

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky
Centrum počítačové grafiky a vizualizace dat
Předmět: ZPG

Pohyb po virtuální planetě (Move around a virtual planet)

Zadání úlohy:

Vytvořit virtuální planetu pomocí trojúhelníkové sítě. Výšková mapa bude načtena z binárního souboru, jedna výška je reprezentovaná jedním bytem, výšky jsou uloženy v řádcích od západu k východu, řádky jsou uloženy od severu k jihu. Velikost mapy bude minimálně 128x128 s tím, že vzdálenost mezi jednotlivými body je 2 metry. Pozorovatel bude chodit po terénu a jeho oči budou ve výšce 1,85m. Dorazí-li pozorovatel na konec mapy, nebude moci pokračovat dále. Pozorovatel nesmí vidět ani vstupovat do terénu, proto bude použita interpolace mezi jednotlivými body. Rychlost pohybu bude 3m/s. Výchozí pozice bude uprostřed mapy. Veškeré pohyby a animace budou závislé na reálném čase. Planeta bude osvětlena sluncem tak, že jeden den bude trvat dvě minuty (1min. den , 1min. noc). Během dne se bude měnit intenzita osvětlení (intenzita slunečního svitu). Během noci bude terén osvětlen tak, aby bylo něco vidět. Přechod mezi dnem a nocí musí být plynulý. Pozorovatel se bude pohybovat konstantní rychlostí vůči času. Klávesou „W“ bude umožněn pohyb vpřed, klávesou „S“ pohyb vzad, klávesou „A“ úkrok vlevo, a klávesou „D“ úkrok vpravo. Jednotlivé pohyby lze kombinovat. Pomocí myši se bude moci otáčet a rozhlížet. Rozhlížení nahoru/dolů bude pouze v rozsahu (-90°; +90°). Pomocí klávesy „U“ lze invertovat rozhlížení nahoru/dolů.

Prohlášení:

Prohlašuji, že všechny použité zdroje jsou řádně citovány a předkládaná práce byla vytvořena zcela mnou, pokud není uvedeno jinak.

Hodnocení (upřesní cvičící, resp. vedoucí projektu): <ol style="list-style-type: none">1. vlastní aktivita při zpracování, doplnění zadání o dodatečné prvky2. náročnost řešené problematiky3. úplnost řešení jak z hlediska teoretického, tak i praktické realizace4. úroveň zpracování informačních zdrojů (knihy, časopisy apod.)5. technická kvalita (program funkční, výsledky relevantní, ...)6. forma zpracování	FOTO řešitele
---	---------------

e-mail studenta: rendlk@students.zcu.cz	Datum: 5.12.2011
Kód studenta: A09B0415P	
Jméno studenta: Kamil Rendl	Hodnocení (počet bodů):
Datum odevzdání: 6.12.2011	Jméno cvičícího: Ing. Oldřich Petřík
Podpis studenta:	Podpis cvičícího:

1. Úvod

- 1.1. Planetu vytvoříme jako výškovou mapu, kterou načteme z bytového souboru. Jednotlivé výšky budou reprezentovány jedním bytem. Planeta je obíhána sluncem, které osvětluje různě jednotlivé plochy terénu. Jeden den bude trvat 2min. Při pohybu po planetě musíme zajistit konstantní rychlost pohybu 3m/s všemi směry a při různém pohledu. Pohled a otáčení jsou realizovány pomocí myši. Také je nutné dát pozor na to, aby se pozorovatel nedostal mimo zobrazovanou mapu.
- 1.2. Výškovou mapu vytvořím jako dvourozměrné pole. Výška v daném bodě bude načtena ze souboru a bude uložena na daném místě v poli. Vzdálenost mezi jednotlivými body je 2 metry, proto použiji interpolaci pro určení výšky v místech mezi jednotlivými body, které jsou dány výškovou mapou. Rychlost pohybu budu počítat jako čas mezi jednotlivými framy vynásobenými vzdáleností, kterou ujdeme.

2. Známé metody

2.1. Interpolace:

Jelikož je terén vzorkován po 2 metrech, je nutné dopočítávat jednotlivé výšky mezi stěžejními výškami, které jsou zadány v mapě. Pro interpolaci jsem zvolil metodu Bilineární interpolace [1], kterou jsem vybral z několika metod. Ty jsem našel a nastudoval na internetu. Bilineární interpolace se mi zdála jako nejlepší volba pro toto konkrétní zadání. Tato metoda funguje na principu lineární interpolace. Při výpočtu nejprve dopočítáme lineární interpolaci v x-ovém směru a také v y-ovém. Poté vezmu tyto interpolace a dosadím je do vzorce, který je uveden níže. Výsledek je pak přibližná výška v daném místě.

$$\begin{aligned} f(x, y) \approx & \frac{f(Q_{11})}{(x_2 - x_1)(y_2 - y_1)}(x_2 - x)(y_2 - y) \\ & + \frac{f(Q_{21})}{(x_2 - x_1)(y_2 - y_1)}(x - x_1)(y_2 - y) \\ & + \frac{f(Q_{12})}{(x_2 - x_1)(y_2 - y_1)}(x_2 - x)(y - y_1) \\ & + \frac{f(Q_{22})}{(x_2 - x_1)(y_2 - y_1)}(x - x_1)(y - y_1). \end{aligned}$$

2.2. Pohyb:

Pohyb je v programu dopočítáván z azimutu (směr, kterým je postava právě natočena), koeficientu rychlosti a z počátečního bodu. Při zaznamenání stisku klávesy pro pohyb se provede posun po ose-y podle rovnice.

$$y = y + \text{rychlost} * (\text{float})\text{Math.Sin}((-\text{azimut} * \text{Math.PI} / 180) + \text{Math.PI} / 4)$$

Výsledné y je poté nová pozice osoby.

