



2007年12月14日

第23回 OpenGL によるグラフィックス (12)

情報処理論 (応用)

松山大学 経営学部

檀 裕也

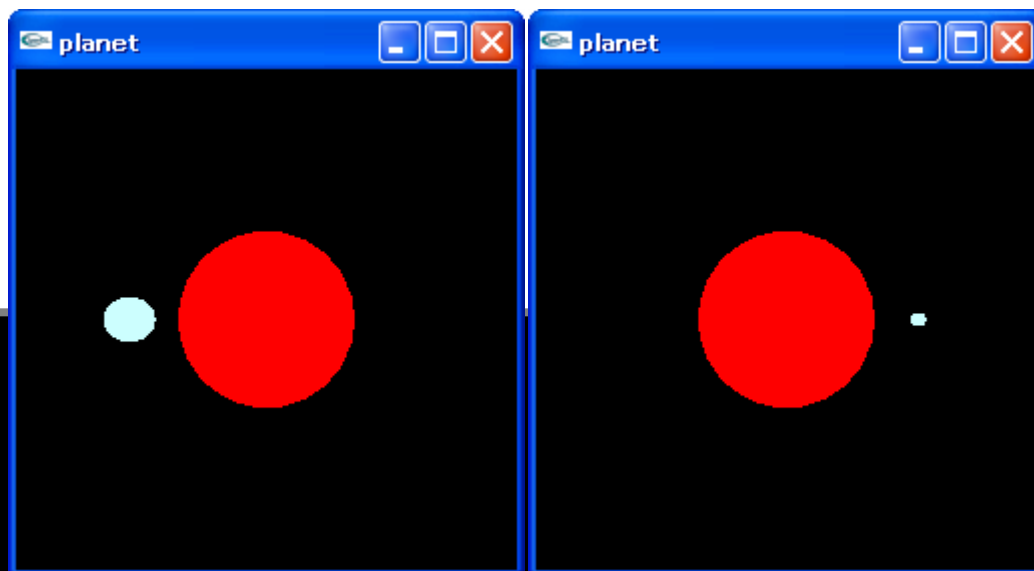
<http://www.cc.matsuyama-u.ac.jp/~dan/education/application/>



前回の実習課題

- 太陽の周りを地球が公転する様子をCGで表現せよ。
 - 宛先: dan@cc.matsuyama-u.ac.jp
 - 件名: CG課題#11

```
C:¥>planet.exe
```



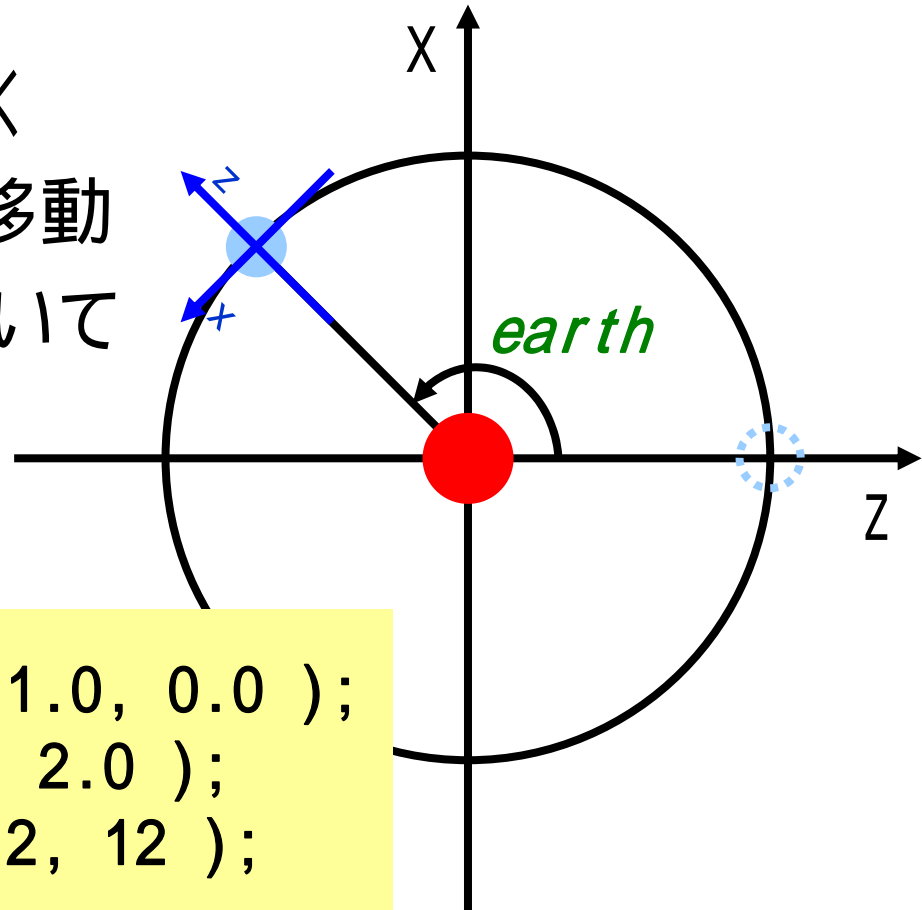


考えかた

- 地球の描画

- 原点に球面を描く
- z 軸方向に平行移動
- y 軸のまわりについて回転させる

ローカル座標系



```
glRotatef( earth, 0.0, 1.0, 0.0 );  
glTranslatef( 0.0, 0.0, 2.0 );  
glutSolidSphere( 0.1, 12, 12 );
```



