

Oggi creiamo 2 script C molto semplici.

Prima iniziamo andando sul nostro Desktop, e creiamo il directory dove inseriremo nostri file.

Poi creiamo un file .c, che ci permetterà di scrivere codice con il linguaggio di programmazione C.

Con l'editore di testo `sudo` andremo dentro il file per editarlo.

```
kali@kali: ~/Desktop/program_c
File Actions Edit View Help
(kali@kali)-[~]
$ Desktop
(kali@kali)-[~/Desktop]
$ sudo mkdir program_c
[sudo] password for kali:
(kali@kali)-[~/Desktop]
$ mkdir program_c
(kali@kali)-[~/Desktop]
$ program_c
(kali@kali)-[~/Desktop/program_c]
$ touch program.c
(kali@kali)-[~/Desktop/program_c]
$ ls
program.c
(kali@kali)-[~/Desktop/program_c]
$ sudo nano program.c
(kali@kali)-[~/Desktop/program_c]
$ gcc -g program.c
/usr/bin/ld: /usr/lib/gcc/x86_64-linux-gnu/13/../../../../x86_64-linux-gnu/Scrt1.o: in function `_start':
(.text+0x17): undefined reference to `main'
collect2: error: ld returned 1 exit status
(kali@kali)-[~/Desktop/program_c]
$ sudo nano program.c
```

"the quieter you become, the more you are"

Qua ho già scritto tutto il programma di moltiplicazione.

`#include <stdio.h>` ci include la libreria `stdio.h`, che ci dà accesso a comandi che useremo nel codice.

`int main()` specifica che questo sarà il codice principale.

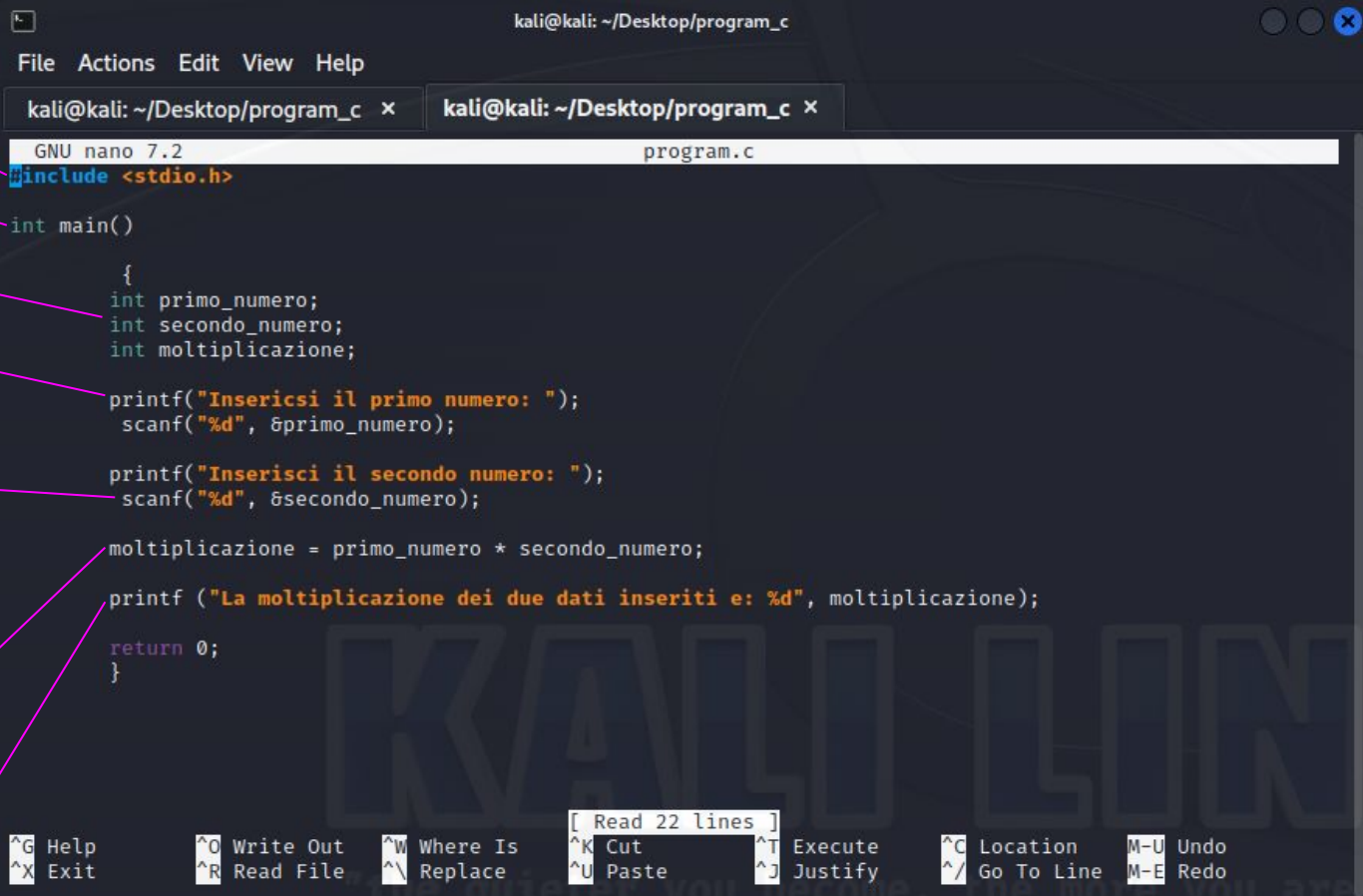
`int` ci indicherà il tipo di funzione.

