

Marzena Nowakowska, Wydział Zarządzania i Modelowania Komputerowego, PŚk

**Temat 2. Typy wbudowane i komunikacja z użytkownikiem**

Zad. 1.

Napisać program *Zad02\_01*, który pobiera dane od użytkownika i wyświetla wizytówkę tego użytkownika.

```
osoba=input("Podaj imię: ")
osoba+=" "+input("Podaj nazwisko: ") # 1
adres=input("Podaj kod pocztowy: ")
adres+=" "+input("Podaj miasto: ")
adres+=", "+input("Podaj resztę adresu: ")
telefon=input("Podaj nr telefonu: ")
maxNapis=adres if len(adres)>=len(telefon) else telefon # 2
if len(maxNapis)<= len(osoba): # 3
    maxNapis=osoba
lZnkw=len(maxNapis)+ 4
print("\n")
print(""*lZnkw) # 4
print("* "+osoba+" "*(lZnkw-3-len(osoba))+"*")
print("* "+adres+" "*(lZnkw-3-len(adres))+"*")
print("* "+telefon+" "*(lZnkw-3-len(telefon))+"*")
print(""*lZnkw)
```

Sprawdzić (notatki, Internet), jak działają wykorzystane narzędzia programistyczne języka Python. Objąć te narzędzia dla instrukcji oznaczonych numerami: 1 (operator przypisania złożonego), 2 (funkcja *len* oraz operator warunkowy), 3 (instrukcja warunkowa), 4 (multiplikacja obiektu).

Zad. 2.

Napisać program *Zad02\_02*, który pobiera od użytkownika jego oceny tytułów (książek, filmów, gier) w skali 1-10, gdzie 1 to ocena najniższa a 10 – najwyższa, dla 3 tytułów. Następnie program ilustruje te oceny w sposób przedstawiony poniżej. Jeżeli potrzeba (szerokość wyjścia), liczbę punktów należy przeskalać. Wyświetlić nie więcej niż *n* znaków tytułu (*n* – ustalone przez projektanta); w tym celu zastosować właściwą metodę klasy *str* (sprawdzić w notatkach lub Internecie).

```
Tytuł pierwszy |*****
Tytuł drugi   |*****
Tytuł ostatni |**
```

Zad 3.

Napisać program *Zad02\_03*, który na podstawie parametrów równania kwadratowego wyświetla informację, ile równanie ma pierwiastków rzeczywistych.

Wariant 1	Wariant 2
<pre>print("Podaj parametry równania kwadratowego") a=float(input("a=")) b=float(input("b=")) c=float(input("c=")) if a ==b == c:     print("Wszystkie parametry są zerami ") else:     delta= b**2-4*a*c     print("Liczba pierwiastków rzeczywistych ", \         "równania kwadratowego: ", \         2 if delta &gt; 0 else (1 if delta ==0 else 0))</pre>	<pre>print("Podaj parametry równania kwadratowego") a=float(input("a=")) b=float(input("b=")) c=float(input("c=")) if a ==b == c:     print("Wszystkie parametry są zerami ") else:     delta= b**2-4*a*c     print("Liczba pierwiastków rzeczywistych ", \         "równania kwadratowego: ", end="")     if delta &gt; 0:         print(2)     elif delta == 0:         print(1)     else:         print(0)</pre>

Sprawdzić wykaz funkcji dostępnych w module *math*. W tym celu w oknie konsoli Spydera wydać polecenia (importować moduł, jeżeli wcześniej tego nie wykonano): *import math; help(math)*. Odpowiedzieć na pytania poniżej.

1. Ile jest funkcji zdefiniowanych w module *math*? Ile jest stałych matematycznych?
2. Ile jest cyfr części ułamkowej jest zaprezentowanych dla stałych matematycznych?
3. Która funkcja zamienia radiany na stopnie, a która stopnie na radiany? Zilustrować odpowiedzi na przykładach.

Uzupełnić program *Zad02\_03* wyliczając pierwiastki rzeczywiste (jeżeli istnieją) i wyświetlić wyniki.

Zad. 4.

Napisać program *Zad02\_04*, który pobiera od użytkownika współrzędne punktu na płaszczyźnie i wyświetla na ekranie, w której ćwiartce układu współrzędnych leży ten punkt.

Zad. 5.

Napisać program *Zad02\_05*, w którym zostaną utworzone listy zawierające dane płaskich figur geometrycznych: boki trójkąta, boki prostokąta, boki trapezu. Program oblicza dla każdej figury pole powierzchni i obwód i tworzy odpowiednie listy zawierające obliczone wartości. Po wykonaniu obliczeń, wyznaczone wartości mają być wyświetlone na ekranie. Wykorzystać metody i operacje na listach. Jeżeli sanieznane, wzory na pola figur płaskich znaleźć w Internecie.

```

from math import sqrt
trjk = [5, 6, 8]
oTrjk = sum(trjk); pom = oTrjk/2
pTrjk = sqrt(pom*(pom- trjk[0])*( pom- trjk[1])*( pom- trjk[2]))
Trjk = []
Trjk.append("Obwód trójkąta: " +str(oTrjk))
Trjk.append("Pole trójkąta: " + str(pTrjk))
print(Trjk)

prstk = [6,10]
oPrstk = sum(prstk)*2
pPrstk = prstk[0]* prstk[1]
Prstk=[]
Prstk.append("Obwód prostokata: " + str(oPrstk))
Prstk.append("Pole prostokata: " + str(pPrstk))
print(Prstk)

trpz = [13, 10, 7, 8] # (2,4,3,6) oznaczają odpowienio: podstawa, podstawa, bok, bok
oTrpz = sum(trpz);
# Wyznaczyć do zmiennej 'pTrpz' pole trapezu na podstawie długości boków trapezu (por. Interent).
# Dokończyć obliczenia dla trapezu i wyświetlić wyniki.
# Wprowadzić zaokrąglenie wyliczonych wartości do dwóch miejsc części ułamkowej – funkcja round.

```

Zmodyfikować program, tak aby dane figur były pobierane od użytkownika. Wykorzystać podaną niżej podpowiedź przykładowego tworzenia listy dla danych do pola trójkąta.

```

# Pobieranie danych od użytkownika
trjk = []
trjk.append(float(input("Podaj bok trójkąta a= ")))
trjk.append(float(input("Podaj bok trójkąta b= ")))
trjk.append(float(input("Podaj bok trójkąta c= ")))

```

Zad. 6.

Napisać program *Zad02\_06*, w którym na podstawie danych pobranych od użytkownika zostaną utworzone trzy listy zawierające dane trójwymiarowych figur geometrycznych odpowiednio: prostopadłościanu, kuli i stożka. Program oblicza dla każdej figury pole powierzchni i objętość i tworzy odpowiednie listy zawierające obliczone wartości. Dla każdej figury powstaje osobna dwuelementowa lista zawierająca odpowiednio: pole powierzchni i objętość. Po wykonaniu obliczeń, wyznaczone wartości mają być wyświetlone na ekranie.