

Instrukcja A2, Python II - Funkcje

1. Proszę napisać program który prosi użytkownika o podanie długości 2 boków trójkąta prostokątnego. Następnie korzystając z funkcji *przeciwprostokatna()* program wyświetla na ekranie komunikat jaka jest długość przeciwprostokątnej tego trójkąta.

```
def przeciwprostokatna(a, b):  
    wynik = (a*a + b*b)**0.5  
    return wynik
```

Proszę zmodyfikować powyższy kod tak, aby dodatkowo liczony był obwód trójkąta oraz jego powierzchnia, do tego celu należy stworzyć nową funkcję obwód oraz nową funkcję powierzchnia.

2. Jak widać na powyższym przykładzie, funkcje mogą przyjmować więcej niż jeden argument. Możliwe jest zdefiniowanie funkcji z argumentami domyślnymi, których wartość zostanie użyta w momencie, kiedy użytkownik nie poda innych wartości argumentów podczas wywołania funkcji. Proszę zapoznać się z poniższą definicją i użyciem funkcji z argumentami domyślnymi i na jej podstawie napisać funkcję o nazwie *welcome*, która przyjmuje jako argumenty imię i nazwisko osoby oraz opcjonalny argument język, który ma domyślną wartość 'pl'. Funkcja ta ma zwrócić ciąg znaków zawierający powitanie osoby w wybranym języku. Jeśli język jest równy 'pl', powitanie ma być w języku polskim, jeśli 'en', to w języku angielskim, a jeśli 'fr', to w języku francuskim. Jeśli język ma inną wartość, funkcja ma zwrócić pusty ciąg znaków.

```
def default_function(a = 1, b = 3, c = 4):  
    return a + b + c  
  
# przykład użycia:  
print(default_function()) # → 8  
print(default_function(10)) # → 17  
print(default_function(1,1,1)) # → 3
```

3. Funkcja może zwracać także więcej niż jedną wartość, która może być bezpośrednio przypisana do zmiennych albo zapisana do listy i używana z wykorzystaniem odniesienia do konkretnego elementu listy:

**AGH**

AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY

```
def suma_roznica_iloczyn(a, b):  
    suma = a + b  
    roznica = a - b  
    iloczyn = a*b  
    return suma, roznica, iloczyn  
  
x,y,z = suma_roznica_iloczyn(2,3)  
liczby = suma_roznica_iloczyn(2,3)  
print(x) # → 5  
print(y) # → -1  
print(z) # → 6  
print(liczby[0], liczby[1], liczby[2]) # → 5 16
```

Napisz funkcję, która obliczy wszystkie kąty trójkąta zdefiniowanego przez podane 3 boki (niezależnie czy trójkąt jest prostokątny czy nie). Do tego celu proszę wykorzystać funkcje trygonometryczne dostępne w bibliotece numpy.

4. Po zdefiniowaniu kilku funkcji, możliwe jest wywołanie jednej funkcji wewnątrz innej:

```
def dodaj(a, b):  
    return a + b  
  
def pomnoz(a, b):  
    return a * b  
  
def roznica(a, b):  
    suma = dodaj(a, b) # Wywołujemy funkcję dodaj  
    iloczyn = pomnoz(a, b) # Wywołujemy funkcję pomnoz  
    return suma - iloczyn  
  
print(roznica(5, 3)) # Wyświetli -4
```

Proszę zdefiniować osobne funkcje pozwalające obliczyć:
-objętość prostopadłościanu,

- pole powierzchni prostopadłościanu,
- masę prostopadłościanu

W tym celu proszę pobrać od użytkownika 3 parametry geometryczne (długości boków w metrach) oraz gęstość w kg/m^3 . Funkcja licząca gęstość powinna korzystać z funkcji obliczającej objętość prostopadłościanu. Wynik działania powinien wyświetlić się na ekranie również w jednostkach układu SI.

6. Proszę zapętlić program z poprzedniego zadania za pomocą odpowiedniej pętli w taki sposób aby po wyliczeniu odpowiedniej wielkości użytkownik był pytany czy chce kontynuować i w przypadku odpowiedzi pozytywnej program ponownie zapyta o dane i wyświetli wynik. W przypadku odpowiedzi negatywnej zakończy działanie.

Zadanie domowe:

Napisać zestaw funkcji pozwalające na obliczenie: objętości, masy, pola powierzchni:

- a) kuli (10%)*
- b) czworościanu foremnego (10%)*
- c) ostrosłupa prostego o podstawie prostokątnej (10%)*
- d) stożka (10%)*
- e) walca (10%)*
- f) elipsoidy (20%)*, **

* jeżeli funkcje obliczające objętości, masy, pola powierzchni działają poprawnie

Program powinien wyświetlać wszystkie wymienione nazwy brył które zostały zaimplementowane, następnie poprosić użytkownika o dokonanie wyboru jednej z nich. Po dokonaniu wyboru program powinien wyświetlić listę właściwości, które wylicza (objętość, masa, pole powierzchni). Po dokonaniu wyboru użytkownik podaje potrzebne parametry (o które zostanie poproszony) a następnie wyświetla wynik wybranej operacji (objętości, masy, pola powierzchni) dla wybranej bryły. (10%).

W całym programie w wyświetlanych komunikatach należy stosować jednostki układu SI (5%)

Program powinien być zapętlony, tzn. po wyliczeniu zadanej wartości dla zadanej bryły powinien pytać użytkownika czy chce kontynuować i wykonać inne obliczenia, podobnie jak w zadaniu 6. (5%)

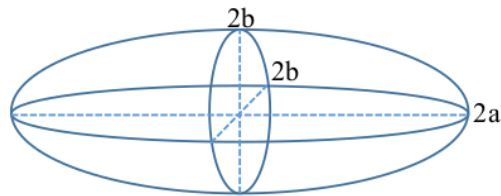
Program powinien posiadać zabezpieczenie przed podaniem wartości niedozwolonych (np. liczb ujemnych lub łańcuchów znaków i innych koniecznych warunków jeśli występują***) i informować o tym fakcie użytkownika w poprzez wyświetlenie odpowiedniego komunikatu, jednocześnie kontynuując działanie i prosząc o podanie poprawnej wartości. (10%)

**Wzór uproszczony na pole powierzchni elipsoidy obrotowej:



AGH

AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY



$$P = 2\pi b \left(b + \frac{a}{\epsilon} \arcsin(\epsilon) \right)$$

$$\epsilon = \sqrt{1 - \frac{b^2}{a^2}}$$

$$a > b$$

a, b – półosie elipsoidy

***Przykład mechanizmu zabezpieczającego przed podaniem niedozwolonego typu danych:

```
c=False
```

```
while c == False:
```

```
try:
```

```
x = float(input())
```

```
if x < 0:
```

```
print("Podana wartość nie może być ujemna")
```

```
if x >= 0:
```

```
c = True
```

```
except:
```

```
print("Podana wartość nie może być ciągiem znaków")
```