`printf` ci lascerà scrivere testo dentro di "" che poi sarà presentato allo usuario dello script come testo da leggere, in questo caso istruzioni di cosa fare.

`scanf` aspetterà un input dalla tastiera, in questo caso % si scrive prima del argomento, l'argomento `d` cioè, una variabile di numero intero. A questo argomento, li daremo il nome di `secondo_numero`, scritto dopo il simbolo `&`.

Qui dichiariamo che la parola `moltiplicazione` sarà il puntatore la variabile chiamata `primo_numero` moltiplicato con l'altra chiamata `secondo_numero` usando il simbolo `*`

Facciamo un `printf` finale, e indichiamo che la variabile da far vedere sarà la moltiplicazione eseguita nella riga scorsa.



```
kali@kali: ~/Desktop/program_c
File Actions Edit View Help
kali@kali: ~/Desktop/program_c x kali@kali: ~/Desktop/program_c x
GNU nano 7.2 program.c
#include <stdio.h>

int main()

{
    int primo_numero;
    int secondo_numero;
    int moltiplicazione;

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

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

    moltiplicazione = primo_numero * secondo_numero;

    printf ("La moltiplicazione dei due dati inseriti e: %d", moltiplicazione);

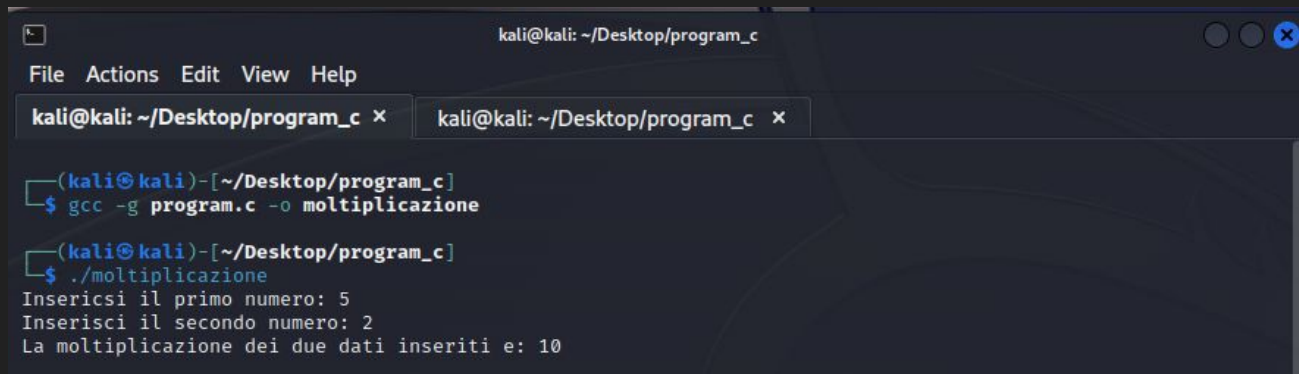
    return 0;
}
```

^G Help ^O Write Out ^W Where Is [Read 22 lines] ^K Cut ^T Execute ^C Location M-U Undo
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo

Una volta salvato il file .c, possiamo andare a compilarlo con il comando **gcc** eseguito da **-g** per indicare il file che vogliamo compilare, e **-o** per dare un nome al nuovo file eseguibile.

