

REPORT 28/03/2025

```
File Actions Edit View Help
GNU nano 8.1 linguaccio.c
#include <stdio.h> // libreria standard del linguaggio C che fornisce funzioni per la gestione delle operazioni di input e output (ad esempio printf e scanf)
#include <math.h> // libreria per una serie di funzioni matematiche (ad esempio: sqrt() e M_PI)

int main () { // usiamo int main perchè indica il tipo di valore che la funzione "main" restituirà al sistema operativo quando il programma termina. Se il programma è terminato correttamente restituirà "return 0;"
float D; // float lo usiamo per lavorare con i numeri reali di 4 byte, "float" permette di gestire numeri con la virgola (decimali), perchè se usassimo "int" funzionerebbe solo con numeri interi, ovvero senza virgola (decimali)
float area_quadrato;
float area_triangolo;
float area_cerchio;

// input dell'utente
printf("qui devi inserire un numero reale: "); // l'utente capisce che deve inserire un numero, senza printf il programma sarebbe confuso, l'utente non saprebbe dove inserire un valore
scanf("%f",&D); // scanf legge l'input dell'utente e salvarlo nella variabile "D", %f legge un numero con la virgola (float), %d significa memorizza il valore in "D", quindi l'utente digita un numero e il programma lo salva in "D"

// calcolo dell'area di un quadrato, triangolo, cerchio
area_quadrato = D * D; // lato per lato
area_triangolo = M_PI * ( D / 2 ) * ( D / 2 ); // p greco moltiplica lato diviso 2, che moltiplica il lato diviso 2
area_cerchio = (sqrt(3) / 4) * D * D; // sqrt è specifica per il tipo di dato "float" più efficiente con numeri a precisione singola rispetto a sqrt, destinato a numeri di tipo "double" ovvero precisione doppia

// stampa dei risultati
printf("Area del quadrato: %.2f\n", area_quadrato); // andiamo a scrivere area del quadrato, (anche gli altri), perchè altrimenti l'utente visualizzerebbe solamente il numero (%.2f) e l'utente potrebbe non sapere a cosa si riferisce
printf("Area del triangolo equilatero: %.2f\n", area_triangolo); // "%.2f" stampa il numero come un numero in virgola con due decimali (esempio: 16.00 e non 16,00000)
printf("Area del cerchio: %.2f\n", area_cerchio); // "\n" inserisce una nuova riga dopo il valore stampato, separando le uscite nel terminale

return 0; // deve dare come valore "0" altrimenti il codice risulterà non corretto/errato
}
```

qui in alto riporta il codice per il calcolo delle aree.

Poi ho usato il comando per eseguirlo:

“gcc -o programma linguaccio.c”

“./programma”

```
(kali㉿kali)-[~]
$ ./programma
qui devi inserire un numero reale: 5
Area del quadrato: 25.00
Area del triangolo equilatero: 19.63
Area del cerchio: 10.83
```

dandoci i valori per ogni figura