



**Computação Gráfica e Interfaces
(LEI) _2025/2026 _ 1ºSemestre**

**Relatório
3º Projeto**

Turno P3

Horário: Quinta-feira - 14H00 às 16H00

Realizado por:

**67912 João Rodrigues
68725 Tomás Silva**

Variáveis de entrada e saída dos Shadres

- **Shader_phong.vert:**

-Entrada (in):

vec4 a_position: Posição do vértice em coordenadas do modelo;

vec3 a_normal: Vetor normal do vértice, usado para a iluminação.

-Uniforms:

mat4 u_projection: Matriz de projeção (perspetiva);

mat4 u_model_view: Matrix ModelView (posição/orientação do objeto com a transformação da câmara);

mat4 u_normals: Matriz usada para transformar as normais corretamente no espaço da câmara.

-Saídas (out):

vec3 v_normal: Normal transformada para o espaço da câmara;

vec3 v_position: Posição do vértice já transformada para o espaço da câmara.

- **Shader_phong.frag:**

-Entrada (in):

Vec3 v_normal: Normal interpolada para o fragmento atual;

vec3 v_position: Posição do fragmento no espaço da câmara.

-Uniforms:

int u_blinn: Define se o specular é calculado com Blinn's Phong (1) ou Phong (0);

int u_n_lights: Número efetivo de luzes usadas na cena;

LightInfo u_lights[MAX_LIGHTS]: Vetor das luzes presentes na cena;

Estrutura da luz, LightInfo:

vec3 ambient: Componente ambiente da luz;

vec3 diffuse: Componente difusa da luz;

vec3 specular: Componente especular da luz;

vec4 position: Posição/Direção da luz;

vec3 axis: Direção do cone para spotlight;

float aperture: Ângulo de abertura do spotlight;

float cutoff: Controla a intensidade do spotlitght;

int onOff: Define se a luz está ativa;

int type: 0=point, 1=directional, 2=spotlight.

MaterialInfo u_material: Material do objeto a ser desenhado;

Estrutura do material, MaterialInfo:

vec3 Ka: Cor ambiente do material;

vec3 Kd: Cor difusa do material;

vec3 Ks: Cor especular do material;

float shininess: Esponente specular (brilho);

-Saída (out):

vec4 color: Cor final calculada para o fragmento.

- **Shader_gouraud.vert:**

-Entrada (in):

 Vec3 a_position: Posição do vértice;

 vec3 a_normal: Normal do vértice.

-Uniforms:

 int u_blinn: Define se o specular é calculado com Blinn's Gouraud (1) ou Gouraud (0);

 int u_n_lights: Número efetivo de luzes usadas na cena;

 LightInfo u_lights[MAX_LIGHTS]: Vetor das luzes presentes na cena;

 Estrutura da luz, LightInfo:

 vec3 ambient: Componente ambiente da luz;

 vec3 diffuse: Componente difusa da luz;

 vec3 specular: Componente especular da luz;

 vec4 position: Posição/Direção da luz;

 vec3 axis: Direção do cone para spotlight;

 float aperture: Ângulo de abertura do spotlight;

 float cutoff: Controla a intensidade do spotlitght;

 int onOff: Define se a luz está ativa;

 int type: 0=point, 1=directional, 2=spotlight.

 MaterialInfo u_material: Material do objeto a ser desenhado;

 Estrutura do material, MaterialInfo:

 vec3 Ka: Cor ambiente do material;

 vec3 Kd: Cor difusa do material;

 vec3 Ks: Cor especular do material;

 float shininess: Esponente specular (brilho);

-Saída (out):

 vec4 v_color: Cor final calculada no vértice e interpolada pelos fragmentos.

- **Shader_gouraud.frag:**

-Entrada (in):

 vec3 v_color: Cor já calculada no vertex shader (shader_gouraud.vert);

-Saídas (out):

 vec4 color: Cor final do fragmento (v_color recebida);

Funcionalidades Extra

Começamos por implementar os desafios propostos, isto é, a possibilidade escolher entre coordenadas da camera ou coordenadas do mundo para a disposição das luzes:



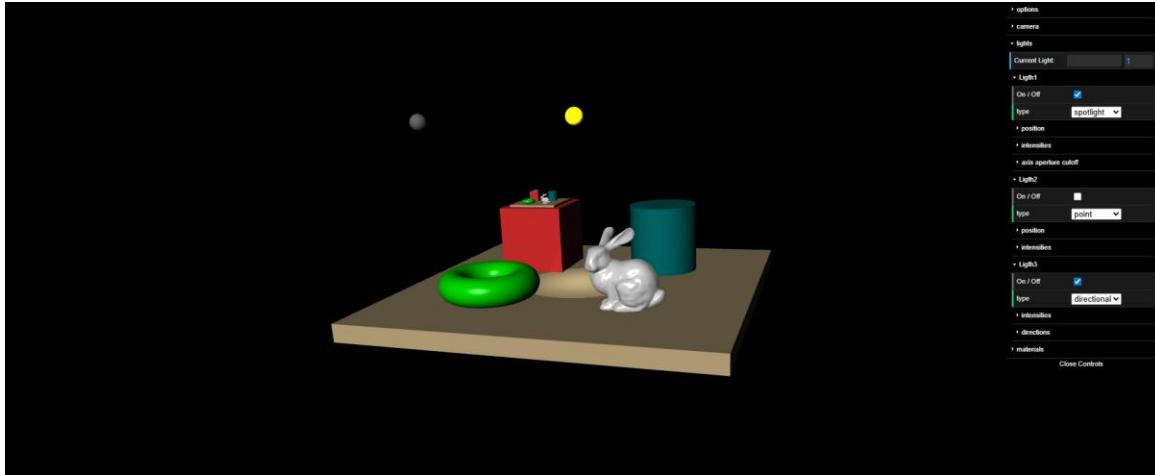
Bem como a implementação dos comandos para w, a, s, d & r para voar pelo cenário e resetar os parametros da câmera (e também reseta os parametros da aba options).

Para além disso adicionamos uma opção para a renderização dos shaders poder ser no formato blinn (ou normal) dado nos slides (H / R):

A screenshot of a software interface showing a checkbox labeled 'Blinn' which is checked.

```
if(u_blinn == 1)
    specularFactor = pow(x: max(x: dot(x: N,y: H), y: 0.0), y: u_material.shininess);
else
    specularFactor = pow(x: max(x: dot(x: N,y: R), y: 0.0), y: u_material.shininess);
```

Contamos ainda com o extra solicitado das esferas desenhadas na fonte das varias luzes (quando em modo point ou spotlight), que apresentam uma cor amarela quando a luz está ligada e acinzentada quando a mesma está apagada:



Por fim, adicionamos uma poção para selecionar uma das luzes, e a luz selecionada é movével através das setas (para frente, trás, esquerda e direita) e pageUp e pageDown (para cima e baixo), para que utilizador não tenha e que escolher coordenada a coordenada manualmente (o que não deixa de ser possível):

