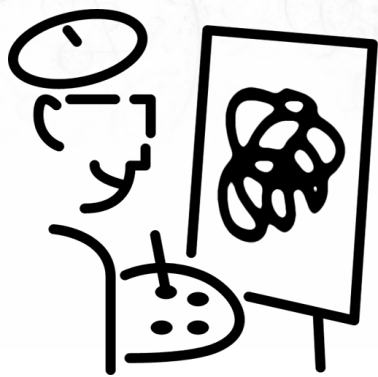


**ARTGORITMY / F**

# **BEZKONTEXTOVÁ GRAFIKA**



## Bezkontextová gramatika

**$G = (N, T, P, S)$**

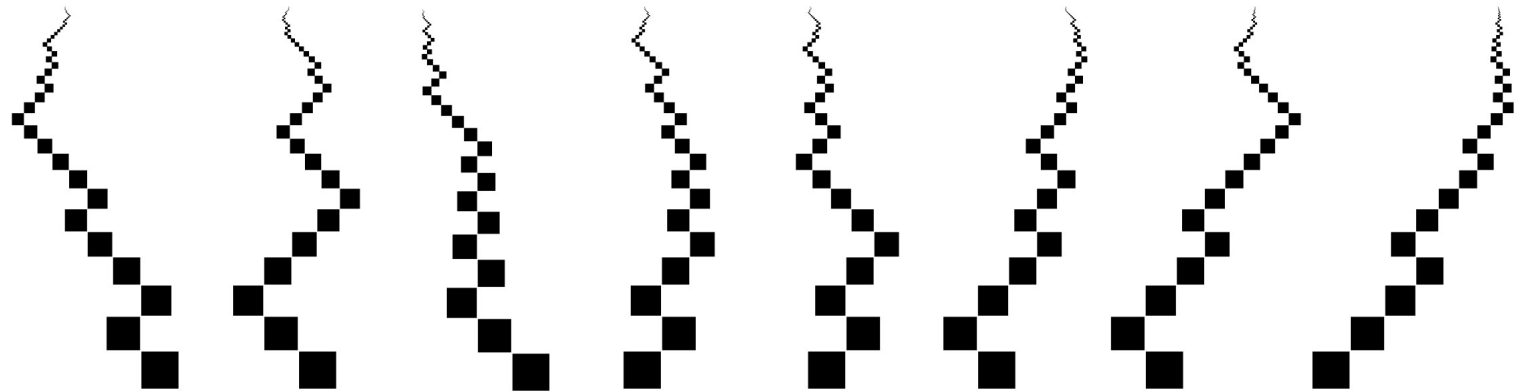
- $N$  konečná množina neterminálů (= konceptů)  
 $T$  konečná množina terminálů (= grafických prvků)  
 $P$  konečná množina pravidel (= tvůrčích postupů)  
 $A \rightarrow \alpha, A \in N, \alpha \in (N \cup T)^*$   
 $S$  počáteční proměnná, axiom (= prázdné plátno)  
 $S \in N$

Posloupností **derivací** ( $\rightarrow$ ) lze do **prázdného plátna** pomocí **tvůrčích postupů**, založených na různých **konceptech**, vykreslovat **grafické prvky** dokud není obraz **akceptován**

## Vizuální gramatika CFDG, *Context Free Design Grammar*

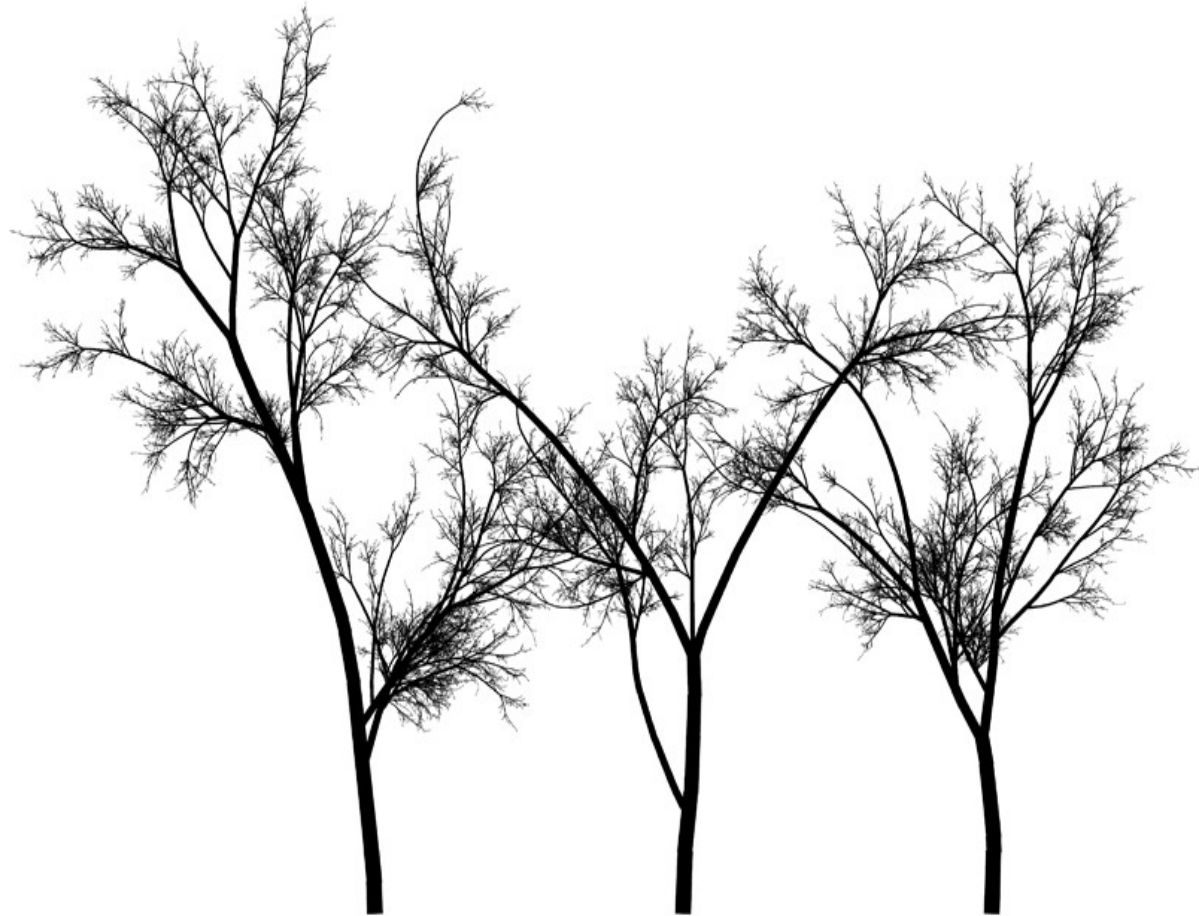
- grafické symboly
- prepisovací pravidla
- náhodný výběr pravidel

```
startshape paint
rule paint {
  SQUARE{}
  paint { s 0.9 x 1 y 1 }
}
rule paint {
  SQUARE{}
  paint { s 0.9 x -1 y 1 }
}
```

