



Modelos de Iluminação

Disciplina: Computação Gráfica (BCC35F)

Curso: Ciência da Computação

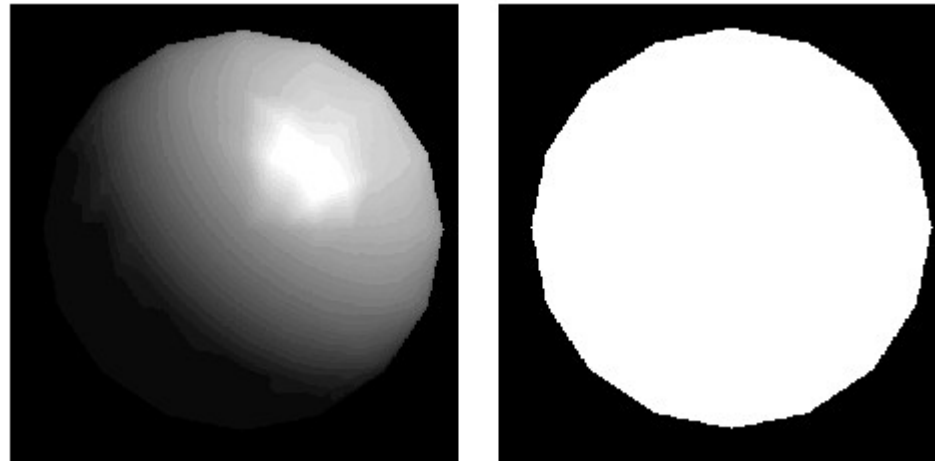
Prof. Walter T. Nakamura
waltertakashi@utfpr.edu.br

Campo Mourão - PR

Baseados nos materiais elaborados pelas professoras Aretha Alencar (UTFPR) e Rosane Minghim (USP)

- 1) Introdução
- 2) Fontes de Luz
- 3) Efeitos de Luz em Superfícies
- 4) Modelos Básicos de Iluminação
 - Programação OpenGL
- 5) Apêndice

- Imagens **realísticas** são criadas usando **projeções perspectivas**, aplicando-se efeitos de iluminação natural às superfícies visíveis por meio de um **modelo de iluminação** (*shading model*)
- Modelos de iluminação são usados para calcular a cor de uma posição iluminada na superfície do objeto
- De forma geral, modelar os **efeitos da luz** sobre um objeto é um processo **complexo**, que envolve princípios físicos



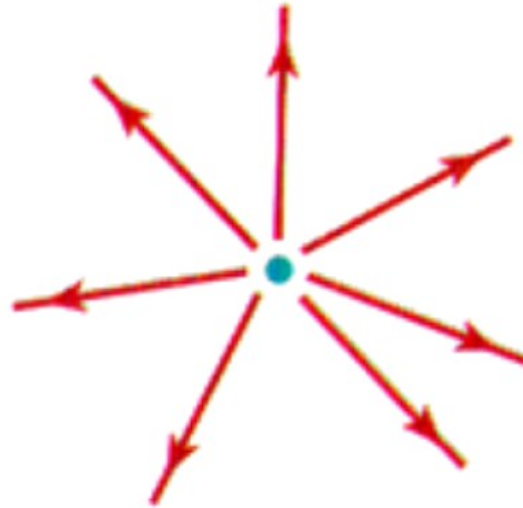
- Os **modelos físicos** envolvem vários fatores, como **propriedades** dos materiais, posição do objeto em relação a luz e outros objetos, e características das fontes de luz:
 - Objetos podem ser opacos ou mais ou menos transparentes, podem ser finos ou grosseiros
 - Fontes de luz podem ter vários formatos, cores e posições
- Os **modelos de iluminação** em computação gráfica são na maioria das vezes **aproximações** das leis físicas que descrevem efeitos de luz sobre superfícies

Fontes de Luz

- Qualquer objeto que emite energia brilhante é uma **fonte de luz** que contribui para os efeitos de luz dos outros objetos na cena
- **Fontes de luz** podem ter diferentes formas e características (posição, cor, direção de emissão, formato, etc.)
- Em aplicações gráficas de **tempo real**, um modelo simples de iluminação normalmente é aplicado por causa do **custo computacional**
 - Propriedades da emissão de luz são definidas usando **valores distintos** para cada componente de cor RGB, descrevendo suas intensidades

Fontes de Luz Pontuais

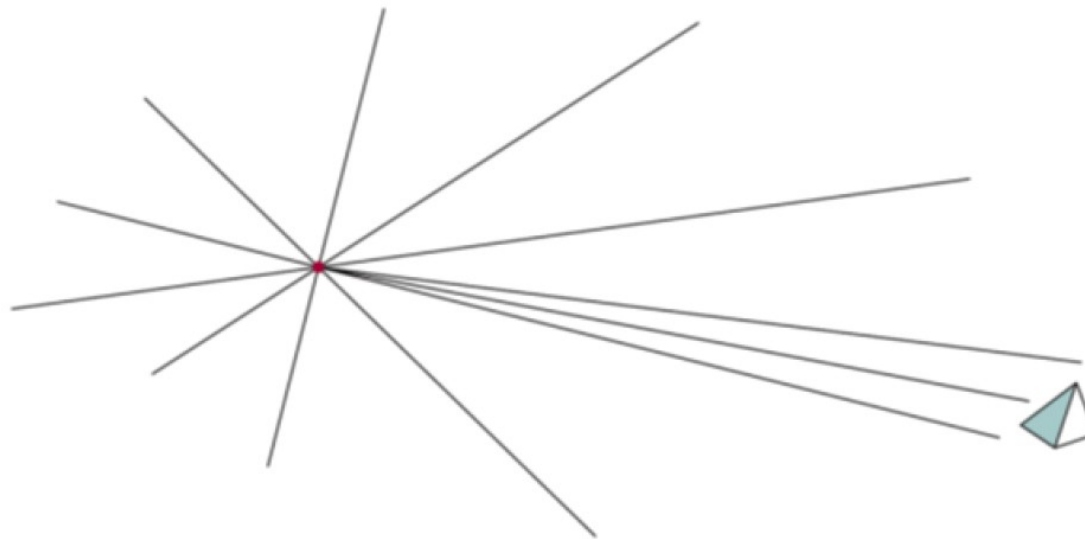
- O modelo mais simples de emissão de luz é **fonte de luz pontual** com uma única cor:
 - Definida por uma **posição** e a **cor** da luz emitida:



- Os raios de luz são gerados em **direções radiais divergentes** a partir do ponto de luz:
 - Indicado para aproximar efeitos de luz quando a **fonte de luz** é pequena em comparação com os objetos da cena.

Fontes de Luz Infinitamente Distantes

- Uma **fonte de luz grande** (ex: o Sol), que está bem longe da cena, pode ser aproximada como um ponto emissor bem distante dos objetos
 - A iluminação é provida em uma **única direção**:



- Uma fonte de luz distante é simulada definindo sua **cor** e uma **direção** da emissão dos raios, não é necessário especificar uma posição

Atenuação Radial da Intensidade

- A energia de radiação de uma fonte de luz a uma distância d_l da origem tem sua **amplitude atenuada** por um fator $1/d_l^2$
 - Uma superfície próxima da fonte recebe uma maior intensidade de luz
 - Para uma **iluminação realística** essa atenuação deve ser levada em consideração.
- Na prática uma atenuação de $1/d_l^2$ para fontes de luz pontuais não produz efeitos realísticos:
 - Tende a definir uma alta variação da intensidade para objetos que estão próximos da fonte de luz e pouca variação quando d_l é grande.

Atenuação Radial da Intensidade

- Para produzir efeitos mais realísticos com fontes pontuais usamos:

$$f_{radatten}(d_l) = \frac{1}{a_0 + a_1 d_l + a_2 d_l^2}$$

- Os valores de a_0 , a_1 e a_2 podem ser ajustados para se produzir efeitos de atenuação desejados
 - Valores grandes podem ser assinalados a a_0 quando d_l é um valor baixo para prevenir $f_{radatten}(d_l)$ de ficar muito grande

Atenuação Radial da Intensidade

- Esse cálculo de atenuação não pode ser empregado quando a fonte pontual estiver no “infinito” porque a distância à fonte é indeterminada.
- Função da Atenuação Radial da Intensidade:

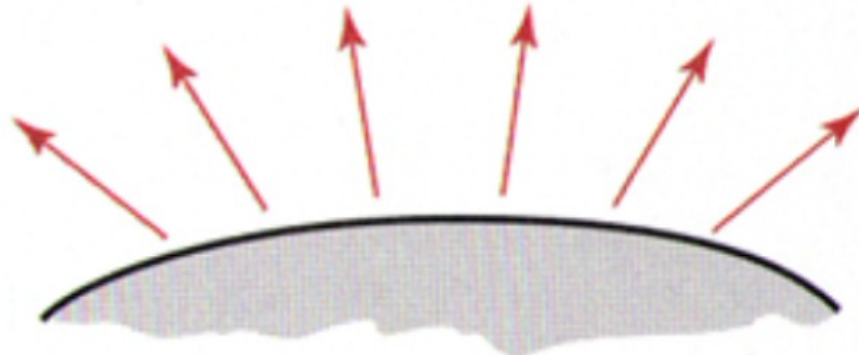
$$f_{radatten}(d_l) = \begin{cases} 1.0, & \text{se a fonte está no infinito} \\ \frac{1}{a_0 + a_1 d_l + a_2 d_l^2}, & \text{se a fonte é local} \end{cases}$$

Efeitos de Luz em Superfícies

- Um modelo de iluminação computa os **efeitos de luz** para uma superfície usando várias **propriedades óticas**
- Quando a superfície é **opaca**, parte da luz é **refletida** e parte absorvida:
 - A quantidade de luz refletida depende do tipo de material da superfície
- Para uma superfície **transparente**, alguma luz é também **transmitida** através da mesma

□ Reflexão Difusa

- Superfícies irregulares tendem a refletir a luz em todas as direções, parecendo igualmente brilhantes a partir de qualquer ponto de vista

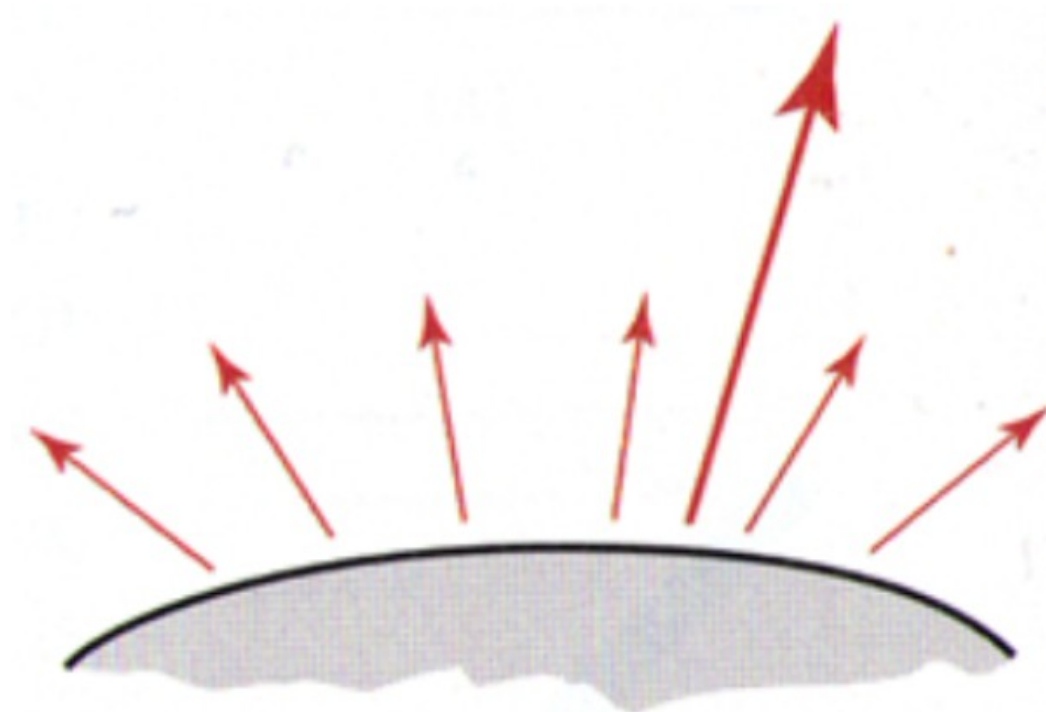


- O que chamamos de **cor de um objeto** é a cor da reflexão difusa quando o objeto é iluminado com uma luz branca:
 - Objetos azuis refletem a componente azul da luz branca, e absorvem as demais componentes
 - Um objeto azul sobre luz vermelha ficará preto, pois toda luz é absorvida

