

Sem vložte zadání Vaší práce.



**FAKULTA
INFORMAČNÍCH
TECHNOLOGIÍ
ČVUT V PRAZE**

Diplomová práce

Engine pro renderování a procedurální generování voxelových světů

Bc. Lukáš Hepner

Katedra softwarového inženýrství
Vedoucí práce: Ing. Adam Veseczký

11. ledna 2022

Poděkování

Doplňte, máte-li komu a za co děkovat. V opačném případě úplně odstraňte tento příkaz.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítacových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 11. ledna 2022

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2022 Lukáš Hepner. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Hepner, Lukáš. *Engine pro renderování a procedurální generování voxelových světů*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.

Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

Klíčová slova Nahraďte seznamem klíčových slov v češtině oddělených čárkou.

Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

Keywords Nahraďte seznamem klíčových slov v angličtině oddělených čárkou.

Obsah

Úvod	1
1 Cíl práce	3
2 Modelování rostlin s využitím L-systémů	5
2.1 L-systém	6
2.2 Interpretace řetězců pomocí želvy	6
2.3 Větvení v L-systémech	6
2.4 Stochastické L-systémy	8
2.5 Implementace	10
2.5.1 Formát L-systému	10
2.5.2 Želva	10
2.5.3 Rozšířená abeceda	11
2.5.4 Ovládání želvy	12
2.6 Modelování rostlin	14
2.6.1 Simulace růstu	16
2.7 Výsledky použití L-systémů	16
Závěr	19
Literatura	21
A Seznam použitých zkratek	23
B Obsah přiloženého CD	25

Seznam obrázků

2.1	Identické stromy	5
2.2	Kvadratické Kochovy ostrovy	7
2.3	Struktury připomínajících rostliny	8
2.4	Využití stochastického L-systému	9
2.5	Nepravidelná koruna akáciového stromu	13
2.6	Akáciový strom rostoucí v přírodě	14
2.7	Model akáciového stromu	15
2.8	Tráva na planinách	15
2.9	Fáze růstu keře	16
2.10	Vegetace vygenerovaná na základě L-systémů	17

Úvod

KAPITOLA **1**

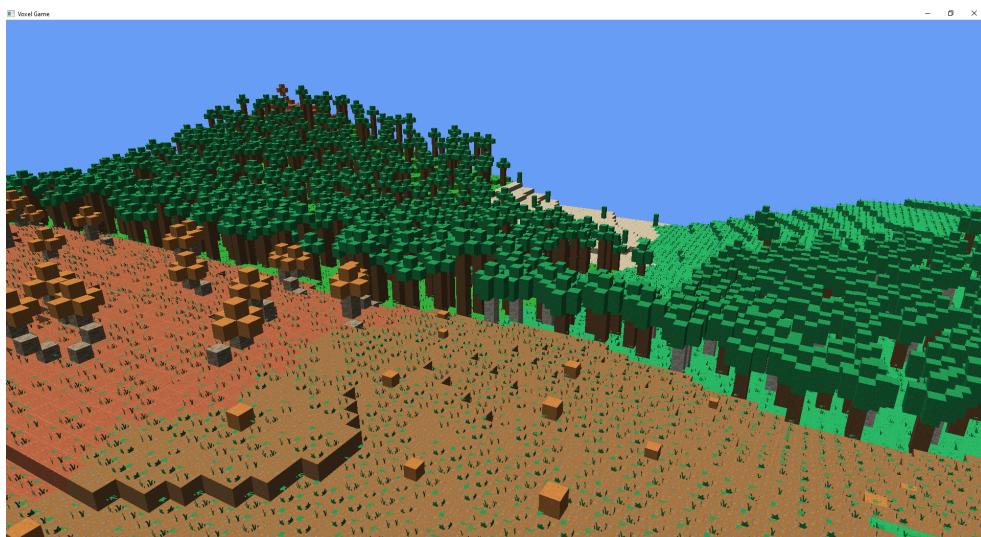
Cíl práce

KAPITOLA **2**

Modelování rostlin s využitím L-systémů

Herní svět obsahuje velké množství vegetace a ač jsou si všechny stromy, keře typově podobné, hráč si velice rychle všimne, že jsou identické. Na obrázku 2.1 vidíme tři druhy stromů, které jsou zkopírovány po scéně. Stromy v levé zadní části mají různou výšku, přesto působí umělým dojmem.