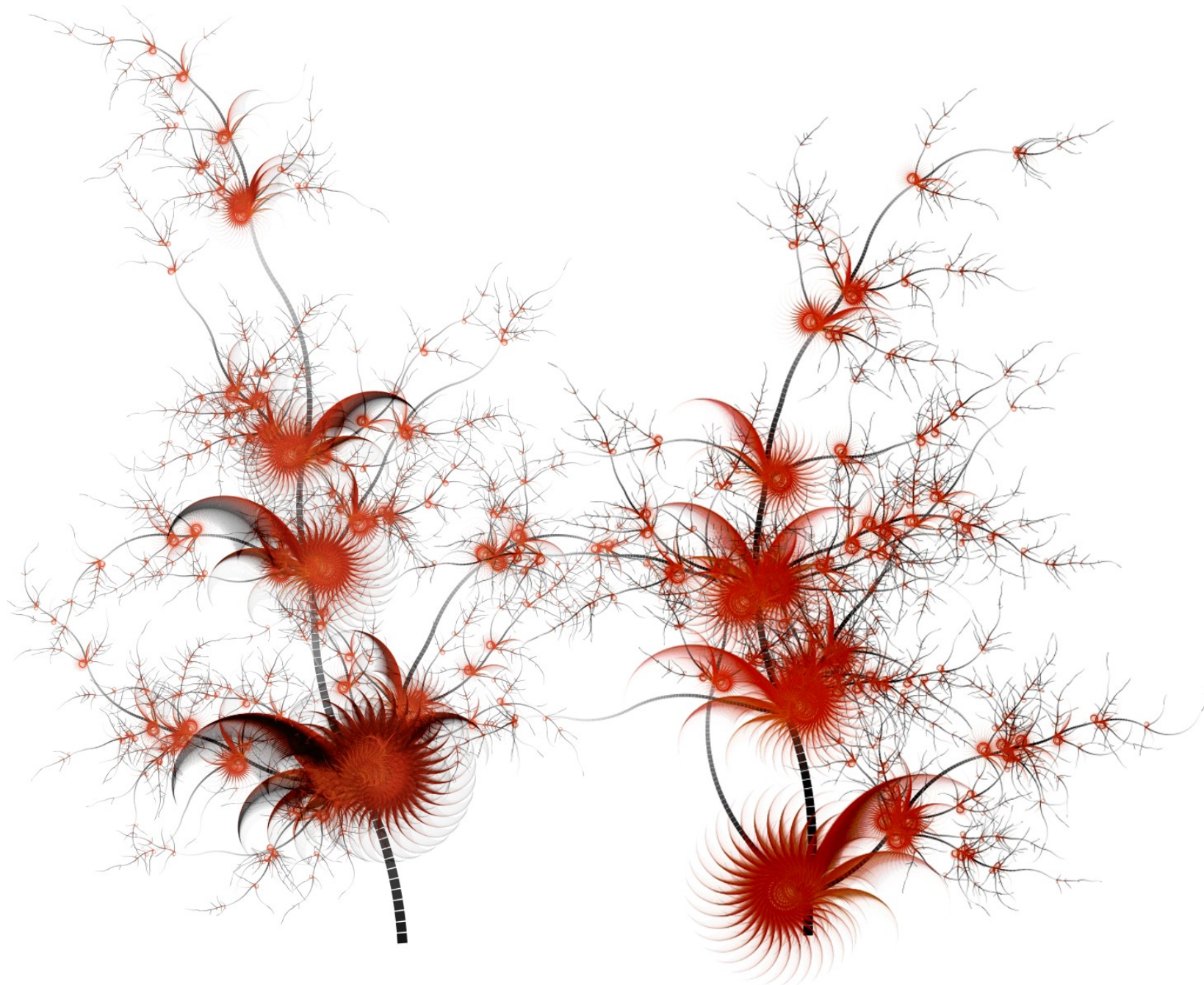


```
startshape FOREST
```

```
rule FOREST {  
  SEED {}  
  SEED {x -20}  
  SEED {x -40}  
}  
  
rule SEED {BRANCH {}}  
rule SEED {BRANCH {rotate 1}}  
rule SEED {BRANCH {rotate -1}}  
rule SEED {BRANCH {rotate 2}}  
rule SEED {BRANCH {rotate -2}}  
rule SEED {FORK {}}  
  
rule BRANCH {RBRANCH {}}  
rule BRANCH {LBRANCH {}}  
  
rule LBRANCH {BLOCK {} LBRANCH {y 0.885 rotate 0.1 size 0.99}}  
rule LBRANCH {BLOCK {} LBRANCH {y 0.885 rotate 0.2 size 0.99}}  
rule LBRANCH {BLOCK {} LBRANCH {y 0.885 rotate 4 size 0.99}}  
rule LBRANCH {BLOCK {} FORK {}}  
  
rule RBRANCH {BLOCK {} RBRANCH {y 0.885 rotate -0.1 size 0.99}}  
rule RBRANCH {BLOCK {} RBRANCH {y 0.885 rotate -0.2 size 0.99}}  
rule RBRANCH {BLOCK {} RBRANCH {y 0.885 rotate -4 size 0.99}}  
rule RBRANCH {BLOCK {} FORK {}}  
  
rule BLOCK {  
  SQUARE {rotate 1}  
  SQUARE {rotate -1}  
  SQUARE {}  
}  
  
rule FORK {  
  BRANCH { }  
  BRANCH {size 0.5 rotate 40}  
}  
rule FORK {  
  BRANCH { }  
  BRANCH {size 0.5 rotate -40}  
}  
rule FORK {  
  BRANCH {size 0.5 rotate -20}  
  BRANCH { }  
}  
rule FORK {  
  BRANCH {size 0.7 y 0.1 rotate 20}  
  BRANCH {size 0.7 y 0.1 rotate -20}  
}
```



