

# PREZENTACJA Z MODUŁU 1

Program obliczający pola figur płaskich

---

Autorzy:

Marcin Ciszek

Konrad Bogacz

# UŻYTE BIBLIOTEKI

```
import os          #biblioteka pozwalająca czyszcic okno konsoli  
import math        #biblioteka zawierająca liczbę PI
```

OS – Biblioteka potrzebna do użycia funkcji `os.system('cls')`

Math – Biblioteka potrzebna do użycia liczby PI w liczeniu pola koła

# SPRAWDZENIE POPRAWNOŚCI WPISANYCH DANYCH

```
def Sprawdzanie(): #funkcja sprawdzajaca czy podana wartosc jest poprawna
    while True:
        try: #probujemy pobrac wartosc
            wartosc = float(input(' >'))
            break
        except ValueError: #pomijamy wartosc ktora prowadzi do bledu programu
            print('\n Wpisz poprawna liczbe!')
    return wartosc #funkcja zwraca wartosc jesli jest ona poprawna
```

# FUNKCJA MENU GŁÓWNEGO

```
def menu(): #funkcja w ktorej zawarte jest menu
    dostępne_wybory = ('0','1','2','3','4') #pula wyborow dostepnych dla uzytkownika
    glowne_menu=''
    Wybierz figure
    1.Kwadrat
    2.Prostokat
    3.Trojkat
    4.Kolo:
    0.Wyjście
    ''' # w ten sposob mozna zapisac pare linijek tekstu
    print(glowne_menu)
    print(" Podaj odpowiadajaca cyfre")
    while True: #petla sprawdzajaca poprawnosc podanego wyboru
        wybor_uzytkownika = input(' >') #pobranie wyboru od uzytkownika
        if wybor_uzytkownika in(dostępne_wybory): #sprawdzenie czy wybor jest dostepny
            return wybor_uzytkownika #zwrocenie wybory jesli znajduje sie na liscie
        else:
            os.system('cls') #wyczyszczenie konsoli
            print(glowne_menu)
            print(' Podaj poprawna cyfre')
```



# FUNKCJA KOŃCZĄCA DZIAŁANIE PROGRAMU

```
def koniec_programu(): #funkcja sprawdzajaca czy uzytkownik chce zakonczyc program
    print("\n Czy chcesz zakonczyc program? T/N")
    while True:
        wybor_uzytkownika = input(' >').upper() # .upper() zamienia male znaki na wielkie
        if wybor_uzytkownika=='T':
            return True
        elif wybor_uzytkownika=='N':
            return False
        else:
            os.system('cls')
            print("\n Czy chcesz zakonczyc program? T/N")
            print("\n Podaj poprawny znak")
```

# FUNKCJA LICZĄCA POLE KWADRATU

```
def kwadrat():  
    print("\n Wybrales kwadrat\n\n Podaj dlugosc boku")  #\n przechodzi do nowej linii  
    dl_bok=Sprawdzenie()  
    pole = dl_bok**2  
    os.system('cls')  
    print(f"\n Pole kwadratu {dl_bok} ^ 2 = {pole}")
```

# FUNKCJA LICZĄCA POLE PROSTOKĄTU

```
def prostokat():  
    print("\n Wybrales prostokat\n\n Podaj dlugosc boku a")  
    dl_bok_a=Sprawdzenie()  
    print(" Podaj dlugosc boku b")  
    dl_bok_b=Sprawdzenie()  
    pole = dl_bok_a*dl_bok_b  
    os.system('cls')  
    print(f"\n Pole prostokatu {dl_bok_a} * {dl_bok_b} = {pole}")
```

# FUNKCJA LICZĄCA POLE TRÓJKĄTA

```
def trojkat():  
    print("\n Wybrales trojkat\n\n Podaj dlugosc podstawy")  
    dl_podstawy = Sprawdzenie()  
    print(" Podaj wysokosc")  
    wysokosc = Sprawdzenie()  
    pole = (dl_podstawy * wysokosc) / 2.0  
    os.system('cls')  
    print(f"\n Pole trojkata ({dl_podstawy} * {wysokosc}) / 2 = {pole}")
```



# FUNKCJA LICZĄCA POLE KOŁA

```
def kolo():  
    print("\n Wybrales kolo\n\n Podaj dlugosc promienia")  
    promien = Sprawdzenie()  
    pole = promien**2*math.pi  
    os.system('cls')  
    print(f"\n Pole kola {promien} ^ 2 * {round(math.pi,4)} = {round(pole,4)}")  
    #zaokraglenie(zmienna,ilosc miejsc po przecinku)
```

# GŁÓWNY PROGRAM

## GEOMETRIA W KROKACH

```
while True: #Zapętla Działanie programu
    os.system('cls')
    wybor_uzytkownika=menu() #wywołanie funkcji która zwróci wybor uzytkownika
    os.system('cls')

    if wybor_uzytkownika == '0': #wybor_uzytkownika jest stringiem, dlatego wybory sa zapisane jako znak, a nie wartosc
        if koniec_programu()==True:
            break #jesli uzytkownik chce zakonczyc program to lamiey petle while

    elif wybor_uzytkownika == '1': #Inaczej else if
        kwadrat()
        if koniec_programu()==True:
            break

    elif wybor_uzytkownika == '2':
        prostokat()
        if koniec_programu()==True:
            break

    elif wybor_uzytkownika == '3':
        trojkat()
        if koniec_programu()==True:
            break

    elif wybor_uzytkownika == '4':
        kolo()
        if koniec_programu()==True:
            break
```