

Wstęp do programowania w języku Python

Kolokwium

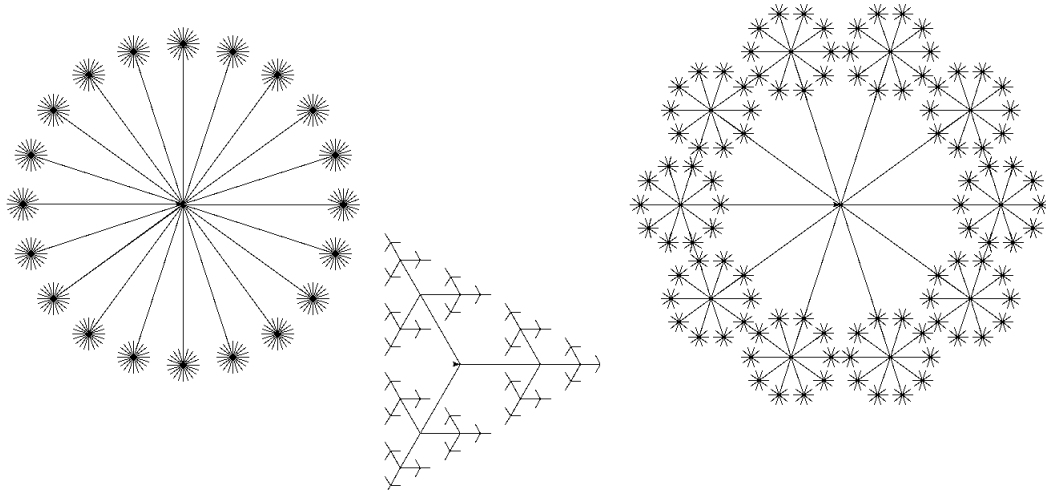
23 stycznia 2013

Za kolokwium można dostać 105 punktów (100 zapowiadanych oraz 5 bonusowych). W rozwiązaniach istotna jest przede wszystkim czytelność kodu.

Zadanie 1.(15pkt) Napisz w Pythonie funkcję `klepsydra(n)`, która za pomocą instrukcji `print` wypisuje klepsydrę zawierającą `n` gwiazdek w pierwszym rzędzie. Poniżej wynik działania wywołania funkcji `klepsydra(7)`

```
*****
 *****
  ****
   ***
    **
   ***
  *****
 *****
```

Zadanie 2.(15pkt) Napisz w Pythonie rekurencyjną funkcję `gwiazda`, za pomocą której można wykonywać rysunki takie, jak poniższe. Wybierz takie parametry funkcji, żeby mieć jak największą kontrolę nad wyglądem rysunku. Podaj przybliżone wartości parametrów, które dały takie rysunki.



Zadanie 3.(20pkt) Napisz funkcję `dzielniki`, która dla liczby naturalnej większej niż 1, zwraca listę pierwszych dzielników tej liczby.

Zadanie 4.(10pkt) Co wypisze poniższy program:

```
def f1(n):
    digits = sorted(set(str(n)))
    return "".join(digits) + "*" + "".join(digits)

def f2(s, wordSet):
    res = set()
    for i in range(len(s)):
        for j in [2,3,4,5,6,7,8]:
            if s[i:i+j] in wordSet:
                res.add(s[i:i+j])
    return res
```

```
def f3(s, wordSet):
    return [w for w in wordSet if w in s]

print f1(321)
print f1(55555555222)

s = "matematykapustkinieznosi"

ws = ["ma","temat","tematyka", "kapusta", "kapustki","matematyka","znosi","kinie"]

print f2(s,ws)
print f3(s,ws)
```

Zadanie 5.(20pkt) *Tabliczka działań* nazwiemy zbiór napisów postaci:

`<liczba> <operator> <działanie> = <liczba>`

przedstawiających prawdziwe fakty arytmetyczne. Napisz wyrażenie w Pythonie, którego wartość będzie tabliczka działań, w której argumenty są liczbami od 1 do 100, a działaniami są `+-/`. Dodatkowo wymagamy, żeby tabliczka działań zawierała wszystkie możliwe działania dla argumentów z podanego zakresu. Taka pełna tabliczka działań dla zakresu od 1 do 2 jest

```
set(['1+1 = 2', '1-1 = 0', '1*1 = 1', '1/1 = 1', '1+2 = 3', '1-2 = -1',
    '1*2 = 2', '1/2 = 0', '2+1 = 3', '2-1 = 1',
    '2*1 = 2', '2/1 = 2', '2+2 = 4', '2-2 = 0', '2*2 = 4', '2/2 = 1'])
```

Napisz fragment kodu, który dla danej tabliczki działań wypisuje te liczby, która jest najczęstszym wynikiem działania (nazwijmy ją x), wraz z informacją ile, i jakie działania, dają w wyniku x .

Zadanie 6.(20pkt) W języku LISP wyrażenia zapisuje się w notacji prefiksowej, w której najpierw pojawia się znak działania, a po nim ciąg argumentów. Przykładowo `1+2+3*4` zapisalibyśmy jako `(+ 1 2 (* 3 4))`. Powinieneś napisać interpreter wyrażen lispowych, wykonując dwa podpunkty:

- Napisz funkcję, przekształcającą wyrażenie lispowe do odpowiedniego wyrażenia Pythonowego (w powyższym przykładzie do `["+ ", 1, 2, ["*", 3, 4]]`)¹
- Napisz rekurencyjną funkcję, która wylicza wartość wyrażenia Pythonowego.

¹Możesz założyć, że spacje stawiane są tak jak w powyższym przykładzie.