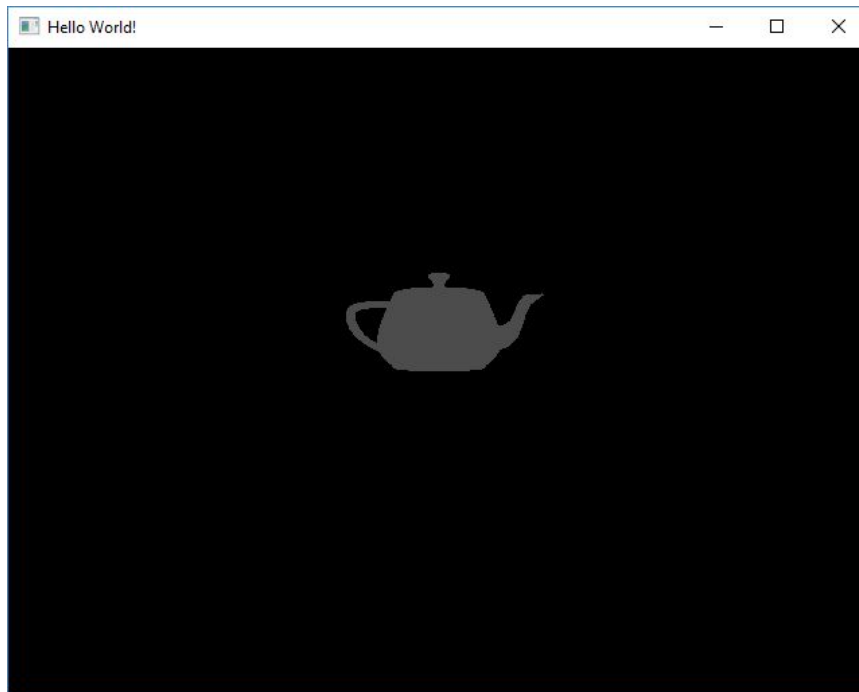


PV112 Java - 3. cvičení

Stáhněte si cv3.zip ze studijních materiálů. Rozbalte a složku otevřete v NetBeans `File -> Open project...` (`Ctrl + Shift + O`). Po úspěšném otevření, zkompilování a spuštění projektu by se mělo zobrazit okno 640x480 pixelů uprostřed obrazovky s nadpisem "Hello World!" a šedou neosvětlenou konvičkou.

Dnes budeme pracovat v souborech `Cv3.java` a `model.fs.glsl` a `model.vs.glsl`. Po dostatečném zorientování ve struktuře třídy doporučuji stiskem `Ctrl + Shift + Mínus` sbalit všechny bloky a rozbalit si pouze metody `init()`, `render()` a `drawModel()`.

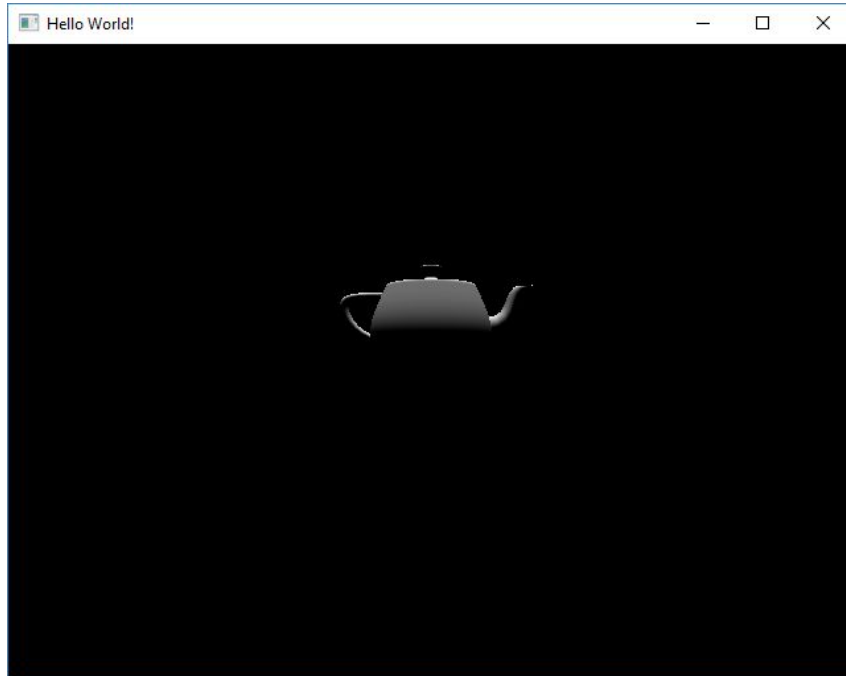
V tomto cvičení navíc máte možnost ovládat pozici kamery; při držení levého (primárního) tlačítka myši se můžete otáčet kolem středu, a při držení pravého tlačítka můžete přibližovat/oddalovat scénu.



