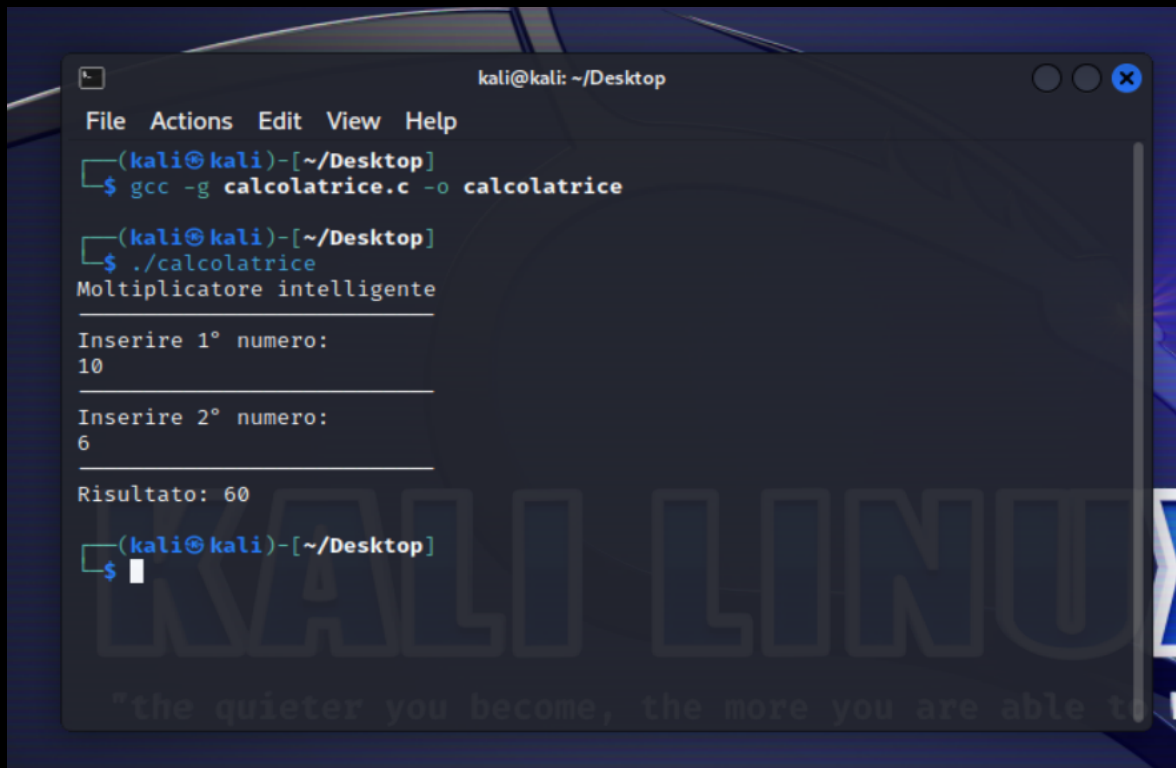


PRATICA S2/L3

Front-end



```
kali@kali: ~/Desktop
File Actions Edit View Help
(kali@kali)-[~/Desktop]
$ gcc -g calcolatrice.c -o calcolatrice

(kali@kali)-[~/Desktop]
$ ./calcolatrice
Moltiplicatore intelligente
Inserire 1° numero:
10
Inserire 2° numero:
6
Risultato: 60

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

La seguente immagine mostra il risultato di un programma sviluppato tramite il linguaggio di programmazione C. si tratta un moltiplicatore basico capace di calcolare la somma di due numeri.

Back-end



```
File Actions Edit View Help
#include <stdio.h>
int main ()
{

    int x;
    int z;
    int y;
    printf("Moltiplicatore intelligente\n");
    printf("_____\n");
    printf("Inserire 1° numero: \n");
    scanf("%d", &x);
    printf("_____\n");
    printf("Inserire 2° numero: \n");
    scanf("%d", &z);

    y = x * z;
    printf("_____\n");
    printf("Risultato: %d\n", y);
    return 0;
}

