

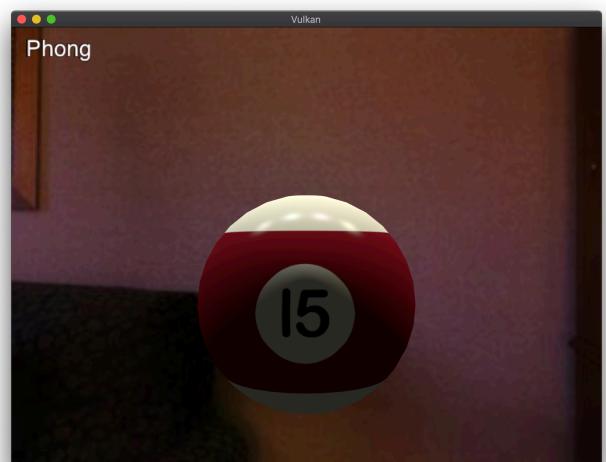
## Assignment 16: BRDF Models

In this assignment you have to complete the Fragment Shader contained in file `shaders/BRDFShader.frag`, to implement the diffuse and specular component of several standard BRDFs. The file must be compiled into `shaders/BRDFFrag.spv` as seen in *Assignment 10*. For each light model, function ending with `..._BRDF()` should return a `vec3` vector containing the RGB color computed by the model. Each function receives the following parameters:

- `vec3 L`: the direction of the light (unitary vector)
- `vec3 N`: the direction of the normal vector (unitary vector)
- `vec3 V`: the direction of the view vector (unitary vector)
- `vec3 C`: the main color required by the model

If needed, BRDF receive in input also other parameters relevant to the particular model. Refer to the comment in the code for them. The following GLSL standard procedures can be helpful in solving this exercise: `dot()` `pow()` `reflect()` `max()` `min()` `acos()` `sin()` `tan()`

The expect results should be similar to the following:



Users can move the view using the same keys as in Assignment 0, and remove the texture pressing the **T** key.

ESC – quit the application		SPACE BAR – move to the next light   <b>T</b> : toggle the texture				
<b>Q</b> : roll left	<b>W</b> : forward	<b>E</b> : roll right	<b>R</b> : up		<b>↑</b> : look up	
<b>A</b> : left	<b>S</b> : backward	<b>D</b> : right	<b>F</b> : down	<b>←</b> : look left	<b>↓</b> : look down	<b>→</b> : look right