

## Aula – Computação Gráfica

### OpenGL Básico

Slides para uso pessoal e exclusivo durante o período de aula. Distribuição ou qualquer uso fora do escopo da disciplina é expressamente proibido.

1

1

#### Visão Geral

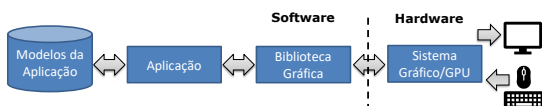
- Framework Gráfica Conceitual
- OpenGL
- GLUT
- Exemplos

2

2

#### Framework Gráfica Conceitual

- Biblioteca gráfica
  - Passo intermediário entre aplicação e os dispositivos
  - Aplicação mapeia seus objetos para visões (imagens)
    - Utiliza biblioteca gráfica
- Interação do usuário resulta em modificações nos modelos
- Essa framework tem mais de 4 décadas
  - Ainda é bem útil



3

## Framework Gráfica Conceitual

- Biblioteca gráfica
  - Exemplos
    - OpenGL™, DirectX™, Windows Presentation Foundation™ (WPF), RenderMan™, HTML5+WebGL
  - Primitivas
    - Caracteres, linhas, polígonos, meshes, etc.
  - Atributos
    - Cor, estilo de linha, material, etc.
  - Luz
  - Transformações

4

4

## OpenGL – O que é?

- Uma API para geração de gráficos 3D e 2D
- Ela provê
  - Primitivas vetoriais
  - Primitivas para imagens
- OpenGL
  - Gera imagens de alta qualidade
  - Usa aceleração gráfica (se disponível)
  - É independente de plataforma e do sistema de janelas
  - Não provê comunicação com o sistema de janelas

5

5

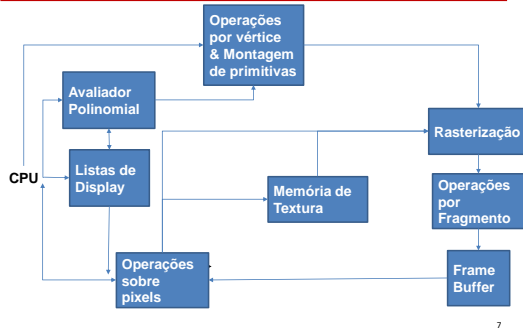
## Sistemas de Janela

- Provê interação humano-computador via janelas de aplicações
- As janelas ficam espalhadas na tela do computador
- Cada aplicação deve controlar/atualizar as suas janelas
- A interação do usuário com as janelas são convertidas em eventos
  - Ex. Arraste do mouse, Teclado pressionado, Janela minimizada, ...
- Aplicação recebe eventos através de rotinas **callback**
  - Redesenhar janela
  - Tratar clique do mouse
- Não existe padrão nas APIs dos Sistema de Janelas
  - MS Windows, LINUX, Apple
  - Pode-se usar uma biblioteca para padronizar a interface
    - Ex.: GLUT

6

6

## Arquitetura do OpenGL



7

## Como desenhar com OpenGL

- OpenGL utiliza o conceito de máquina de estados
- Ele tem funções para
  - Desenhar primitivas geométricas e imagens
  - Alterar/Consultar variáveis de estado
    - Ex.: cor, material, iluminação, etc
- OpenGL está sempre evoluindo
  - Extensões são padronizadas
  - Novas versões são estabelecidas por um comitê
    - Envolve usuários e fabricantes

8

8

## APIs Relacionadas

- GLU (OpenGL Utility Library)
  - Vem com o padrão OpenGL
  - Fornece primitivas mais avançadas
    - NURBS, trianguladores, quádras, etc.
- GLUT (OpenGL Utility Toolkit)
  - API que faz interface com os sistemas de janelas
  - Abstrai camadas proprietárias
  - Boa para prototipação, mas limitada para aplicações reais
  - Não é parte do OpenGL

9

9

## Estrutura de um Programa OpenGL/GLUT

```
#include <GL/glut.h>           Cabeçalho
/* Outros headers */

void display (void) {
    ...
}                               Callbacks
/* Outras */

int main (int argc, char *argv[]) {
    glutInit (&argc, argv);      Inicialização do GLUT
    glutInitDisplayMode( modo );  Inicialização da
    glutCreateWindow( nome_da_janela ); janela da aplicação
    glutDisplayFunc( display );   Registro das
    glutReshapeFunc( reshapeCallback ); Callbacks
    /* Registro de outras rotinas callback */
    glutMainLoop();              Loop principal
    return 0;
}
```