~
~
~ "the quieter you become, the more you are able to h
~

9,40 All
```

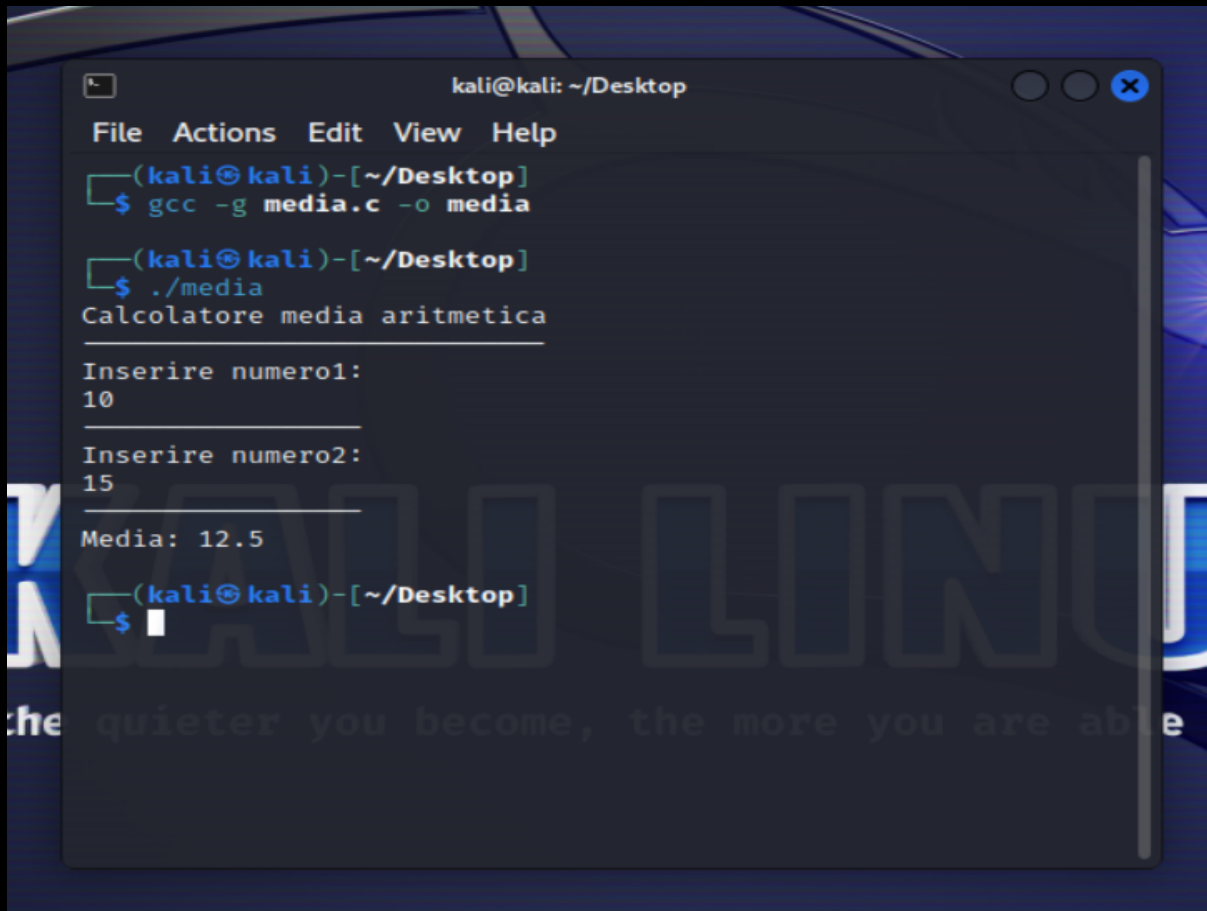
L'immagine seguente invece rappresenta il back-end del programma, ovvero la parte non visibile agli utenti composta da una serie di comandi che il programma utilizza per svolgere le sue funzioni.

Spiegazione funzione comandi:

- il primo comando è preceduto da `<#>`, rappresenta la direttiva del processore, che indica di utilizzare la libreria `<stdio.h>`, contenente le definizioni delle funzioni da noi utilizzate.
- `<Main()>`, molto importante perché indica al programma da dove deve partire per leggere le istruzioni.
- Parentesi graffe `{}` in apertura e in chiusura, la loro funzione è delimitare un blocco di codice.
- `int x/y/z`, rappresentano le variabili.
- `printf("messaggio");`, funzione utilizzata per mostrare sullo schermo i caratteri all'interno degli apici.
- `scanf`, utilizzato per leggere l'input digitato dall'utente.
- `y = x * z`, rappresenta il valore di y, che è uguale alla moltiplicazione tra x e z
- `return 0`, svolge la funzione di restituire il controllo al sistema.

2° Programma

Front-end



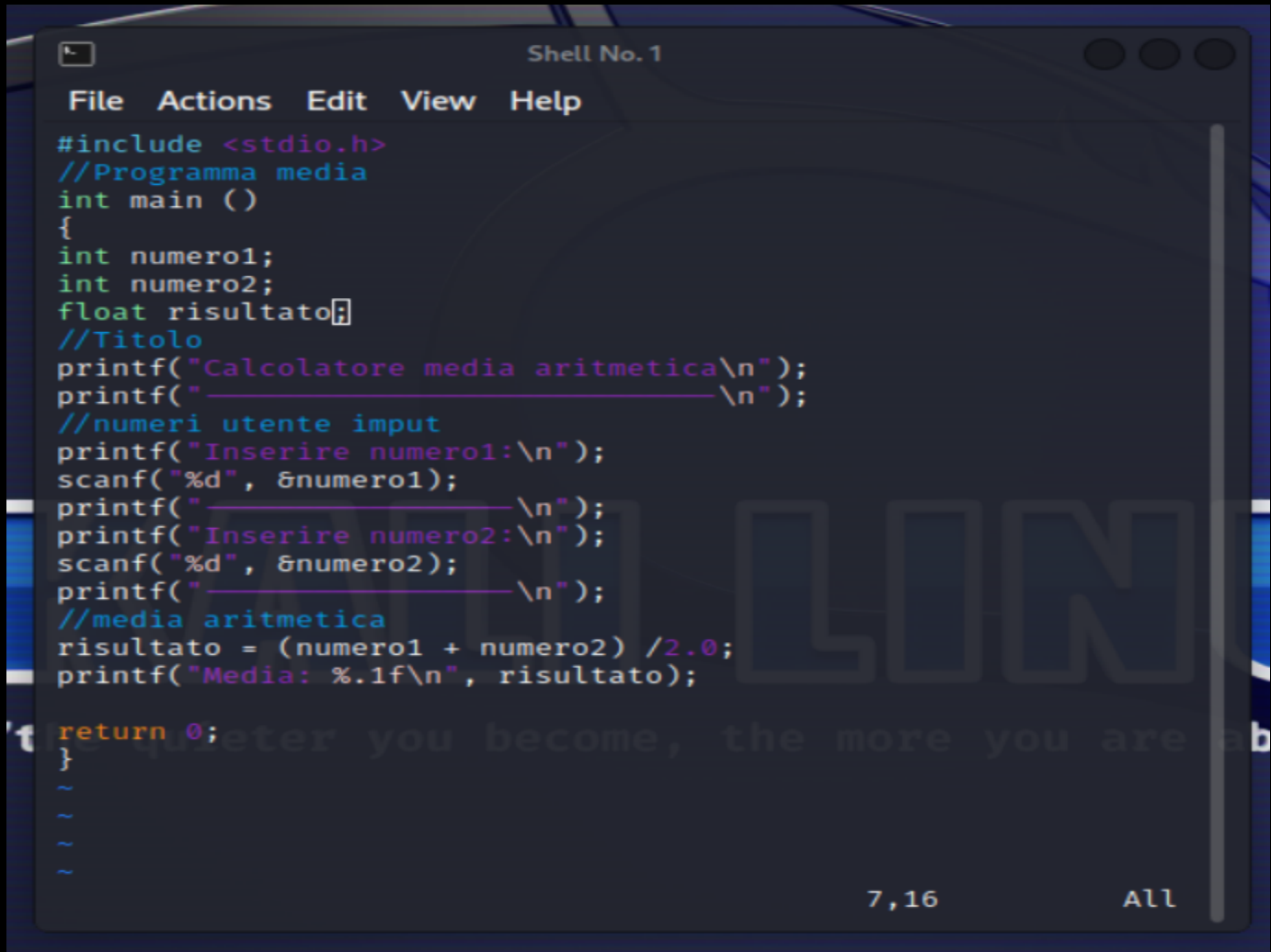
```
kali@kali: ~/Desktop
File Actions Edit View Help
(kali@kali)-[~/Desktop]
$ gcc -g media.c -o media

