

PYTHON – TURTLE 05 (16)

IF - REKURENCJA - FRAKTALE

Gdyby na końcach małych gwiazdek umieścić kolejne, jeszcze mniejsze... i na końcach tych ostatnich... Możemy tworzyć kolejne pętle, ale wygodniej powielić takie samo zadanie za pomocą rekurencji. Rekurencja to sposób tworzenia algorytmów, w którym funkcje wywołują same siebie. Bardzo ważne w rekurencji jest umiejętność zakończenia działania funkcji na odpowiednim etapie. Służy do tego instrukcja warunkowa IF, której zadaniem jest sprawdzenie tzw. warunku logicznego. Gdy warunek jest prawdziwy, to wykonujemy instrukcje.

Wianek rekurencyjny (2)

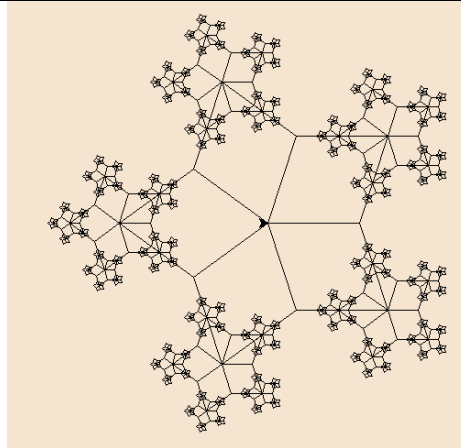
- Wklej funkcje: **romb**, **gwiazda**, **wianekR**

```
from turtle import *
speed(0)

def romb(bok,kat):
    for i in range(2):
        fd(bok);rt(kat);fd(bok);rt(180-kat)

def gwiazda(bok,ile):
    kat=360/ile
    for i in range(ile):
        romb(bok,kat)
        rt(kat)

def wianekR(bok,ile,rek):
    kat=360/ile
    gwiazda(bok,ile)
    if rek>0:
        for i in range(ile):
            fd(bok);rt(kat);fd(bok)
            wianekR(bok/3,ile,rek-1)
            bk(bok);lt(kat);bk(bok)
            rt(kat)
```



Funkcja **wianekR** jest prawie identyczna jak **wianek** z poprzedniej lekcji

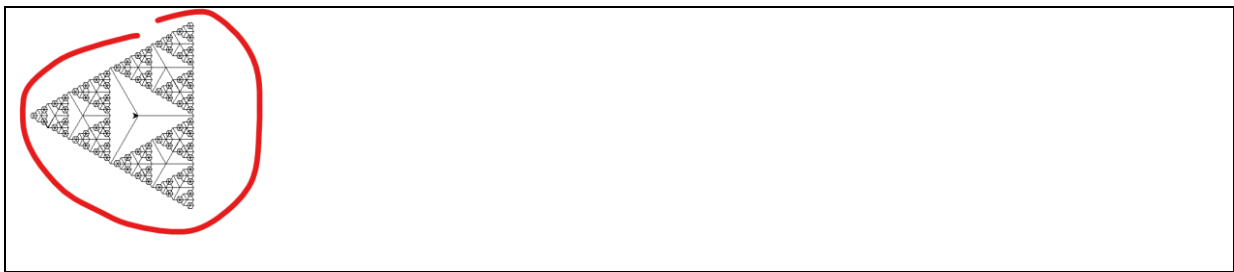
- w parametrach wywołania funkcji pojawia się nowa zmienna **rek**, która mówi o poziomie rekurencji
- przed pętlą rysującą kolejną warstwę gwiazdek umieszczamy instrukcję warunkową, która sprawdza poziom rekurencji **rek** – funkcja kończy działanie i przestaje rysować wianek, gdy poziom staje się równy zero
- w pętli rysowaliśmy mniejsze gwiazdki, a teraz wywołujemy funkcję rekurencyjnie **wianekR** – samą siebie, z nowymi parametrami, mniejszy bok i poziom rekurencji mniejszy o jeden

- Uruchom funkcję poleceniem: **wianekR(100,5,3)**
UWAGA - w czasie tworzenia rysunku możesz pracować nad kolejnym ćwiczeniem
- Wklej do ramki** zrzut ekranu z programem i rysunkiem

Większy wianek rekurencyjny (2)

W poprzednim przykładzie małe wianki są 3 razy mniejsze od poprzedniego **wianekR(bok/3,ile,rek-1)**

- Popraw funkcję **wianekR**, aby każdy nowy wianek był tylko **2 razy mniejszy**
- Uruchom funkcję poleceniem: **wianekR(100,3,4)**
UWAGA - w czasie tworzenia rysunku możesz pracować nad kolejnym ćwiczeniem
- Wklej do ramki** zrzut ekranu z rysunkiem



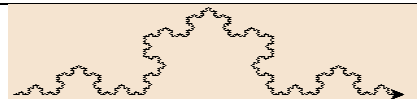
Fraktale

Fraktale wykorzystywane są w grafice i animacji komputerowej do tworzenia gier i filmów, w ekonomii do analizy zachowań na giełdzie, w medycynie do analizy obrazów tomograficznych... i wielu innych dziedzinach działalności człowieka.

