

# 計算機実験 2 ～ OpenGLの使い方 ～ 7月1日

高田 雅美@E359  
takata@ics.nara-wu.ac.jp

```

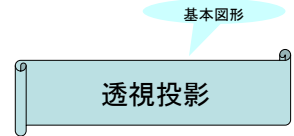
#include <GL/glu.h>
void display(void){
    glClear(GL_COLOR_BUFFER_BIT);
    glPushMatrix();
    glTranslatef( 0.0, 0.0, -3.0 );
    glColor3f( 1.0, 1.0, 1.0 );

    glEnd();
    glPopMatrix();
    glFlush();
}

void myInit( char *programe){
    int width = 500, height = 500;
    float aspect = (float)width / (float)height;
    glutInitWindowPosition( 0, 0 );

    glLoadIdentity();
    gluPerspective( 45.0, aspect, 1.0, 7.0 );
    glMatrixMode( GL_MODELVIEW );
}

```



```

#include <GL/glu.h>
void display(void){
    glClear(GL_COLOR_BUFFER_BIT);
    glPushMatrix();
    glTranslatef( 0.0, 0.0, -3.0 );
    glColor3f( 1.0, 1.0, 1.0 );

    glEnd();
    glPopMatrix();
    glFlush();
}

void myInit( char *programe){
    int width = 500, height = 500;
    float aspect = (float)width / (float)height;
    glutInitWindowPosition( 0, 0 );

    glLoadIdentity();
    gluPerspective( 45.0, aspect, 1.0, 7.0 );
    glMatrixMode( GL_MODELVIEW );
}

```

ヘッダファイルの追加  
透視投影は上位関数

```

#include <GL/glu.h>
void display(void){
    glClear(GL_COLOR_BUFFER_BIT);
    glPushMatrix();
    glTranslatef( 0.0, 0.0, -3.0 );
    glColor3f( 1.0, 1.0, 1.0 );

    glEnd();
    glPopMatrix();
    glFlush();
}

void myInit( char *programe){
    int width = 500, height = 500;
    float aspect = (float)width / (float)height;
    glutInitWindowPosition( 0, 0 );

    glLoadIdentity();
    gluPerspective( 45.0, aspect, 1.0, 7.0 );
    glMatrixMode( GL_MODELVIEW );
}

```

透視投影の設定  
GluPerspective

- 第一引数 fovy
  - ◆ 視点の角度
- 第二引数 aspect=w/h
  - ◆ 断面幅と高さの比
- 第三引数 near
  - ◆ 視点から頂面の距離
- 第四引数 far
  - ◆ 視点から底面の距離

```

#include <GL/glu.h>
void display(void){
    glClear(GL_COLOR_BUFFER_BIT);
    glPushMatrix();
    glTranslatef( 0.0, 0.0, -3.0 );
    glColor3f( 1.0, 1.0, 1.0 );

    glEnd();
    glPopMatrix();
    glFlush();
}

void myInit( char *programe){
    int width = 500, height = 500;
    float aspect = (float)width / (float)height;
    glutInitWindowPosition( 0, 0 );

    glLoadIdentity();
    gluPerspective( 45.0, aspect, 1.0, 7.0 );
    glMatrixMode( GL_MODELVIEW );
}

```

glPushMatrix()  
描画図形をスタックへ  
glTranslatef(...)  
図形の幾何変換  
基準位置の平行移動  
glPopMatrix()  
スタックから取り出し

幾何関数

- 移動させたい描画図形の前に定義
- 平行移動
  - glTranslatef( tx, ty, tz )
  - tx:x軸方向の移動, ty:y軸方向, tz:z軸方向
- 回転
  - glRotatef( θ, x, y, z )
  - θ:回転角度, xyz:回転軸のベクトル
- 拡大・縮小
  - glScalef( sx, sy, sz )
  - 拡大率(負の値=座標系の反転)

### 課題3

- 透視投影法
- 各軸の最小値 -100
- 各軸の最大値 100
- X軸とY軸を描く(黒色)
- 正六角形
  - ◆1つ目 原点を中心に描く
  - ◆2つ目 1つ目に対して
    - ◆正面に対して30度回転
    - ◆中心から20X軸移動
    - ◆中心から20Y軸移動

```
void display(void){
    glColor3f( 1.0, 1.0, 1.0);
    glutSolidSphere( 1.0, 20, 20 );
    glColor3f(0.0, 1.0, 0.0);
    glRectf( -1.0, -1.5, 1.0, 1.5);
    glPopMatrix();
}
```

透視投影

プリミティブ関数

#### 定義済みの図形

- glutSolidSphere( ... )
  - ◆プリミティブ関数
  - ◆球体を描く
  - ◆球の半径, 描画分割数
- glRectf( ... )
  - ◆四角形を描く
  - ◆x, y, x, y

```
void display(void){
    glColor3f( 1.0, 1.0, 1.0);
    glutSolidSphere( 1.0, 20, 20 );
    glColor3f(0.0, 1.0, 0.0);
    glRectf( -1.0, -1.5, 1.0, 1.5);
    glPopMatrix();
}
```

#### プリミティブ関数

- glutWire\*
  - ◆ワイヤ(輪郭線)のみで描画される
- glutSolid\*
  - ◆面で描画される
- ◆ \*
- ◆ 球: Sphere( 半径, 緯度分割数, 経度分割数 )
- ◆ 立方体: Cube( 一辺の長さ )
- ◆ トーラス: Torus(内半径, 外半径, 分割数, 分割数)
- ◆ 円錐: Cone( 半径, 高さ, 分割数, 分割数 )
- ◆ 円柱: Cylinder( 半径, 高さ, 分割数 )
- ◆ ティーポット: Teapot( サイズ )

### 課題4

- ワイヤーのティーポット
- ソリッドの円錐
  - ◆円錐であることがわかるように回転
- 2つの図が重ならないように配置