Stromy v reálném světě jsou si podobné – rozeznáváme jednotlivé druhy stromů, přesto neexistují dva stejné stromy. Pokud druh stromu zapíšeme formální gramatikou nazývanou L-systém, docílíme podobné struktury stromů, které se budou lišit v detailech.



Obrázek 2.1: Identické stromy

2.1 L-systém

L-systém nebo také Lindenmayerův systém je paralelní přepisovací systém vyvinutý maďarským teoretickým biologem a botanistou Aristidem Lindenmayerem v roce 1968. L-systém je typ formální gramatiky skládající se z abecedy, přepisovacích pravidel a počátečního axiomu. Pomocí postupného derivování počátečního axiomu je možné simulovat vývoj rostliny v čase [1].

2.2 Interpretace řetězců pomocí želvy

Řetězce lze graficky reprezentovat pomocí želvy, konsumující symboly abecedy. Každý symbol určuje akci, kterou má želva vykonat. Želva se může pohybovat ve 2D nebo 3D prostoru. Ve 2D si můžeme interpretaci představit jako želvu, držící tužku, pohybující se po papíře.

Želvu lze reprezentovat jako trojici (x, y, α) , kde (x, y) představuje kartézské souřadnice reprezentující polohu v prostoru a α úhel kam želva směruje. Zadáním délky kroku d a změny úhlu δ lze želvu ovládat pomocí následujících symbolů.

- F – Posun dopředu o délku d . Stav želvy se změní na (x', y', α) , kde $x = x + d \cos \alpha$ a $y = y + d \sin \alpha$. Mezi body (x, y) a (x', y') je nakreslena čára.
- + – Rotace doleva o úhel δ . Nový stav želvy $(x, y, \alpha + \delta)$.
- - – Rotace doprava o úhel δ . Nový stav želvy $(x, y, \alpha - \delta)$.

Nechť je definován následující L-systém. Bud' ω počáteční axiom, p přepisovací pravidlo, $\delta = 90^\circ$ a d zmenšené čtyřnásobně pro každý obrázek [2].

$$\begin{aligned}\omega &: F - F - F - F \\ p &: F \rightarrow F - F + F + FF - F - F + F\end{aligned}$$

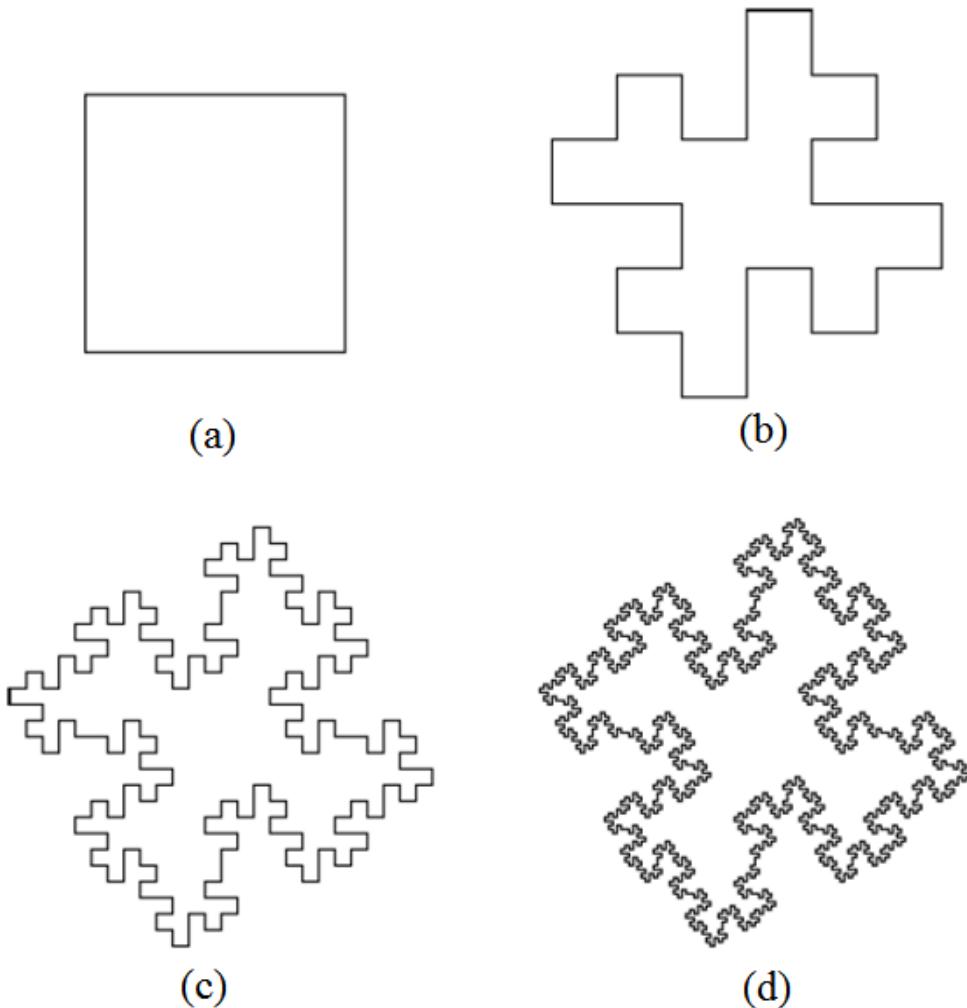
Želva interpretující daný L-systém generuje kvadratické Kochovy ostrovy 2.2. Obrázky jsou vygenerovány derivacemi o délce 0 až 3.