Funkcja Koch (2)

- Przepisz do edytora funkcję **koch** z ramki

```
def koch(bok, poz):
    if poz==0: fd(bok)
    else:
        koch(bok/3, poz-1); lt(60)
        koch(bok/3, poz-1); rt(120)
        koch(bok/3, poz-1); lt(60)
        koch(bok/3, poz-1)
```

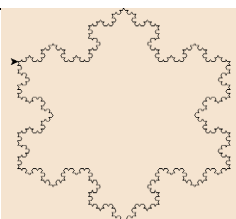


- Uruchom funkcję poleceniem: **koch(300,0)**
długość 300, poziom 0
- Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem: **koch(300,1)**
długość 300, poziom 1
- Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem: **koch(300,2)**
długość 300, poziom 2
- Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem: **koch(300,3)**
długość 300, poziom 3
- Wklej do ramki** zrzut ekranu z rysunkiem

Płatek Kocha (2)

- Przepisz do edytora funkcję **PLATEK** z ramki
Funkcja przedstawia „Płatek Koch’a”

```
def platek(bok, poz):
    for i in range(3):
        koch(bok, poz)
        rt(120)
```



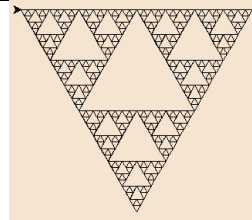
- Uruchom funkcję poleceniem: **platek(300,0)**
długość 300, poziom 0
- Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem: **platek(300,1)**
długość 300, poziom 1

- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem: **platek(300,2)**
długość 300, poziom 2
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem: **platek(300,3)**
długość 300, poziom 3
- **Wklej do ramki** zrzut ekranu z rysunkiem

Trójkąt Sierpińskiego (2)

- **Wklej do edytora** funkcję **sierp** z ramki. Funkcja przedstawia „Trójkąt Sierpigo”

```
def sierp(bok,poz):
    if poz==0:
        for i in range(3):fd(bok);rt(120)
    else:
        sierp(bok/2,poz-1);fd(bok/2)
        sierp(bok/2,poz-1);lt(60);bk(bok/2);rt(60)
        sierp(bok/2,poz-1);rt(60);bk(bok/2);lt(60)
```



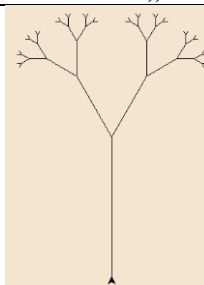
- Uruchom funkcję poleceniem:**sierp(300,0)**
długość 300, poziom 0
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**sierp(300,1)**
długość 300, poziom 1
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**sierp(300,2)**
długość 300, poziom 2
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**sierp(300,3)**
długość 300, poziom 3
- **Wklej do ramki** zrzut ekranu z rysunkiem

Drzewo binarne (2)

- **Wklej do edytora** funkcję **drzewoB** z ramki. Funkcja przedstawia „Drzewo binarne”

```
def drzewoB(b,n):
    if n==0:fd(b);bk(b)
    else:
        fd(b);lt(30)
        drzewoB(b/2,n-1)
        rt(60)
        drzewoB(b/2,n-1)
        lt(30);bk(b)

lt(90);bk(200)
```



- Uruchom funkcję poleceniem:**drzewoB(300,0)**
długość 300, poziom 0
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**drzewoB(300,1)**

długość 300, poziom 1

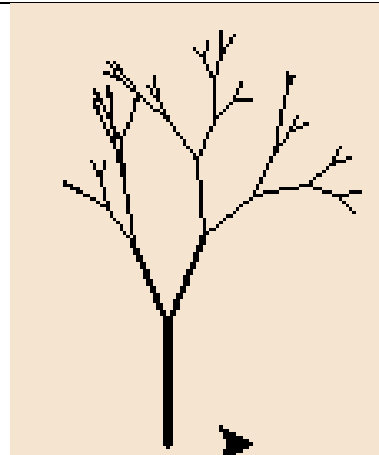
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**drzewoB(300,2)**
długość 300, poziom 2
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**drzewoB(300,3)**
długość 300, poziom 3
- **Wklej do ramki** zrzut ekranu z rysunkiem

Drzewo losowe (2)

- **Wklej do edytora** funkcję **drzewko** z ramki
importujemy dodatkowo funkcję losującą

```
from random import *
def drzewko(bok):
    pensize(bok/20)
    kat=randint(-30,30)
    skala=randint(6,8)/10
    fd(bok*skala)
    if (bok > 10):
        lt(kat)
        drzewko(bok*skala)
        rt(2*kat)
        drzewko(bok*skala)
        lt(kat)
        bk(bok*skala)

lt(90);bk(200)
```



- Uruchom funkcję poleceniem:**drzewko(50)**
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**drzewko(100)**
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**drzewko(150)**
- **Wklej do ramki** zrzut ekranu z rysunkiem
- Uruchom funkcję poleceniem:**drzewko(200)**
- **Wklej do ramki** zrzut ekranu z rysunkiem

Las (2)

- Narysuj las drzew posługując się algorytmem:



obrót w lewo o 90

cofnij się o 200

powtórz 10 razy

drzewko(randint(50,100))

podnieś pisak

obrót w prawo o 90

do przodu o randint(20,50)

obrót w lewo o 90

opuść pisak

każde drzewo o losowej długości i przesunięte o losową odległość

- **Wklej rysunek do ramki**
- **Wklej do ramki** zrzut ekranu z instrukcjami