➔ ContextFree

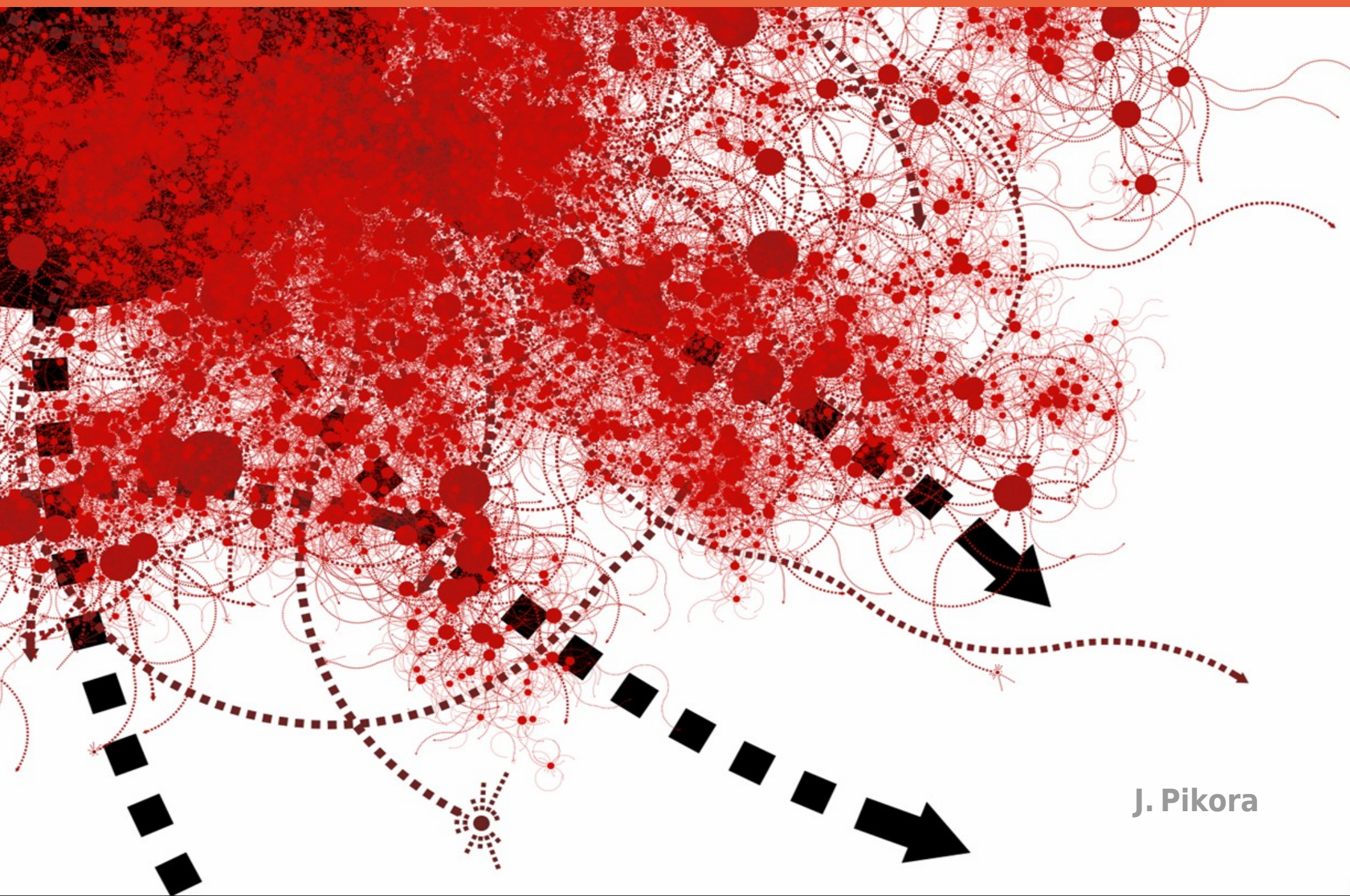


Předchozí obrázek je vykreslen pouze ze čtverců  
(s různou velikostí, barvou a průhledností)

