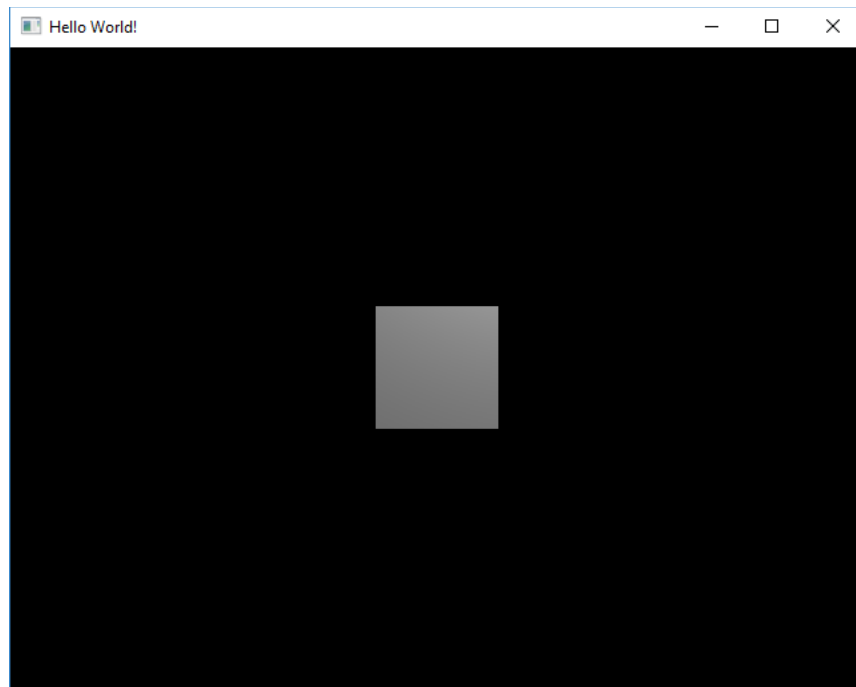


PV112 Java - 4. cvičení

Stáhněte si cv4.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 kostkou. Dnes budeme pracovat v souborech Cv4.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().

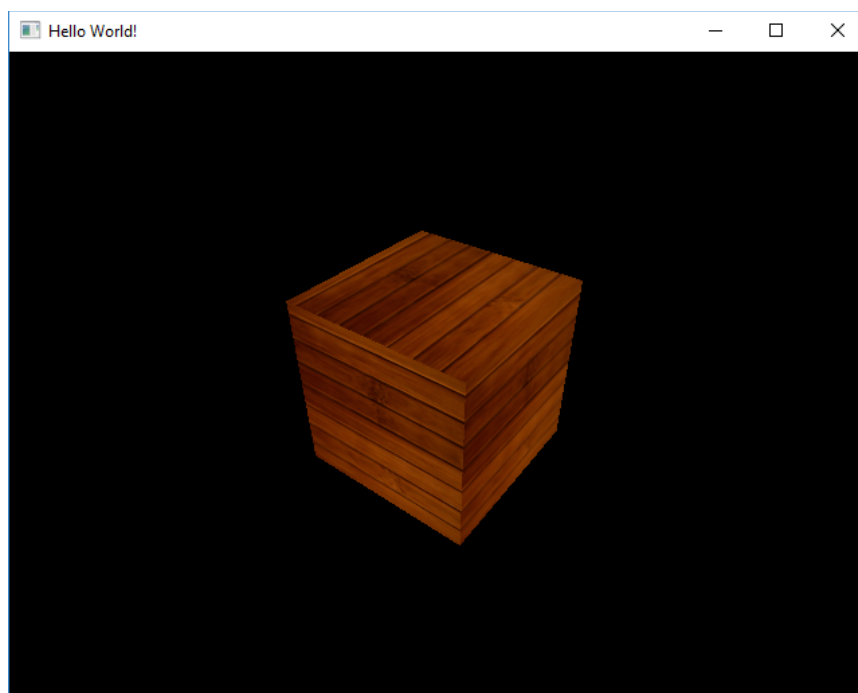
V tomto cvičení navíc máte možnost ovládat pozici kamery; při držení levého tlačítka myši se můžete otáčet kolem středu, a při držení pravého tlačítka můžete přibližovat/oddalovat scénu.



