

UNIVERSIDADE ESTADUAL DO OESTE DO PARANÁ – UNIOESTE
CAMPUS UNIVERSITÁRIO DE CASCAVEL
CURSO DE CIÊNCIA DA COMPUTAÇÃO

Disciplina: Computação Gráfica.
Prof^o: Adair Santa Catarina.

TRABALHO 2 DE COMPUTAÇÃO GRÁFICA – TURMA 2023

Neste trabalho o acadêmico deverá desenvolver um modelador 3D elementar. A modelagem de objetos deverá ser realizada pela técnica de revolução, onde um perfil bidimensional é rotacionado em 360° ao redor de um dos eixos principais para construir um objeto tridimensional. Tome como exemplo a Figura 1, onde o perfil 2D (1a) foi rotacionado ao redor do eixo horizontal resultando no objeto 3D (1b).

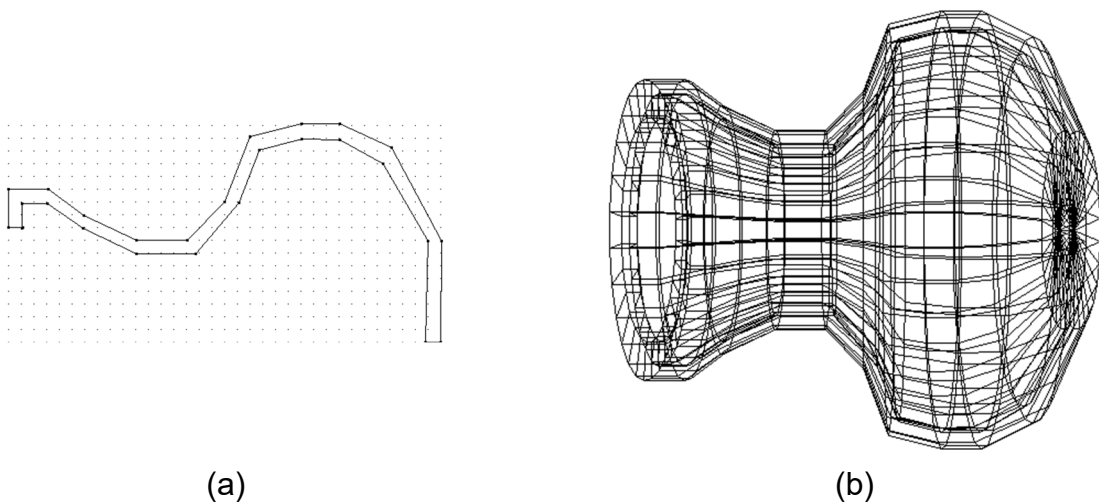
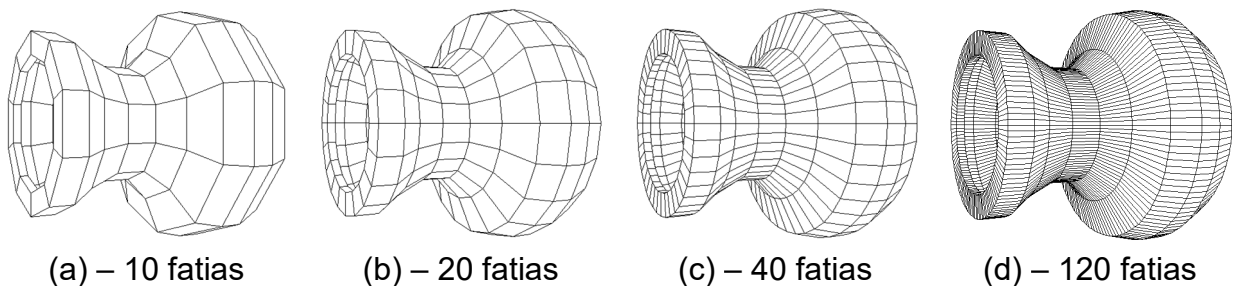


Figura 1 – Um perfil 2D (a) e o objeto 3D gerado pela revolução do perfil (b)

Note que, na Figura 1b, o objeto 3D está apresentado na técnica *wireframe*, sem ocultação de superfícies. O número de fatias usadas na revolução foi igual a 30, ou seja, a cada $360^\circ/30 = 12^\circ$ criou-se um paralelo para delimitar as faces do objeto. Outros exemplos de objetos 3D gerados com diferentes números de fatias são apresentados na técnica *wireframe*, com ocultação de superfícies, na Figura 2.