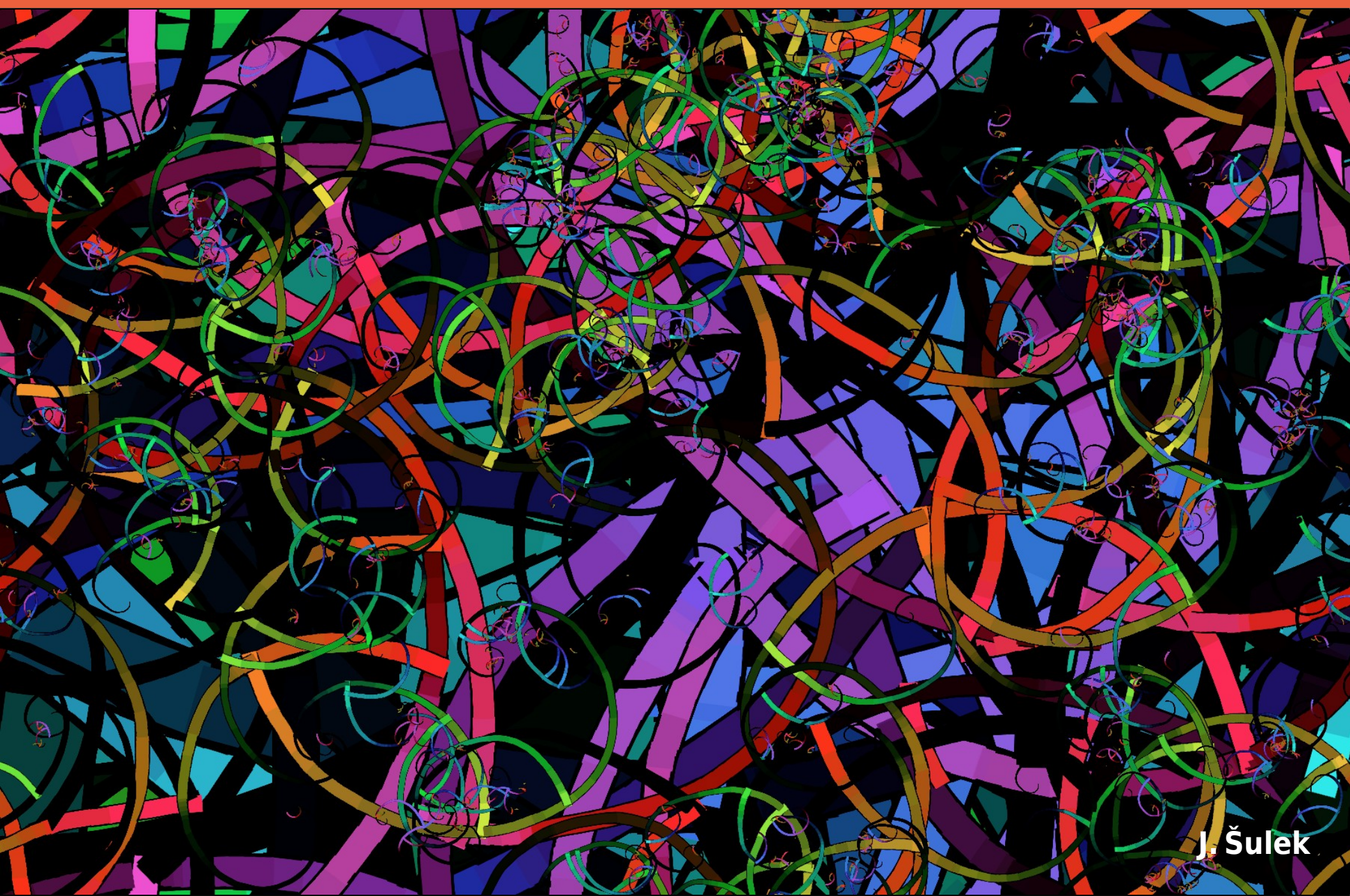
Pouze na vyznačených místech se ‚štětec dotkne plátna‘,  
vše ostatní je záležitostí výpočtů

```
startshape PLANT
rule PLANT {
  SQUARE {}
  PLANT {y 1.2 size 0.99 rotate 1.5 brightness 0.03}
}
rule PLANT 0.05 {
  PLANT {flip 90}
}
rule PLANT 0.05 {
  SQUARE {}
  PLANT {y 1.2 size 0.99 rotate 1.5 brightness -0.5 flip 90}
  PLANT {y 1.2 x 1.2 size 0.6 rotate -60 brightness -0.5}
  PLANT {y 1.2 x -1.2 size 0.5 rotate 60 brightness -0.5 flip 90}
  FLOWER {size 0.5}
}
rule SPROUT {
  LEAF {}
  BUD {}
}
```

```
rule LEAF {
  SHRIVEL {}
  LEAF {y 1 rotate 1.5 size 0.99 alpha -0.05}
}
rule BUD {
  SHRIVEL {}
  BUD {y 1 rotate -10.5 size 0.99 alpha -0.01}
}
rule SHRIVEL 0.01 {
  SQUARE {}
  LEAF {size 0.5}
}
rule FLOWER {
  SPROUT {hue 10 saturation 0.8 brightness 0.8}
  FLOWER {size 0.7 rotate 27.8 hue 5 alpha -0.1}
}
```