10

10

## GLUT – Registrando Callbacks

- Callbacks são chamadas quando eventos ocorrem.
- A call-back precisa ser registrada para ser chamada no loop
  - Padrão de nome do GLUT  
glut{Nome do evento}Func (callback)
  - Exemplo da do evento responsável por desenhar a janela  
glutDisplayFunc (Desenho);

11

11

## GLUT – Callback de desenho

- É disparada sempre que se precisa desenhar a janela
  - Ex. automaticamente ao maximizar a janela
  - Ex. a pedido do programador com o comando
    - glutPostRedisplay()
- Obrigatorio ter uma se utilizar o GLUT

12

12

## GLUT - Callbacks

- Outras callbacks como exemplo
  - De teclado
 

```
void keyboard (unsigned char key, int x, int y)
```
  - De mouse
 

```
void mouse(int button, int state, int x, int y)
void motion(int x, int y)
void passiveMotion(int x, int y)
```
  - Eventos ociosos (nenhum evento ocorrendo)
 

```
void idle (void)
```

13

13

## GLUT - Inicialização

- Inicialização do GLUT
 

```
glutInit (int* argc, char** argv)
```

  - Permite passar comando para o GLUT

14

14

## Inicialização das Janelas

- `glutInitDisplayMode (int modo)`
  - As janelas possuem recursos que podem ou não ser ativados
  - O parâmetro *Modo* controla os recursos da janela criada
  - Ele é um “ou” bit-a-bit de constantes:
    - GLUT\_RGB cores dos pixels serão expressos em RGB
    - GLUT\_DOUBLE bufferização dupla (ao invés de simples)
    - GLUT\_DEPTH buffer de profundidade (z-buffer)
    - GLUT\_ACCUM buffer de acumulação
    - GLUT\_ALPHA buffer de cores terá componente alfa
- `glutInitWindowPosition (int x, int y)`
  - Define a posição inicial do canto superior esquerdo da janela
- `glutInitWindowSize (int width, height)`
  - Define o tamanho da janela em pixels

15

15

## Inicialização das Janelas

- 
- `int glutCreateWindow (char* nome)`
    - Cria uma nova janela
    - `nome` define o nome para rotular a janela
    - O retorno inteiro identifica a janela

16

16

## Inicialização das Janelas

- 
- `int glutCreateWindow (char* nome)`
    - Cria uma nova janela
    - `nome` define o nome para rotular a janela
    - O retorno inteiro identifica a janela

17

17

## Loop Principal

- 
- Antes do loop principal
    - Pode-se inicializar parâmetros estáticos do sistema
      - Ex. Cor do fundo, Tipo de sombreado de desejado, ...
  - Avaliação da fila de eventos é feita com:
    - `glutMainLoop (void)`
    - Ela “nunca” retorna

18

18

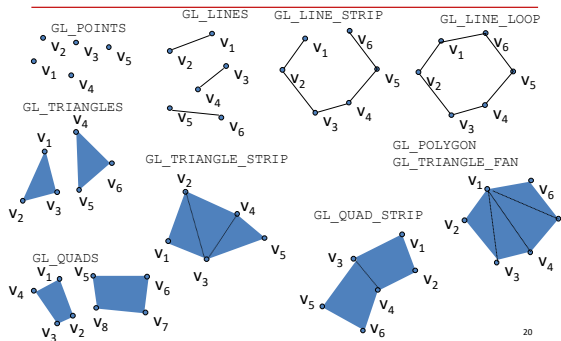
## OpenGL – Primitivas de desenho

- Entre `glBegin()` e `glEnd()` apenas alguns comandos podem ser usados. Ex.:
  - `glMaterial`
  - `glNormal`
  - `glTexCoord`
  - `glVertex`
    - Ex. `glVertex3f(0.25, 0.75, 0.0);`
- O vértice assume as propriedades das variáveis de estado atuais
  - Portanto, antes de chamar o vértice escolha
    - cor, material, normal, etc

19

19

## OpenGL – Primitivas de desenho



20

20

## Perguntas ?????

21

21