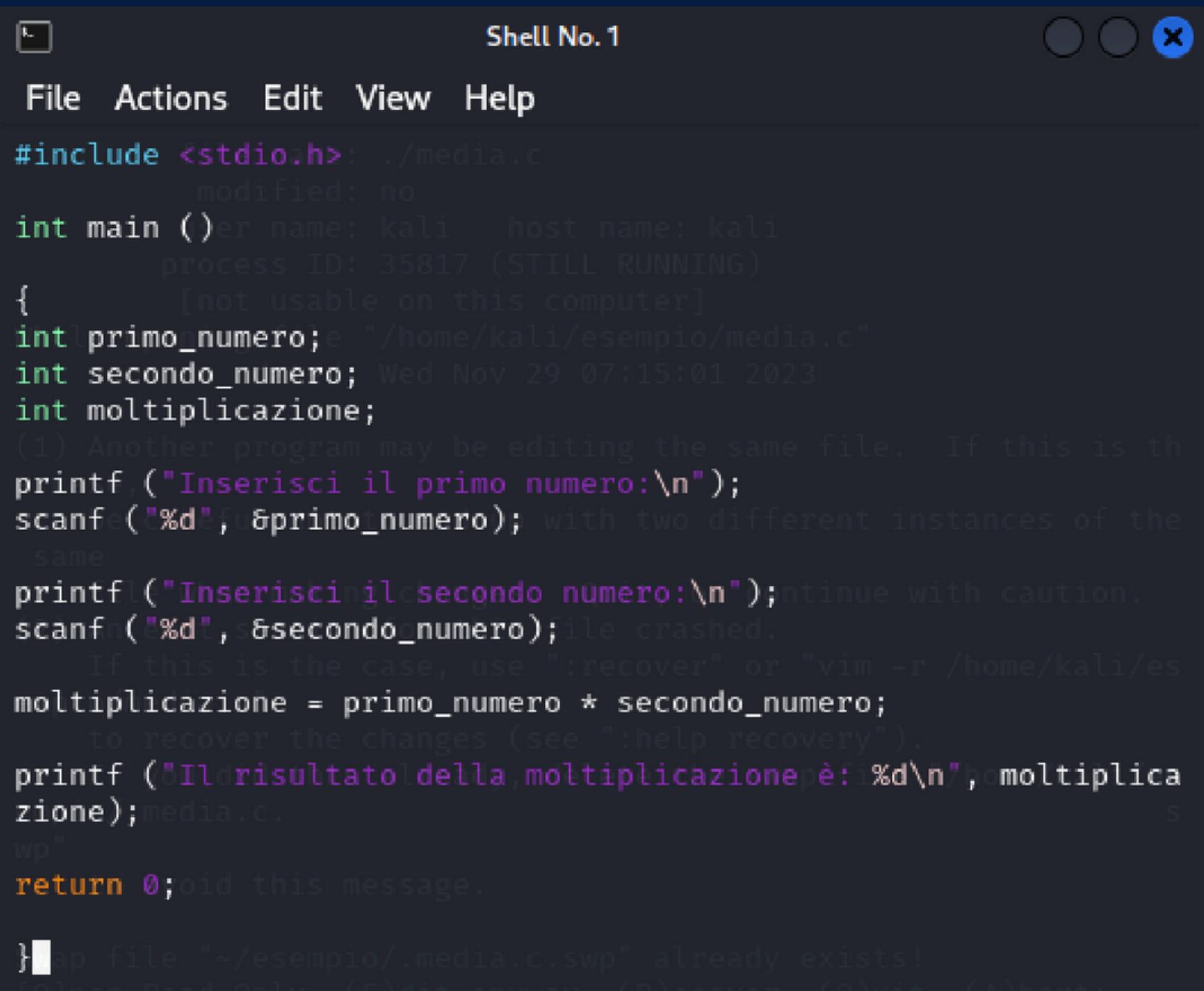
```
(kali㉿kali)-[~/esempio]
$ gcc -g media.c -o 2
```

Step 3 – Andremo dunque ad editare i nostri due file con il comando **<nano ./"nome file">**

```
(kali㉿kali)-[~/esempio]
$ nano ./1
```

```
(kali㉿kali)-[~/esempio]
$ nano ./2
```

Step 4 - Creeremo il primo codice relativo ad un programma che esegua una moltiplicazione tra due valori scelti dall'utente



The screenshot shows a terminal window titled "Shell No. 1". The window has a dark theme with light-colored text. The code displayed is as follows:

```
#include <stdio.h>
int main ()
{
    int primo_numero;
    int secondo_numero;
    int moltiplicazione;
    printf("Inserisci il primo numero:\n");
    scanf("%d", &primo_numero);
    printf("Inserisci il secondo numero:\n");
    scanf("%d", &secondo_numero);
    moltiplicazione = primo_numero * secondo_numero;
    printf ("Il risultato della moltiplicazione è: %d\n", moltiplicazione);
    return 0;
}
```