Dopo aver creato il nuovo file eseguibile, usiamo il comando **./** eseguito dal nome del file per avviare.

Funziona!



```
kali@kali: ~/Desktop/program_c
File Actions Edit View Help
kali@kali: ~/Desktop/program_c x kali@kali: ~/Desktop/program_c x
(kali@kali)-[~/Desktop/program_c]
$ gcc -g program.c -o moltiplicazione
(kali@kali)-[~/Desktop/program_c]
$ ./moltiplicazione
Inserisci il primo numero: 5
Inserisci il secondo numero: 2
La moltiplicazione dei due dati inseriti e: 10
```

Ora facciamo uno script per avere la media fra 2 numeri interi.

Il codice é molto semplice a quello per la moltiplicazione, con le piccole differenze di un paio di nomi di puntatori.

In questa operazione, sommiamo il primo numero con il secondo, e poi dividiamo il risultato in 2, così come abbiamo fatto alla elementaria.

kali@kali: ~/Desktop/program_c

File Actions Edit View Help

kali@kali: ~/Desktop/program_c x kali@kali: ~/Desktop/program_c x

GNU nano 7.2 program2.c

```
#include <stdio.h>

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

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

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

    media = (primo_numero + secondo_numero) /2;

    printf ("La media dei due dati inseriti e: %d", media);

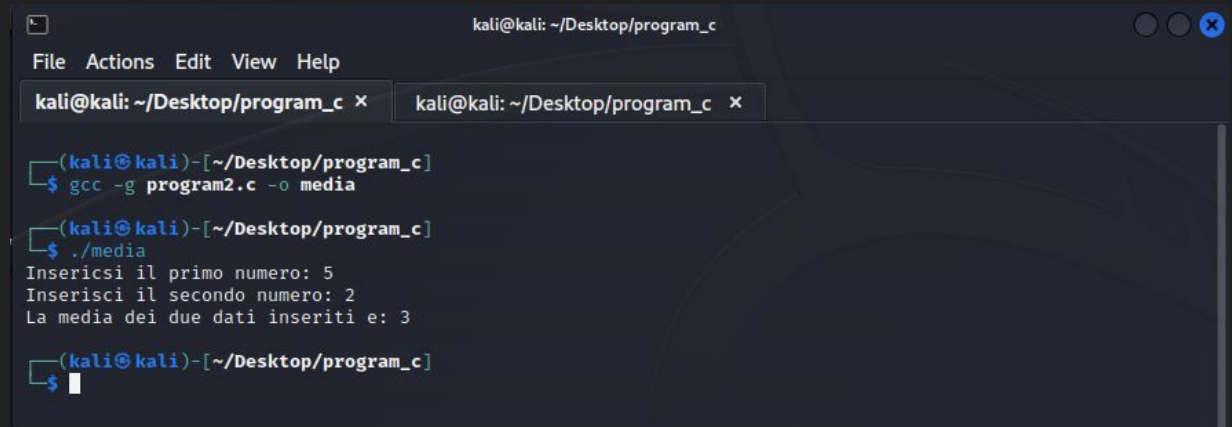
    return 0;
}
```

[Read 23 lines]

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo
^X Exit	^R Read File	^\\ Replace	^U Paste	^J Justify	^/_ Go To Line	M-E Redo

Compiliamo come
abbiamo fatto
prima, eee...

Funziona!



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

(kali@kali)-[~/Desktop/program_c]
$ gcc -g program2.c -o media

(kali@kali)-[~/Desktop/program_c]
$ ./media
Inserisci il primo numero: 5
Inserisci il secondo numero: 2
La media dei due dati inseriti e: 3

(kali@kali)-[~/Desktop/program_c]
$
```

Thanks for
watching!