Zkuste se na konvičku podívat z více úhlů, a uvidíte, že nemá zatím žádné „stínování“.

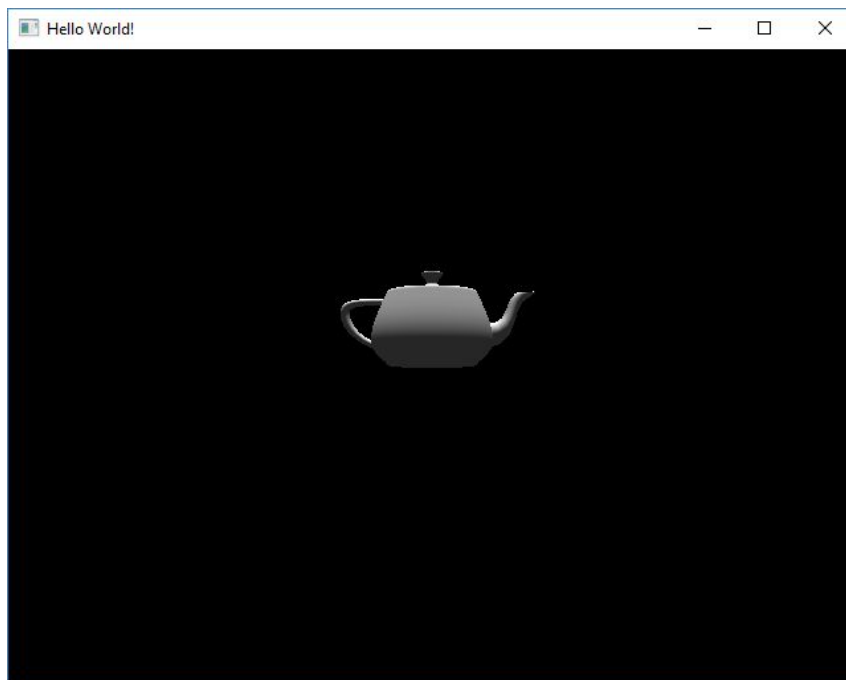
Task 1 – směrové světlo, difúzní složka světla

Jako první světlo si naimplementujeme směrové světlo (*directional light*). Směrové světlo nemá žádný pevně daný zdroj, svítí pouze určitým směrem na všechny objekty ve scéně (jako např. slunce). Pro výpočet světla budeme muset získat normály v souřadnicích světla (world space), proto je budeme muset ve vertex shaderu přetransformovat pomocí transponované inverze model matice. Následně ve fragment shaderu musíme spočítat tzv. difúzní faktor, který je udán rovnicí $\max(N \cdot L, 0)$ a tímto vynásobit difúzní složku světla. Prohlédněte si konvičku kamerou pod různými úhly.



Task 2 – ambientní složka světla

Pro přidání světla i na nepřímo osvětlené části musíme přidat tzv. ambientní složku světla. Ta nám udává, kolik světla je nepřímo přijímáno z okolí. Nastavte si ji na nějakou tmavou šedou, a uvidíte konvičku celou. Toto světlo přimícháme k difúznímu světlu, a dostaneme následující výsledek.



