

Projekt PV112

Minimální požadavky:

- Alespoň 10 objektů ve scéně, z nichž alespoň 8 bude složitějších (načtených ze souboru). Objekty se mohou opakovat, ale scéna musí mít minimálně 3 různé objekty (např. stoly, židle a skříně). Zbylé objekty mohou být generovány procedurálně či uloženy přímo v kódu (kostka, koule, obdélník, konvička apod.)
- Alespoň 8 objektů musí být otexturováno alespoň 3 různými texturami (např. 2 stoly z třeshňového dřeva, 5 židlí z dubu a otexturovaný koberec na podlaze). Zbylé objekty mít texturu nemusí (např. stříbrný přibor).
- Alespoň jeden objekt musí být (viditelně) animován. Může měnit pozici, barvu, intenzitu osvětlení apod.
- Scéna musí obsahovat alespoň 2 světla, musí být osvětleno alespoň 8 objektů.

Toto jsou minimální požadavky. Každá aplikace musí navíc něco reprezentovat (viz dále) a tím si naklade větší požadavky na některé z výše uvedených oblastí.

Aplikace musí něco reprezentovat, možnosti jsou:

- Jednoduchá hra: Takové aplikace mohou mít mnoho stejných opakujících se objektů, mnoho stejných textur, mnoho stejných jednoduchých animací, jednoduché osvětlení a fixní kameru. Hra se hlavně vyznačuje tím, že se dá hrát, tedy hráč má nějaký úkol a ten lze splnit. Hra má také často nějakou jednoduchou detekci kolizí, aby se hráč nemohl dostat někam, kam nemá. Hra **obsahuje** nějaké jednoduché GUI, ve kterém se zobrazují věci jako princip hry, ovládání, počet získaných bodů, aktuální obtížnost, čas, úspěš/neúspěš apod.
 - Příklady: Snake, Bludiště, Sokoban, Hanojské věže
- Hezká scéna: Takové aplikace mohou mít velmi málo animací (například jen kyvadlo u hodin či rybička v akváriu) a jednoduché ovládání, vyznačují se ale tím, že jsou propracované a "hezké". Mají mnoho různých objektů, mnoho různých textur, kameru, se kterou si můžou scénu prohlížet, několik světel a komplikované shadery.
 - Příklad: pokoj, v něm stůl s počítačem a několika židlemi, lampička, lustr, koberec na podlaze, šatní skříň, polička s knihami, postel s peřinami, na stěnách tapety, několik obrazů a kyvadlové hodiny ukazující aktuální čas. Světla jsou výhradně bodová a kuželová, s útlumem.
- Líbivý efekt: Takové aplikace mají relativně málo objektů, ale mají propracovaný efekt, kterým může být například hořící pochodně, tryskající fontána, či střelba a následný výbuch rakety. Tyto aplikace se vyznačují propracovanou animací, která může obsahovat vznikající a zanikající objekty (plameny pochodně, kapky fontány), mnoho měnících se textur, reprezentující například plameny, osvětlení měnící se podle intenzity plamene pochodně, a často vše toto dohromady. Efekt půjde velice jednoduše ovládat, například vystřelit z rakety či pohybovat pochodní. Ostatní objekty ve scéně jsou velice jednoduché a slouží hlavně k tomu, aby se na nich mohl ten efekt projevit (jsou například osvětlovány tou pochodní).

Aplikace navíc musí obsahovat alespoň 4 z následujících technik:

- environment mapping se skyboxem
- toon shading aplikovaný na celou scénu (vyjma věcí, u kterých to nedává smysl, jako například GUI)
- bump mapping / normal mapping / parallax mapping / parallax occlusion mapping
- blending
- kuželová světla
- mlha
- útlum světla / ambientní a pomocná světla
- větší množství (hlavních) světla (10 a více), například chodník osvětlený řadou pouličních lamp, nemusí ale všechna světla svítit na jeden objekt.
- jednoduchá detekce kolizí (aby hráč neprošel stěnami bludiště, aby had snědl nějakou věc apod.)
- jednoduchá fyzika (padání věcí gravitací), ale ať je tam alespoň zrychlení, rovnoměrný přímočarý pohyb jako fyziku nebereme
- projekční textury (projektor apod.)
- procedurální textury
- procedurální geometrie (generování terénu podle výškové mapy, deformace objektu podle jeho umístění ve světě apod.)
- rovinné zrcadlo
- view frustum culling / portal culling, ale pouze na rozsáhlejších scénách obsahujících více než 50 složitějších objektů
- instancování / částicový systém, pouze pokud pracuje alespoň s 50 jednoduchými objekty
- zvuky (zvuk sebrání potravy hadem, zvuk výbuchu rakety, apod.)
- pokud vás něco zajímavého napadne, napište nám a pokud vám to odsouhlasíme, tak to můžete použít