Efeitos de Luz em Superfícies

□ Reflexão Especular

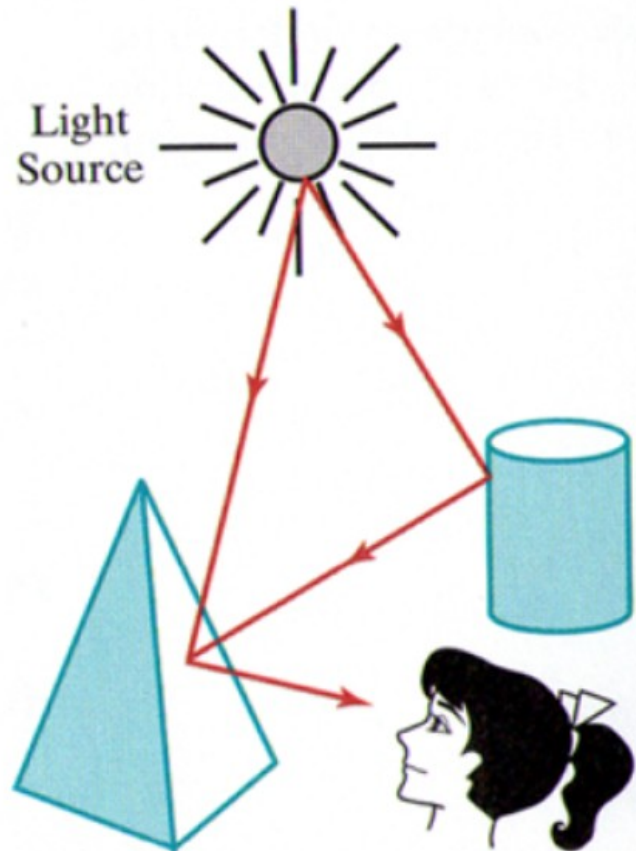
- Além da reflexão difusa, parte da luz refletida é concentrada em uma **região mais brilhante**
- O efeito de realce é mais pronunciado em superfícies de materiais brilhantes.



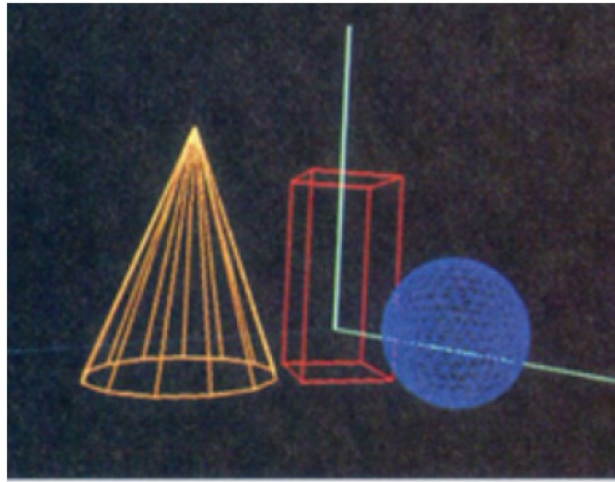
Efeitos de Luz em Superfícies

□ Luz de Fundo ou Ambiente

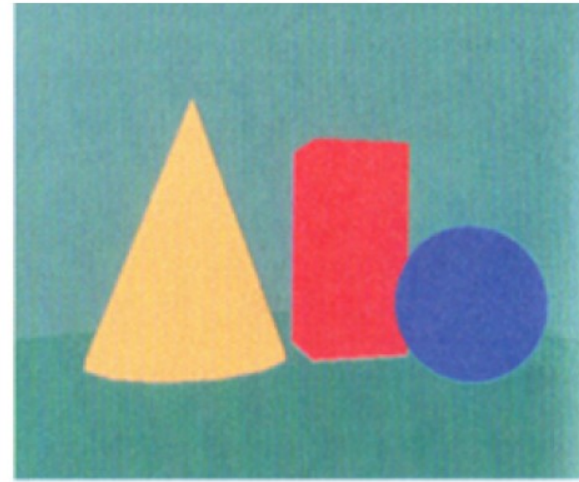
- Efeito de iluminação produzido pela luz refletida de várias superfícies na cena
 - A luz total refletida de uma superfície é a soma das contribuições da luz refletida pelas outras superfícies.



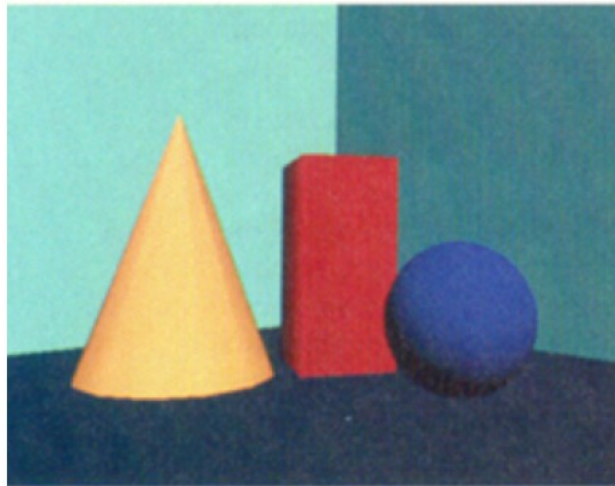
Exemplo



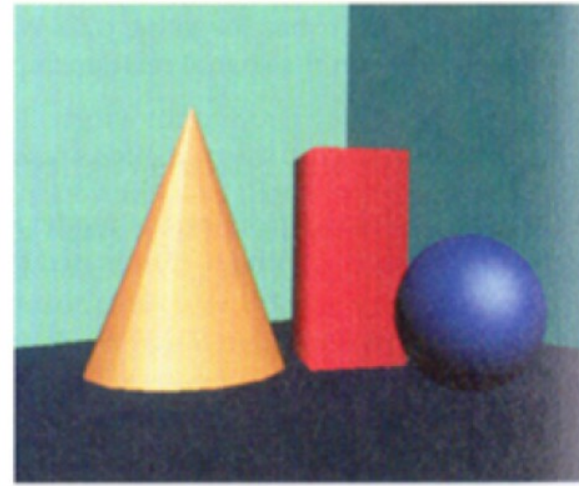
(a)



(b)



(c)



(d)

Figura: (a) cena wire-frame, (b) usando somente luz ambiente, (c) reflexão difusa com luz ambiente e uma única fonte de luz pontual, e (d) reflexão difusa e especular com luz ambiente e uma única fonte de luz pontual.

Efeitos de Luz em Superfícies

- **Modelos precisos** de iluminação computam toda a interação entre a radiação de **luz** e o **material** dos objetos
 - Esses efeitos podem ser aproximados usando modelos empíricos com bons resultados.
- Essa **interação** normalmente é **computacionalmente muito cara**, então aproximações para a iluminação ambiente são definidas.

- **Luz de fundo** pode ser incorporada definindo um nível de **brilho geral** para a cena
 - Produz luz ambiente uniforme para todos os objetos, aproximando as reflexões difusas de todas superfícies da cena
- O nível de luz ambiente em uma cena é definido por um parâmetro de intensidade I_a
- A quantidade de luz refletida dependerá das propriedades óticas (material) das superfícies
- Luz ambiente pode ser modelada assumindo que a luz incidente é **espalhada com igual intensidade** em todas as direções independente da direção de visão

- Um parâmetro k_a (**coeficiente de reflexão ambiente**) pode ser usado para determinar a fração da luz incidente que será espalhada como reflexões difusas para gerar a iluminação ambiente de cada objeto na cena. A iluminação ambiente é constante, igual o nível de luz ambiente I_a multiplicada por k_a
- Para fontes de luz monocromáticas, $0, 0 \leq k_a \leq 1, 0$:
 - Superfícies brilhantes apresentam valores de k_a altos – intensidade de reflexão próxima da incidente
 - Valores de k_a próximos de 0 definem superfícies que absorvem a luz

□ Equação de Luz Ambiente

- Para efeitos de luz de fundo, as superfícies são completamente iluminadas pela luz ambiente I_a definida para a cena, e a contribuição dessa para reflexão difusa é

$$I_{ambdiff} = k_a \cdot I_a$$

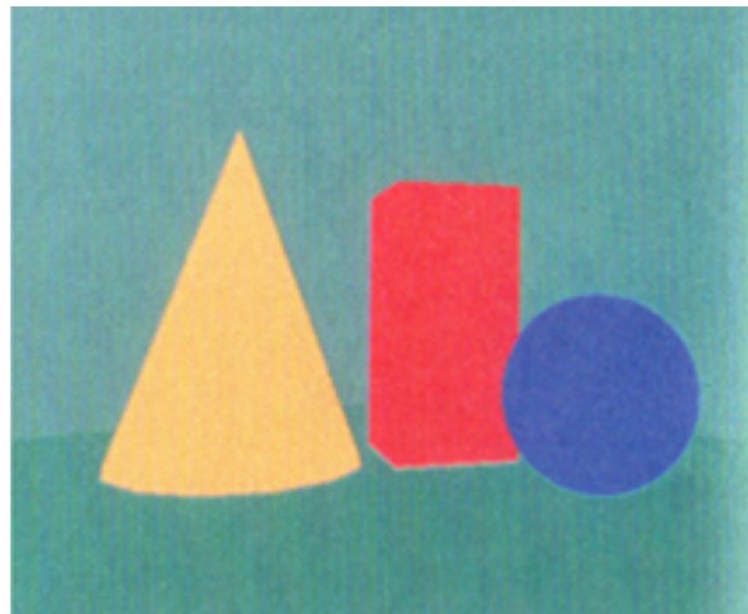


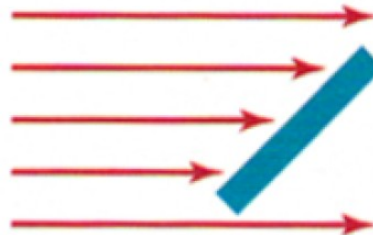
Figura: Se somente luz ambiente é considerada, o efeito de iluminação obtido em uma cena é pouco interessante.

Reflexão Difusa

- Quando uma superfície é iluminada por uma fonte de luz com intensidade I_l , a quantidade de **luz incidente** depende da **orientação da superfície** relativa a direção da luz



(a)

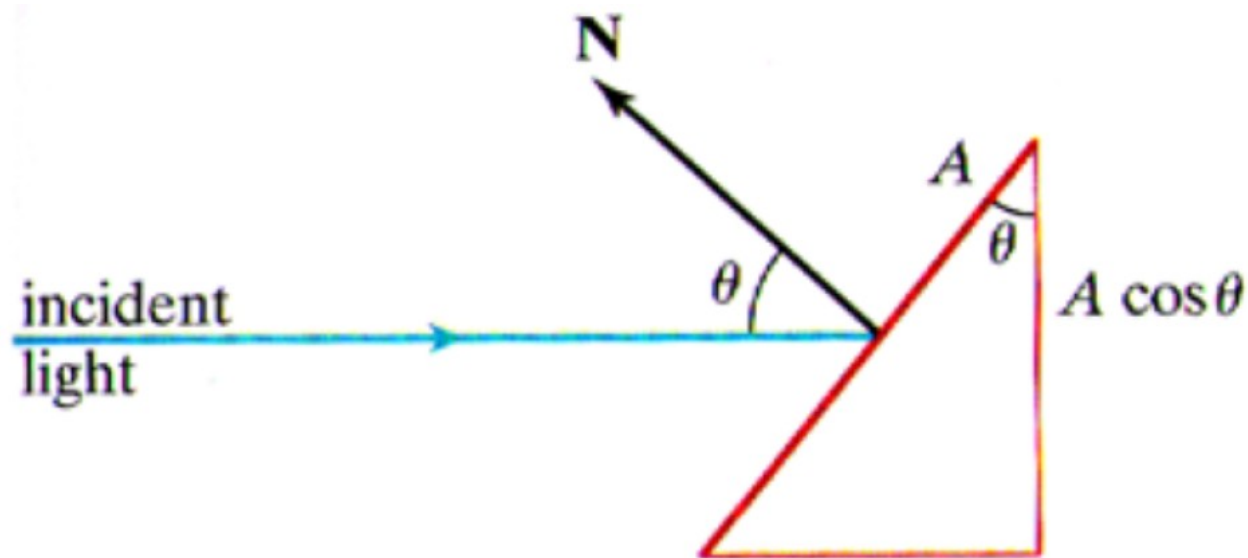


(b)

- O número de raios intersectando a superfície é proporcional a área da projeção perpendicular da superfície na direção da luz incidente

Reflexão Difusa

- Considerando um **ângulo de incidência** θ entre a direção da luz incidente e a normal da superfície:



- A área projetada é proporcional a $\cos \theta$

- Assim podemos modelar a quantidade de luz incidente de uma fonte com intensidade I_l como:

$$I_{l,incident} = I_l \cos \theta$$

- Com isso podemos modelar a reflexão difusa de uma fonte de luz com intensidade I_l como:

$$I_{l,diff} = k_d I_{l,incident}$$

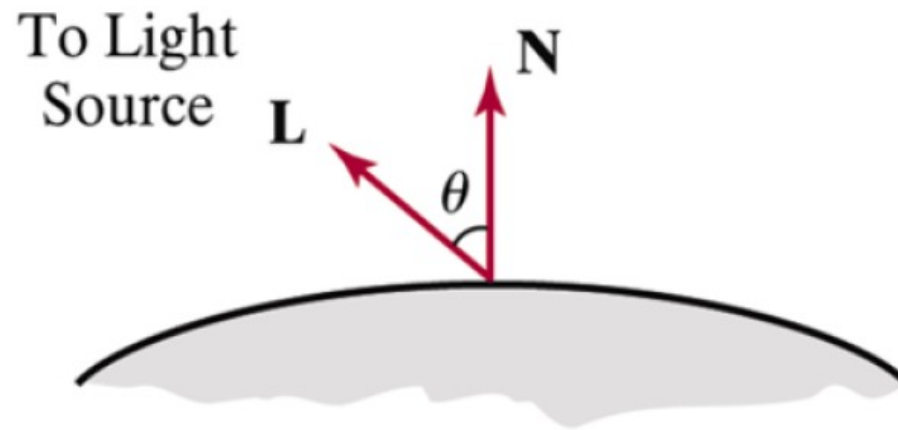
$$I_{l,diff} = k_d I_l \cos \theta$$

Reflexão Difusa

- Conforme o **ângulo de incidência** aumenta, a **iluminação** a partir da fonte de luz diminui
- Uma superfície somente será iluminada quando $0^\circ \leq \theta \leq 90^\circ$, quando $\cos \theta < 0.0$, a luz estará atrás da superfície

Reflexão Difusa

- Considerando N como o vetor unitário normal a superfície e L o vetor unitário de direção da luz, então $\cos \theta = N \cdot L$ (produto escalar)



- Equação de Reflexão Difusa**
 - A equação de reflexão difusa para uma única fonte de luz fica:

$$I_{l,diff} = \begin{cases} k_d I_l (N \cdot L), & \text{se } N \cdot L > 0 \\ 0.0, & \text{se } N \cdot L \leq 0 \end{cases}$$

Reflexão Difusa

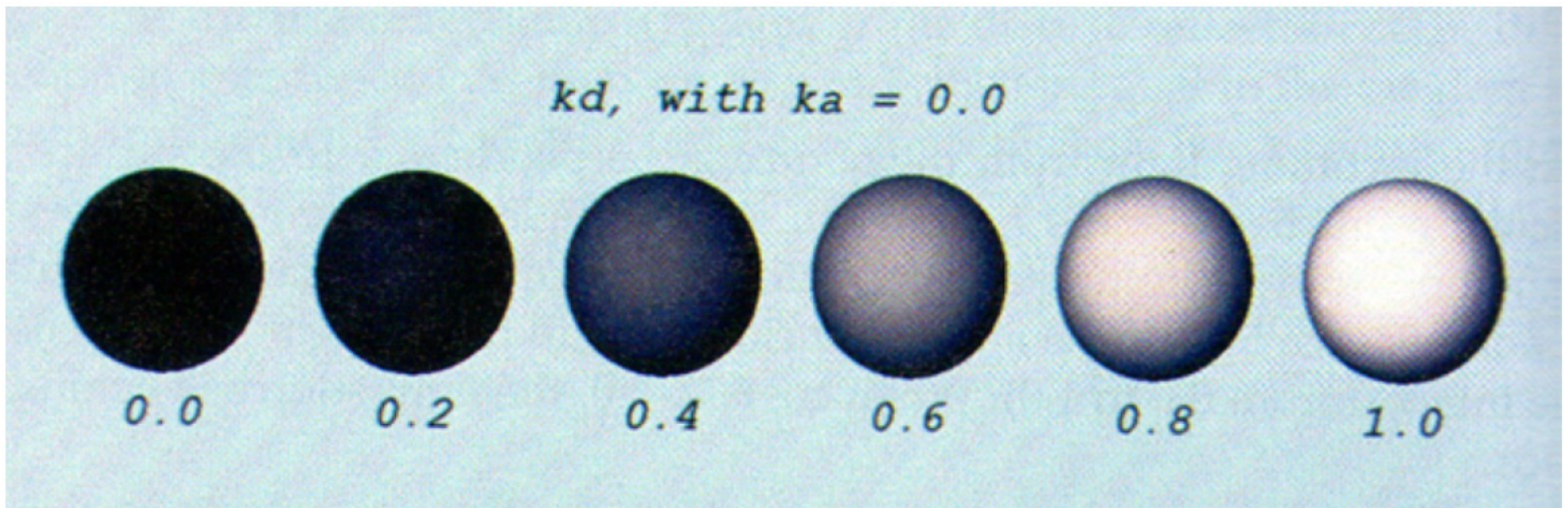
- O vetor unitário L é calculado para fontes de luz pontuais usando a posição da superfície, P_{surf} , e a posição da fonte de luz, P_{source} :

$$L = \frac{P_{source} - P_{surf}}{|P_{source} - P_{surf}|}$$

- Uma fonte de luz no infinito não tem posição, somente a direção de propagação
 - Emprega-se o negativo da direção de emissão para a direção do vetor L

Reflexão Difusa

- Exemplo de iluminação difusa variando k_d entre 0 e 1:
 - Uma única fonte de luz pontual
 - Sem luz ambiente



Reflexão Difusa

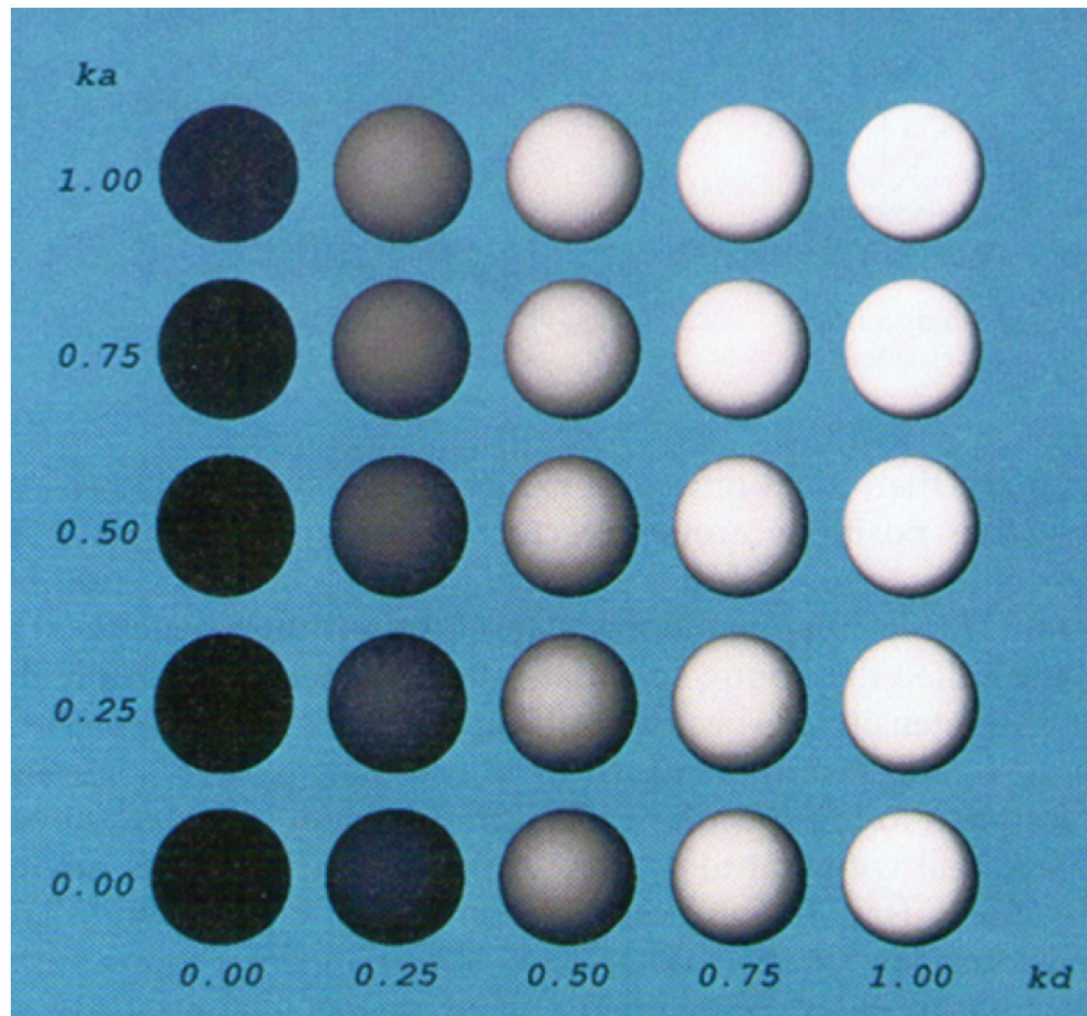
- Podemos **combinar cálculos** de fonte de luz ambiente e pontual para se obter a **reflexão difusa** em uma posição da superfície
- Introduz o **coeficiente de reflexão ambiente** k_a para cada superfície para modificar a intensidade I_a da luz ambiente.
- Equação da Reflexão Difusa Total**
 - Podemos escrever a equação de reflexão difusa total de uma única fonte pontual como:

$$I_{diff} = \begin{cases} k_a I_a + k_d I_l (N \cdot L), & \text{se } N \cdot L > 0 \\ k_a I_a, & \text{se } N \cdot L \leq 0 \end{cases}$$

- Onde k_a e k_d dependem das propriedades do material da superfície.

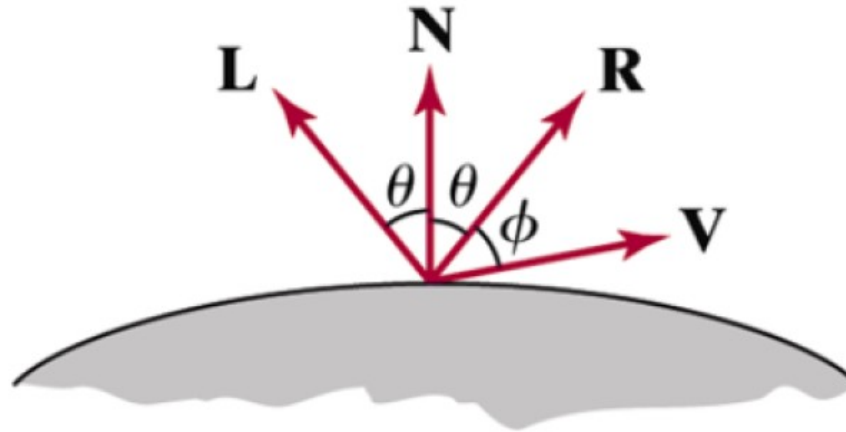
Reflexão Difusa

- Exemplo de iluminação difusa variando k_a e k_d :



Reflexão Especular e Modelo de Phong

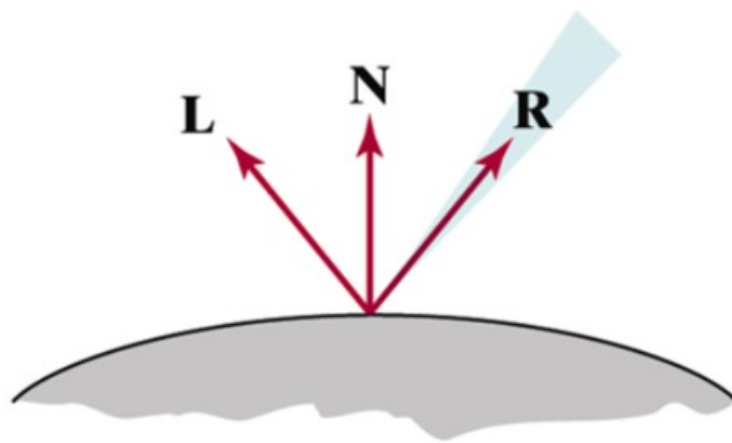
- A **reflexão especular** vista em uma superfície brilhante é o resultado da reflexão total (ou quase) da luz incidente em uma **área concentrada** ao redor de um **ângulo de reflexão especular**



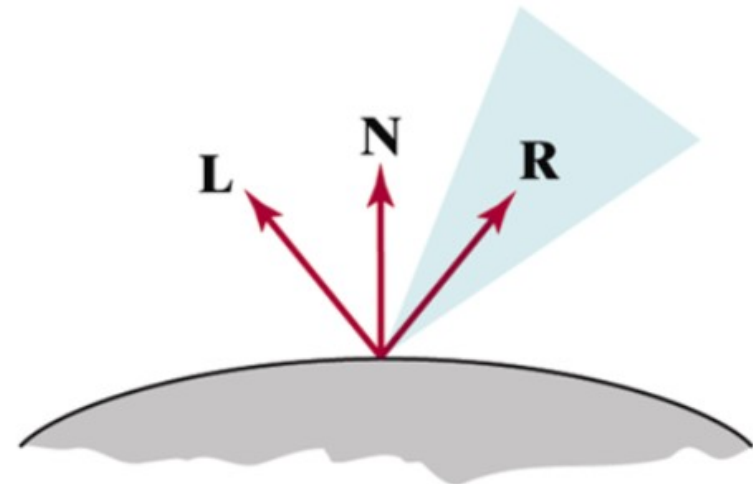
- O ângulo de reflexão especular θ é igual ao ângulo de incidência da luz, opostos a normal da superfície N
- O vetor unitário R representa a direção de reflexão especular ideal
- O vetor unitário L aponta na direção da fonte de luz
- O vetor unitário V aponta na direção do visualizador.