2.3 Větvení v L-systémech

S danými přepisovacími pravidly není možné generovat větvící se struktury. Želva vždy pokračuje od své poslední pozice. Říše rostlin je dominovaná větvícími se strukturami, potřebujeme proto matematické vyjádření této skutečnosti.

Větvení v řetězci můžeme reprezentovat pomocí dvou symbolů [a], kde [značí začátek větve a] konec větve [3].

Symboly jsou interpretovány želvou následovně:



Obrázek 2.2: Kvadratické Kochovy ostrovy

- [– Ulož atributy želvy do zásobníku.
-] – Načti atributy želvy ze zásobníku a smaž je z vrcholu zásobníku (operace pop). Při této operaci není nakreslená žádná čára.

Díky nově přidaným symbolům lze generovat struktury připomínající rostliny. Struktury na obrázku 2.3 jsou generované následujícími L-systémy:

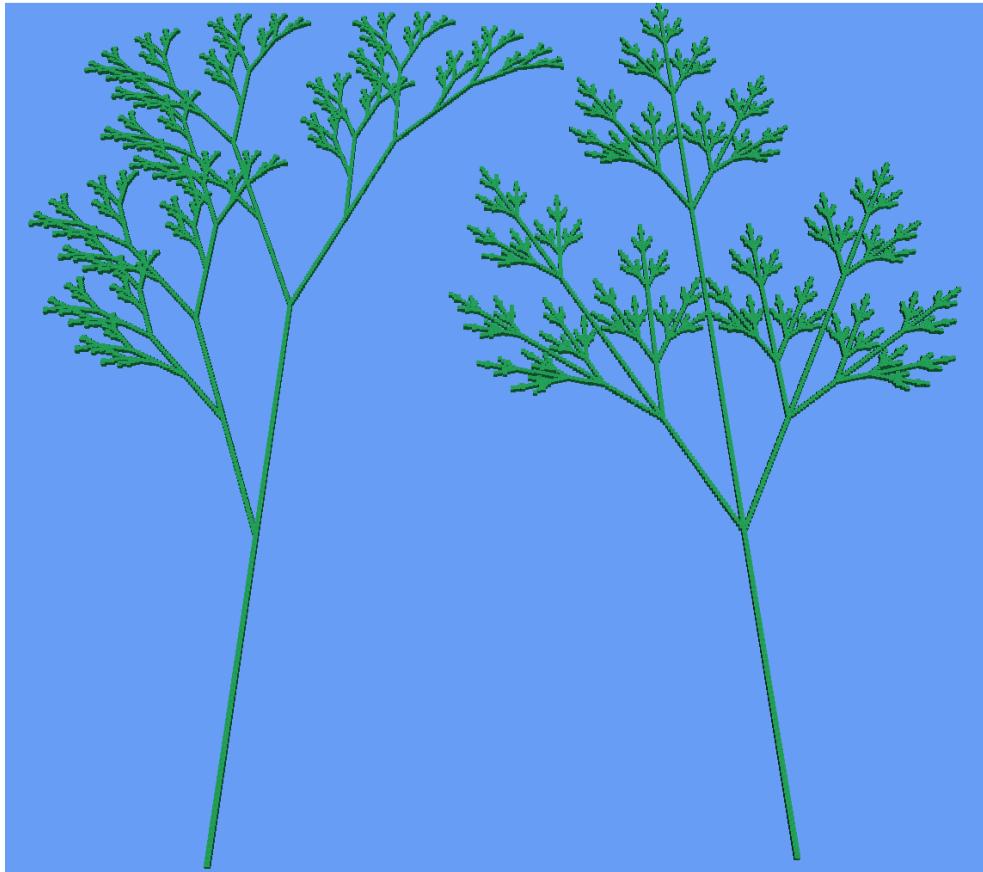
$$1. \delta = 20^\circ$$

$$\omega : E$$

$$p1 : F \rightarrow FF$$

