

ESERCIZIO S3/L3

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 = $2 \cdot \pi \cdot r$).
- Rettangolo (perimetro = base*2 + altezza*2).

```
1 import math
2
3 def calcolo_perimetro():
4     print("Andiamo a calcolare il perimetro di una figura geometrica\n")
5     print("Quadrato >>> Q \n Cerchio >>> C \n Rettangolo >>> R \n")
6
7     print("Quale figura scegli?")
8     sc = input(">>> ")
9
10    if sc == "Q":
11        print("Il quadrato!")
12        lato = float(input("Inserire il valore del lato "))
13        print("Il perimetro del quadrato è ", lato*4)
14
15    elif sc == "C":
16        print("Il cerchio!")
17        r = float(input("Inserire il valore del raggio "))
18        ris = 2*r*math.pi
19        print("La circonferenza del cerchio è ", round(ris, 2))
20
21    elif sc == "R":
22        print("Il rettangolo!")
23        b = float(input("Inserire il valore della base "))
24        a = float(input("Inserire il valore dell'altezza "))
25        print("Il perimetro del rettangolo è ", b*2+a*2)
26
27    else:
28        print("Riprova")
29
30    calcolo_perimetro();
31
```

Come prima cosa abbiamo importato la libreria math per poter utilizzare le sue funzioni come esempio il pi greco che andremo ad usare nel calcolo della circonferenza del cerchio.

Con "def calcolo_perimetro()" identifichiamo il nome della nostra funzione "perimetro", il programma propone all'utente di scegliere tra diverse figure e di calcolarne perimetro e area. Questo è reso possibile grazie alla funzione "print" che ci mostra a schermo il contenuto all'interno delle sue parentesi e al comando "input" che dà la possibilità all'utente di inserire dei valori.

Specificando il tipo di variabile "float" indichiamo a Python di convertire l'input dell'utente in un numero decimale che altrimenti sarebbe interpretato come stringa, non permettendoci di effettuare operazioni matematiche.

In seguito andremo a utilizzare le condizioni if, elif e else, che consentono al programma di eseguire operazioni diverse in base alla scelta dell'utente, garantendo una corretta esecuzione del calcolo del perimetro per ogni figura geometrica.

Avendo la possibilità di ottenere un risultato con diversi numeri dopo la virgola, si utilizza la funzione "round(numero, ndigit)" dove in questo caso il numero sta per il risultato del calcolo e ndigit sono le cifre che vogliamo visualizzare dopo la virgola.

Per concludere, richiamiamo la funzione "calcolo_perimetro();" per l'esecuzione del programma.

```
(kali@kali)-[~/Desktop]
$ python python.py
Andiamo a calcolare il perimetro di una figura geometrica

Quadrato >>> Q
Cerchio >>> C
Rettangolo >>> R

Quale figura scegli?
>>> Q
Il quadrato!
Inserire il valore del lato 6
Il perimetro del quadrato è 24.0

(kali@kali)-[~/Desktop]
$ python python.py
Andiamo a calcolare il perimetro di una figura geometrica

Quadrato >>> Q
Cerchio >>> C
Rettangolo >>> R

Quale figura scegli?
>>> C
Il cerchio!
Inserire il valore del raggio 6.5
La circonferenza del cerchio è 40.84

(kali@kali)-[~/Desktop]
$ python python.py
Andiamo a calcolare il perimetro di una figura geometrica

Quadrato >>> Q
Cerchio >>> C
Rettangolo >>> R

Quale figura scegli?
>>> R
Il rettangolo!
Inserire il valore della base 10
Inserire il valore dell'altezza 6
Il perimetro del rettangolo è 32.0
```

TEAM 5:

Zhongshi Liu (Team Leader)

Michael Andreoli

Donato Tralli

Stefano Cesaroni

Otman Hmich

Ayoub Mgoun