Reflexão Especular e Modelo de Phong

- Em um refletor ideal (espelho perfeito), a luz incidente é refletida somente na direção de reflexão especular, e será visível somente quando V e R coincidirem ($\phi = 0^\circ$)
- Objetos que não são refletores ideais exibem reflexão especular ao redor de R em um intervalo finito de posições de visão:
 - Superfícies **brilhantes** tem um campo menor de reflexão especular
 - Superfícies **foscas** tem um campo maior de reflexão especular



Shiny Surface
(Large n_s)



Dull Surface
(Small n_s)

Reflexão Especular e Modelo de Phong

- **Modelo de Phong** define a intensidade da reflexão especular proporcional a $\cos^{n_s} \varphi$, com $0^\circ \leq \varphi \leq 90^\circ$
- O **expoente de reflexão especular** n_s (shininess) é determinado pelo tipo de superfície:
 - Superfícies brilhantes apresentam valores altos de n_s (p.ex. 100 ou mais)
 - Superfícies rugosas apresentam valores baixos de n_s (p.ex. 1)
 - Para refletores perfeitos $n_s \rightarrow \infty$

Reflexão Especular e Modelo de Phong

- Diferentes gráficos de $\cos^{n_s} \phi$ usando diferentes valores de expoente especular n_s :

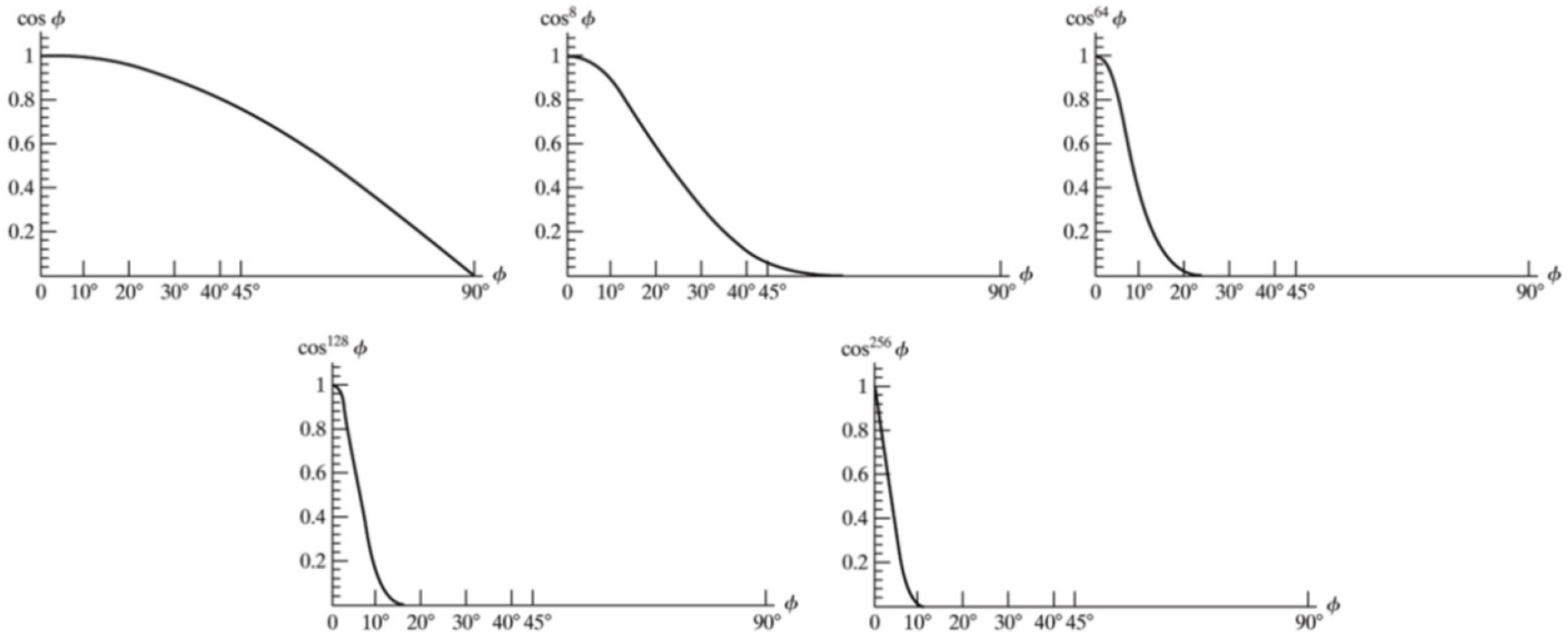


Figura: Mostrando o efeito de n_s no intervalo de ângulos para os quais podemos esperar ver reflexões especulares.

Reflexão Especular e Modelo de Phong

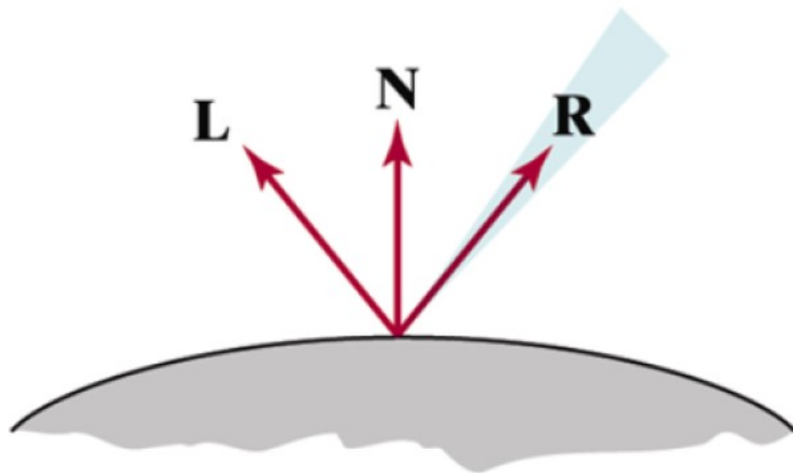
- A **intensidade da reflexão especular** também depende das propriedades dos **materiais** da superfície
 - É possível aproximar a intensidade da reflexão especular para um dado **material** usando o **coeficiente de reflexão especular** k_s

Reflexão Especular e Modelo de Phong

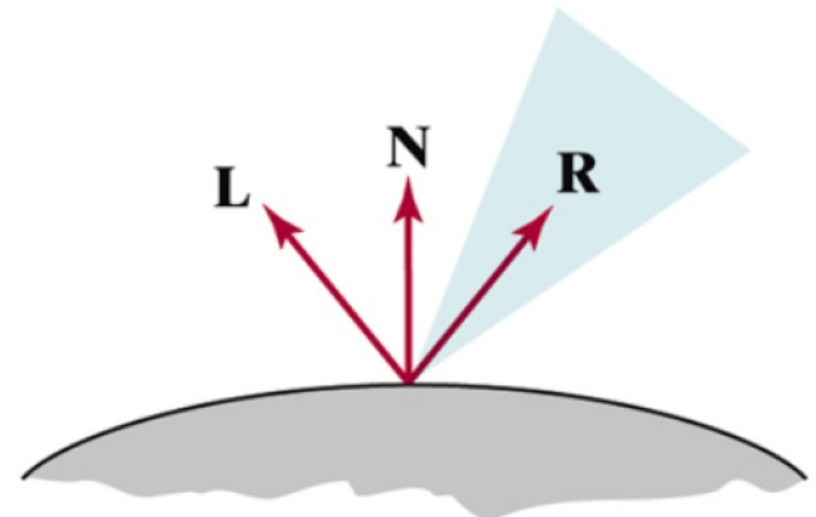
- Usando coeficiente de reflexão especular, k_s , podemos escrever o modelo de Phong de **reflexão especular**:

$$I_{l,spec} = k_s I_l \cos^{n_s} \phi$$

- Onde I_l é a intensidade da fonte de luz e ϕ é o ângulo de visão relativo a R



Shiny Surface
(Large n_s)



Dull Surface
(Small n_s)

Reflexão Especular e Modelo de Phong

- Como V e R são vetores unitários, então $\cos \varphi = V \cdot R$
- Se V e L estiverem do mesmo lado da normal N ($V \cdot R \leq 0$) ou a fonte de luz estiver atrás da superfície ($N \cdot L \leq 0$), efeitos especulares não precisam ser calculados.
- **Equação da Reflexão Especular**
 - Assumindo que o coeficiente de reflexão especular é constante para qualquer material, podemos calcular para uma única fonte de luz:

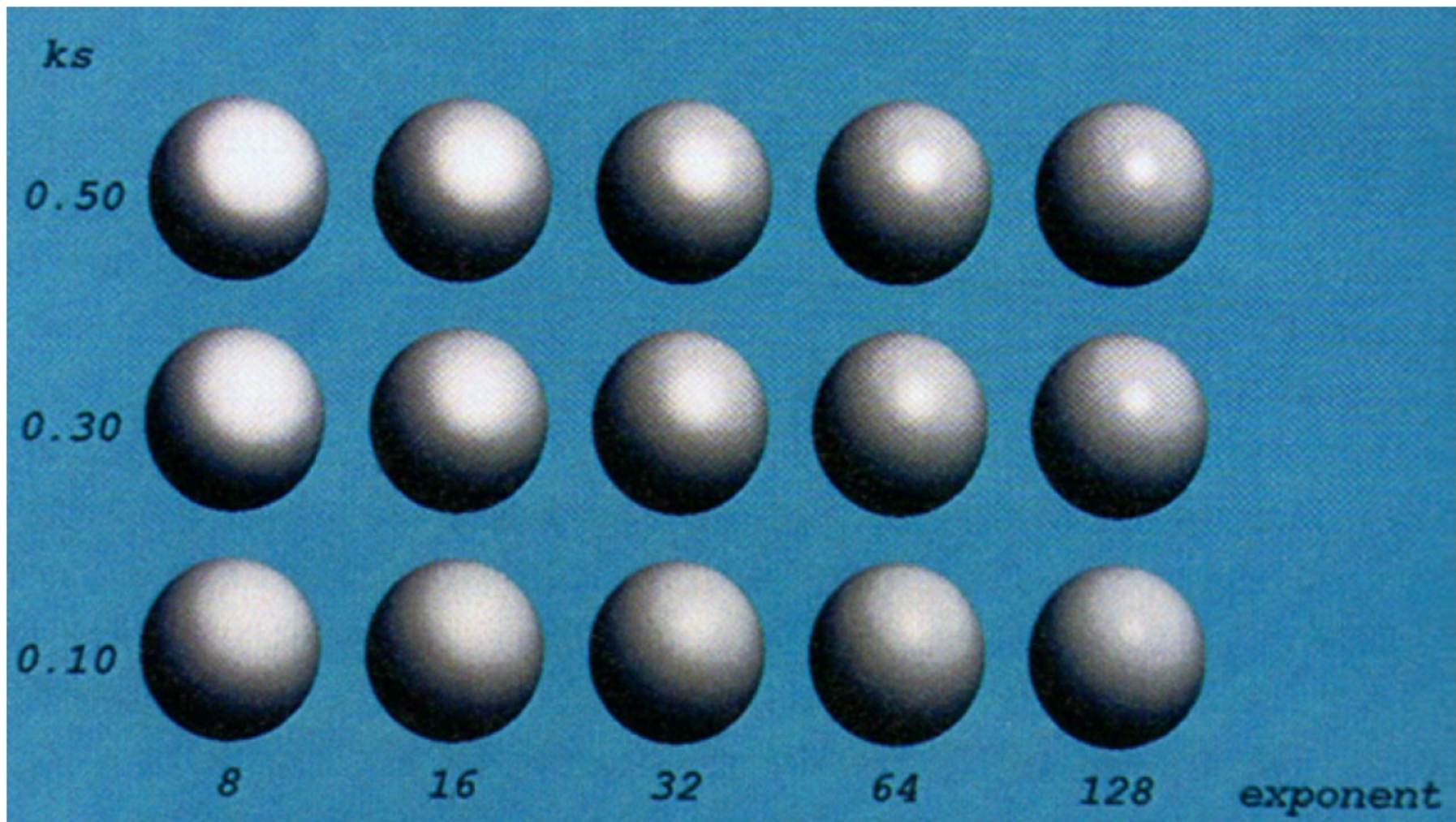
$$I_{l,spec} = \begin{cases} k_s I_l (V \cdot R)^{n_s}, & \text{se } V \cdot R > 0 \text{ e } N \cdot L > 0 \\ 0.0, & \text{se } V \cdot R \leq 0 \text{ ou } N \cdot L \leq 0 \end{cases}$$

- A direção de R pode ser calculada como $R = N(2N \cdot L) - L$, ou seja, a partir dos vetores N e L ¹

¹Dedução da fórmula disponível no Apêndice

Reflexão Especular e Modelo de Phong

- Exemplos de reflexão especular variando k_s e n_s em uma superfície esférica iluminada por uma única fonte de luz pontual:



Reflexão Especular e Modelo de Phong

- O vetor V é calculado usando a posição da superfície, P_{surf} , e a posição de visão, P_{view} :

$$V = \frac{P_{view} - P_{surf}}{|P_{view} - P_{surf}|}$$

- Se apenas **uma direção de visão** é usada para todas as posições na tela, o **cálculo** da iluminação especular é **acelerado**
 - Mas o resultado final é pior.

