# Assignment 11: Direct Light Models

In this assignment you have to complete the Fragment Shader contained in file `shaders/BlinnShader1.frag` to `shaders/BlinnShader3.frag`, to implement *Direct*, *Point* and *Spot* light models. The file must be compiled into `shaders/BlinnFrag1.spv` to `shaders/BlinnFrag3.spv`, as previously seen. Each fragment shader requires a different light model, and uses different parameters, all specified in the comments.

The following GLSL standard procedures can be helpful in solving this exercise:

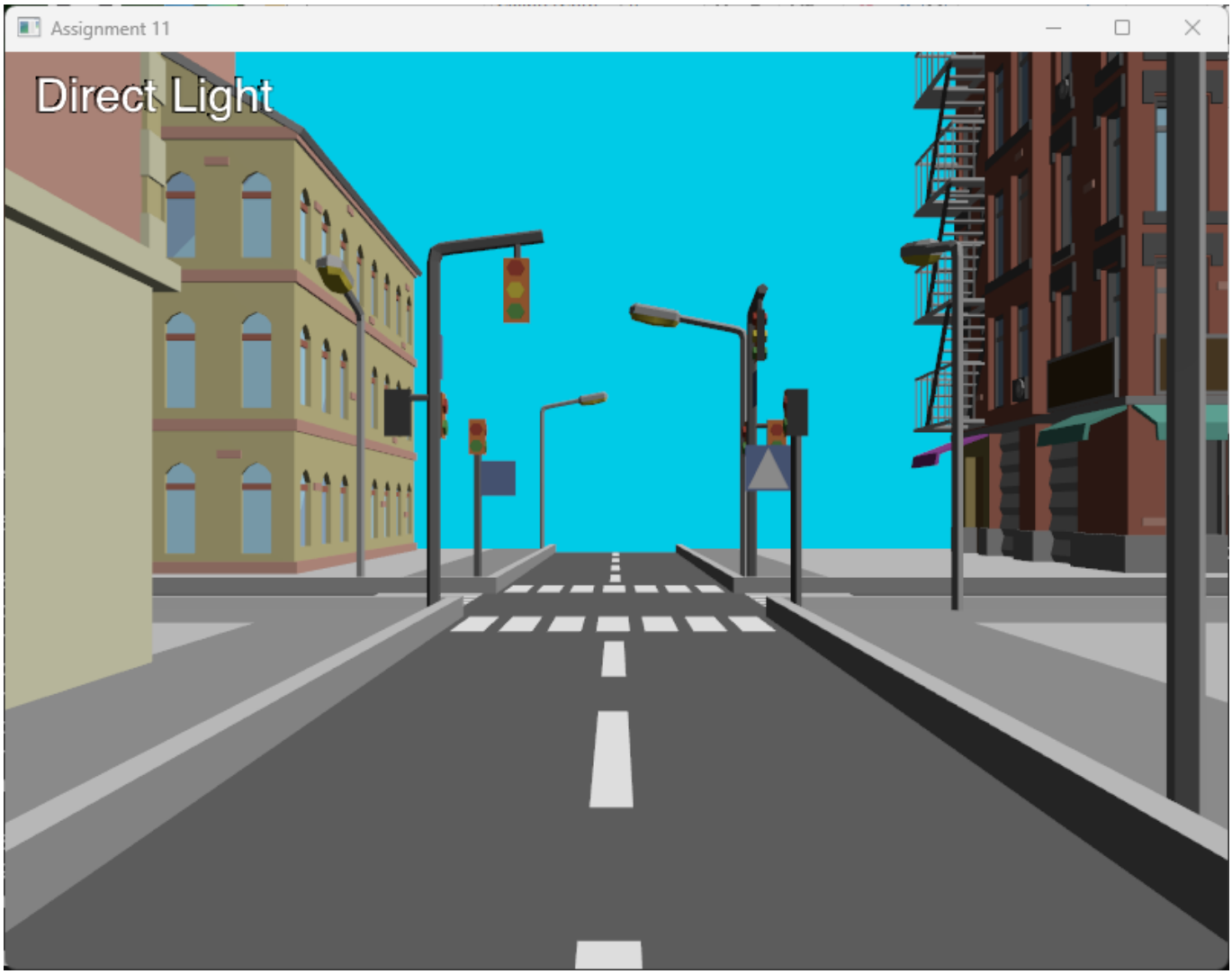
*`normalize()`*
*`pow()`*
*`dot()`*
*`length()`*
*`clamp()`*
*`max()`*
*`min()`*