$$p2 : E \rightarrow F[+E]F[-E] + E$$

2. MODELOVÁNÍ ROSTLIN S VYUŽITÍM L-SYSTÉMŮ



Obrázek 2.3: Struktury připomínající rostliny generované pomocí závorkovaného systému

$$2. \delta = 25,7^\circ$$

$$\omega : E$$

$$p1 : F \rightarrow FF$$

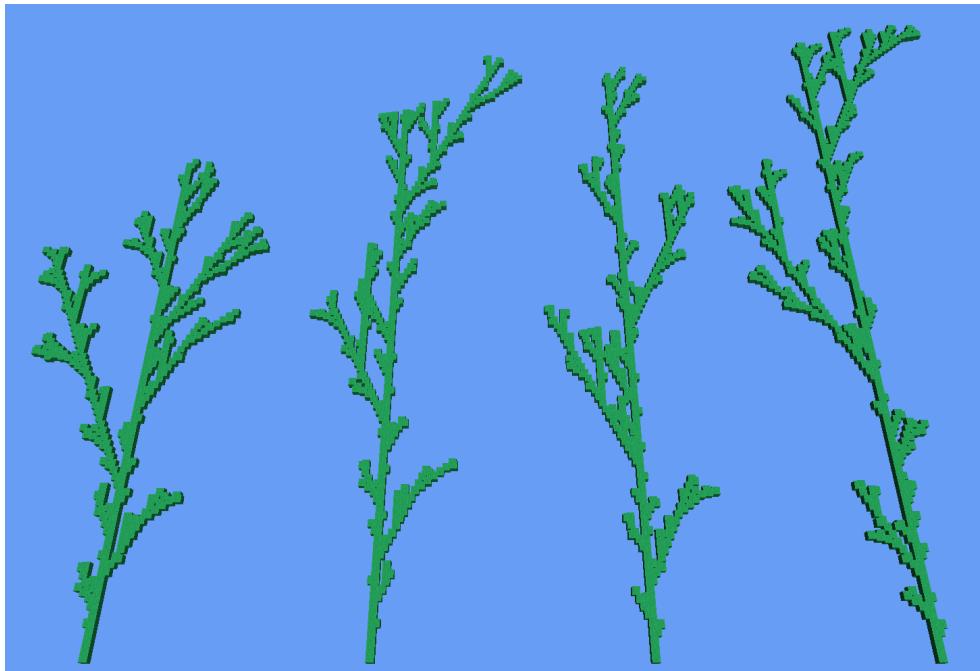
$$p2 : E \rightarrow F[+E][-E]FE$$

L-systém 1 generuje rostlinu vlevo, L-systém 2 generuje rostlinu vpravo.

2.4 Stochastické L-systémy

Rostliny generované deterministickým L-systémem jsou všechny stejné. Jejich použití v scéně by vytvářelo stejný efekt, který je popsán na začátku kapitoly.

K předejití tohoto efektu je nutné zavést variace v rámci druhu. Náhodná interpretace řetězce má limitované využití. Změna úhlu větvení, šířky a výšky



Obrázek 2.4: Využití stochastického L-systému

segmentů rostliny zachovávají topologii struktury, ze které je generovaná. Stochastické L-systémy mohou měnit topologii struktury [4].