Pro osu x je vzorec obdobný, jen funkci sinus nahradíme funkcí cosinus.

$$x = x + \text{rychlost} * (\text{float})\text{Math.Cos}((-\text{azimut} * \text{Math.PI} / 180) + \text{Math.PI} / 4)$$

2.3. Kolize s pevnými předměty:

Každý neprůchodný prvek, který se nachází na mapě, je uložen v poli Vektorů⁴. Každý prvek zde vypadá jako Vector4(x,y,šířka,délka), kde x a y udávají výchozí bod na mapě, kde se nachází ona překážka, a šířka a délka nastavují délku a šířku překážky. Při každém pohybu se projde toto pole s kolizemi a zjistí se případná kolize. Dojde-li ke kolizi osoby s předmětem, nastaví se výchozí pozice před kolizí.

2.4. Animace vrat

Při zastavení postavy před vraty v okolí 2metrů a stisku mezerníku dojde k otevírání vrat. Tato animace je nastavena na dobu 3sec, za kterou by se měla vrata otevřít. Animace je prováděna jako zmenšování čtverce s texturou vrat. Toto zmenšování čtverce je závislé na čase. Po celkovém otevření vrat je možný vstup do domu. Po opuštění prostoru vrat se dveře automaticky zavřou.

2.5. Animace dveří:

Hlavní vstupní dveře nemají žádnou animaci. Při otevření dojde pouze k překreslení čtverce s texturou dveří na jinou pozici. Při opuštění prostoru dveří dojde k překreslení dveří na výchozí pozici. Hlavní dveře jsou závislé na sběru klíčů. Každé dveře jdou otevřít pouze s klíčem daného čísla, které je uvedené vlevo od dveří. Dveře ve štítu domu otevírat nejdu, jsou pouze jako kulisa.

2.6. Klíče:

Jednotlivé klíče jsou předměty, které lze sbírat a jsou vždy označeny číslem. Při sebrání klíče např. s číslem 2 nás tento klíč opravňuje otevírat hlavní dveře, vedle kterých je vlevo nahoře napsáno číslo 2. Obdobně to platí i pro ostatní klíče a dveře. Nebudeme-li mít klíč se správným číslem, není možné dveře otevřít.

2.7. Sběr klíčů:

Při průchodu objektem klíče dojde k odstranění tohoto objektu ze scény. Tím je zaručeno, že již máme tento klíč sebrán.

3. Seznam prvků dodělaných nad rámec základního zadání.

Textury –	svět i objekty jsou otexturované
Sběr předmětů –	Na mapě se nachází čtyři klíče, které když sebereme, můžeme procházet dveřmi s daným číslem
Animace vrat –	garážová vrata se otevírají pomocí animace.
Pevné objekty -	objekty které jsou neprůstupné (alespoň 10 objektů (4x dům, 4x velký plot kolem domu a 4x malý plot před domem.))
Herní motiv -	Nutnost sbírání klíčů, aby bylo možné procházet vstupními dveřmi.

4. Ovládání programu

Pomocí myši se lze otáčet kolem své osy, předklánět a zaklánět.

Klávesy pro pohyb jsou: „W“ – pro pohyb vpřed
„S“ – pro pohyb vzad
„D“ – pro pohyb vpravo
„A“ – pro pohyb vlevo

Klávesy : „U“ – pro inverzi předklonu a záklonu
„Mezerník“ – při stisku mezerníku v okolí dveří nebo vrat dojde k jejich otevření

5. Závěr

Stěžejní body základního zadání jsem, myslím, splnil. Načtení a vytvoření výškové mapy pomocí trojúhelníkové sítě funguje bez problémů. Výpočet normálových vektorů pro jednotlivé vrcholy je také bezproblémový. Pohyb pozorovatele po planetě funguje dobře. Planeta je otexturovaná. Slunce, které obíhá planetu, je reprezentováno objektem bod. Osvětlení planety sluncem funguje bez problémů. Přechody mezi dnem a nocí jsou plynulé. Slunce obíhá planetu přesně v daném poměru 1 minuta den, 1 minuta noc.

Problematiku některých věcí jsem konzultoval s Janem Moulisem, Václavem Rajtmajerem a Jakubem Rinkešem.

6. Literatura

Funkce pro interpolaci (online publikace):

[1] Bilineární interpolace. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 23. 11. 2007, 11:34, last modified on 27. 2. 2011, 12:38 [cit. 2011-12-07].

Dostupné z WWW: http://cs.wikipedia.org/wiki/Biline%C3%A1rn%C3%AD_interpolace