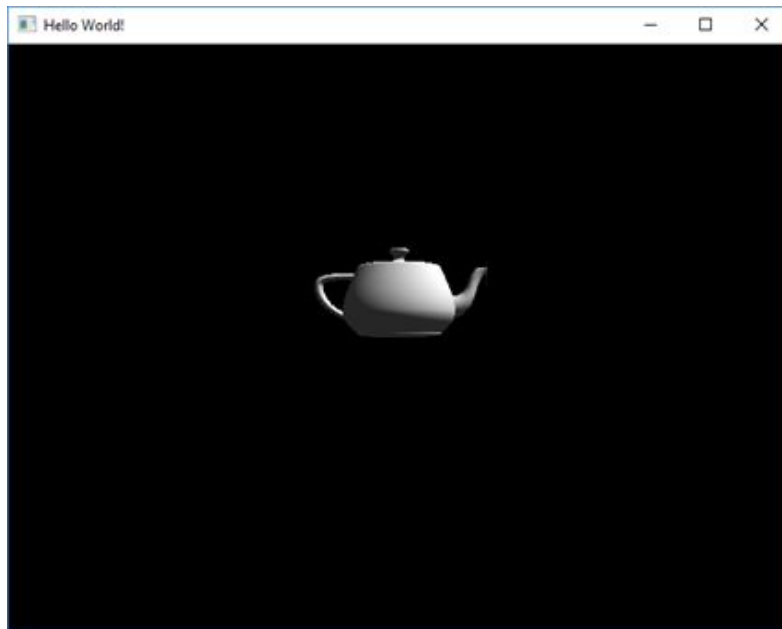
Task 3 – uniformní proměnné

Mít vlastnosti světla zadrátované v shaderech není velmi šťastné řešení – proto je na čase udělat menší refactoring. Za pomoci uniformních proměnných umožněte nastavit pozici, ambientní barvu a difúzní barvu přímo z Java kódu.

Třídni proměnné máte už připravené, stačí vám tedy získat jejich pozici pomocí

`glGetUniformLocation` z vašeho model programu a při renderování nastavit těmto uniformním proměnným hodnoty.

Hodnoty difúzní a ambientní složky nechte stejné jako byly natvrdo zadrátované v shaderu, pozici ale nastavte na (2, 3, 1, 0).

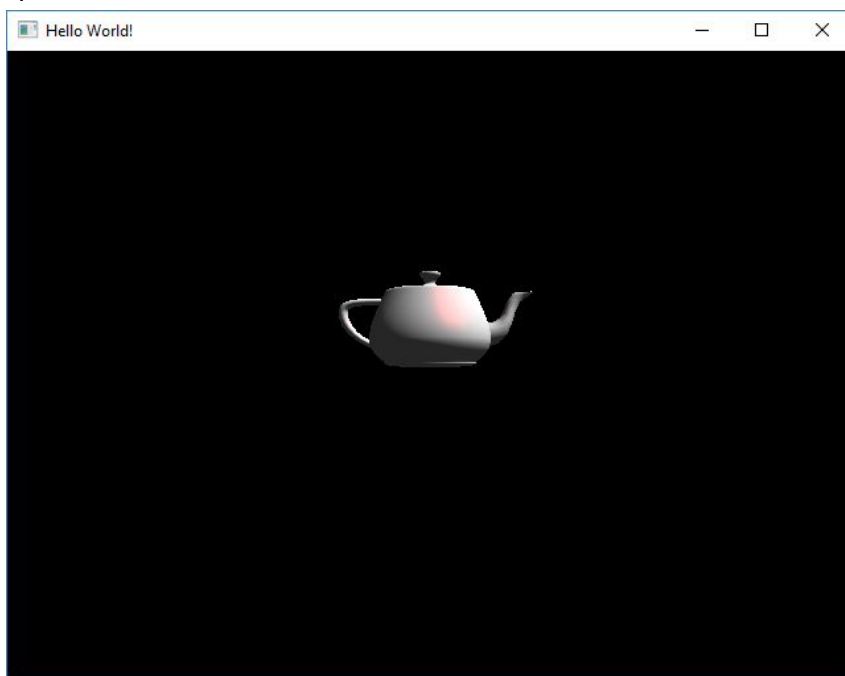


Task 4 – přidání spekulární složky světla

Spekulární složka světla udává odlesky, které záleží na pozici pozorovatele – musíme si tedy do shaderu předat kromě spekulární barvy světla i pozici kamery.

Musíte si spočítat tzv. half-vector, který je součtem směru pohledu a směru světla, a spočítáme spekulární faktor, kterým vynásobíme spekulární barvu světla.

Nastavíme si v Javovém kódu spekulární barvu na čistě červenou, a v fragment shaderu barvu přimícháme k naší spočtené barvě.