Task 1 – Vytvoření textury a otexturování objektu

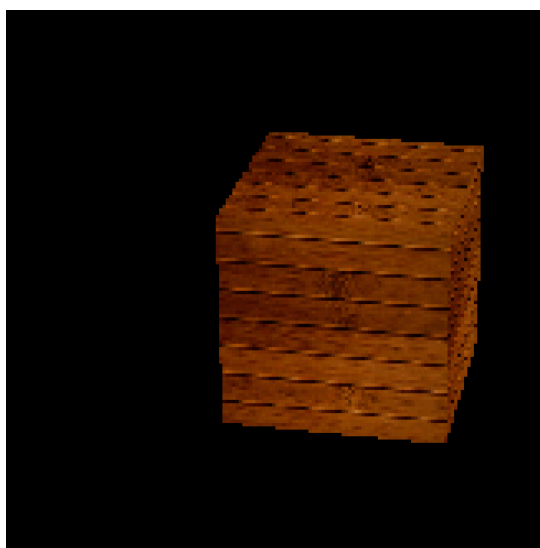
Jako první přidáme texturu bez osvětlení. Použijeme texturu uloženou v souboru *resources/textures/wood.jpg*, která obsahuje dřevo.

V tomto kroku nebudeme používat osvětlení, pouze nanese barvu textury na kostku na základě texturovacích souřadnic kostky.



Task 2 – Lineární filtrování

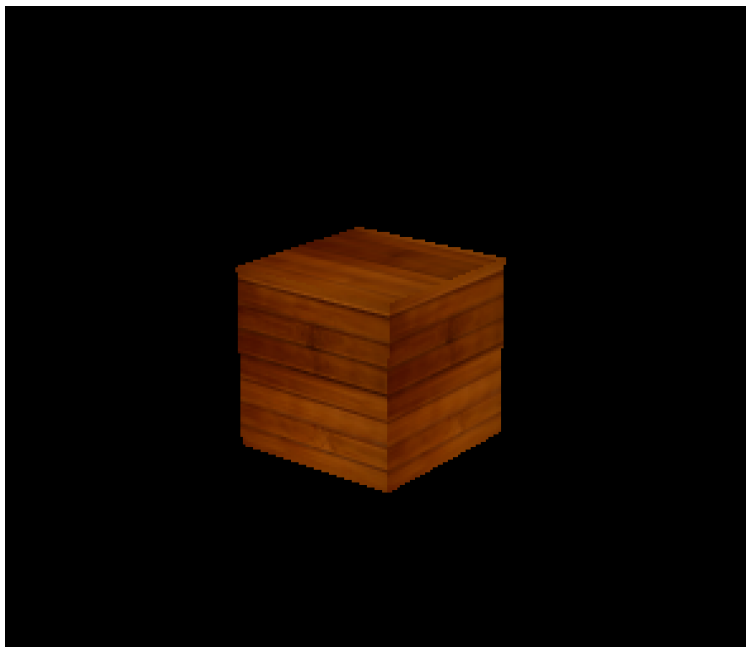
V rámci prvního tasku jsme nastavili filtrování textur na `GL_NEAREST`. V tomto kroku změníme oba filtry na `GL_LINEAR`.



Task 3 – Mipmapy

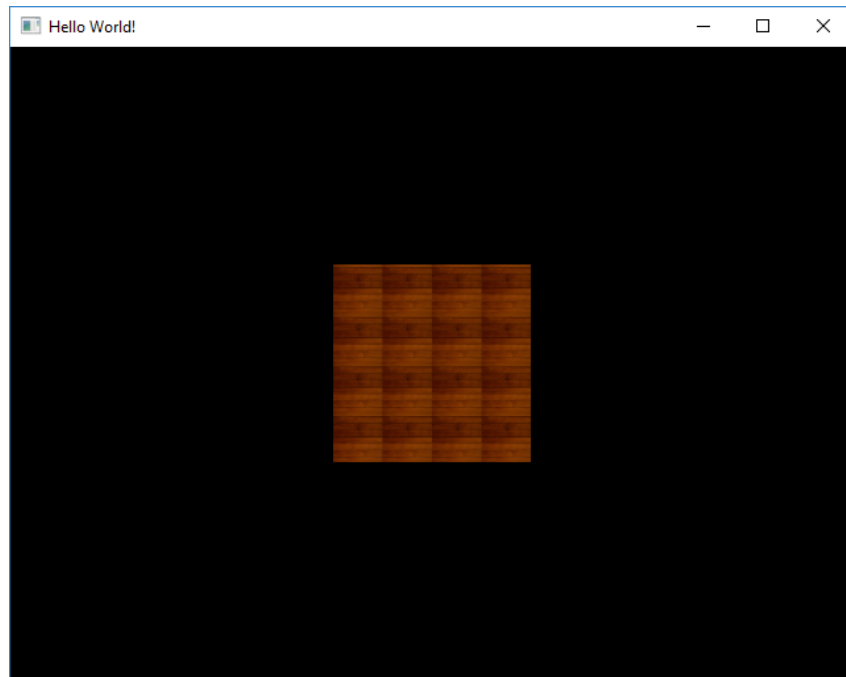
V tomto kroku změníme filtr `GL_TEXTURE_MIN_FILTER` na `GL_LINEAR_MIPMAP_LINEAR`, a necháme vygenerovat mipmapy pro texture (pomocí funkce `glGenerateMipmap`), což nám umožní použít mipmapping.