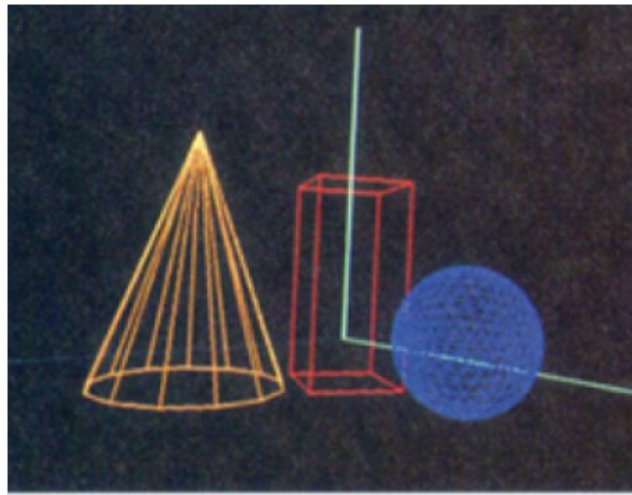
□ Equação da Reflexão Especular e Difusa Combinadas

- Para uma única fonte de luz, podemos modelar a combinação das reflexões difusa e especular como:

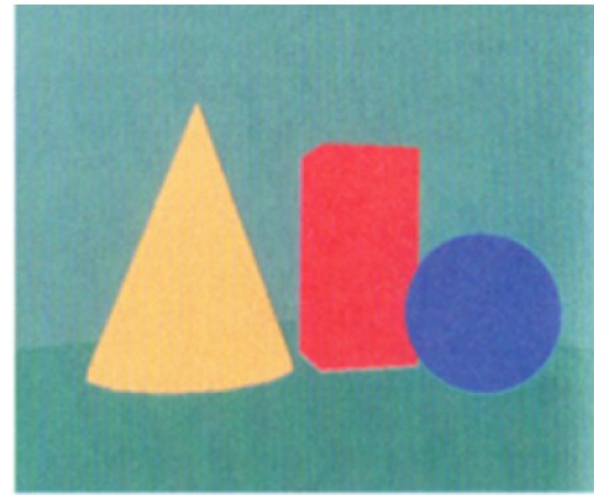
$$\begin{aligned} I &= I_{diff} + I_{spec} \\ &= (k_a I_a + k_d I_l (N \cdot L)) + k_s I_l (V \cdot R)^{n_s} \end{aligned}$$

- A superfície será iluminada somente pela luz ambiente quando a fonte de luz estiver atrás da superfície ($N \cdot L \leq 0$)
- Se V e L estiverem do mesmo lado da normal N ($V \cdot R \leq 0$) ou a fonte de estiver atrás da superfície, efeitos especulares não precisam ser calculados

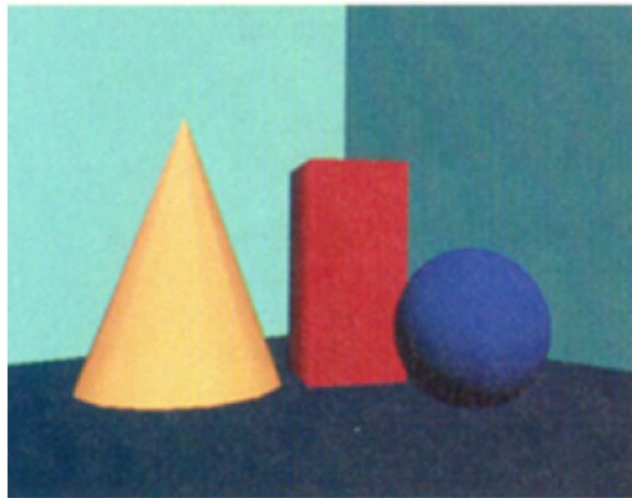
Reflexão Especular e Difusa Combinadas



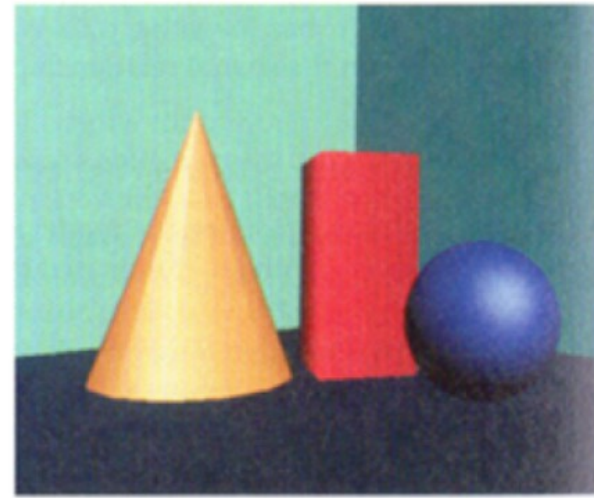
(a)



(b)



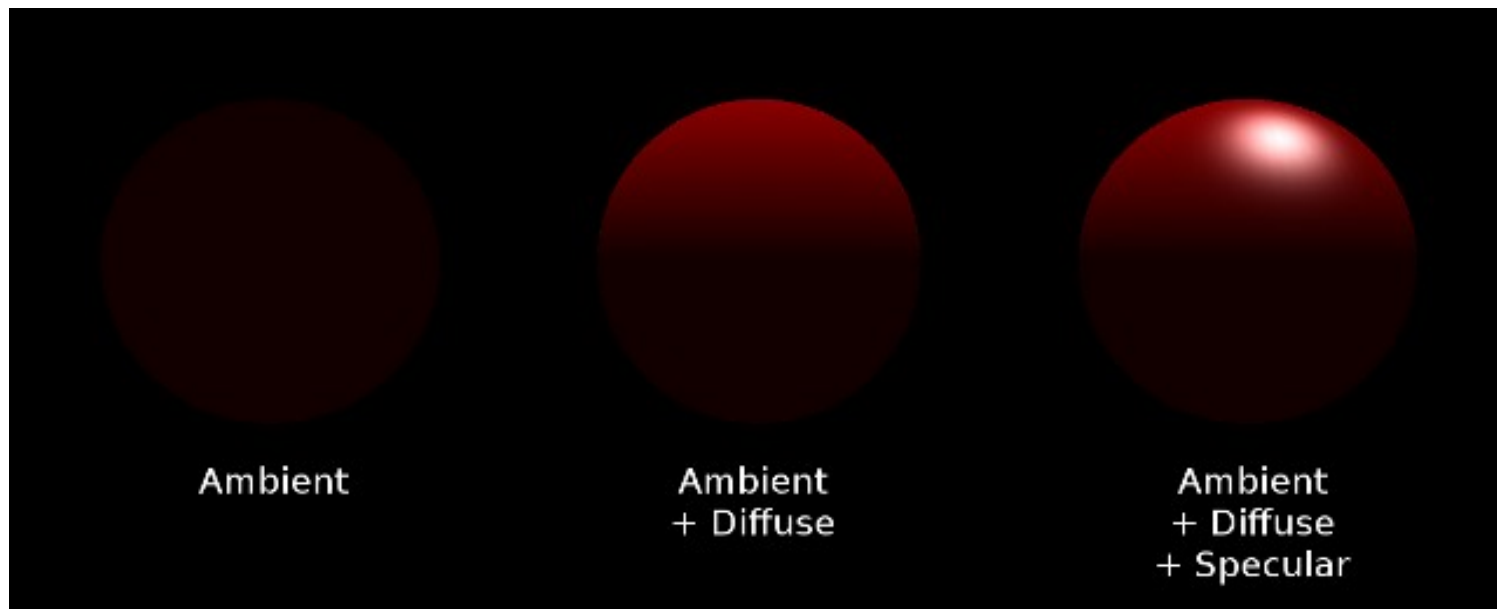
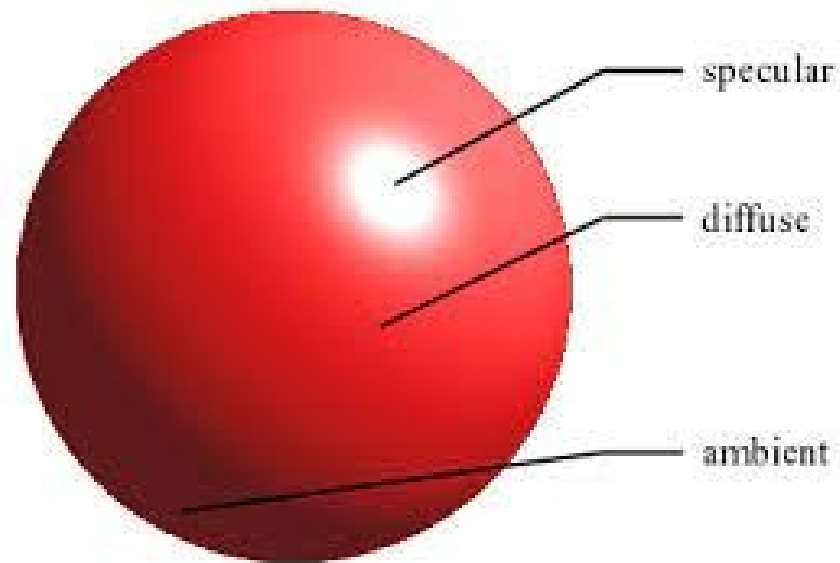
(c)



(d)

Figura: Cena wire-frame (a), usando somente luz ambiente (b), reflexão difusa com luz ambiente e uma única fonte de luz pontual (c) e reflexão difusa e especular com luz ambiente e uma única fonte de luz pontual (d).