解答例 (1 / 4)

planet.c

```
#include <stdlib.h>
#include <GL/glut.h>
#include <GL/gl.h>
#include <GL/glu.h>
#include <math.h>

double ey = 0.0;
double ez = 3.0;
double theta = 0.3;
double earth = 0.0;

void display( void )
{
    glClear( GL_COLOR_BUFFER_BIT );

    glLoadIdentity();
    gluLookAt( ez * sin( theta ), ey, ez * cos( theta ),
              0.0, 0.0, 0.0, 0.0, 1.0, 0.0 );

    glColor3f( 1.0, 0.0, 0.0 );
    glutSolidSphere( 1.0, 24, 24 );    // 太陽を描く
}
```



解答例 (2 / 4)

```
glRotatef( earth, 0.0, 1.0, 0.0 );  
glTranslatef( 0.0, 0.0, 2.0 );  
glColor3f( 0.8, 1.0, 1.0 );  
glutSolidSphere( 0.1, 12, 12 ); // 地球を描く  
  
glFlush();  
}
```

```
void myKeyBoard( unsigned char key, int x, int y )  
{  
    switch( key ){  
        case 0x1B:  
            exit( 0 );  
        case 'y':  
            ey -= 1.0;  
            break;  
        case 'Y':  
            ey += 1.0;  
            break;  
        case 'z':  
            ez -= 0.2;  
            break;
```



解答例 (3 / 4)

```
case 'Z':
    ez += 0.2;
    break;
case 'r':
    theta -= 0.1;
    break;
case 'R':
    theta += 0.1;
    break;
case 'e':
    earth -= 5.0;
    break;
case 'E':
    earth += 5.0;
    break;
}
glutPostRedisplay();
}

int main( int argc, char *argv[] )
{
    glutInit( &argc, argv );
```



解答例 (4 / 4)

```
glutInitDisplayMode( GLUT_RGB );  
glutInitWindowSize( 250, 250 );  
glutCreateWindow( argv[0] );  
  
glClearColor( 0.0, 0.0, 0.0, 0.0 );  
glMatrixMode( GL_PROJECTION );  
glLoadIdentity();  
glFrustum( -1.0, 1.0, -1.0, 1.0, 1.0, 20.0 );  
glMatrixMode( GL_MODELVIEW );  
  
glutDisplayFunc( display );  
glutKeyboardFunc( myKeyBoard );  
glutMainLoop();  
  
return 0;  
}
```