(a) – 10 fatias (b) – 20 fatias (c) – 40 fatias (d) – 120 fatias
Figura 2 – Objetos 3D gerados por revolução do mesmo perfil 2D com diferentes números de fatias

O objeto 3D pode, agora, ser operado com transformações geométricas, como a translação, escala e rotação, por exemplo. A figura 3 mostra o objeto 3D modificado por transformações geométricas.

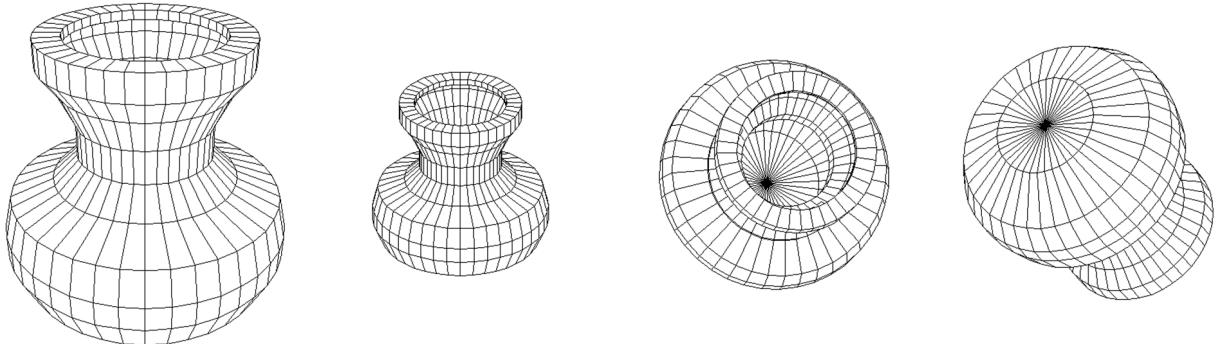
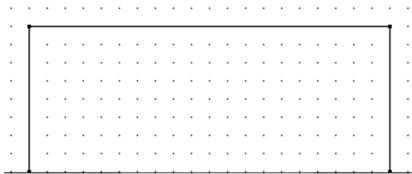
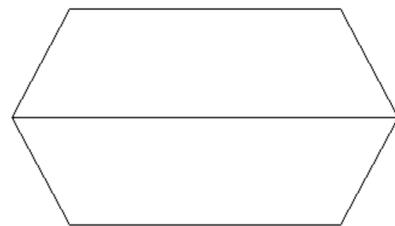


Figura 3 – O objeto 3D operado por diferentes transformações geométricas

Usando o processo de revolução é possível criar poliedros simples, como o prisma retangular apresentado na Figura 4b. O perfil 2D (Figura 4a) foi rotacionado em 4 fatias ($360^\circ/4=90^\circ$).



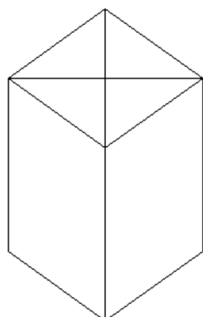
(a)



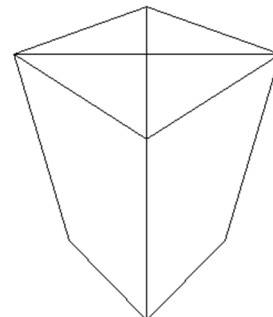
(b)

Figura 4 – O perfil 2D (a) empregado na criação de um prisma retangular (b)

A apresentação de um objeto em 3D, na tela do computador, exige a execução de um pipeline de operações gráficas. Neste pipeline devemos escolher o tipo de projeção que será empregada. Na Figura 5 o prisma retangular é apresentado em projeção paralela (Figura 5a) e em projeção perspectiva (Figura 5b).



(a)



(b)

Figura 5 – O prisma retangular desenhado em projeção paralela (a) e em projeção perspectiva (b)

Note que, na Figura 5a, todos os lados do prisma são paralelos, enquanto na Figura 5b é possível notar o efeito da projeção em perspectiva com um ponto de fuga. As arestas verticais perderam o paralelismo, levando à interpretação visual de que a face superior está mais próxima do observador que a face inferior do prisma.

Os processos de modelagem, transformações e projeções, demonstrados até o momento, usaram a representação *wireframe* para apresentação visual dos objetos 3D. Entretanto, para conferir aspecto mais realista aos objetos torna-se necessário colorizá-los ou sombreá-los. Três são os algoritmos tradicionais de sombreado local: constante, Gouraud e Phong. A Figura 6 mostra o vaso modelado anteriormente em sombreado constante (a) e em sombreado Gouraud (b).