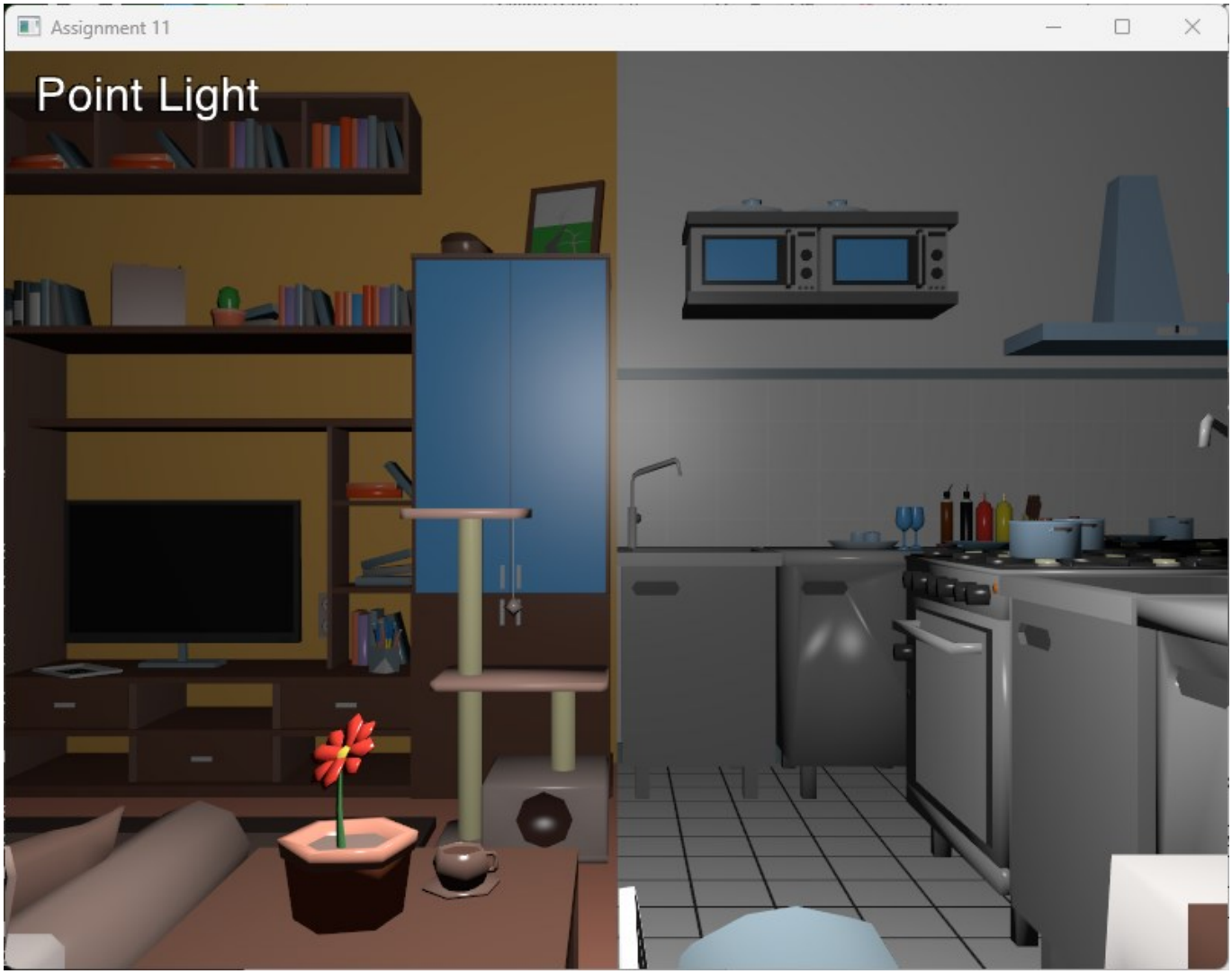
If you need help about GLSL, you can refer to the following tutorial:

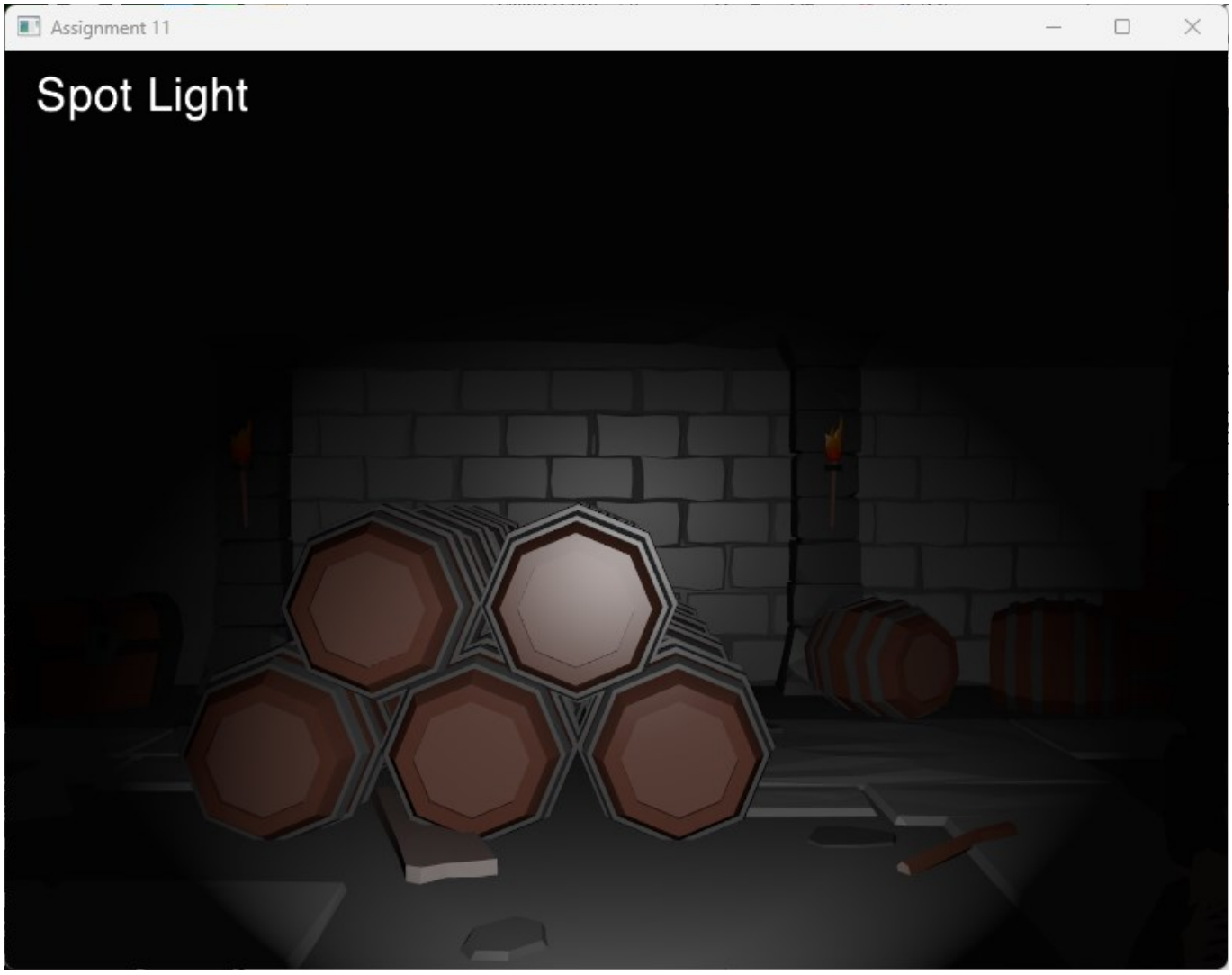
https://cgvr.cs.uni-bremen.de/teaching/cg2_07/literatur/glsl_tutorial/index.html

Starting from the section “Data Types and Variables”, at around 1/3 of the page. Please ignore what is presented before since it refers to a very old version of OpenGL which uses concepts that are now deprecated and not valid for Vulkan.

The expect results should be similar to the following:




Assignment 11
Point Light




Users can move the view using the same keys as in Assingnment 0.

| ESC – quit the application | | SPACE BAR – move to the next light | | | | |
|---|---|---|---|---|---|---|
| **Q**: roll left | **W**: forward | **E**: roll ri | **R**: up | | ↑: look up | |
| **A**: left | **S**: backward | **D**: right | **F**: down | ←: look left | ↓: look down | →: look right |