Zvětšovací filtr ponecháme na původní hodnotě.



Task 4 – Transformace texturovacích souřadnic

Texturovací souřadnice na naší kostce byly v mezích $[0.0, 1.0]$. Přetransformujte je do mezí $[-2.0, 2.0]$ abyste dostali takto otexturovanou kostku.

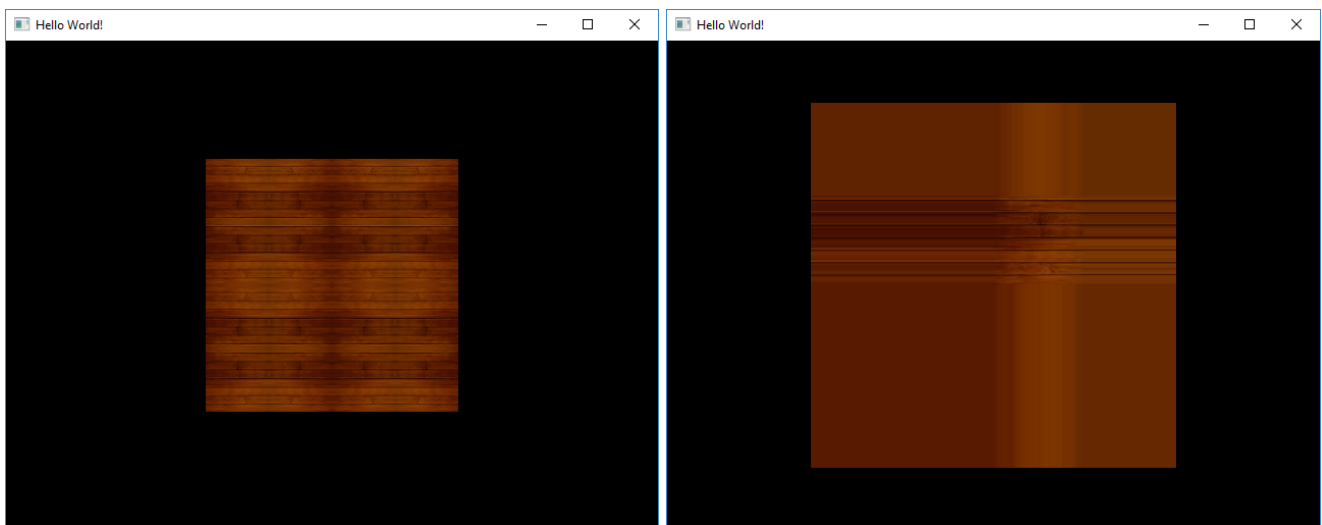


Task 5 – Opakování textury

Stejně jako existují filtrovací módy, existují i módy pro případy, že nám texturovací souřadnice přesáhnou interval $[0.0, 1.0]$. Pro to existuje několik módů (je nutné je nastavovat v obou osách, tzn. `GL_TEXTURE_WRAP_S` i `GL_TEXTURE_WRAP_T`):