-systém, který byl do ted' používán nemohl mít více přepisovacích pravidel pro stejný symbol abecedy. Pokud má stochastický L-systém vice přepisovacích pravidel, je z nich vybrána jedna s pravděpodobností $1/n$, kde n je počet přepisovacích pravidel pro daný symbol abecedy¹.

Scéna 2.4 byla vygenerována za pomocí stochastického L-systému, kde:

$$\delta = 25, 7^\circ$$

$$\omega : F$$

$$p1 : F \rightarrow F[+F]F[-F]F$$

$$p2 : F \rightarrow F[+F]F$$

$$p3 : F \rightarrow F[-F]F$$

¹Tato definice se liší, od definice uvedené ve [4]. Tento způsob náhodného výběru je použit v implementaci, kde pravděpodobností distribuci zastupuje několikanásobné zopakování přepisovacího pravidla.

2.5 Implementace

2.5.1 Formát L-systému

L-systém může být načten ze souboru pomocí třídy `LSystemParser`. L-systém musí mít následující formát:

```
yaw_angle pitch_angle shring_ratio
axiom
letter > production
.
.
.
letter > production
```

Soubor může obsahovat za sebou jdoucí L-systémy. `LSystemParser` je vrátí jako pole. Soubor může obsahovat komentáře na nových řádcích, začínající symbolem `#`.

Třída `LSystem` obsahuje gramatiku a tři atributy specifikující úhel náklonu podle osy y (yaw), podle osy x (pitch) a změnu velikosti bloku. Poslední atribut je využit při rozvětvení rostliny. Potomci mateřské větve by se měly řídit postulátem Leonarda da Vinci: „všechny větve stromu, v každé úrovni jeho růstu jsou v součtu jejich tloušťky rovné tloušťce kmene pod nimi.“ V případě dvojitého rozvětvení, tloušťky mateřské větve w_1 , tloušťky potomků w_2 dostaneme rovnici [5]:

$$w_1^2 = 2w_2^2$$

$$\frac{w_2}{w_1} = \frac{1}{\sqrt{2}} \approx 0,707$$

Hodnotu 0,7 je možné nalézt v L-systémech modelující keře.

2.5.2 Želva

Třída `Turtle` rozšiřuje pohyb želvy — popsané v kapitole Interpretace řetězců pomocí želvy — do 3D prostoru. Vnitřní stav želvy určují následující atributy:

- Pozice v prostoru.
- Velikost bloku vytvořeného želvou.
- Barva použitá pro kreslení (výstupní pole).
- Yaw — rotace podle osy y.
- Pitch — rotace podle osy x.

Želva si udržuje tři navzájem kolmé směrové vektory (nahoru, dopředu, doprava) jednotkové délky, které využívá pro pohyb po scéně. Vektory jsou aktualizované po každé rotaci. Pro výpočet je nutné znát vektor směrující kolmo vzhůru vůči scéně (WORLD_UP). Generovaný svět je plochý, proto lze tento vektor nahradit konstantním vektorem $(0, 1, 0)$.

```
glm::vec3 front;
front.x = cos(glm::radians(Yaw_)) * cos(glm::radians(Pitch_));
front.y = sin(glm::radians(Pitch_));
front.z = sin(glm::radians(Yaw_)) * cos(glm::radians(Pitch_));

Front_ = glm::normalize(front);
Right_ = glm::normalize(glm::cross(Front_, WORLD_UP));
Up_ = glm::normalize(glm::cross(Right_, Front_));
```

Délku x v rovině určené osami x a z lze spočítat jako délku přilehlé odvěsnky. $\cos(Yaw) = x/h$, kde h je délka přepony. Víme, že vektor má jednotkovou délku, proto $h = 1$. Stejný postup aplikujeme pro rovinu určenou osami x a y .

Tímto způsobem dopočítáme délky y a z vektoru směrujícího dopředu a normalizujeme ho. Jelikož jsou na sebe vektory kolmé, využijeme vektorového součinu, jehož výsledkem je vektor kolmý k oběma původním vektorům. Všechny vektory je nutné normalizovat, aby se předešlo jejich zkracování s tím, jak se Pitch blíží $\pm 90^\circ$.

