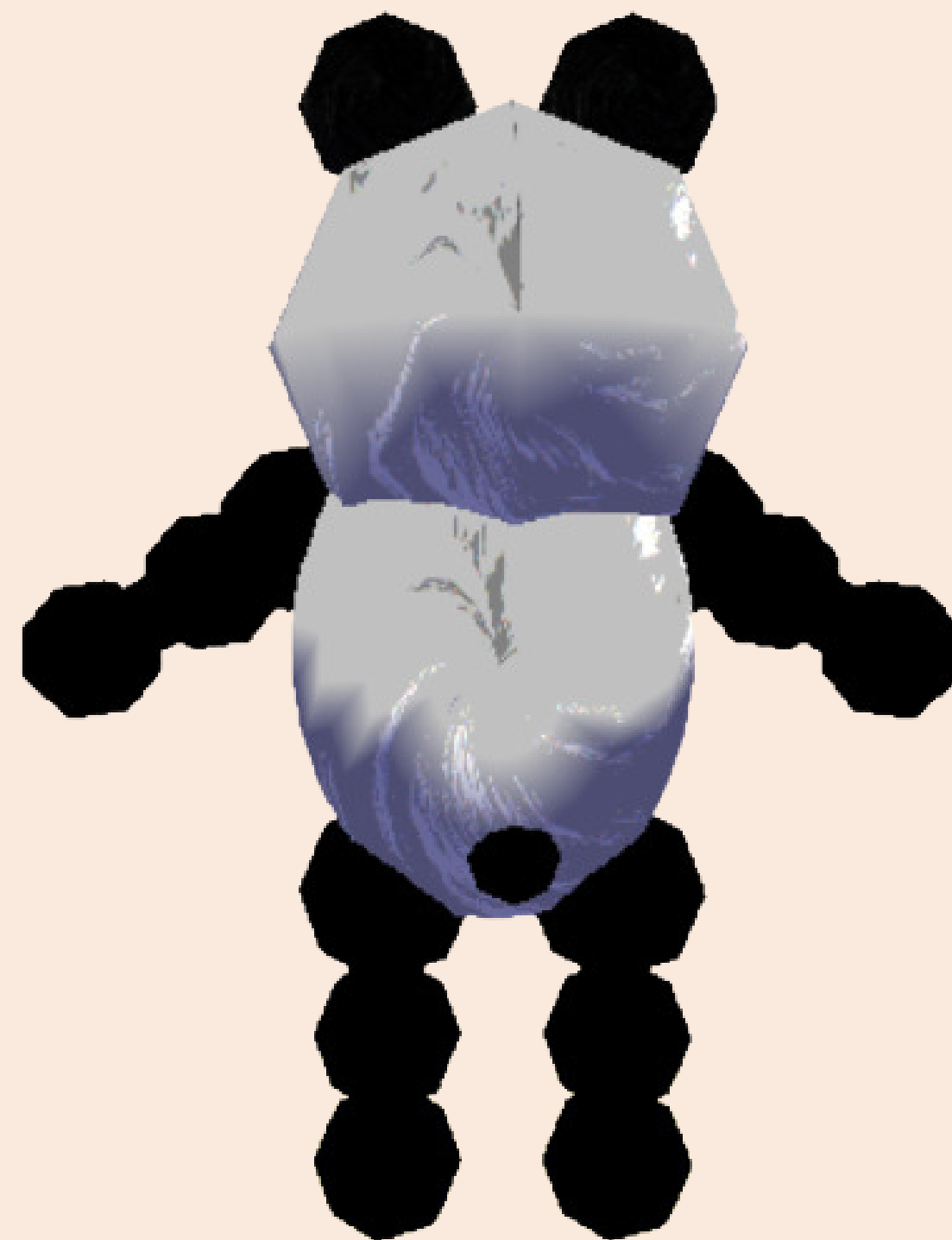


animal
PANDA

Projeto final da disciplina de Computação Gráfica
Acadêmica: Barbara Rodrigues
Universidade Tecnológica Federal do Paraná



Estrutura corporal



Estrutura corporal

Panda::Desenha(){
braco esquerdo
cabeça
braco direito
orelha esquerda
orelha direita
rabinho
perna esquerda
perna direita
tronco
}

ORELHA

```
glEnable(GL_TEXTURE_2D);  
glBindTexture(GL_TEXTURE_2D, texture[1]);  
gluQuadricTexture(q, true);  
glTexCoord2f(0.0f, 0.0f);  
    glTranslatef(-3 * grossura, 4 * grossura, 0.0);  
    gluSphere(q, 1.5 * grossura, 8, 8);  
glDisable(GL_TEXTURE_2D);
```

