



2007年12月12日

## 第22回 OpenGL によるグラフィックス (11)

# 情報処理論 (応用)

松山大学 経営学部

檀 裕也

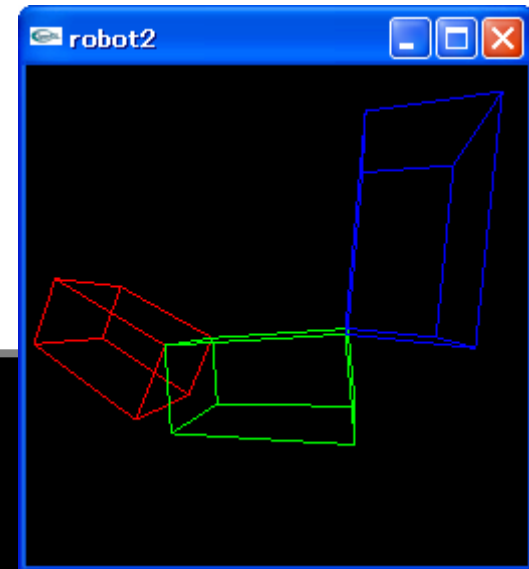
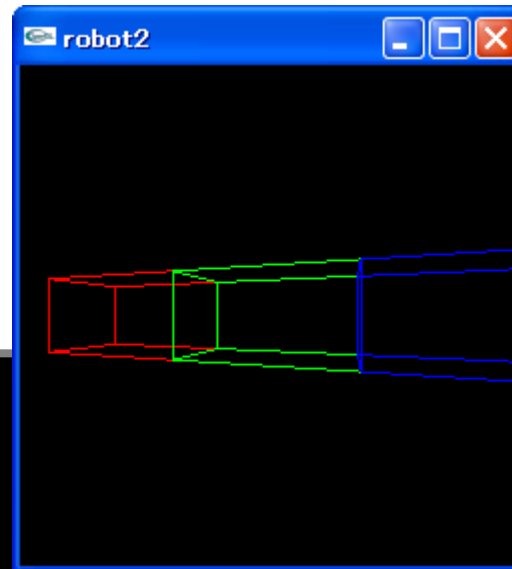
<http://www.cc.matsuyama-u.ac.jp/~dan/education/application/>



# 前回の実習課題

- 例題のプログラムにブロックを1つ追加せよ。
  - 宛先: dan@cc.matsuyama-u.ac.jp
  - 件名: CG課題#10

```
C:¥>robot2.exe
```





# 解答例 ( 1 / 6 )

**robot2.c**

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

double z = 3.0;
double theta = 0.3;
double shoulder = 0;
double elbow = 0;
double finger = 0;

void display( void )
{
    glClear( GL_COLOR_BUFFER_BIT );
    glLoadIdentity();
    gluLookAt( z * sin( theta ), 0.0, z * cos( theta ),
              0.0, 0.0, 0.0, 0.0, 1.0, 0.0 );

    glTranslatef( -2.0, 0.0, 0.0 );

    glColor3f( 1.0, 0.0, 0.0 );
```



# 解答例 (2 / 6)

```
glTranslatef( -1.0, 0.5, 0.0 );  
glRotatef( shoulder, 0.0, 0.0, 1.0 );  
glTranslatef( 1.0, -0.5, 0.0 );  
glPushMatrix();  
    glScalef( 2.0, 1.0, 1.0 );  
    glutWireCube( 1.0 );  
glPopMatrix();
```

```
glTranslatef( 2.0, 0.0, 0.0 );
```

```
glTranslatef( -1.0, 0.5, 0.0 );  
glRotatef( elbow, 0.0, 0.0, 1.0 );  
glTranslatef( 1.0, -0.5, 0.0 );  
glColor3f( 0.0, 1.0, 0.0 );  
glPushMatrix();  
    glScalef( 2.0, 1.0, 1.0 );  
    glutWireCube( 1.0 );  
glPopMatrix();
```