J. Pikora



J. Šulek



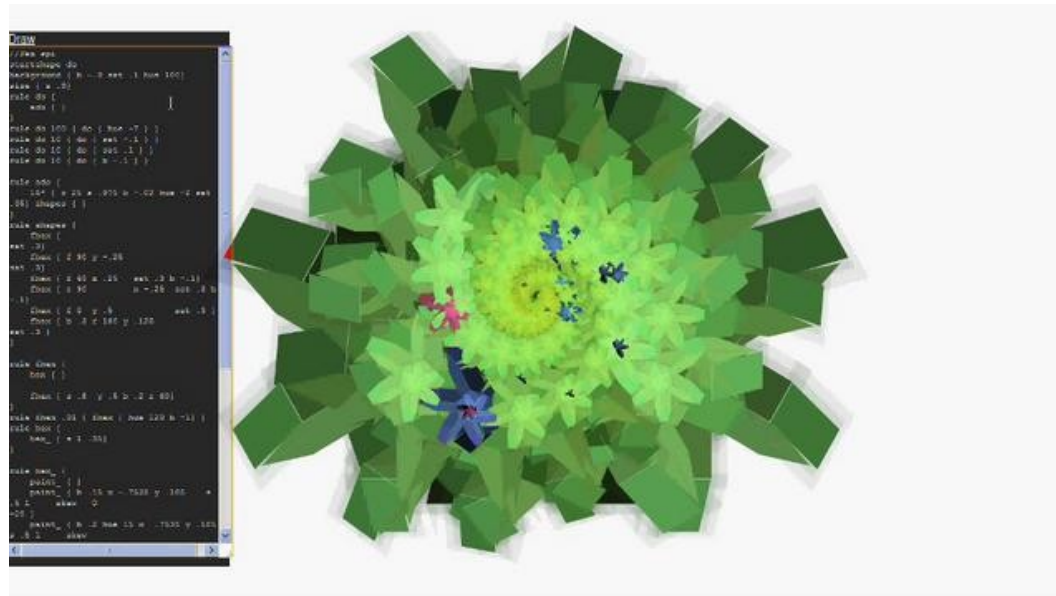
## ➔ Clnk

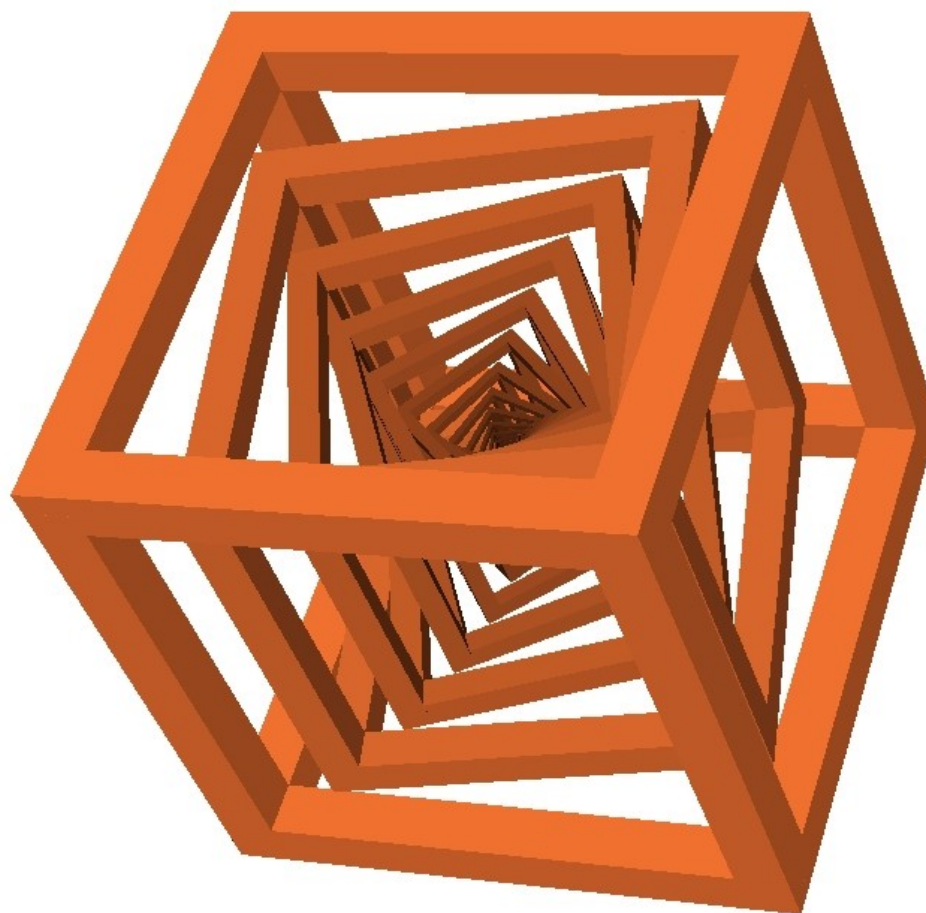
= jiný interpret jazyka CFDG, používá dialekt CFDG2