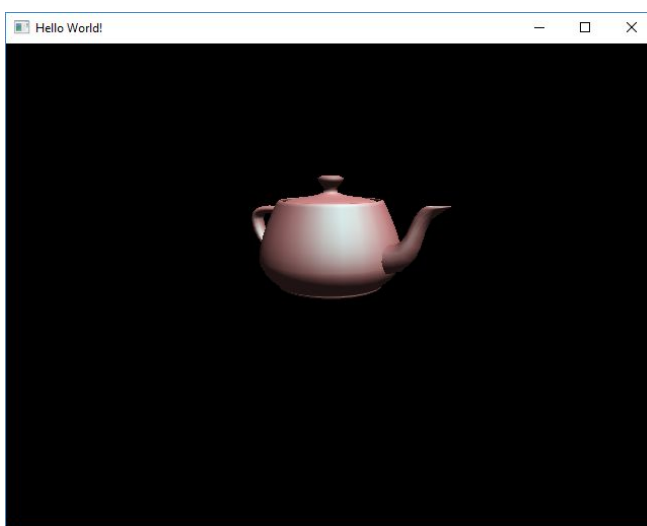
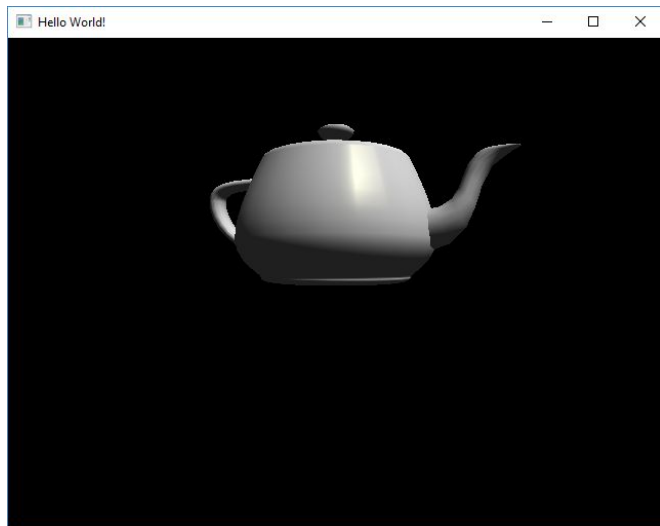


Task 5 – přidání materiálů

Barvy objektů ve scéně by měl kromě světla ovlivňovat i sám objekt; k tomu použijeme tzv. materiály. Přidání materiálů provedeme zase přes uniformní proměnné, pro analogické složky ke složkám světla, s tím, že materiál bude mít navíc složku „shininess“, která se počítá pro sílu spekulárního odlesku. Vyzkoušejte si libovolné barvy složek materiálu a různé hodnoty pro shininess. Shininess nechte v intervalu $[1; \infty]$. Stejně tak si pohrajte s hodnotami složek světla. Zkuste si odpovědět na následující dvě otázky:

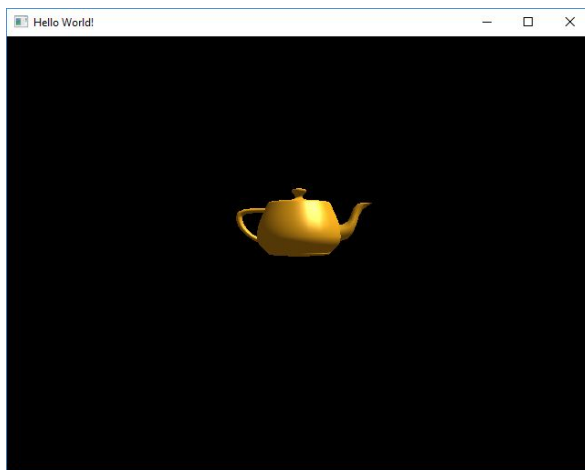
Proč čistě červená spekulární barva světla a čistě zelená spekulární barva materiálu nevydává žádný odlesk?

Jak úplně „vypnout“ spekulární odlesky materiálu?



Task 6 – „reálné“ materiály

Místo nereálného materiálu s náhodnými čísly zkusme použít nějaký material z následující tabulky, která obsahuje RGBA hodnoty pro ambientní, difúzní a spekulární složku materiálu, včetně shininess parametru. Vyberte si jeden material (např. váš oblíbený kov nebo slitinu), a aplikujte jej na vaši konvičku. Prozkoumejte pomocí kamery, jak se konvička leskne z různých úhlů.