Reflexão Especular e Difusa Combinadas



- **Equação da Reflexão Especular e Difusa Combinadas para Múltiplas Fontes de Luz**
 - É possível usar a quantidade de fontes de luz que se deseja, para isso soma-se as contribuições de reflexão difusa e especular de cada fonte:

$$I = I_{ambdiff} + \sum_{l=1}^n [I_{l,diff} + I_{l,spec}]$$

$$I = k_a I_a + \sum_{l=1}^n I_l [k_d (N \cdot L_l) + k_s (V \cdot R_l)^{n_s}]$$

Modelo de Iluminação Básico com Atenuação da Intensidade

□ Equação para Modelo de Iluminação Básico com Atenuação da Intensidade

- É possível formular uma modelo de iluminação monocromático geral como:

$$I = I_{ambdiff} + \sum_{l=1}^n f_{l,radatten}(I_{l,diff} + I_{l,spec})$$

- Como visto anteriormente:

$$f_{l,radatten}(d_l) = \frac{1}{a_0 + a_1 d_l + a_2 d_l^2}$$

- Onde:

- d_l é a distância entre o ponto na superfície e a fonte de luz com índice l
- e a_0 , a_1 e a_2 são as constantes de atenuação

Modelo de Iluminação Básico com Atenuação da Intensidade

- Onde

$$I_{ambdiff} = k_a I_a$$

- E

$$I_{l,diff} = \begin{cases} 0.0, & \text{se } N \cdot L_l \leq 0 \text{ (fonte de luz atrás do objeto)} \\ k_d I_l (N \cdot L_l), & \text{caso contrário} \end{cases}$$

- E por fim

$$I_{l,spec} = \begin{cases} 0.0, & \text{se } V \cdot R_l \leq 0 \text{ ou } N \cdot L_l \leq 0 \\ k_s I_l (V \cdot R_l)^{ns}, & \text{caso contrário} \end{cases}$$

Considerações sobre Cor RGB

- Para **cores RGB**, as intensidades são modeladas com **vetores de 3 elementos** que designam os componentes vermelho, verde e azul:

$$I_l = (I_{lR}, I_{lG}, I_{lB})$$

$$I_a = (I_{aR}, I_{aG}, I_{aB})$$

- Similarmente os **coeficientes de reflexão** são também especificados para as 3 componentes de cor:

$$k_a = (k_{aR}, k_{aG}, k_{aB})$$

$$k_d = (k_{dR}, k_{dG}, k_{dB})$$

$$k_s = (k_{sR}, k_{sG}, k_{sB})$$

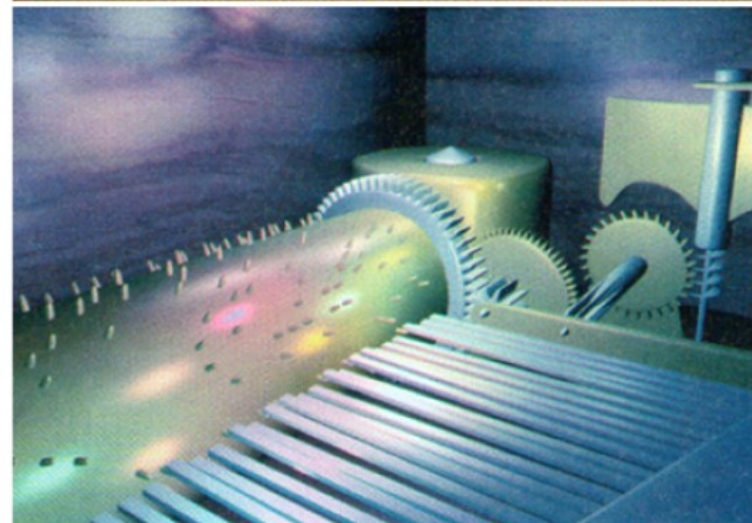
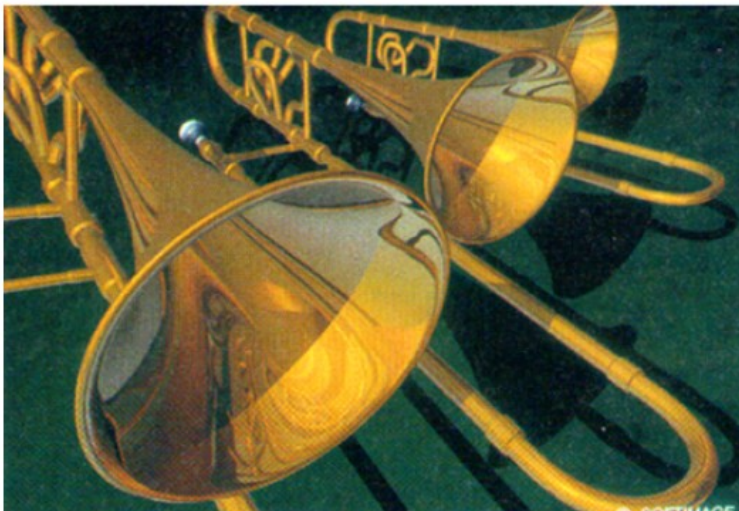
- Assim cada componente de cor da superfície é calculada separadamente.

Considerações sobre Cor RGB

- Para definir a **cor de uma superfície**, os coeficientes de reflexão difusa podem ser usados:
 - Por exemplo, para a definição de uma superfície azul, um valor diferente de zero para a componente k_{dB} deve ser escolhido enquanto as outras componentes são zeradas $k_{dR} = k_{dG} = 0.0$
 - Somente a luz azul é refletida, as outras são absorvidas pela superfície

Considerações sobre Cor RGB

- Exemplos de reflexão considerando diferentes **materiais** e **múltiplas fontes (coloridas) de luz**:



- As **rotinas de iluminação** são ativadas usando:

```
glEnable (GL_LIGHTING) ;
```

- **Múltiplas fontes de luz** podem ser adicionadas a uma cena e **várias propriedades** podem ser associadas a cada fonte usando:

```
glLight* (light_name, light_property, property_value) ;
```

- Um sufixo *i*, *iv*, *f* ou *fv* é adicionado ao nome da função dependendo do **tipo de dado** do valor da propriedade

- O parâmetro `light_name` recebe um identificador:

`GL_LIGHT0, GL_LIGHT1, GL_LIGHT2, ..., GL_LIGHT7`

- Depois de todos os parâmetros de uma luz terem sido definidos, essa deve ser ligada usando:

```
glEnable(light_name);
```


Especificando a Posição e Tipo de uma Fonte de Luz

- Para designar a **posição** de uma fonte de luz usa-se o flag `GL_POSITION` e passa-se um vetor de 4 elementos
 - Os primeiros 3 elementos do vetor definem sua posição em coordenadas do mundo.
- Quarto elemento do vetor é usado para definir o **tipo de fonte de luz**:
 - Fonte próxima da cena (posição):
 - Quarto elemento do vetor diferente de 0.0
 - Fonte distante da cena (direção):
 - Quarto elemento do vetor igual a 0.0
 - O raio de luz está na direção da linha partindo do ponto (x, y, z) até a origem.

Especificando a Posição e Tipo de uma Fonte de Luz

- O seguinte exemplo define duas fontes de luz, uma local e uma distante:

```
//fonte local
GLfloat light0_pos[4] = {2.0, 0.0, 3.0, 1.0};
//fonte distante
GLfloat light1_pos[4] = {0.0, 1.0, 0.0, 0.0};
//define posição da luz local
glLightfv(GL_LIGHT0, GL_POSITION, light0_pos);
glEnable(GL_LIGHT0);
//define direção da luz distante
glLightfv(GL_LIGHT1, GL_POSITION, light1_pos);
glEnable(GL_LIGHT1);
```

- Os valores padrão de uma fonte de luz são (0.0, 0.0, 1.0, 0.0)
 - Fonte de luz distante e luz na direção negativa de z

Especificando a Posição e Tipo de uma Fonte de Luz

- A **posição da luz** é incluída na descrição da cena, sendo transformada em **coordenadas de visão** junto com a descrição dos objetos

Especificando as Cores da Fonte de Luz

- A cor da luz é definida especificando as diferentes cores RGBA
 - O componente **alpha** só é usado quando se ativa a transparência na cena
- Para cada fonte de luz especifica-se sua contribuição para efeitos de luz ambiente, difusa e especular

```
GLfloat white[4] = {1.0, 1.0, 1.0, 1.0};  
GLfloat black[4] = {0.0, 0.0, 0.0, 1.0};  
  
glLightfv(GL_LIGHT0, GL_AMBIENT, black);  
glLightfv(GL_LIGHT0, GL_DIFFUSE, white);  
glLightfv(GL_LIGHT0, GL_SPECULAR, white);
```

- Por padrão, para a **luz 0** a propriedade de **luz ambiente** é preta e branca para as propriedades difusa e especular
 - Para as outras luzes todas as propriedades são pretas

Exemplo 1 – Especificando as Cores da Fonte de Luz

```
#include <GL/glut.h>

void lighting(){
    float position[4] = {2.0f, 2.0f, 2.0f, 1.0f};
    float white[4] = {1.0f, 1.0f, 1.0f, 1.0f};
    float black[4] = {0.0f, 0.0f, 0.0f, 1.0f};
    glLightfv(GL_LIGHT0, GL_POSITION, position);
    glLightfv(GL_LIGHT0, GL_AMBIENT, black);
    glLightfv(GL_LIGHT0, GL_DIFFUSE, white);
    glLightfv(GL_LIGHT0, GL_SPECULAR, white);

    //ativa a iluminação
    glEnable(GL_LIGHTING);
    glEnable(GL_LIGHT0);
}
```

Exemplo 1 – Especificando as Cores da Fonte de Luz

```
int init(){
    glClearColor(1.0f, 1.0f, 1.0f, 1.0f); //define a cor de fundo
    glEnable(GL_DEPTH_TEST); //habilita o teste de profundidade

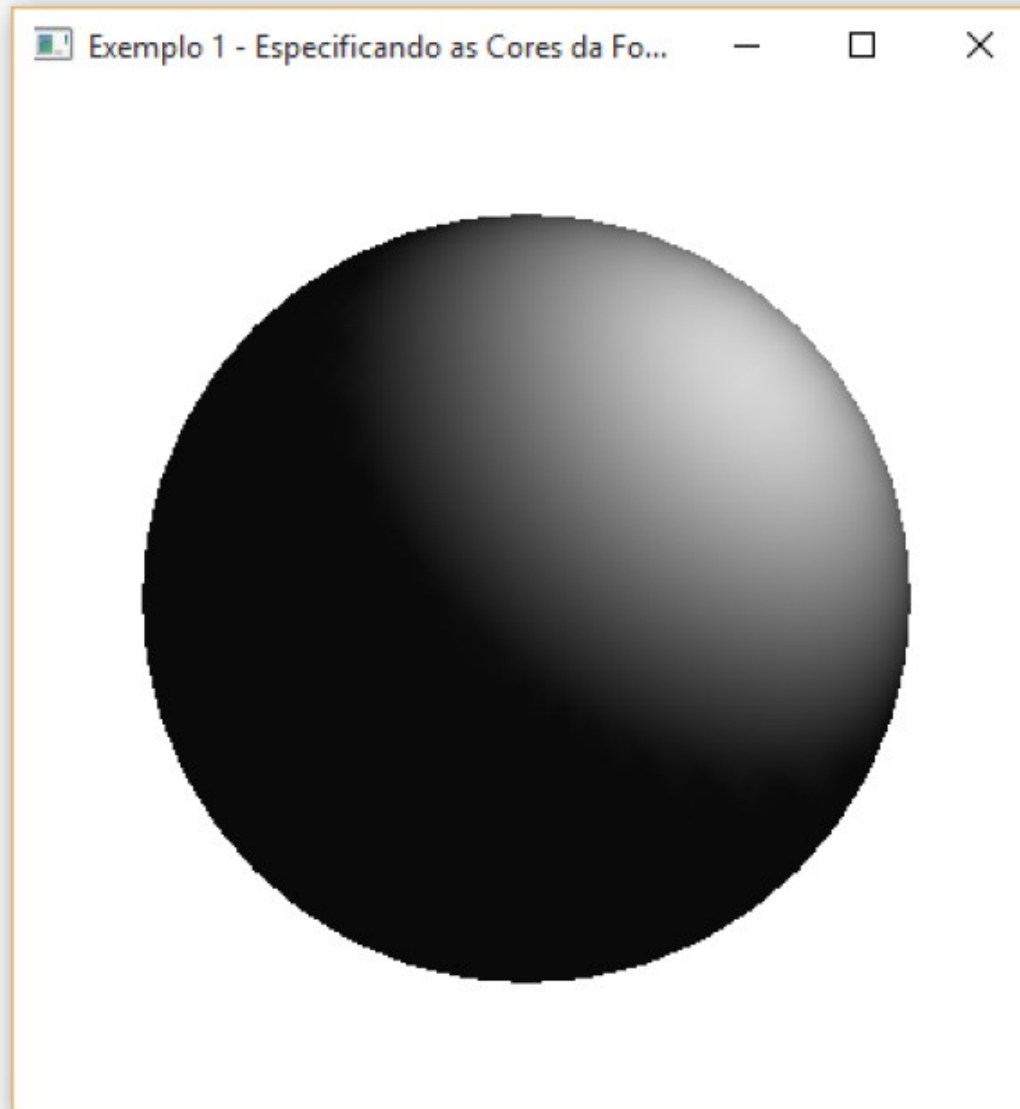
    glMatrixMode(GL_MODELVIEW); //define que a matriz é a model view
    glLoadIdentity(); //carrega a matriz de identidade
    gluLookAt(0.0, 0.0, 1.0, //posição da câmera
              0.0, 0.0, 0.0, //para onde a câmera aponta
              0.0, 1.0, 0.0); //vetor view-up
    glMatrixMode(GL_PROJECTION); //define que a matriz é a de projeção
    glLoadIdentity(); //carrega a matriz de identidade
    glOrtho(-2.0, 2.0, -2.0, 2.0, -2.0, 2.0);

    lighting();
}
```