Cabeça, tronco, orelhas e rabinho: os sólidos foram criados no próprio desenha()

RABINHO

```
glEnable(GL_TEXTURE_2D);  
glBindTexture(GL_TEXTURE_2D, texture[1]);  
glTexCoord2f(1.0f, 0.0f);  
    glTranslatef(-1, -8 * grossura, -2);  
    glutSolidSphere(0.75 * grossura, 8, 8);  
glDisable(GL_TEXTURE_2D);
```

CABEÇA

```
glEnable(GL_TEXTURE_2D);  
glBindTexture(GL_TEXTURE_2D, texture[0]);  
GLUquadricObj *q = gluNewQuadric();  
gluQuadricTexture(q, true);  
    glTranslatef(-1 * grossura, 0.0, 0.0);  
    gluSphere(q, 4 * grossura, 8, 8);  
glTexCoord2f(0.0f, 0.0f);  
glDisable(GL_TEXTURE_2D);
```

Estrutura corporal



```
Panda::Desenha(){  
    braco esquerdo  
    cabeça  
    braco direito  
    orelinha esquerda  
    orelinha direita  
    rabinho  
    perna esquerda  
    perna direita  
    tronco  
  
}
```

Braços e pernas: foram criados em classes separadas e chamadas no Desenha().

CLASSE BRAÇO

```
class Braco  
{  
public:  
    Braco(float comprimento, float largura);  
    void desenha() { a.desenha(); }  
    void setCurvatura(float curvatura, bool reto, int tipo);  
    float getCurvatura() { return a.getAngulo() * 100 / 90; }  
    void setBracoReto(float curvatura);  
    float getBracoReto() { return a.getAngulo() * 100 / 90; }  
  
protected:  
    Osso a, b;  
};
```

CLASSE PERNA

```
class Perna  
{  
public:  
    Perna(float comprimento, float largura);  
    void desenha() { a.desenha(); }  
    void setCurvatura(float curvatura, bool reto, int tipo);  
    float getCurvatura() { return a.getAngulo() * 100 / 90; }  
    void setBracoReto(float curvatura);  
    float getBracoReto() { return a.getAngulo() * 100 / 90; }  
  
protected:  
    Osso a, b;  
};
```

