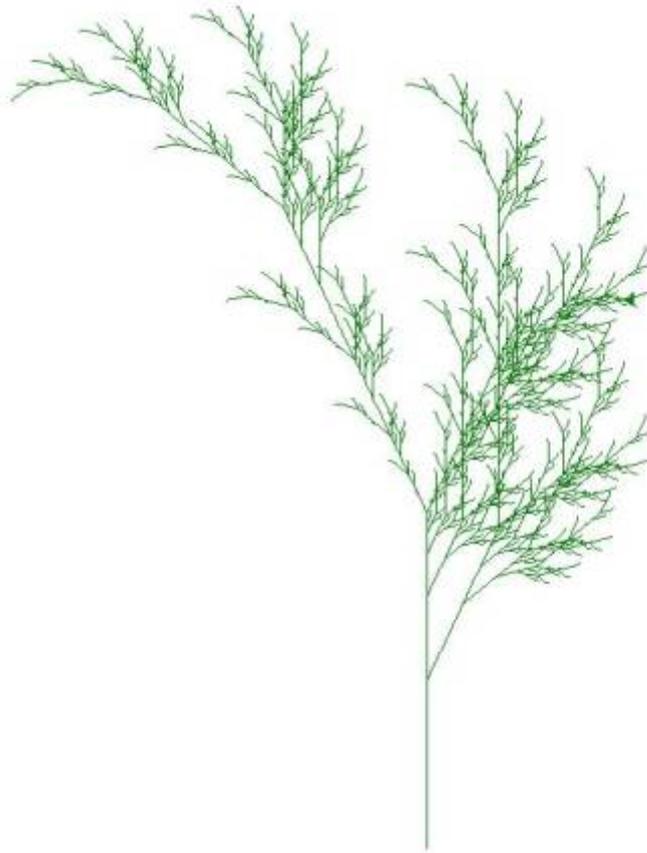


# Fraktale w pythonie

[Wikipedia: L-systems](#)



```
from turtle import *

tracer(0)
start="X"
dlugosc=4
kat=25

stos=[]
slZam={'X':'F+[[X]-X]-F[-FX]+X', 'F':'FF'}

iteracje=6
zolw='zolv'

def LSBuduj(st,ile,sl):
    nowy=""
    for litera in st:
        if litera in st:
            if litera in sl.keys():
                nowy+=sl[litera]
            else:
```

```
        nowy+=litera

    if ile>1:
        ile-=1
        return LSBuduj(nowy,ile,sl)
    else:
        return nowy
DoWykonania=LSBuduj(start,iteracje,slZam)
Polecenia={}
Polecenia['F']=[zolv+'.pd()',zolv+'.fd('+str(dlugosc)+'')']
Polecenia['+']=[zolv+'.right('+str(kat)+'')']
Polecenia['-']=[zolv+'.left('+str(kat)+'')']
Polecenia['[']=[ 'stos.append(('+zolv+'.xcor(),' +zolv+'.ycor(),' +zolv+'.heading()))']
Polecenia[']']=[zolv+'.pu()',zolv+'.setx(stos[len(stos)-1][0])',
                zolv+'.sety(stos[len(stos)-1][1])',
                zolv+'.setheading(stos[len(stos)-1][2])',
                'stos.pop()']

print(Polecenia)

zolv=Turtle()
zolv.pu()
zolv.goto(0,-300)
zolv.color('green')
zolv.pd()
zolv.setheading(90)
zolv.speed(0)
for litera in DoWykonania:
    if litera in Polecenia.keys():
        for rozkaz in Polecenia[litera]:
            eval(rozkaz)
update()
```