(kali@kali)-[~/Desktop]
$ ./media
Calcolatore media aritmetica
Inserire numero1:
10
Inserire numero2:
15
Media: 12.5

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

Il programma rappresenta un calcolatore di media aritmetica fra due numeri interi.

Back-end



```
Shell No. 1
File Actions Edit View Help
#include <stdio.h>
//Programma media
int main ()
{
    int numero1;
    int numero2;
    float risultato;
    //Titolo
    printf("Calcolatore media aritmetica\n");
    printf("-----\n");
    //numeri utente input
    printf("Inserire numero1:\n");
    scanf("%d", &numero1);
    printf("-----\n");
    printf("Inserire numero2:\n");
    scanf("%d", &numero2);
    printf("-----\n");
    //media aritmetica
    risultato = (numero1 + numero2) /2.0;
    printf("Media: %.1f\n", risultato);

    return 0;
}
~
~
~
7,16 All
```

La serie di comandi utilizzata è molto simile al programma rappresentato in precedenza con delle piccole modifiche. Ho utilizzato la variabile **float** per definire il risultato, in modo tale da avere la possibilità di generare un esito decimale.

Di conseguenza ho utilizzato `%1f` per indicare la variabile float ed eliminare eventuali cifre oltre il primo numero dopo la virgola.

Come risultato ho indicato che è uguale alla somma di numero1 e numero2 diviso 2, in modo da ottenere l'esito desiderato.