```
glTranslatef( 2.0, 0.0, 0.0 );
```

```
glTranslatef( -1.0, 0.5, 0.0 );
```



# 解答例 (3 / 6)

```
glRotatef( finger, 0.0, 0.0, 1.0 );
glTranslatef( 1.0, -0.5, 0.0 );
glColor3f( 0.0, 0.0, 1.0 );
glPushMatrix();
    glScalef( 2.0, 1.0, 1.0 );
    glutWireCube( 1.0 );
glPopMatrix();

glFlush();
}
```

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



# 解答例 (4 / 6)

```
case 'r':
    theta -= 0.1;
    break;
case 'R':
    theta += 0.1;
    break;
case 's':
    shoulder -= 5.0;
    break;
case 'S':
    shoulder += 5.0;
    break;
case 'e':
    if( elbow >= 5.0 ){
        elbow -= 5.0;
    }
    break;
case 'E':
    if( elbow <= 180.0 - 5.0 ){
        elbow += 5.0;
    }
    break;
```



# 解答例 (5 / 6)

```
case 'f':
    if( finger >= 5.0 ){
        finger -= 5.0;
    }
    break;
case 'F':
    if( finger <= 180.0 - 5.0 ){
        finger += 5.0;
    }
    break;
}
glutPostRedisplay();
}

int main( int argc, char *argv[] )
{
    glutInit( &argc, argv );
    glutInitDisplayMode( GLUT_RGB );
    glutInitWindowSize( 250, 250 );
    glutCreateWindow( argv[0] );

    glClearColor( 0.0, 0.0, 0.0, 0.0 );
```



# 解答例 (6 / 6)

```
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

---

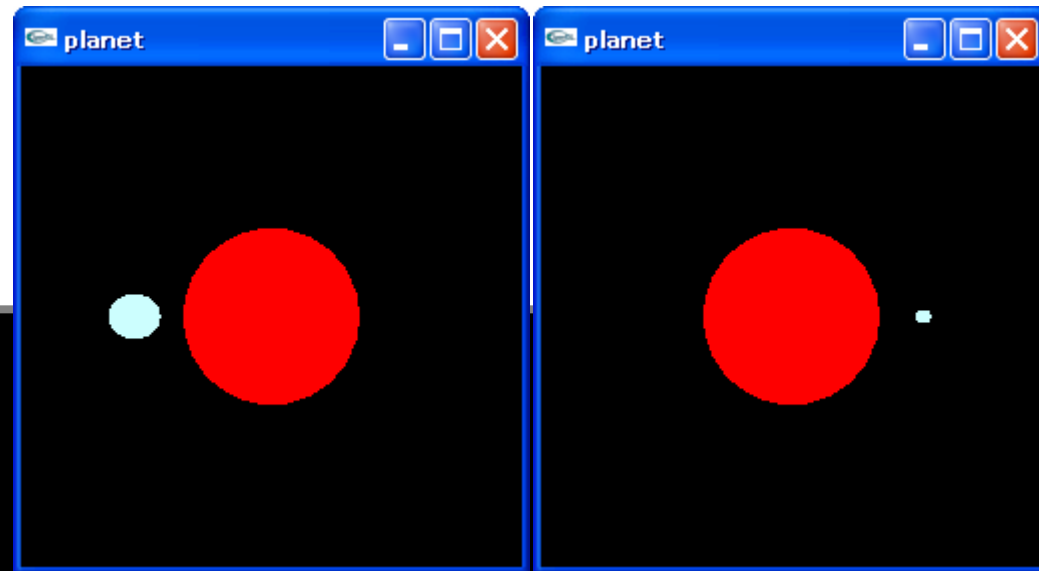
- OpenGL によるグラフィックス (11)
  - 座標系の保存と復元  
太陽系のシミュレーション
  - 開発環境の構築



# 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
```





# 次回の予定

---

- 日時： 12月14日（金）  
4時限
- OpenGL によるグラフィックス  
– モデリング（続き）