K osvětlení, technikám apod.:

- ***Objekty, osvětlení a použité techniky musí do scény pasovat a musí dávat smysl.*** Například nebereme stůl, který si jen tak pro nic za nic svítí, jen aby se splnila podmínka počtu světla ve scéně. Stejně tak nepřijmeme konvičku umístěnou doprostřed místnosti, která na sobě odráží oblohu, nebo normal mapping, který nereaguje správně na osvětlení. Hořící stůl vydávat světlo může, ale měl by být doprovázen nějakou animací plamenů. Stejně tak může svítit potrava, kterou má had sníst, aby byla ve scéně zvýrazněna a byla lépe vidět.
- Objekty, které nemají být osvětlovány (plameny, žárovka, text na obrazovce) by osvětleny být neměly.

Bonusové body:

- Za věci, na kterých si dáte záležet, ať už to bude vizuální věc (částice, které se rozprsknou vždy, když had něco sežere) či programátorská věc (nekonečné, náhodně generované bludiště). Max. 5 bodů.

Termín a forma odevzdávání:

- Individuální prezentace projektů, v termínech od **15. 5. 2017** (poslední týden semestru)
- Nutné projekt odevzdat a obhájit před tím, než půjdete na zkoušku! Termíny pro obhájení projektu budou vypsané v ISu, podobně jako zkouškové termíny.

FAQ, neboli otázky studentů:

- Co je složitý objekt, jak moc má být složitý objekt složitý, je stůl složený z 5 krychlí složitý objekt?
 - Jako složitý objekt chápeme objekt načítaný z externího souboru (OBJ apod.). Tato podmínka v zadání má zamezit tomu, abychom obdrželi projekty složené jen z hromady kostek a koulí. Proto prosím nepodvádějte například tím, že si kostku uložíte do OBJ a načtete. Chceme vidět něco „zajímavějšího“ než věci ze cvičení.
- Může být hra ošklivá?
 - „Může“, ale jen pokud to bude hra. Každý z těch druhů aplikací se zaměřuje na nějakou věc (hra na interaktivitu, hezká scéna na techniky a shadery, efekt na animace). Proto hra nemusí mít dokonale propracované osvětlení nebo úžasné animace. Ale musí se dát hrát, na rozdíl od ostatních.
- Můžeme používat knihovny třetích stran?
 - Ano, ale. Můžete používat určité ty knihovny, co se používají na cvičeních (a ty by vám měli stačit). Můžete ale také používat další knihovny pro načítání speciálních formátů obrázků, geometrií apod., ale jen pokud ty knihovny provedou pouze „C++/Java“ část a funkce OpenGL nechají na vás. Zjednodušeně, pokud je výstupem pole dat, tak je to OK, pokud je výstupem OpenGL buffer, OpenGL textura, objekt s metodou Draw(position), tak to už OK není.
- Můžeme použít jiné programovací jazyky?
 - Ano, ale. Pokud se budete držet C++ / Javy a jen nebudete psát v tom jazyce, ve kterém máte cvičení, tak se nic nestane. Pokud zvolíte něco jiného (C#, Python, Prolog, Haskell, ...) a budete mít s tím problémy, dost možná vám nebudeme schopni pomoci a budete odkázáni sami na sebe. Pokud to ale zvládnete, tak to taky přežijeme. Nezapomeňte ale, že ten svůj projekt nám budete prezentovat.
- Může to být 2D?
 - Ne, ale stačí alespoň 2,5D. OpenGL je hlavně pro 3D grafiku a chceme, aby se v projektu vyskytovaly 3D objekty, 3D transformace atd. Ty ovšem jste schopni použít i ve 2,5D plošinovce, proto nám 2,5D plošinovka vadit nebude.