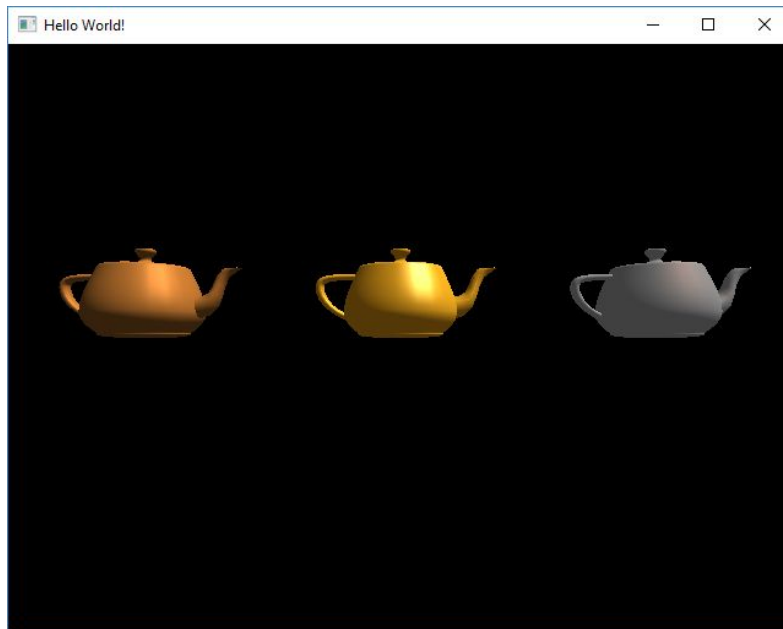


Material	AMBIENT	DIFFUSE	SPECULAR	SHININESS
Brass	0.329412 0.223529 0.027451 1.0	0.780392 0.568627 0.113725 1.0	0.992157 0.941176 0.807843 1.0	27.8974
Bronze	0.2125 0.1275 0.054 1.0	0.714 0.4284 0.18144 1.0	0.393548 0.271906 0.166721 1.0	25.6
Polished Bronze	0.25 0.148 0.06475 1.0	0.4 0.2368 0.1036 1.0	0.774597 0.458561 0.200621 1.0	76.8
Chrome	0.25 0.25 0.25 1.0	0.4 0.4 0.4 1.0	0.774597 0.774597 0.774597 1.0	76.8
Copper	0.19125 0.0735 0.0225 1.0	0.7038 0.27048 0.0828 1.0	0.256777 0.137622 0.086014 1.0	12.8
Polished Copper	0.2295 0.08825 0.0275 1.0	0.5508 0.2118 0.066 1.0	0.580594 0.223257 0.0695701 1.0	51.2
Gold	0.24725 0.1995 0.0745 1.0	0.75164 0.60648 0.22648 1.0	0.628281 0.555802 0.366065 1.0	51
Polished Gold	0.24725 0.2245 0.0645 1.0	0.34615 0.3143 0.0903 1.0	0.797357 0.723991 0.208006 1.0	83.2
Pewter	0.105882 0.058824 0.113725 1.0	0.427451 0.470588 0.541176 1.0	0.333333 0.333333 0.521569 1.0	9.84615

Task 7 – více konviček, různé materiály

Přidáme 2 další konvičky, a vyzkoušíme si další 2 materiály z tabulky výše. Konvičky umístěte na souřadnice $(-5, 0, 0)$ a $(5, 0, 0)$ pomocí transformací model matice.

Můžete si vybrat libovolné materiály, a prozkoumejte je opět pomocí kamery.



Task 8 – bodové světlo

Doteď jsme používali pouze směrové světlo. Další druh světla je světlo bodové – toto světlo svítí z jednoho bodu do okolí, lze si jej představit např. jako žárovku. Pro přidání podpory tohoto druhu světla musíme v našem fragment shaderu upravit výpočet směru světla, kterým svítí. Pro rozlišení mezi směrovým a bodovým světlem použijeme čtvrtou komponentu (w) vektoru pozice světla – pokud bude nulová, jedná se o světlo směrové, pokud nenulová, jedná se o světlo bodové.

Zkuste si opět různé pozice zdroje bodového světla a prohlédněte scénu s kamerou.

