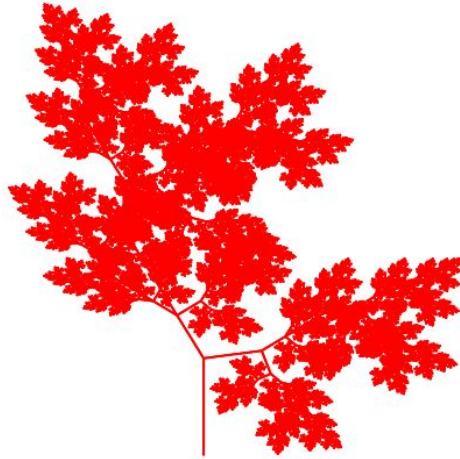


**Zadanie rastlina
Daniel Kazankov**

I_love_Canada



K zhotoveniu rastliny som použil parametre:

L-system generator options

Grammar Options	Curve Options	Colors
AF Skip Draw these symbols. Skip these symbols.	400 400 Curve width. Curve height.	white Background color.
A Starting axiom.	8 How many iterations to perform?	red Curve color.
F=FF Rewrite rule 1.	27 Angle Angle of rotation.	
A=F[-AF+[A]++A][+++A] Rewrite rule 2.	Up Starting direction of movement.	
Rule 3 Rewrite rule 3.	2 Thickness of a curve.	
Rule 4 Rewrite rule 4.	5 Padding around the image.	

Vysvetlenie pri 1 iterácii :

F=FF => dĺžka stoniek

A=F[-AF+[A]++A][+++A] => **A** je pravá stonka a **+++** je jej uhlové zakryvenie

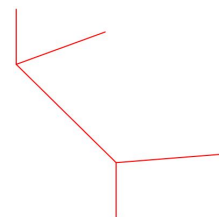
A=F[-AF+[A]++A][+++A] => **+** určuje pravú stranu

A=F[-AF+[A]++A][+++A] => **-** určuje rast ľavej stonky

A=F[-AF+[A]++A][+++A] => **AF** určuje dĺžku ľavej stonky

A=F[-AF+[A]++A][+++A] => **[A]** stredová stonka rastúca z ľavej stonky

A=F[-AF+[A]++A][+++A] => **+++A** pravá stonka rastúca z ľavej stonky, **+++** určuje zakryvenie jej uhla



Javorový strom som vytvoril skúšaním viacerých pravidiel pri rôznom počte iterácii. L systém je výborný nástroj k tvorbe rastlín simulujúcich realitu. Rastliny môžu doslova rásť z jedného jednoduchého kódu. K výsledku som sa dopracoval cez sériu pokusov, v ktorých zohrala rolu čiastočne logika a čiastočne náhoda pri skúšaní viacerých iterácii. Verím, že tento systém by bol použiteľný aj pri 3D modeloch.

V mojej hlave nastáva otázka, ktorú už riešil Turing pri komputačných strojoch. Môže sa naša realita skladať tiež z komplikovaného kódu, ktorý pri obrovskom počte iterácii sa vetví do našej reality?

Otázka späť s Blendrom: Ako môžem v Blendry, čo najefektívnejšie stvoriť realistický model s čo najmenším počtom polygónov (v l-systéme iterácii), tak aby som neprepálil systémové požiadavky, na ktorých musí bežať moja hra?