K zamezení převrácení os jsou z definičního oboru Pitch vyjmuty násobky 90° .

```
if (Helpers::Math::Equal(cos(glm::radians(Pitch_)), 0.0f))
    Pitch_ -= 0.01f;
```

Výsledná nepřesnost je menší než maximální rozdíl dvou čísle typu float ϵ , která jsou považována za stejná. Funkce Equal porovnává desetinná čísla s přesností ϵ .

Želva vystavuje metody pro pohyb ve všech třech osách využívající vektorů Up_, Right_, Forward_. K pozici želvy je přičten patřičný vektor naškálovaný délkou pohybu. Např.:

```
void LSystems::Detail::Turtle::MoveForward(float dz) {
    Position_ += Front_ * dz;
}
```

2.5.3 Rozšířená abeceda

Následující symboly abecedy mají speciální význam pro jejich interpretaci.

- U a u — Posuň želvu nahoru.

2. MODELOVÁNÍ ROSTLIN S VYUŽITÍM L-SYSTÉMŮ

- F a f — Posuň želvu dopředu.
- x — Zmenši želvu.
- X — Zvětši želvu.
- S — Nastav původní velikost želvy.
- + — Rotuj želvu doleva podle osy y.
- - — Rotuj želvu doprava podle osy y.
- ^ — Rotuj želvu nahoru podle osy x.
- & — Rotuj želvu dolů podle osy x.
- [— Ulož kopii želvy na vrchol zásobníku.
-] — Vyjmi želvu z vrcholu zásobníku.
- 0 – 9 — Přepni výstupní pole.

L-systém může obsahovat jakýkoliv jiný ASCII symbol — mimo bílých znaků a # — určený pro expanzi přepisovacích pravidel. Není želvou interpretován.

2.5.4 Ovládání želvy

Implementace želvy se nachází ve jmenném prostoru `LSystems::Detail`, uživatel by ji neměl využívat přímo, ale je pro něj připravena třída `LSystemExecutor` zajišťující generování herních objektů z poskytnutého L-systému.

`LSystemExecutor` umožňuje generovat herní objekty na základě stochastického L-systému — topologie struktury výsledného modelu se může měnit mezi jednotlivými voláními generátoru na základě parametru `salt`. L-systém lze náhodně interpretovat na základě těchto parametrů:

- Rozsah počtu provedených derivací.
- Variace v rotaci želvy. K úhlu, o který se má želva otočit, se přičte $x * \text{původní úhel}$, kde x je z $[-\text{angleVar}, \text{angleVar}]$. Defaultní hodnota `angleVar` je 0,2.
- Výchozí velikost generovaných objektů. Lze určit rozsahem korespondujícím s počtem derivací.

Přidání náhodného úhlu má výrazný efekt na organický vzhled rostliny. Na obrázku 2.5 lze vidět akáciové stromy vyznačující se plochou korunou. V definici L-systému jsou všechny listy ve stejné výšce, výsledný rozdíl ve výškách je způsoben opakováním rotováním želvy. V zápisu lze vidět, že se

2.5. Implementace



Obrázek 2.5: Nepravidelná koruna akáciového stromu

želva otočí o 45° nahoru (^), pokládá větve (u, U), skloní se o 45° (&) a pokládá listy (F). Listy by tak měly být ve stejné rovině, ale nejsou.

L-systém generující akácie:

```
45.0 45.0 0.8
# make sure the plant has splits
# random lenght stem - then split
^uA1S&F+F+F+F
U > uU
A > +uuE
A > -uE
A > +uE
A > -E
A > uuE
A > uE
# top of the plant
E > x [++++UE1S&F+F+F+F] +UE
E > x [++++++UE1S&F+F+F+F] ++UE
E > x [++UE1S&F+F+F+F] -UE
```

Výstupem generátoru je 2D pole obsahující herní objekty rozdělené podle čísla výstupního bufferu, který měla želva při generování. Část enginu je tak odstíněna od textur, které jsou definované v části procedurálního generátoru. Díky tomuto rozdělení modelu je možné snadno měnit textury pro jednotlivá pole. Procedurální generátor tohoto využívá a používá stejný model – jiný běh generátoru – pro vytváření bříz a dubů, lišících se texturou kmene a listů.