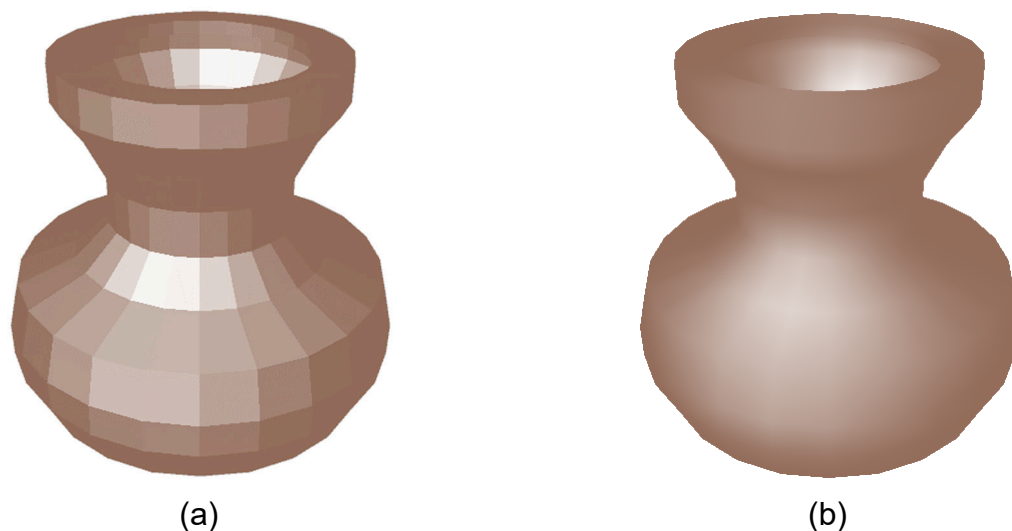


Figura 6 – Um vaso em sombreado constante (a) e em sombreado Gouraud (b)

No sombreado constante (Figura 6a) uma cor é calculada para cada face visível do objeto, sendo replicada em todos seus pixels, possibilitando visualizar as arestas entre as faces do objeto. No sombreado Gouraud (Figura 6b), uma cor é calculada para cada vértice do objeto e interpolada nos demais pixels da face. Como a interpolação de cores é linear, não se evidenciam as arestas limites entre as faces do objeto, conferindo ao objeto um aspecto contínuo e não mais uma aproximação poligonal.

Entretanto, o sombreado Gouraud, por interpolar cores calculadas nos vértices, torna-se incapaz de reproduzir o efeito especular que pode acontecer dentro de uma face do objeto. O efeito especular caracteriza-se por apresentar regiões de brilho concentrando em uma região restrita da face poligonal. A Figura 7 mostra um objeto em sombreado Gouraud e em sombreado Phong. Na Figura 7b, em sombreado Phong, é nítida a região com efeito especular na face frontal do objeto, o que não acontece na Figura 7a em sombreado Gouraud.

O sombreado Phong consegue representar o efeito especular porque a cor é calculada em cada pixel da face e não interpolada a partir da cor dos vértices. O que se interpola no sombreado Phong é um vetor normal à superfície em cada pixel, a partir de vetores normais médios unitários calculados nos vértices dos objetos. Usando este

vetor normal interpolado a cor pode ser calculada nos pixels, simulando uma curvatura das superfícies planares.

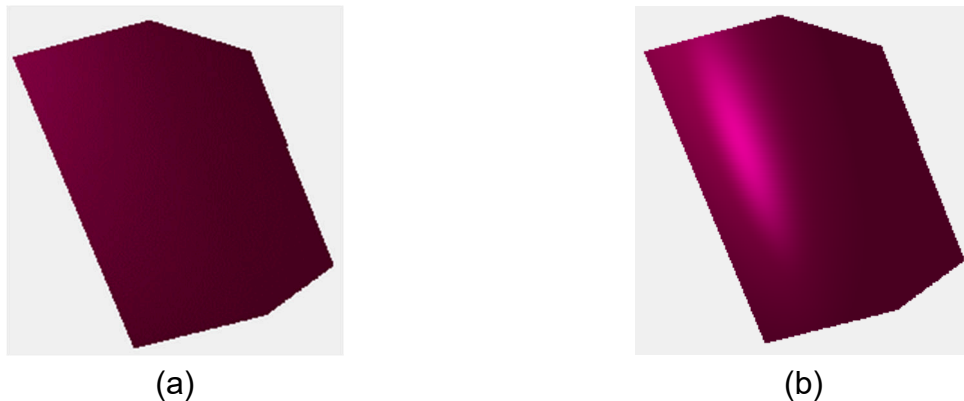


Figura 7 – Um prisma retangular em sombreado Gouraud (a) e Phong (b)

