

**Traccia: Si scriva un programma in Python che in base alla scelta dell'utente permetta di calcolare il perimetro di diverse figure geometriche (scegliete pure quelle che volete voi).**

**Per la risoluzione dell'esercizio abbiamo scelto:**

**Quadrato (perimetro = lato\*4.**

**Cerchio (circonferenza = 2\*pi greco\*r).**

**Rettangolo (perimetro= base\*2 + altezza\*2**

Uso WLS per comodità negli screenshot e nel fare copia incolla

Prima di iniziare a scrivere il codice facciamo alcune considerazioni.

- Facciamo un po' di error handling ed cerchiamo, nei limiti del possibile e del sensato, di estendere il range d'uso
- Devo fare attenzione agli input nella scelta della figura geometrica, quali: nomi errati, maiuscole
- Devo fare attenzione a input non numerici nell'input dei lati/raggio **(TODO)**
- Importo la libreria math per avere pigreco
- Voglio avere un codice simil ordinato quindi creo varie funzioni che vengono chiamate

```
marco@PORTATILE: ~/python x + v
GNU nano 7.2                                perimetro.py
import math

def quadrato():
    lato = float(input("Inserisci la lunghezza del lato del quadrato: "))
    perimetro = lato * 4
    print(f"Il perimetro del quadrato è: {perimetro}")

def rettangolo():
    base = float(input("Inserisci la lunghezza di uno dei due lati del rettangolo: "))
    altezza = float(input("Inserisci la lunghezza dell'altro lato del rettangolo: "))
    perimetro = 2*(base+altezza)
    print(f"Il perimetro del rettangolo è: {perimetro}")

def cerchio():
    raggio = float(input("Inserisci il raggio del cerchio: "))
    perimetro = raggio * 2 * math.pi
    print(f"La circonferenza del cerchio è: {perimetro}")

def calcola_perimetro():
    print("Scegli la figura geometrica per calcolare il perimetro (quadrato, cerchio, rettangolo):")
    choice = input("Inserisci il nome della figura: ").strip().lower()
    match choice:
        case "quadrato":
            quadrato()
        case "rettangolo":
            rettangolo()
        case "cerchio":
            cerchio()
        case _:
            print("Scelta non valida.")
            calcola_perimetro()

calcola_perimetro()
```

Guardiamo insieme il codice, dall'alto al basso

## Importazione del modulo

Qui stiamo importando il modulo `math`, che ci servirà per calcolare la circonferenza del cerchio usando il valore di  $\pi$  (pi greco).

## Funzione per il quadrato

Questa funzione chiede all'utente di inserire la lunghezza del lato di un quadrato. Poi calcola il perimetro moltiplicando la lunghezza del lato per 4 e stampa il risultato.

## Funzione per il rettangolo

Questa funzione chiede all'utente di inserire le lunghezze dei due lati di un rettangolo. Poi calcola il perimetro moltiplicando la base per l'altezza e stampa il risultato. (Nota: qui sembra che ci sia un errore, perché dovrebbe essere  $(base + altezza) * 2$ ).

## Funzione per il cerchio

Questa funzione chiede all'utente di inserire il raggio di un cerchio. Poi calcola la circonferenza moltiplicando il raggio per 2 e poi per  $\pi$  (pi greco) e stampa il risultato.

## Funzione principale per scegliere la figura geometrica

Questa funzione principale chiede all'utente di scegliere una figura geometrica tra quadrato, rettangolo e cerchio. In base alla scelta, chiama la funzione corrispondente.

Se l'utente inserisce una scelta non valida, l'utente viene informato e la funzione richiama se stessa.

### Esecuzione della funzione principale

Infine, viene chiamata la funzione `calcola_perimetro()` che avvia tutto il processo.

Alcuni pezzi di codice interessanti:

```
choice = input("Inserisci il nome della figura: ").strip().lower()
match choice:
    case "quadrato":
        quadrato()
    case "rettangolo":
        rettangolo()
    case "cerchio":
        cerchio()
    case _:
        print("Scelta non valida.")
        calcola_perimetro()
```

**.strip():** Questo metodo rimuove eventuali spazi bianchi all'inizio e alla fine della stringa inserita dall'utente. Ad esempio, se l'utente inserisce " quadrato ", diventerà "quadrato".

**.lower():** Questo metodo converte tutti i caratteri della stringa in minuscolo. Quindi, se l'utente inserisce "QUADRATO", diventerà "quadrato".

**match choice::** Questo è un costrutto di controllo che confronta il valore di choice con diversi casi.

**case "quadrato":** Se choice è uguale a "quadrato", allora esegue il blocco di codice sotto questo caso.

```
def rettangolo():
    base = float(input("Inserisci la lunghezza di uno dei due lati del rettangolo :"))
    altezza = float(input("Inserisci la lunghezza dell'altro lato del rettangolo: "))
    perimetro = 2*(base+altezza)
    print(f"Il perimetro del rettangolo è: {perimetro}")
```

**float(...):** Converte l'input dell'utente da stringa a numero decimale (float), in modo da poter eseguire calcoli matematici.

Si utilizza una f-string (stringa formattata) per inserire il valore del perimetro all'interno della stringa di output. Le f-string, o formatted string literals, sono un modo molto comodo e leggibile per formattare le stringhe in python, permettono di inserire espressioni all'interno delle stringhe, racchiudendole tra parentesi graffe {}.

Per mancanza di tempo non ho messo il controllo sugli input. Si può fare creando una funzione da chiamare nella definizione delle variabili lato/altezza/raggio. Questa funzione può essere strutturata con un while e restituisce come output il numero una volta che questo è effettivamente un numero e non un carattere di natura differente