Nové grafické prvky: úsečka, text

Nové transformace: stíny, průsvitnost, rozostření

Interakční prvky: detekce pohybu a kliknutí myši





➔ **StructureSynth** — generování v prostoru

## Vizuální jazyk *EisenScript*

Na rozdíl od CFGD není čistě bezkontextový, umožňuje např. ‚stárnutí pravidel‘ — ukončení pravidla nebo nahrazení jiným pravidlem po určitém počtu iterací

```
set maxdepth 600
set background #FFF
{ h 30 sat 0.7 } spiral
{ ry 180 h 30 sat 0.7 } spiral

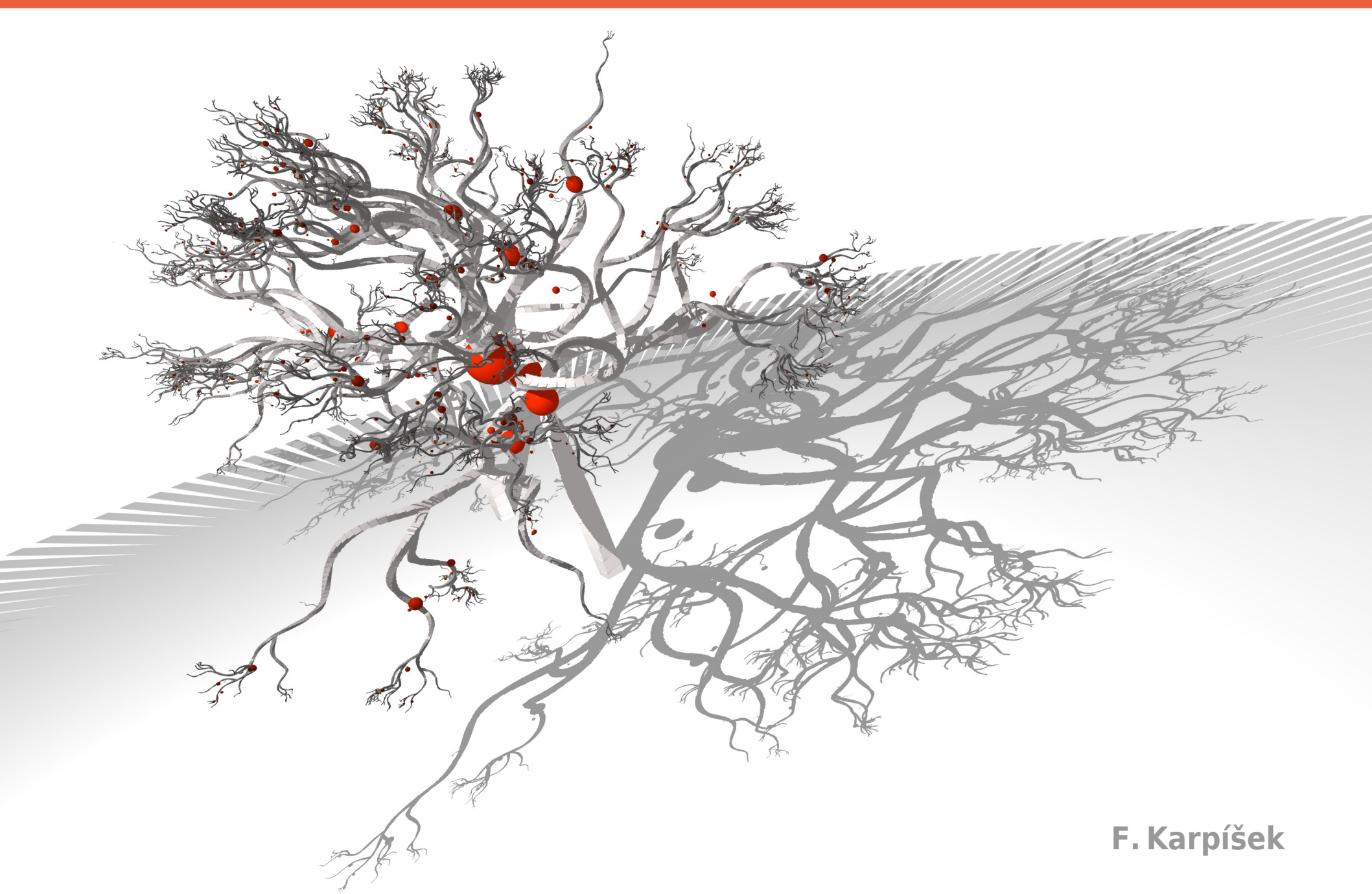
rule spiral w 100 {
  box
  { y 0.4 rx 1 s 0.995 b 0.995 } spiral
}

rule spiral w 100 {
  box
  { y 0.4 rx 1 ry 1 s 0.995 b 0.995 } spiral
}

rule spiral w 100 {
  box
  { y 0.4 rx 1 rz -1 s 0.995 b 0.995 } spiral
}

rule spiral w 6 {
  { rx 15 } spiral
  { ry 180 h 3 } spiral
}
```