2. MODELOVÁNÍ ROSTLIN S VYUŽITÍM L-SYSTÉMŮ



Obrázek 2.6: Akáčiový strom rostoucí v přírodě [6]

2.6 Modelování rostlin

Při modelování vegetace bylo třeba velkého množství pokusů a ladění, kdy rostlina nevypadala přirozeně, ale nebylo jasné, v jaké časti gramatiky je problém. Nejvíce se mi osvědčila technika nalezení reálné rostliny obrázek 2.6 a následné pokusy o její napodobení obrázek 2.7.

Velice se osvědčilo pravidlo:

$$U > uU$$

Díky němuž jsou větve blíže k zemi delší než větve navazující na korunu stromu. Pokud má rostlina význačné části je vhodné je modelovat samostatně (kmen, větve, koruna) viz L-systém generující akácie.

Tato technika ne vždy přinášela ovoce. Modelování trávy rostoucí na planinách se projevilo jako problém. Tráva neměla dostatečnou hustotu a nezapadala do kresleného vzhledu obrázek 2.8. S navyšujícím se počtem herních objektů dramaticky rostla spotřeba paměti. Jeden herní objekt s texturou trávy byl nahrazen desítkami herních objektů, z kterých se skládal model trávy. Tento problém by mohl být řešen přesunem vytváření modelu na grafickou kartu, přidáním vertexů v geometry shaderu.

Výsledné modely svou topologií připomínaly strukturu keřů. Byly proto upraveny — přidáním listí, změnou větvení — a využity v generátoru keřů.

2.6. Modelování rostlin



Obrázek 2.7: Model akáciového stromu



Obrázek 2.8: Tráva na planinách

2. MODELOVÁNÍ ROSTLIN S VYUŽITÍM L-SYSTÉMŮ



Obrázek 2.9: Fáze růstu keře

2.6.1 Simulace růstu

Křovinatý biotop je porostlý dvěma druhy keřů rozděleným do třech fází růstu. Ty jsou simulovány opakováním derivování počátečního axiomu. Keře nejmenšího vzrůstu jsou derivovány 2x, největší keře jsou derivovány 4x. Tloušťka kmene koreluje lineárně s počtem provedených derivací. Větší stromy mají širší kmen a dorůstají vyšší výšky.

Každý druh si zachovává své typické vlastnosti. Na obrázku 2.9² lze pozorovat stejné zakončení větví — rozdělení do dvou větví rostoucích na opačné strany — a podobný úhel v jakém se větve oddělují od kmene. Topologie rostliny se díky stochastickému L-systému mění mezi jednotlivými jedinci.

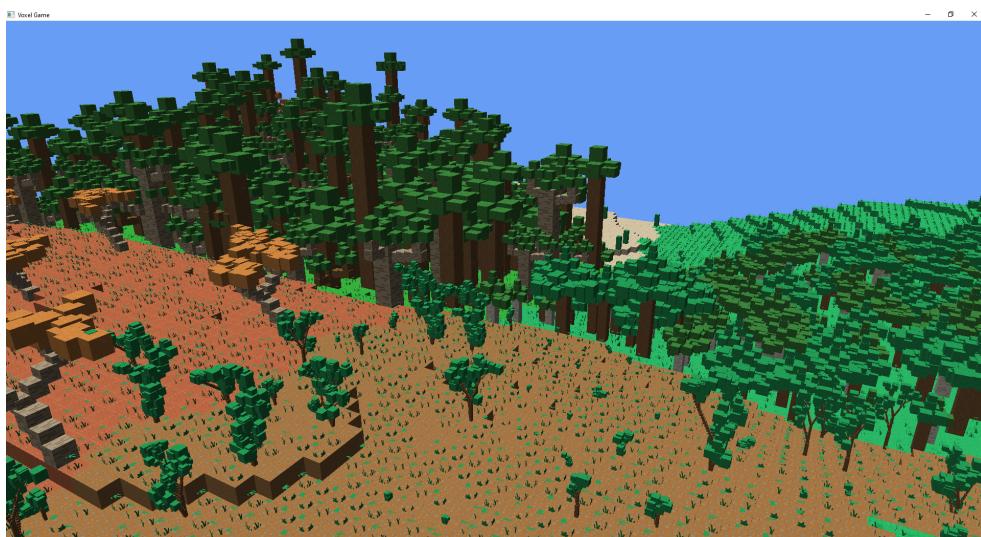
Díky těmto krokům biotop obsahuje rostliny navzájem podobného vzhledu, lišících se v drobných detailech.