Neste trabalho o que se pede é a implementação de um software para:

- 1) Modelar objetos 3D a partir da revolução de perfis (geratrizes) 2D modelados pelo usuário, com diferentes números de fatias, apresentando-os na técnica *wireframe* com ocultação de superfícies (usar o algoritmo do pintor associado ao algoritmo da visibilidade pelo cálculo da normal);
- 2) Apresentar os objetos 3D modelados (dois ou mais objetos) em uma única janela em projeção paralela ou perspectiva, de acordo com a escolha do usuário do software. Use esta janela única para apresentar os objetos nos diferentes tipos de sombreado a serem implementados;
- 3) Transformar geometricamente o objeto (translações e rotações nos eixos x, y e z; escala uniforme, com o mesmo fator nos 3 eixos);
- 4) Apresentar o objeto 3D em sombreado constante, com ocultação de superfícies incluindo o algoritmo *z-buffer*;
- 5) Apresentar o objeto 3D em sombreado Gouraud, com ocultação de superfícies incluindo o algoritmo *z-buffer*;
- 6) Apresentar o objeto 3D em sombreado Phong (simplificado), com ocultação de superfícies incluindo o algoritmo *z-buffer*;
- 7) Todos os parâmetros relacionados com câmera, *window*, *viewport*, luzes e materiais que compõem os objetos devem ser editáveis pelo usuário do software em tempo de execução;
- 8) **Requisito fundamental:** Implementar o software usando o *pipeline* e algoritmos ensinados ao longo da disciplina (sigam a planilha de cálculos disponibilizada nas aulas práticas).

Data de entrega: 29/05/2024

A entrega deverá ser realizada através do e-mail prof.stacatarina@gmail.com. Agrupem todos os arquivos do código fonte em um único arquivo .zip e façam o envio. Não incluam no arquivo .zip arquivos binários (executáveis) pois o *Gmail* recusará o recebimento destas mensagens. Fiquem atentos à confirmação de recebimento que será

realizada pelo professor; caso não recebam confirmação de recebimento em até 48 h após o envio do trabalho entendam que o e-mail foi bloqueado por algum dos servidores envolvidos no envio/recepção da mensagem.

Incluam na mensagem de envio seu nome completo e as instruções para conseguir executar sua aplicação (Sistema operacional, compilador e versão, bibliotecas necessárias e como instalá-las, etc.). Caso seja por demais complexa a configuração de ambiente criem um repositório no *GitHub*, *Drive*, *OneDrive* ou outro serviço online e enviem o link onde seja possível baixar tudo que é necessário para sua aplicação funcione. Isto não exige a necessidade de enviar o arquivo .zip com o código fonte, apenas simplifica o processo de preparação do ambiente para execução de sua aplicação.

Critérios de Correção (Qualidade do software – 70%):

- a) Não cumpriu o requisito 8. Nota = 0 (zero pontos);
- b) Software é executado sem erros e atende completamente os requisitos 1 e 8. Nota = 10;
- c) Software é executado sem erros e atende completamente os requisitos 1, 2 e 8. Nota = 20;
- d) Software é executado sem erros e atende completamente os requisitos 1, 2, 3 e 8. Nota = 30;
- e) Software é executado sem erros e atende completamente os requisitos 1, 2, 3, 4, 7 e 8. Nota = 60;
- f) Software é executado sem erros e atende completamente os requisitos 1, 2, 3, 4, 5, 7 e 8. Nota = 80;
- g) Software é executado sem erros e atende completamente todos os requisitos. Nota = 100.

Critérios de Correção (Arguição – 30%):

Agendar horário para apresentação do trabalho com o professor (será disponibilizado um documento compartilhado para agendamento das apresentações, que deverão acontecer antes do dia 03/06/2024;

- a) Apresentação em dupla e oral – trazer a aplicação funcionando em computador próprio ou instalar a aplicação em um computador de laboratório do curso (responsabilidade dos apresentadores). Caso o computador ou a aplicação desenvolvida não esteja funcionando no início da apresentação agendada a nota da arguição será igual a 0 (zero);
- b) Responder corretamente e com a devida profundidade as questões realizadas pelo professor;
- c) A leitura de comentários do código fonte, como resposta às questões realizadas pelo professor será considerada insuficiente, o que levará a nota 0 (zero) na arguição.

Este trabalho poderá ser desenvolvido em dupla. Quaisquer dúvidas que persistirem poderão e deverão ser esclarecidas através de consultas ao professor da disciplina.