Exemplo 1 – Especificando as Cores da Fonte de Luz

```
void display() {  
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);  
    glMatrixMode(GL_MODELVIEW);  
    glutSolidSphere(1.5, 40, 40);  
  
    glFlush();  
}  
  
int main(int argc, char** argv) {  
    glutInit(&argc, argv);  
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB | GLUT_DEPTH);  
  
    glutInitWindowPosition(200, 0);  
    glutInitWindowSize(400, 400);  
    glutCreateWindow("Exemplo 1 - Definindo as Cores da Fonte de Luz");  
  
    init();  
    glutDisplayFunc(display);  
    glutMainLoop();  
    return 0;  
}
```

Exemplo 1 – Especificando as Cores da Fonte de Luz



Especificando os Coeficientes de Atenuação Radial

- É possível especificar os **coeficientes de atenuação radial** a_0 , a_1 , a_2 usando:

```
//define a0  
glLightfv(GL_LIGHT0, GL_CONSTANT_ATTENUATION, 0.5);  
//define a1  
glLightfv(GL_LIGHT0, GL_LINEAR_ATTENUATION, 0.15);  
//define a2  
glLightfv(GL_LIGHT0, GL_QUADRATIC_ATTENUATION, 0.1);
```

- Os valores dos coeficientes podem ser **inteiros** ou de **ponto flutuante positivos**
 - Os valores padrão para a atenuação radial são $a_0 = 1$, $a_1 = 0$, $a_2 = 0$ (atenuação desativada)

Exemplo 2 - Especificando os Coeficientes de Atenuação Radial

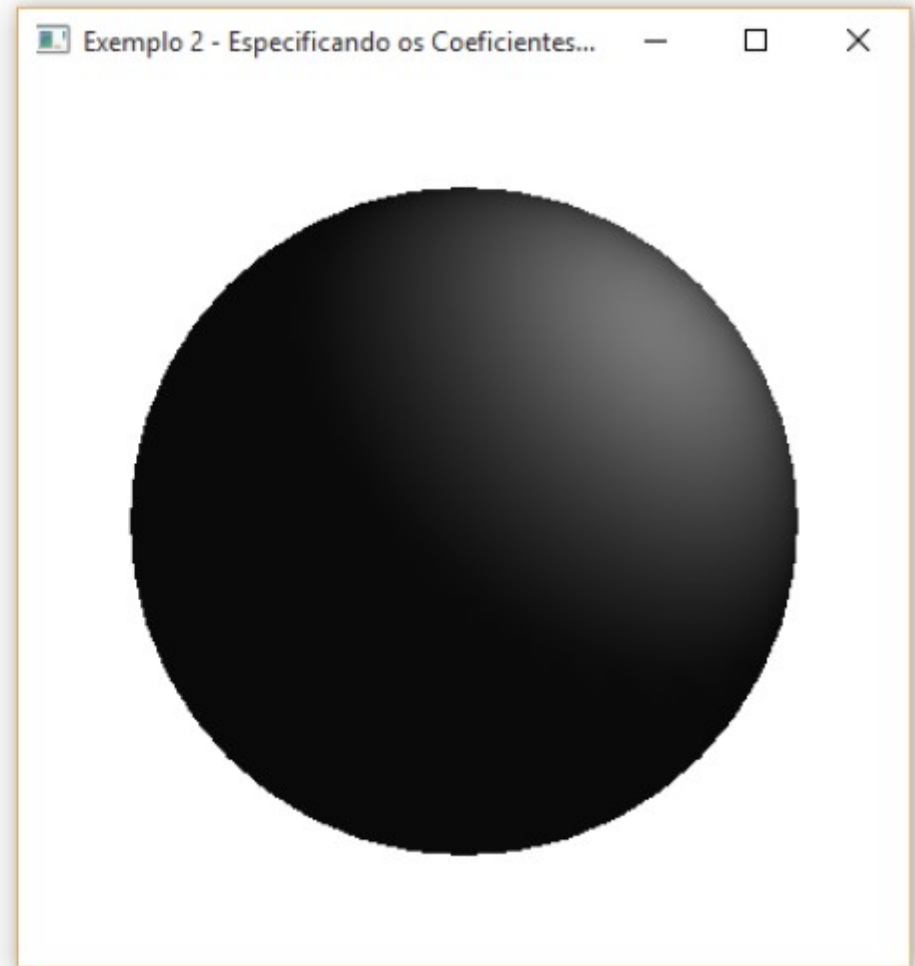
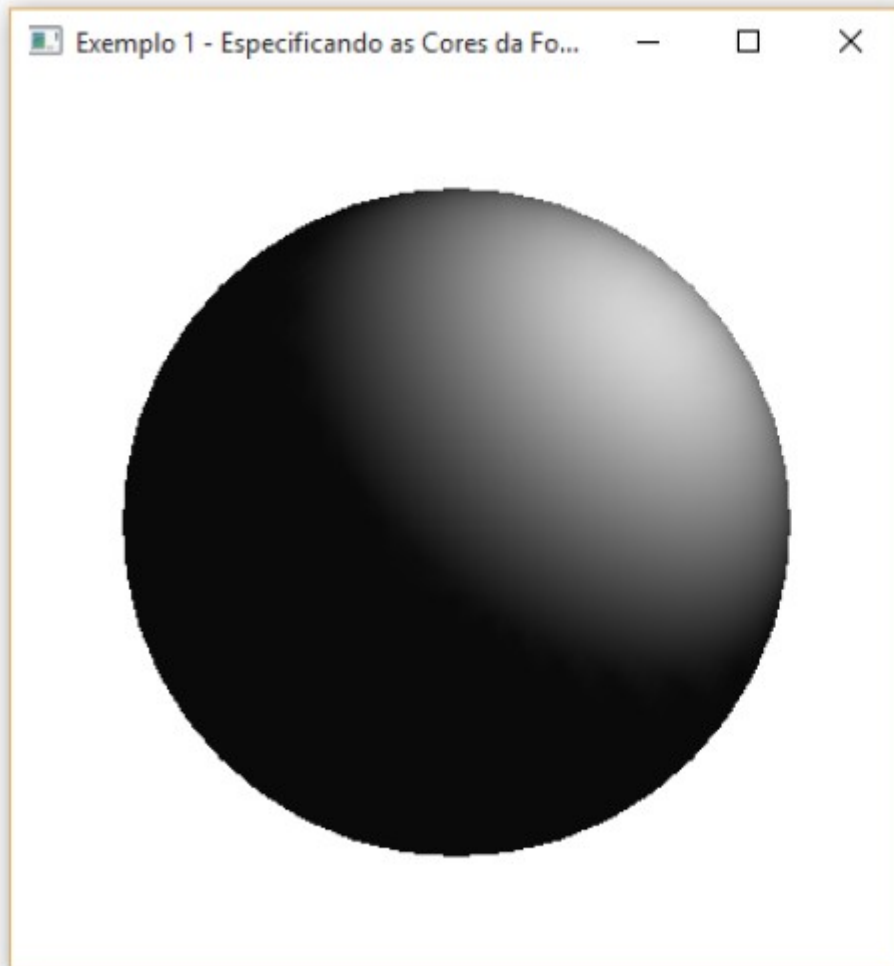
```
void lighting(){
    //uma fonte de luz local
    float position[4] = {2.0f, 2.0f, 2.0f, 1.0f};
    float white[4] = {1.0f, 1.0f, 1.0f, 1.0f};
    float black[4] = {0.0f, 0.0f, 0.0f, 1.0f};

    glLightfv(GL_LIGHT0, GL_POSITION, position);
    glLightfv(GL_LIGHT0, GL_AMBIENT, black);
    glLightfv(GL_LIGHT0, GL_DIFFUSE, white);
    glLightfv(GL_LIGHT0, GL_SPECULAR, white);

    //ativa a atenuação
    glLightf(GL_LIGHT0, GL_CONSTANT_ATTENUATION, 0.5f); //define a0
    glLightf(GL_LIGHT0, GL_LINEAR_ATTENUATION, 0.5f); //define a1
    glLightf(GL_LIGHT0, GL_QUADRATIC_ATTENUATION, 0.1f); //define a2

    //ativa a iluminação
    glEnable(GL_LIGHTING);
    glEnable(GL_LIGHT0);
}
```

Exemplo 2 - Especificando os Coeficientes de Atenuação Radial



Parâmetros de Iluminação Global

- Vários **parâmetros de luz** podem ser definidos globalmente usando:

```
glLightModel*(param_name, param_value);
```

- O sufixo pode ser *i*, *iv*, *f* ou *fv* dependendo do tipo de parâmetro
- Podemos definir globalmente:
 - O nível de **luz ambiente**
 - Como os **brilhos especulares** são calculados
 - Se o **modelo de iluminação** deve ser aplicado na parte de trás dos polígonos

Parâmetros de Iluminação Global

- Por exemplo, para definir uma **luz ambiente** independente das fontes de luz existentes usamos:

```
float global_ambient[4] = {0.9, 0.9, 0.9, 1.0};  
glLightModelfv(GL_LIGHT_MODEL_AMBIENT, global_ambient);
```

- Por padrão essa cor é **branca** de baixa intensidade (0.2, 0.2, 0.2, 1.0)

Exemplo 3 – Parâmetros de Iluminação Global

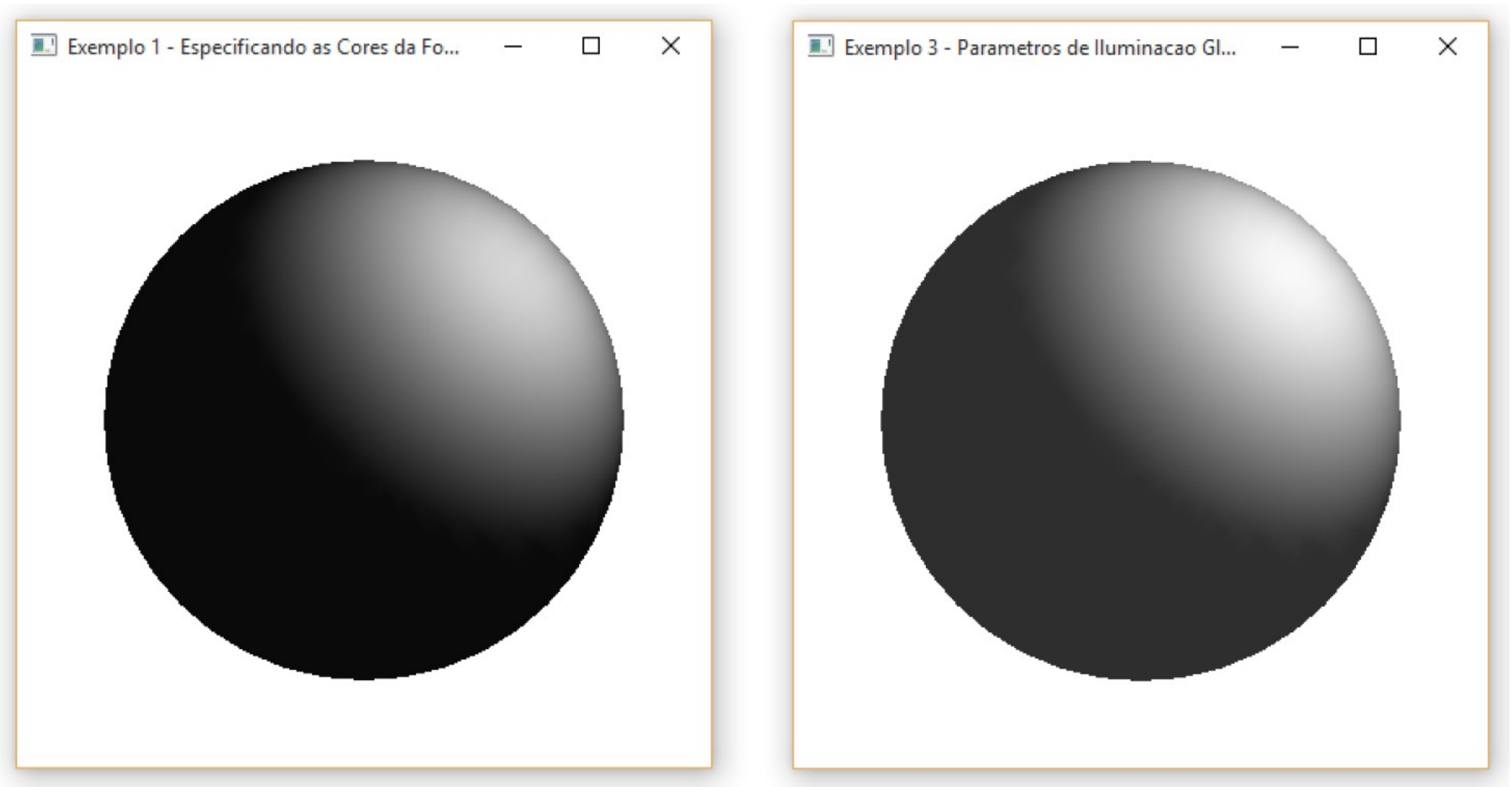
```
void lighting(){
    //uma fonte de luz local
    float position[4] = {2.0f, 2.0f, 2.0f, 1.0f};
    float white[4] = {1.0f, 1.0f, 1.0f, 1.0f};
    float black[4] = {0.0f, 0.0f, 0.0f, 1.0f};

    glLightfv(GL_LIGHT0, GL_POSITION, position);
    glLightfv(GL_LIGHT0, GL_AMBIENT, black);
    glLightfv(GL_LIGHT0, GL_DIFFUSE, white);
    glLightfv(GL_LIGHT0, GL_SPECULAR, white);

    //ativa luz ambiente global
    // LA independente de fonte de luz existentes
    float global_ambient[4] = {0.9, 0.9, 0.9, 1.0};
    glLightModelfv(GL_LIGHT_MODEL_AMBIENT, global_ambient);

    //ativa a iluminação
    glEnable(GL_LIGHTING);
    glEnable(GL_LIGHT0);
}
```

