

1 Advanced Graphics (PB)

In this question, you are asked to write OpenGL GLSL code. Correct GLSL syntax is not required and is not part of the marking scheme. If you are unsure of a method name or syntax, use a reasonable approximation. Full marks will be awarded if you make your intention clear with the use of suitable comments.

An OpenGL program is using the following GLSL shader fragment program:

```
#version 330

const vec3 BLACK = vec3(0, 0, 0);
const vec3 WHITE = vec3(1, 1, 1);
const vec3 BLUE = vec3(0, 0, 1);

uniform vec3 eyePosition;      // Camera position
uniform vec3 lightPosition;   // Light source position

in vec3 position;             // Fragment position
in vec3 normal;               // Fragment normal

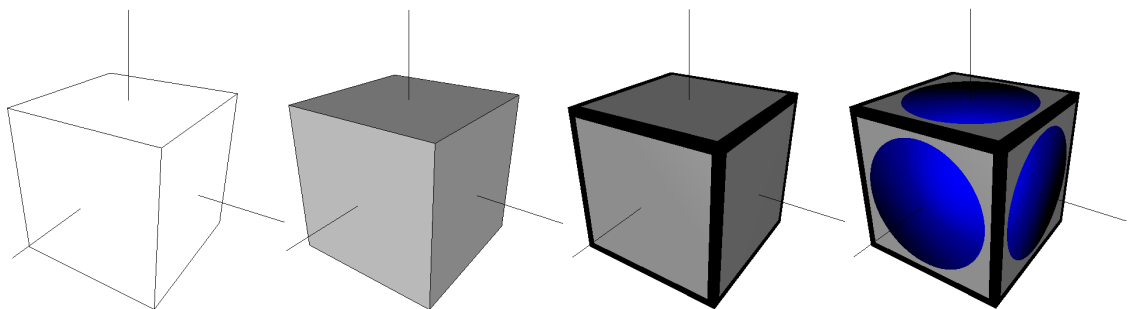
out vec4 fragmentColor;       // GLSL fragment output color

void main() {
    vec3 color = WHITE;

    // [YOUR CODE HERE]

    fragmentColor = vec4(color, 1);
}
```

The shader is being used to render a cube. The cube is centered at the origin, axis-aligned, and has edge length 2; its corners are from $(-1, -1, -1)$ to $(1, 1, 1)$.



- (a) **Diffuse shading:** Fill in the body of the shader fragment program to implement diffuse shading. [4 marks]
- (b) **Specular shading:** Extend your solution to implement specular shading as well as diffuse shading. Shade the surface with a 75% diffuse, 25% specular balance between the two methods. [4 marks]
- (c) **Procedural texturing:** Extend your solution to render thick black edges on all faces of the cube (see illustration). The edges should be approximately 0.1 units thick. [6 marks]
- (d) **Procedural texture effects:** Extend your solution to create the illusion of a dark blue sphere of radius 1.3 units emerging through the sides of the cube. [6 marks]