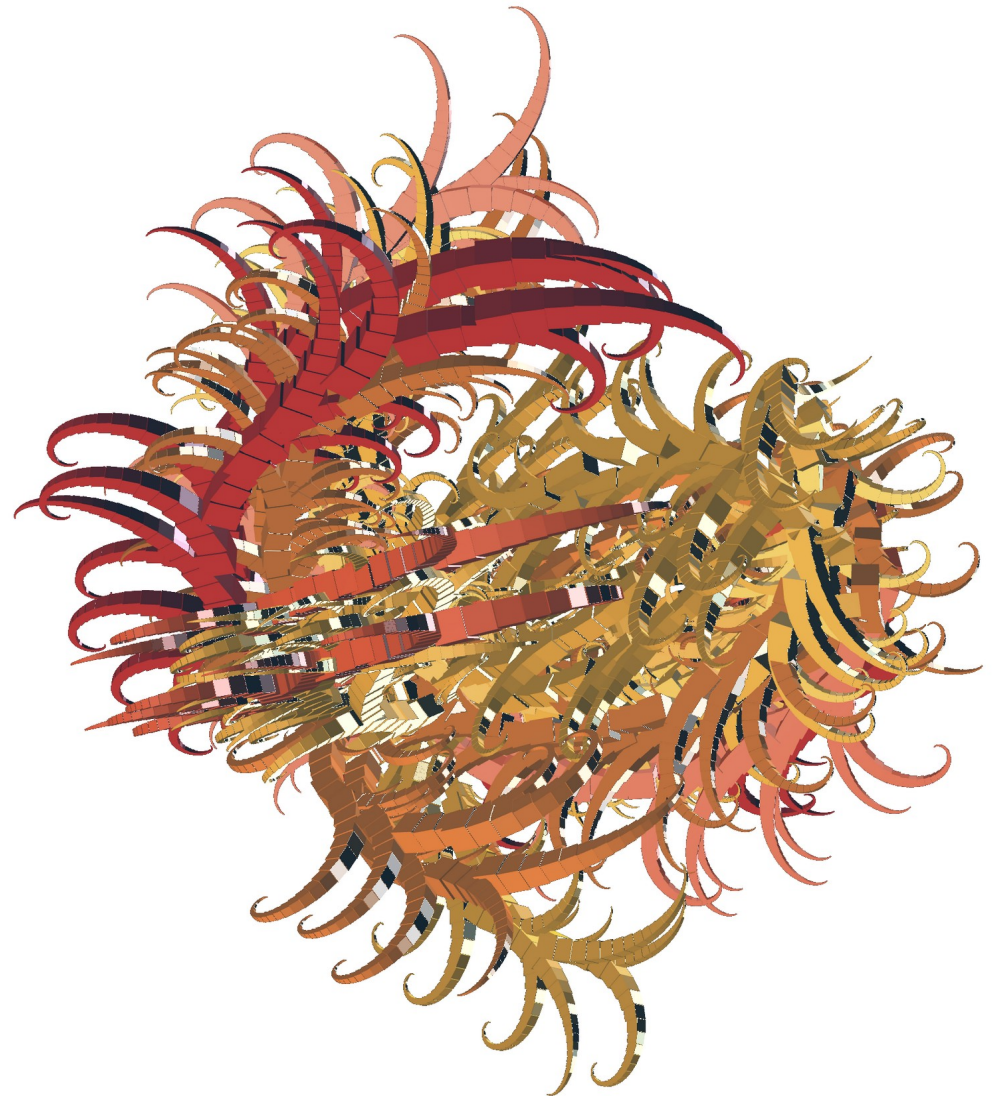


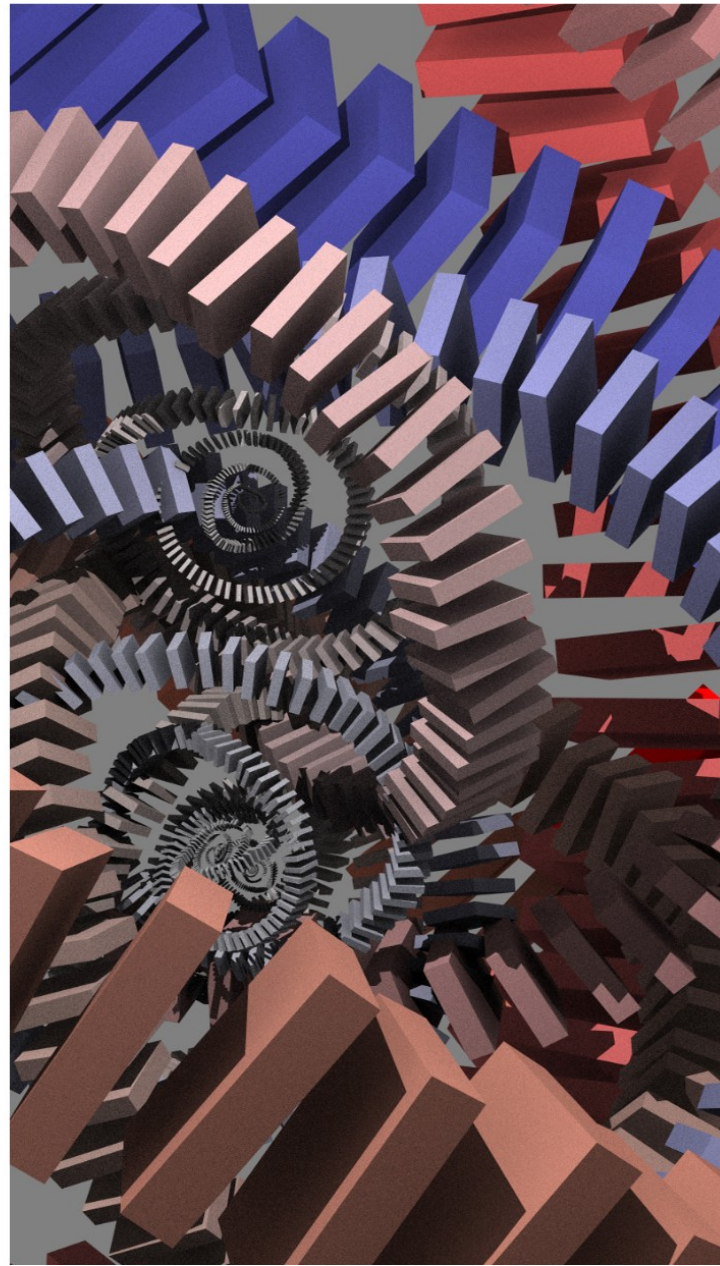
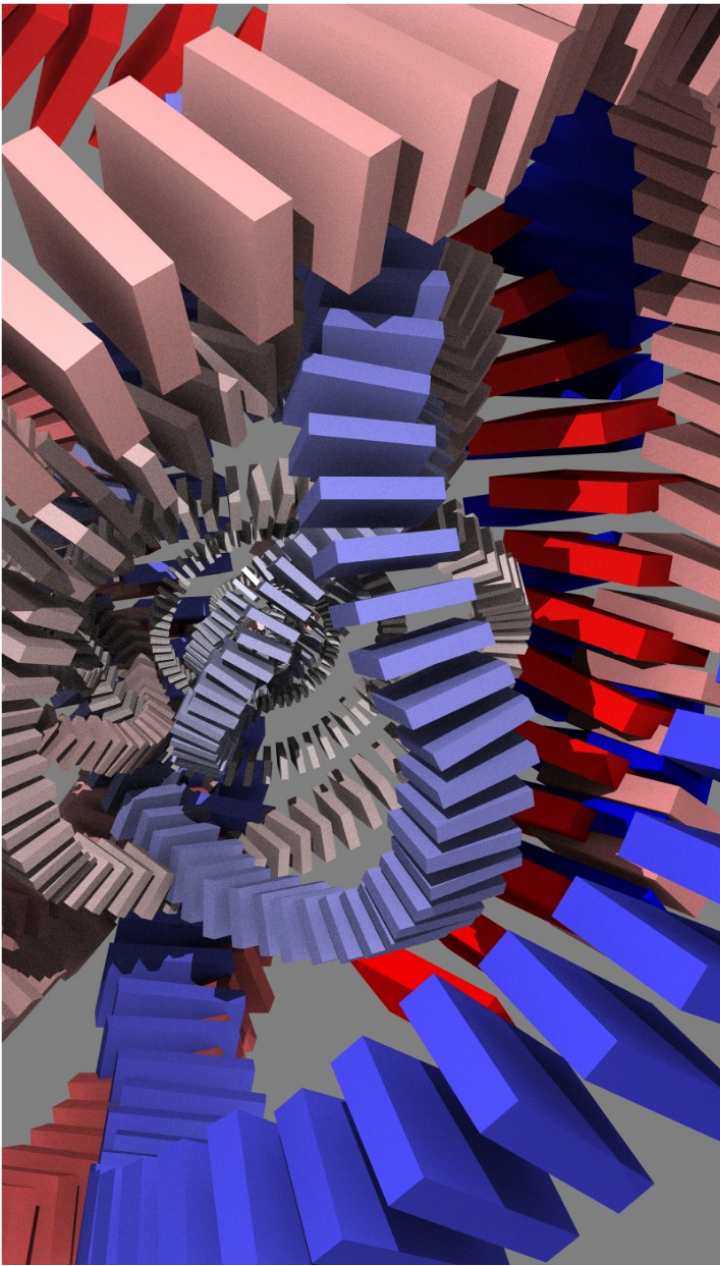