Exemplo 3 – Parâmetros de Iluminação Global



Parâmetros de Iluminação Global

- Para o cálculo da reflexão especular é necessário **determinar o vetor V** (da superfície para a posição de visão)
 - Podemos acelerar o processamento fazendo V constante independente da posição da superfície
- O valor padrão para V é a direção de z , (0.0, 0.0, 1.0), mas podemos usar a posição de visão corrente fazendo:

```
glLightModeli(GL_LIGHT_MODEL_LOCAL_VIEWER, GL_TRUE);
```

- Isso tornará o processo computacionalmente mais caro, mas o resultado será mais realístico
 - Para desabilitar o cálculo de V , fazer essa função igual a `GL_FALSE`

Propriedades da Superfície

- As propriedades óticas das superfícies são definidas usando:

```
glMaterial*(surface_face, surface_property,  
            property_value);
```

- O sufixo pode ser *i*, *iv*, *f* ou *fv* dependendo do tipo de parâmetro
- O parâmetro `surface_face` indica a qual face o material se designa e pode ser:
 - `GL_FRONT`, `GL_BACK` e `GL_FRONT_AND_BACK`
- O parâmetro `surface_property` identifica o parâmetro da superfície e pode ser:
 - k_a , k_d , k_s ou n_s

Propriedades da Superfície

- Os flags `GL_AMBIENT`, `GL_DIFFUSE` e `GL_SPECULAR` são usados para definir os coeficientes de reflexão da superfície
 - Normalmente os valores dos coeficientes difuso e ambiente são os mesmos, para isso usamos `GL_AMBIENT_AND_DIFFUSE`
- Os valores padrão para os coeficientes são:
 - Luz ambiente (k_a): (0.2, 0.2, 0.2, 1.0)
 - Luz difusa (k_d): (0.8, 0.8, 0.8, 1.0)
 - Luz especular (k_s): (1.0, 1.0, 1.0, 1.0)
- Para definir o expoente de reflexão especular (n_s) usamos `GL_SHININESS` entre 0 e 128
 - O valor padrão é 0

Propriedades da Superfície

- Por exemplo, para definir os coeficientes de reflexão ambiente, difusa e especular podemos fazer:

```
float diffuse[4] = {0.65, 0.65, 0.0, 1.0};  
float specular[4] = {0.9, 0.9, 0.9, 1.0};  
float shininess = 65.0;  
  
glMaterialfv(GL_FRONT, GL_AMBIENT_AND_DIFFUSE, diffuse);  
glMaterialfv(GL_FRONT, GL_SPECULAR, specular);  
glMaterialf(GL_FRONT, GL_SHININESS, shininess);
```

Exemplo 4 – Propriedades da Superfície

```
void display(){
    //limpa o buffer
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

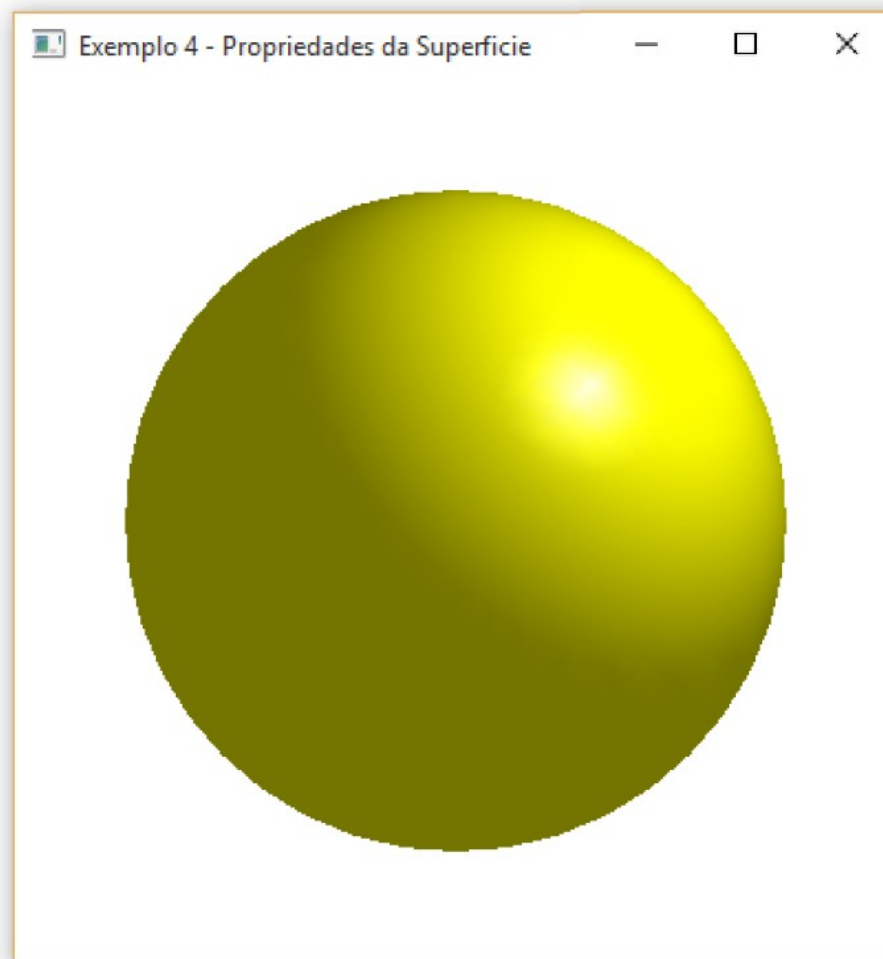
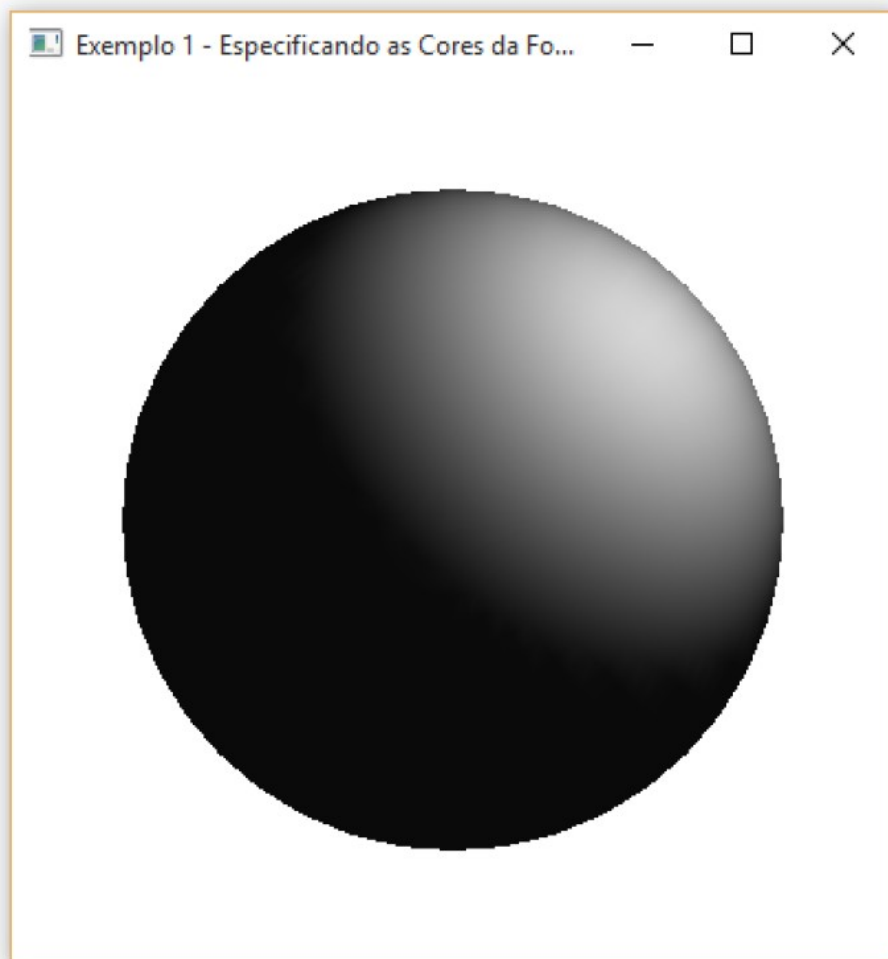
    //define material da superfície
    float kd[4] = {0.65f, 0.65f, 0.0f, 1.0f};
    float ks[4] = {0.9f, 0.9f, 0.9f, 1.0f};
    float ns = 65.0f;

    glMaterialfv(GL_FRONT, GL_AMBIENT_AND_DIFFUSE, kd);
    glMaterialfv(GL_FRONT, GL_SPECULAR, ks);
    glMaterialf(GL_FRONT, GL_SHININESS, ns);

    //define que a matrix é a a model view
    glMatrixMode(GL_MODELVIEW);
    glutSolidSphere(1.5, 40, 40);

    //força o desenho das primitivas
    glFlush();
}
```

Exemplo 4 – Propriedades da Superfície



- 1) Faça um programa que simule o brilho do Sol refletindo no objeto da tela. À medida que o ângulo do Sol for se aproximando de 180, a intensidade de sua luz deve ir reduzindo. Ao passar de 0 graus, deve voltar a aumentar a intensidade.
 - **Dica:** Para determinar a posição X e Y do Sol, utilize as coordenadas polares:
 - $X = \text{raio} * \cos(\text{angulo})$
 - $Y = \text{raio} * \sin(\text{angulo})$

Reflexão Especular e Modelo de Phong

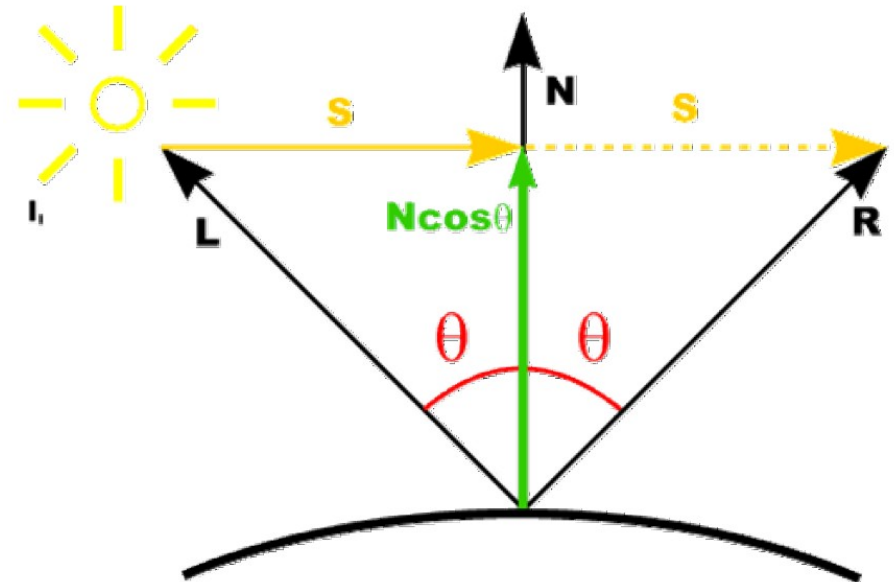
- A projeção de L na direção de N é $(N \cos \theta)$ (projeção escalar), então:

$$R - S = N \cos \theta$$

$$R = N \cos \theta + S$$

$$L + S = N \cos \theta$$

$$S = N \cos \theta - L$$



- De forma que o vetor de reflexão especular R é obtido fazendo:

$$R = N \cos \theta + N \cos \theta - L$$

$$R = 2N \cos \theta - L$$

$$R = N(2N \cdot L) - L$$