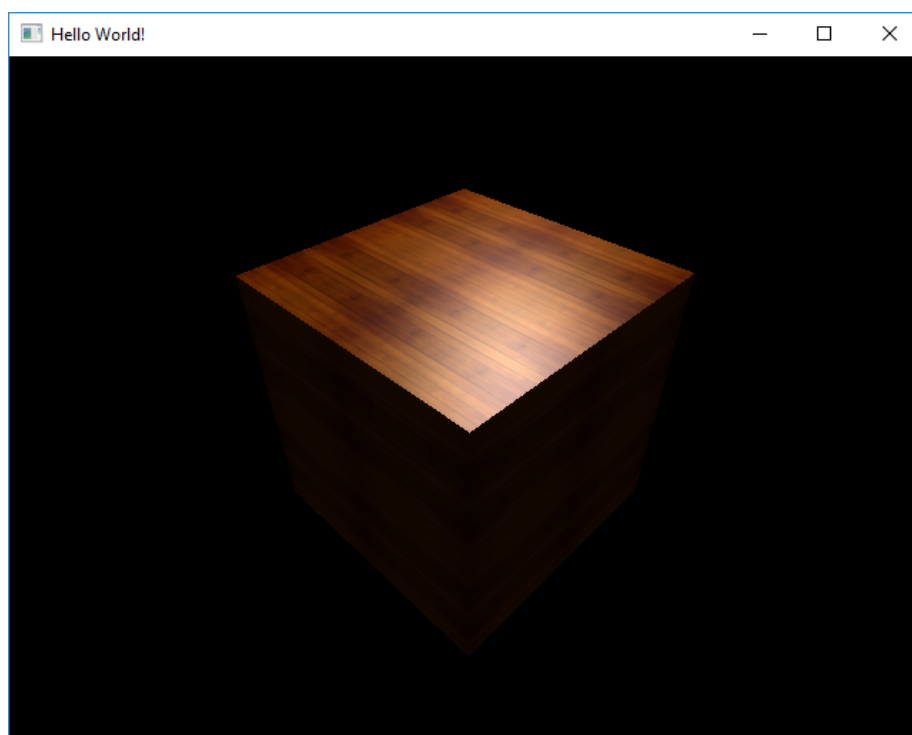
- `GL_REPEAT`
- `GL_MIRRORED_REPEAT`
- `GL_CLAMP_TO_EDGE`
- `GL_CLAMP_TO_BORDER`

Vyzkoušejte si je postupně všechny, včetně různých kombinací pro různé osy.



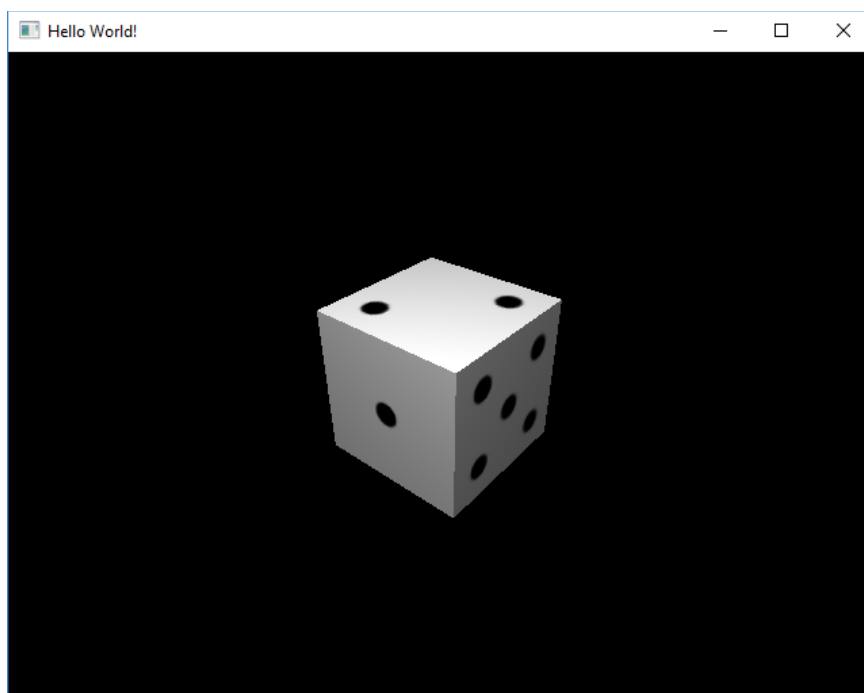
Task 6 – Kombinace textury a osvětlení

Nyní je na čase zase přidat osvětlení. Můžete k tomu použít funkci `phong()`, připravenou v fragment shaderu. Předějte jako ambientní a difúzní barvu materiálu barvu textury.