F. Karpíšek

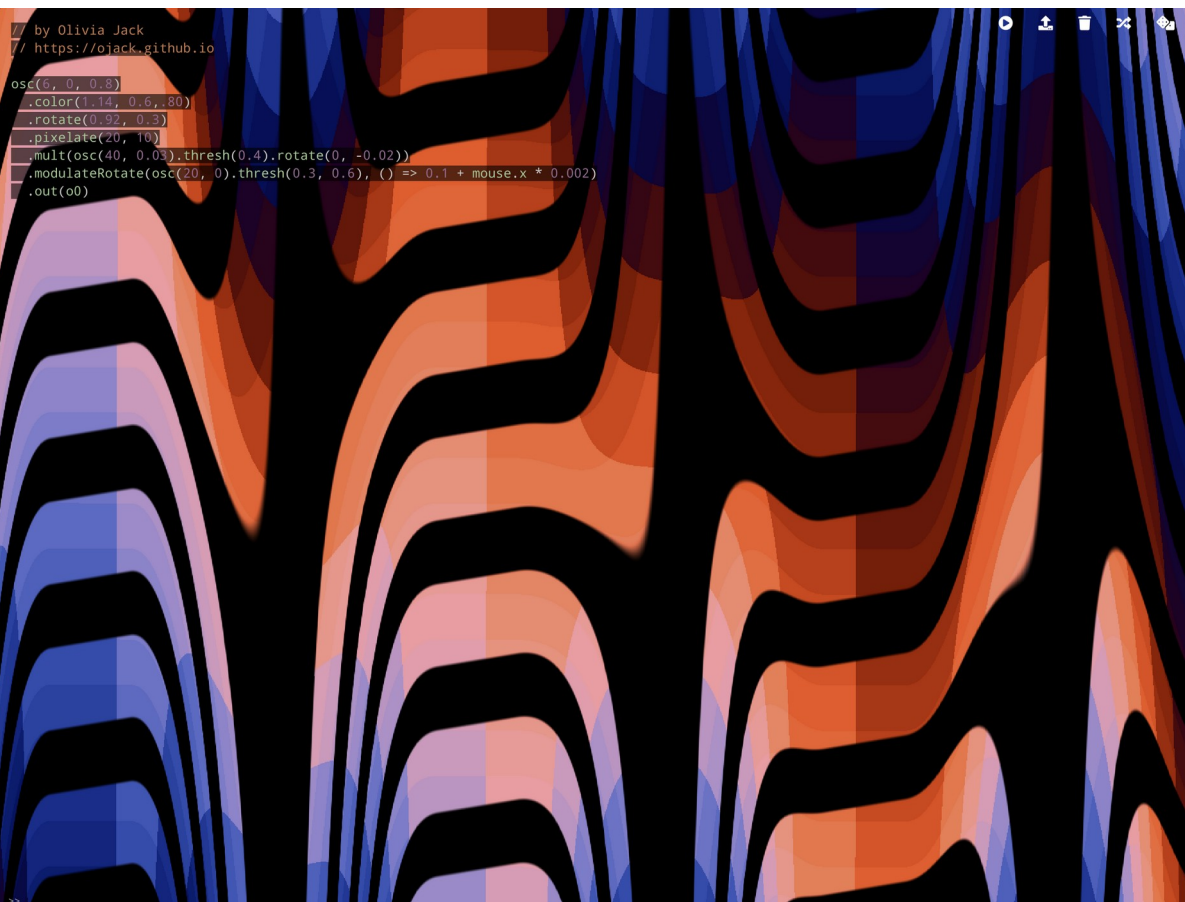
## ➔ Eisenscript Editor

```
{ s 0.2 } aa
rule aa maxdepth 4 {
  { y 1 } arc
  { y -1 } arc
  { ry 15 y 2 s 0.9 rz 35 hue 10 } aa
  { ry 15 y -2 s 0.9 rz -35 hue 10 } aa
}
rule spike {
  35 * { z 0.9 ry 5 s 0.9 } grid
}
rule spike {
  35 * { z 0.9 ry -5 s 0.9 } grid
}
rule spike {
  35 * { z 0.9 ry 10 s 0.9 } grid
}
rule spike {
  35 * { z 0.9 ry -10 s 0.9 } grid
}
rule arc {
  20 * { ry 12 }
  1 * { x 10 }
  spike
}
```





A. Štelclová



## Live coding — generování dynamických struktur v reálném čase

➔ [Livecodelab](#)

➔ [Hydra](#)

➔ [Shadertoy](#)

➔ [ASCII Play](#)



## Zadání

**Vytvořte grafiku  
(grafickou sérii nebo  
videosekvenci) pomocí  
bezkontextové gramatiky  
nebo živého kódování**

## Výstup

**Alespoň jedna grafika / video  
vč. generující gramatiky**