2.7 Výsledky použití L-systémů

Definice L-systémů jsou krátké (≈ 10 řádků na jeden druh rostliny) a produkují velké množství rozdílných jedinců. Předchozí manuální definování rostlin v kódu bylo náročnější, a i při přidání více jedinců pro každý druh, by bylo snadné najít stejné. Odebráním definic rostlin z kódu se zvýšila jeho čitelnost — definice L-systémů jsou zdroje dat, který engine konzumuje. Nutnost komplikace při změně modelu byla odstraněna a zvýšila se rychlosť iterace, s kterou je možné upravovat model.

²Pro větší názornost byly odstraněny části modelu představující listy.

2.7. Výsledky použití L-systémů



Obrázek 2.10: Vegetace vygenerovaná na základě L-systémů

Vytvořením vlastního formátu pro zápis modelu rostliny se oddělila závislost na programovacím jazyce. Modely tak může vytvářet jiný člen týmu bez znalosti programování a překladu kódu.

Generováním rostlin za běhu programu se snížila rychlosť jeho běhu. Tento problém lze mitigovat cachováním rostlin obsahujících velké množství herních objektů. Toto bylo provedeno pro keře. Před spuštěním generace terénu je naplněn buffer obsahující keře vygenerované na základě seedu. Buffer musí být dostatečně velký na to, aby nedošlo ke snížení diverzity rostlin. Při vytváření keře na scéně je vybrán náhodný index do bufferu, závislý na pozici keře. Vybraný model je zkopirován a přesunut na dané místo.

Při porovnání scény 2.1 ze začátku kapitoly si lze všimnout přirozenějšího vzhledu krajiny. Koruny stromů se mohou překrývat, scéna díky tomu působí více organicky — stromy v přírodě nemají přesně stanové hranice, kde končí jeden a začíná druhý. Výsledná scenérie 2.10 působí méně jednolitě díky rozdílným vývojovým stádiím rostlin.

Závěr

Literatura

- [1] Togelius, J.; Shaker, N.; Nelson, M. J.: Grammars and L-systems with applications to vegetation and levels. In *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*, editace N. Shaker; J. Togelius; M. J. Nelson, Springer, 2016, str. 75, [cit. 2022-01-10].
- [2] Przemyslaw Prusinkiewicz, A. L.: Graphical modeling using L-systems. In *The Algorithmic Beauty of Plants*, Springer, 1996, str. 7, [cit. 2022-01-10].
- [3] Przemyslaw Prusinkiewicz, A. L.: Graphical modeling using L-systems. In *The Algorithmic Beauty of Plants*, Springer, 1996, str. 24, [cit. 2022-01-10].
- [4] Przemyslaw Prusinkiewicz, A. L.: Graphical modeling using L-systems. In *The Algorithmic Beauty of Plants*, Springer, 1996, str. 28, [cit. 2022-01-10].
- [5] Przemyslaw Prusinkiewicz, A. L.: Modeling of trees. In *The Algorithmic Beauty of Plants*, Springer, 1996, str. 57, [cit. 2022-01-10].
- [6] Fotografie akácie [online]. [cit. 2022-01-11]. Dostupné z: https://commons.wikimedia.org/wiki/File:Umbrella_thorn_acacia_or_israeli_babool_tree_plant_acacia_tortillis.jpg

Seznam použitých zkrátek

GUI Graphical user interface

XML Extensible markup language

Obsah přiloženého CD

```
readme.txt ..... stručný popis obsahu CD
├── exe ..... adresář se spustitelnou formou implementace
├── src
│   ├── impl ..... zdrojové kódy implementace
│   └── thesis ..... zdrojová forma práce ve formátu LATEX
└── text
    ├── thesis.pdf ..... text práce ve formátu PDF
    └── thesis.ps ..... text práce ve formátu PS
```