

EC212 - Computação Gráfica

Danilo Peixoto Ferreira
(danilopeixoto@gec.inatel.br)
Instituto Nacional de Telecomunicações
03 de outubro de 2019

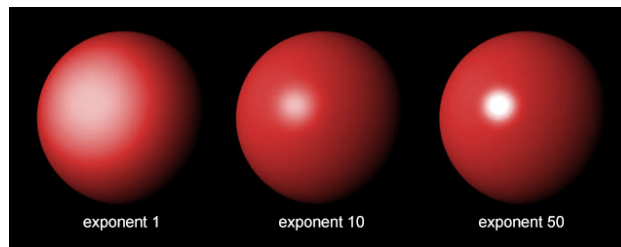
Orientações

Envie o arquivo `blinn_phong.frag` (**Questão 1**) para o endereço de e-mail danilopeixoto@gec.inatel.br com assunto [EC212] Teste 2 e corpo contendo NOME e MATRÍCULA. Entregue a resolução da **Questão 2 e 3** em um documento assinado com NOME e MATRÍCULA.

Questões

1. (40 pontos) Implemente o modelo de reflexão especular de Blinn-Phong no *shader* de fragmento.
 - a) Exporte o parâmetro *exponent* do material especular (`MATERIAL.exponent`) para o *shader* de fragmento (`material.exponent`). Utilize a função `glUniform1f`.
 - b) Implemente o modelo de Blinn-Phong no *shader* de fragmento:

O modelo de reflexão de Blinn-Phong tem como objetivo representar o brilho de superfícies especulares através do parâmetro *exponent*. Exemplo:

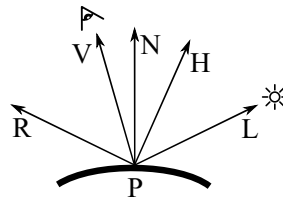


O modelo é calculado pelas equações abaixo:

$$H = \frac{L + V}{|L + V|}$$

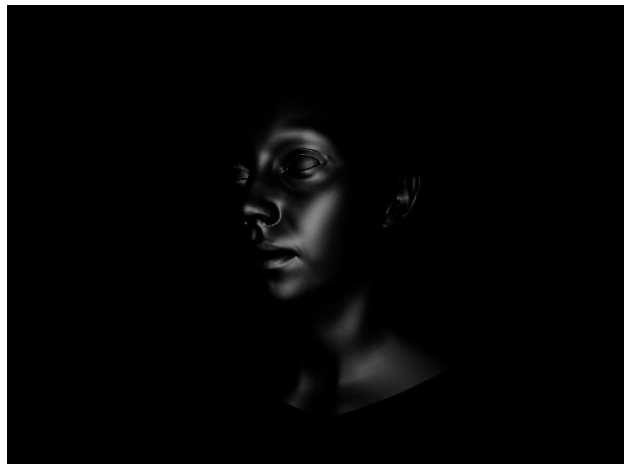
$$L_o = \frac{L_i}{r^2} \cdot K_s \cdot \max(N \cdot H, 0)^n$$

- L Vetor unitário apontando para luz
- V Vetor unitário apontando para câmera
- N Vetor unitário normal à superfície
- P Ponto da superfície
- H Vetor unitário metade representando a normal de micro imperfeições da superfície
- L_o Radiância total (cor do *shader* de fragmento)
- L_i Radiância da luz (cor da luz)
- r Distância entre ponto da superfície e luz
- K_s Cor do material especular
- n Parâmetro *exponent* do material especular



Todos os vetores são unitários.

c) Verifique o sombreamento resultante:



2. (40 pontos) Calcule as coordenadas 2D do ponto da curva Bézier quadrática para $t = 0.25$ utilizando o algoritmo De Casteljau:

```
func lerp(point a, point b, float t) -> point:
    return a * (1 - t) + b * t

func casteljau(uint degree, float t, uint index, point[] points) -> point:
    if degree == 0:
        return points[index]

    point p0 = casteljau(degree - 1, t, index, points)
    point p1 = casteljau(degree - 1, t, index + 1, points)

    return lerp(p0, p1, t)
```

Utilização:

```
point p = casteljau(2, 0.25, 0, points)
```

Lista de pontos de controle:

- P_0 (0 0)
- P_1 (100 100)
- P_2 (200 0)

3. (20 pontos) O que é minificação e magnificação no contexto de texturas? Qual método pode ser aplicado para resolver os artefatos ocasionados por estes efeitos? Explique-o.