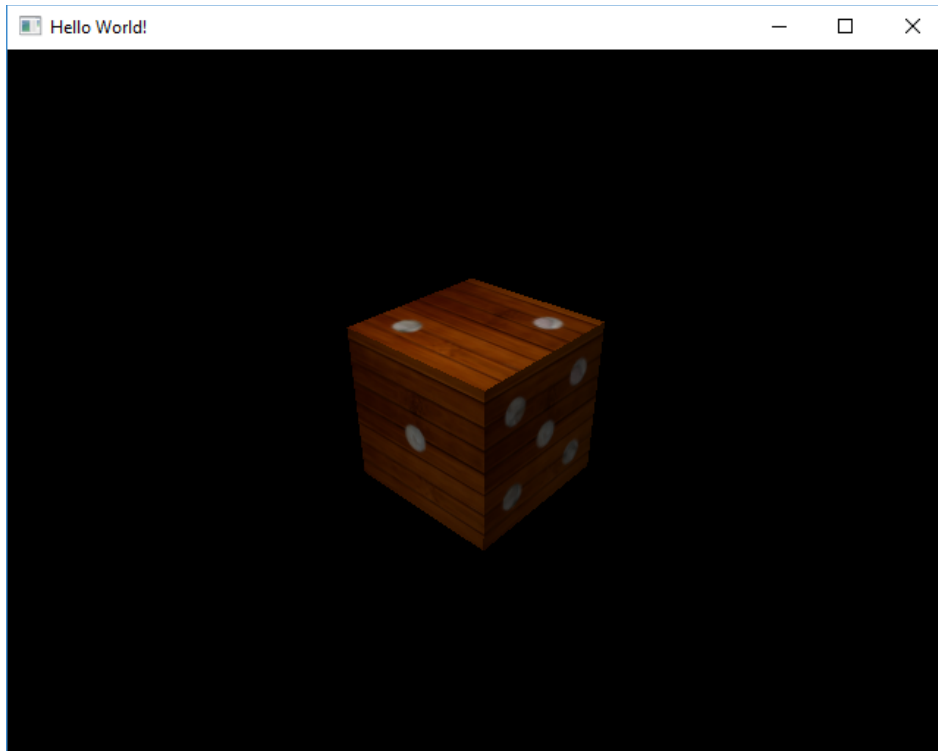
Task 7 – Hrací kostka

Jako další úkol přidáme texturu kostky – textury pro hrací kostku jsou ve více souborech, a načteme je postupně.



Task 8 – Multitexturing

Na základě barev textur kostky nanese barvy kamenů a dřeva. Musíme si tedy načíst kamennou texturu a následně na základě barvy kostky aplikovat texturu dřeva nebo texturu kamení.

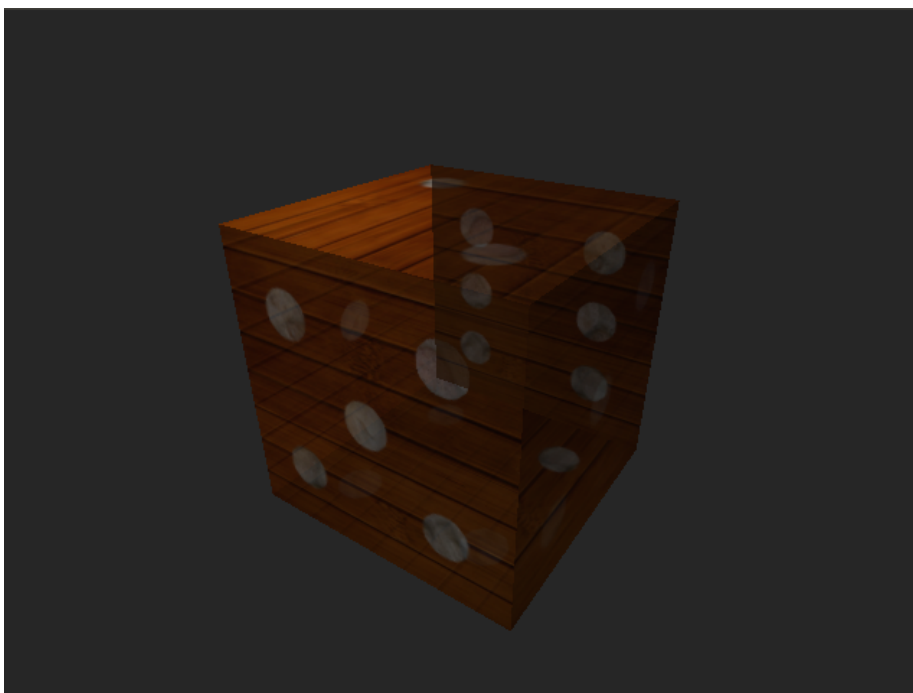


Task 9 – Blending (bonus)

Zapnutím míchání (`GL_BLEND`) a nastavením neprůhlednosti (alpha) fragmentů na menší číslo než `1.0` vykreslíme kostku průhledně. Nastavíme funkci míchání tak, aby se míchání provádělo podle alpha hodnoty nově přichozícího (do výsledného obrazu) fragmentu.

Aby byly vykresleny všechny plošky, vypněte hloubkový test (`GL_DEPTH_TEST`).

Níže zachycený artefakt byl viditelný při alpha 0.7.



Task 10 – Jednoduché seřazení plošek (bonus)

Seřazení plošek jednoduchých (konvexních) objektů je možné dosáhnout vykreslením nejprve zadních a poté předních plošek. Poté se budou fragmenty míchat ve správném pořadí.