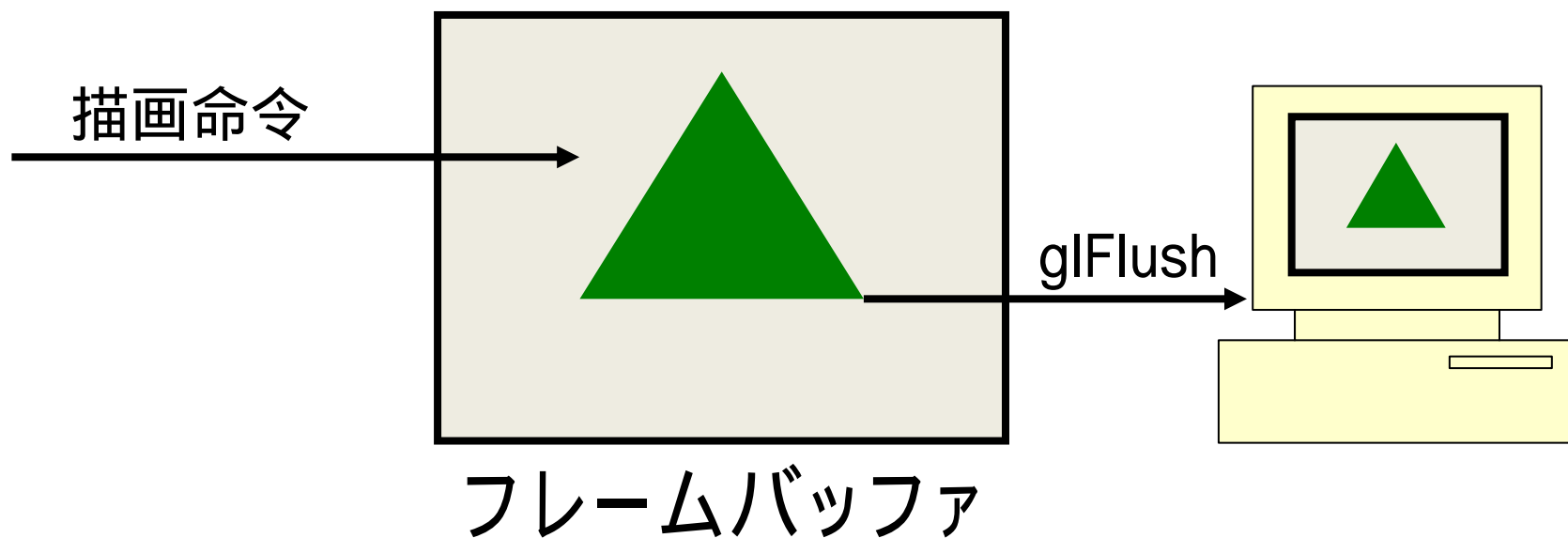
今回の授業内容

- OpenGL によるグラフィックス (12)
 - zバッファ法
 - 座標系の保存と復元
太陽と地球と月
 - 開発環境の構築



フレームバッファ

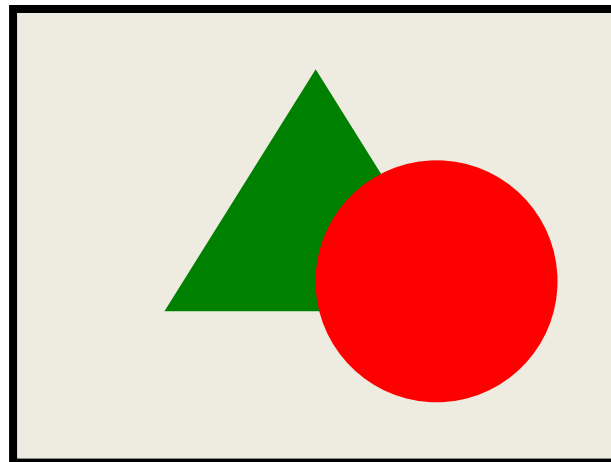
- 画面上の点に対応する色情報を保存する
 - 描画命令でフレームバッファに色情報の書き込み
 - glFlush でフレームバッファの内容を画面に表示





zバッファ法

- 視点から描画物体までの距離を記憶しておく
- 2つ以上の物体が重なる場合、
 - 近いもの（見えるもの）を表示する
 - 遠いもの（見えないもの）は表示しない





ソースコードの記述

- main 関数

```
glutInitDisplayMode( GLUT_RGB | GLUT_DEPTH );
```

- display 関数

```
glClear( GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT );
```

```
glEnable( GL_DEPTH_TEST );
```

描画命令

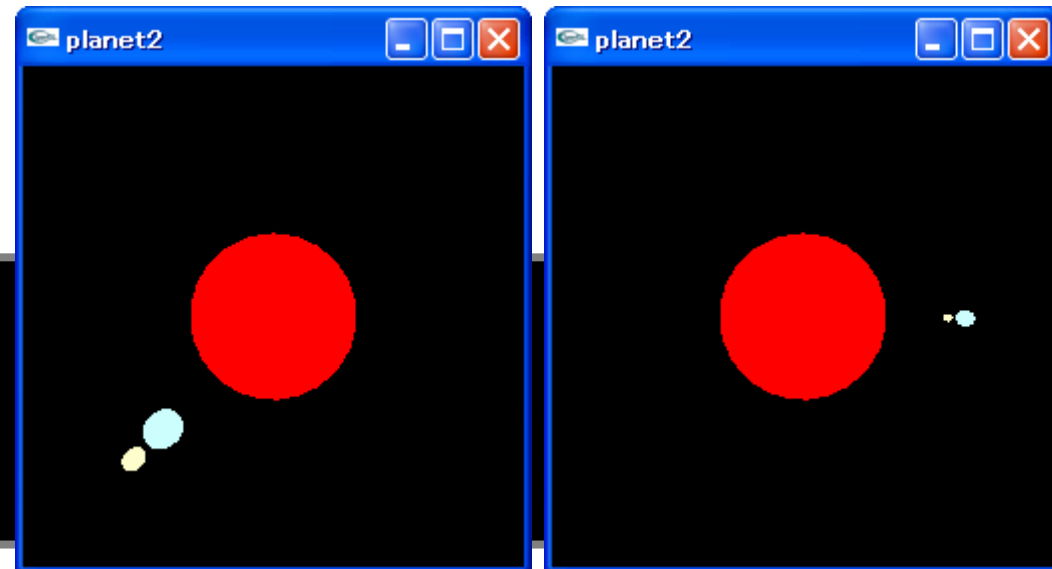
```
glDisable( GL_DEPTH_TEST );
```



実習課題

- 太陽の周りを地球が公転し、さらに地球の周りを月が公転する様子をCGで表現せよ。
 - 宛先: dan@cc.matsuyama-u.ac.jp
 - 件名: CG課題#12

```
C:¥>planet2.exe
```





次回の予定

- 日時： 12月19日（水）
5時限
- OpenGL によるグラフィックス
– モデリング（続き）