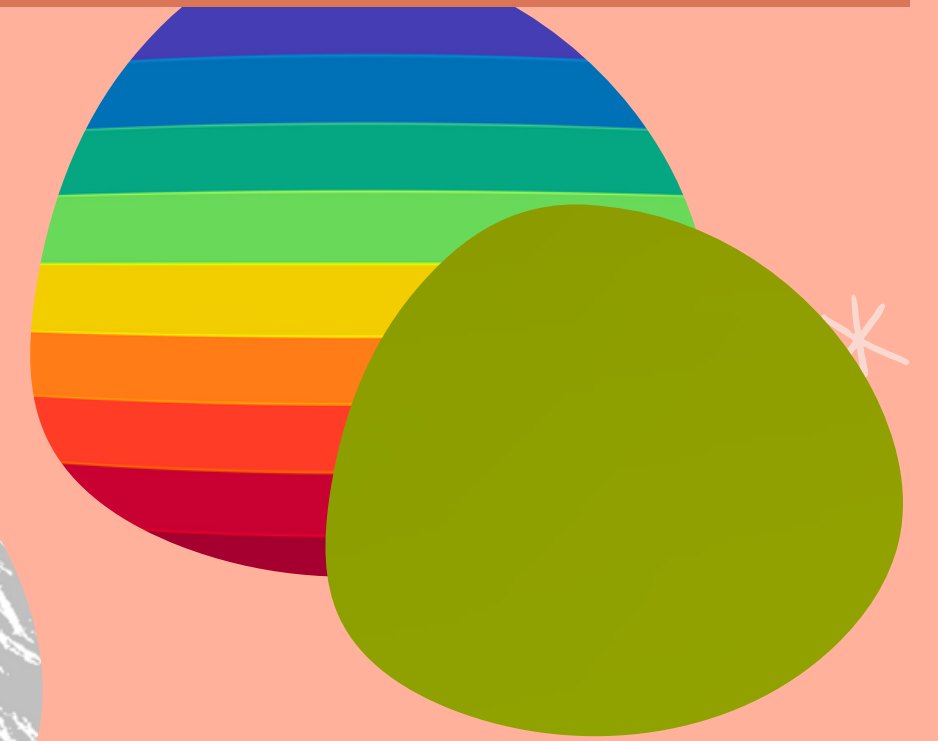
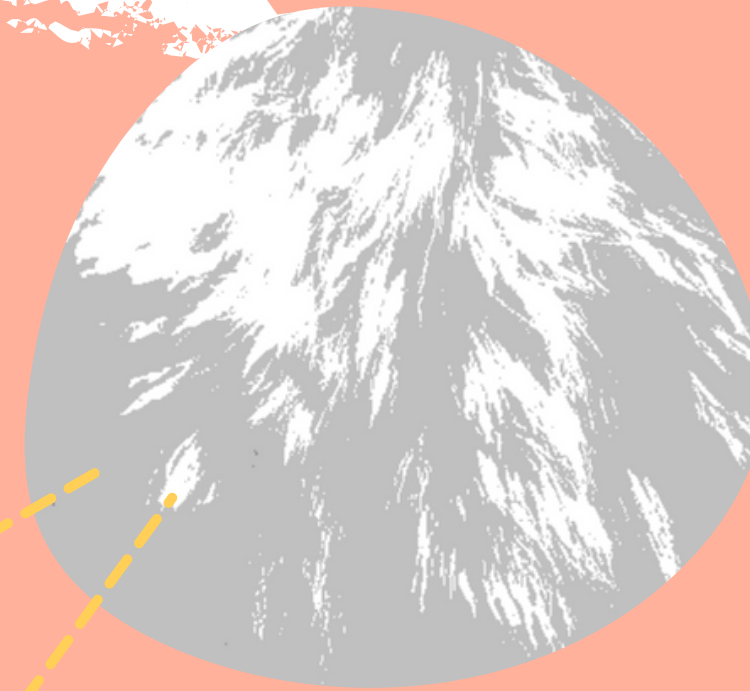
CHAMADA

```
glEnable(GL_TEXTURE_2D);  
glBindTexture(GL_TEXTURE_2D, texture[1]);  
glTexCoord2f(1.0f, 0.0f);  
glTranslatef(-4.0 * grossura, -3, 1);  
    glRotatef(120, 0.0, 0.0, 1.0);  
    glRotatef(-20, 0.0, 1.0, 0.0);  
    glRotatef(curvatura[0] * 0.9, 1.0, 0.0, 0.0);  
    glRotatef(curvaturaZ[0] * 0.9, 0.0, 0.0, 1.0);  
    glutSolidSphere(1.5 * grossura, 8, 8);  
bEsquerdo.desenha();  
glutSolidSphere(1.5 * grossura, 8, 8);  
glDisable(GL_TEXTURE_2D);
```

```
Braco bEsquerdo;  
Braco bDireito;  
Perna pEsquerda;  
Perna pDireita;
```

Textura e iluminação

CHÃO: EXPECTATIVA VS REALIDADE



ILUMINAÇÃO

```
GLfloat diffuseLight[] = {1.0f, 1.0f, 1.0f, 1.0};
GLfloat ambientLight[] = {0.2f, 0.2f, 0.4f, 1.0};
GLfloat lightPos[] = {0.0f, 500.0f, 100.0f, 1.0f};
glEnable(GL_LIGHTING);
glLightfv(GL_LIGHT0, GL_AMBIENT, ambientLight);
glLightfv(GL_LIGHT0, GL_SPECULAR, diffuseLight);
glLightfv(GL_LIGHT0, GL_POSITION, lightPos);
glEnable(GL_LIGHT0);
glEnable(GL_COLOR_MATERIAL);
glColor3f(1.0f, 1.0f, 1.0f);
glMaterialfv(GL_FRONT_AND_BACK, GL_SPECULAR, diffuseLight);
glMateriali(GL_FRONT_AND_BACK, GL_SHININESS, 0);
```


Movimentação

Lista de comandos para os movimentos:

q - abrir braço esquerdo

a - fechar braço esquerdo

w - abrir braço direito

s - fechar braço direito

e - mexer perna esquerda para frente

d - mexer perna esquerda para trás

r - mexer perna direita para frente

f - mexer perna direita para trás

h - home

m - corre

n - anda

b - equilibra

v - luta

MOVIMENTOS

```
void home();  
void deita();  
void anda();  
void corre();  
void equilibra();  
void luta();
```

Os movimentos trabalhados estão centrados nos braços e pernas. Alguns trabalham os ângulos do panda em relação a janela, para dar a sensação de movimento real.