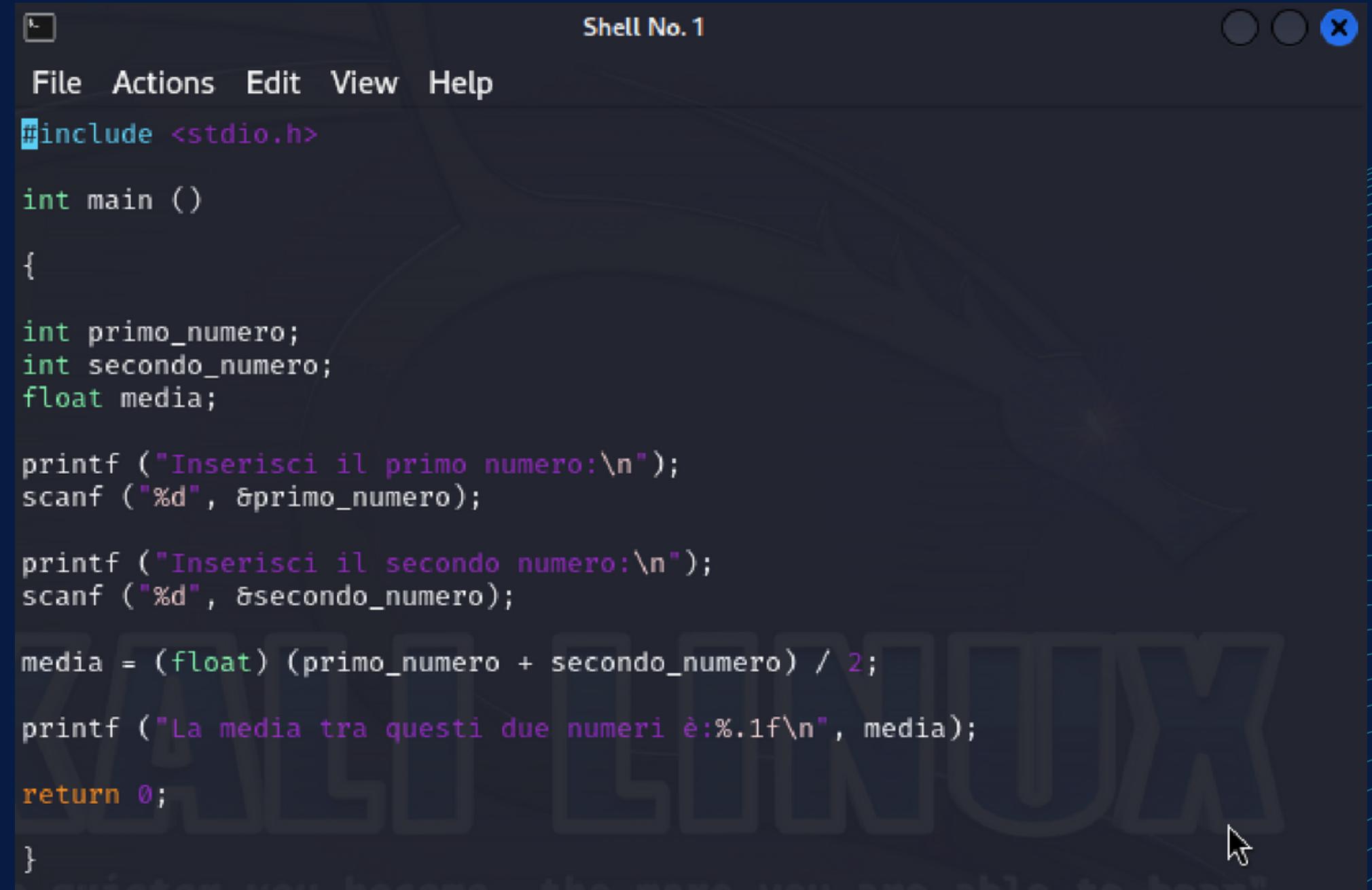
The terminal also displays some system information at the top, including the host name "kali", process ID "35817", and the date and time "Wed Nov 29 07:15:01 2023". A warning message about another program editing the file is present.

Step 5 – Lancieremo quindi il codice con il con il comando comando `./1` e ne testeremo la funzionalità.

Come possiamo vedere dallo screen qua a destra se proviamo a fare una moltiplicazione tra 2 e 3 ci restituirà 6 come risultato, quindi il codice funziona correttamente

```
(kali㉿kali)-[~/esempio]
$ ./1
Inserisci il primo numero:
2
Inserisci il secondo numero:
3
Il risultato della moltiplicazione è: 6
```

Step 6 - Fatto ciò scriveremo ora un codice che eseguirà una media tra due valori scelti dall'utente



The image shows a terminal window titled "Shell No. 1". The window has a dark background with light-colored text. At the top, there is a menu bar with "File", "Actions", "Edit", "View", and "Help". Below the menu, the code is displayed in a monospaced font:

```
#include <stdio.h>

int main ()
{
    int primo_numero;
    int secondo_numero;
    float media;

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

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

    media = (float) (primo_numero + secondo_numero) / 2;

    printf ("La media tra questi due numeri è:%.1f\n", media);

    return 0;
}
```

Step 7 - Lanceremo quindi anche questo codice con **./2** e ne testeremo la funzionalità.
Come vedete se proviamo a fare una media tra i valori 4 e 5 come risultato avremo 4.5 dunque il codice è corretto

```
(kali㉿kali)-[~]
$ ./2
Inserisci il primo numero:
4
Inserisci il secondo numero:
5
La media tra questi due numeri è:4.5
```