

生産情報システム工学

#02 OpenGLを使った グラフィックスプログラミング

2015/04/22

溝口 知広 准教授(居室：61-408室)

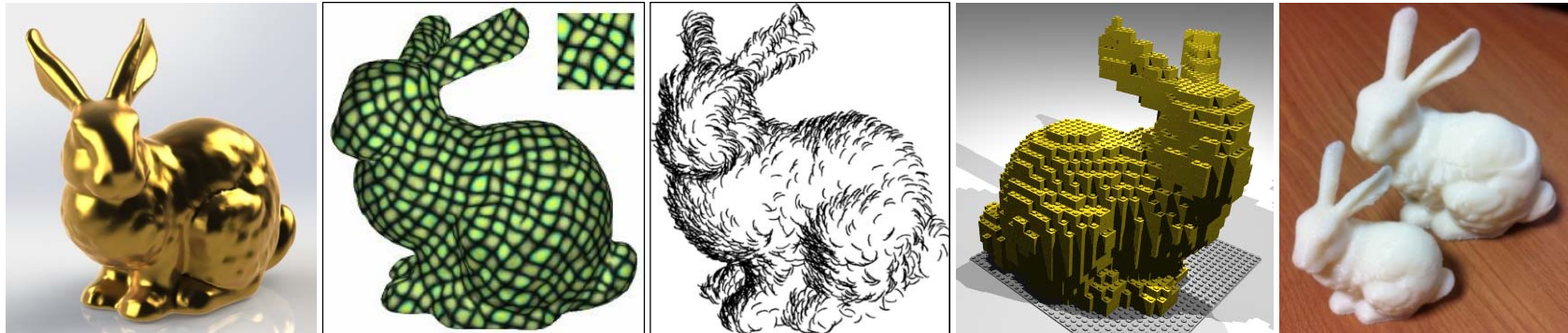
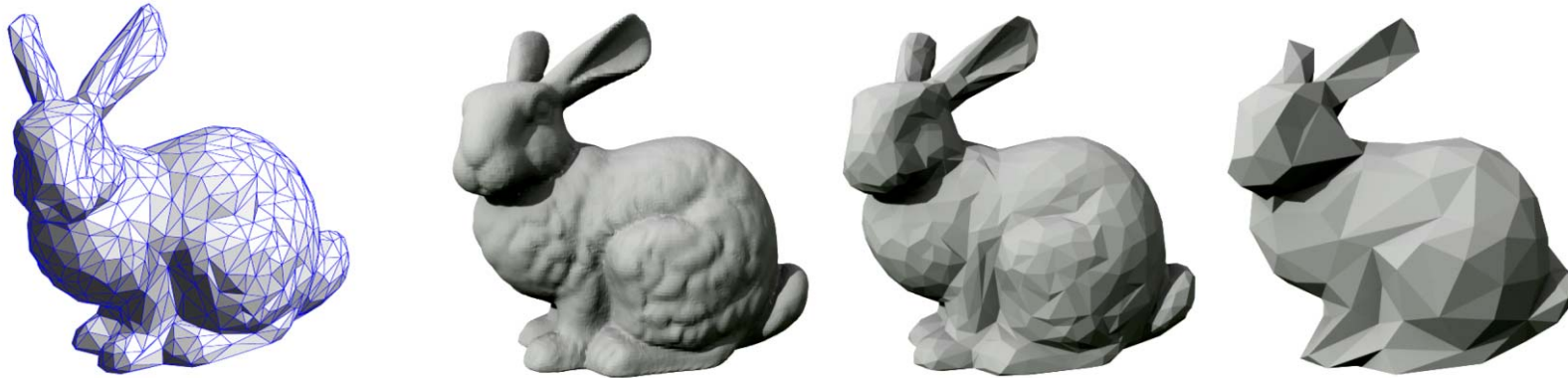
mizo@cs.ce.nihon-u.ac.jp



コンピュータ・グラフィクス(CG)

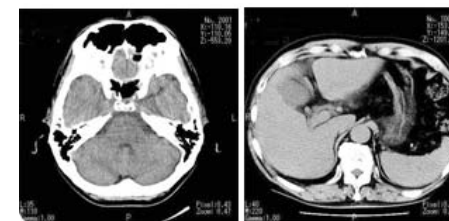
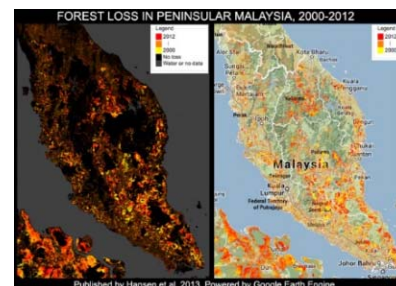
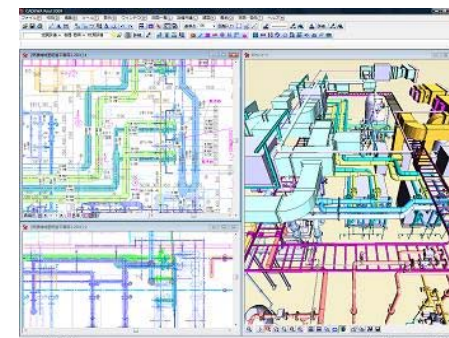
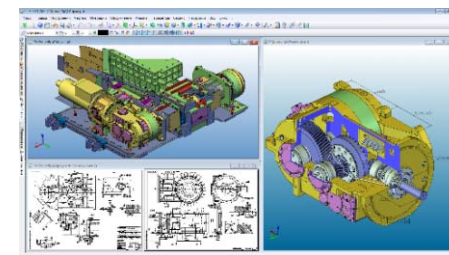
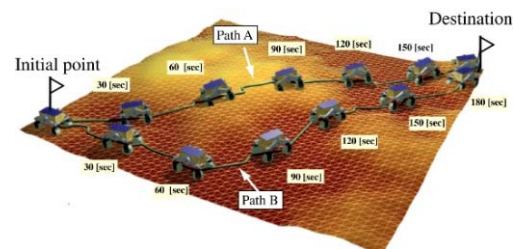


コンピュータ・グラフィックス(CG)



計算幾何学・CGの応用分野

- ロボティクス
- ものづくり
(CAD/CAM/CAE)
- 集積回路(VLSI)
- 建築・土木
- 農業・林業
- 医療(CT, MRI)

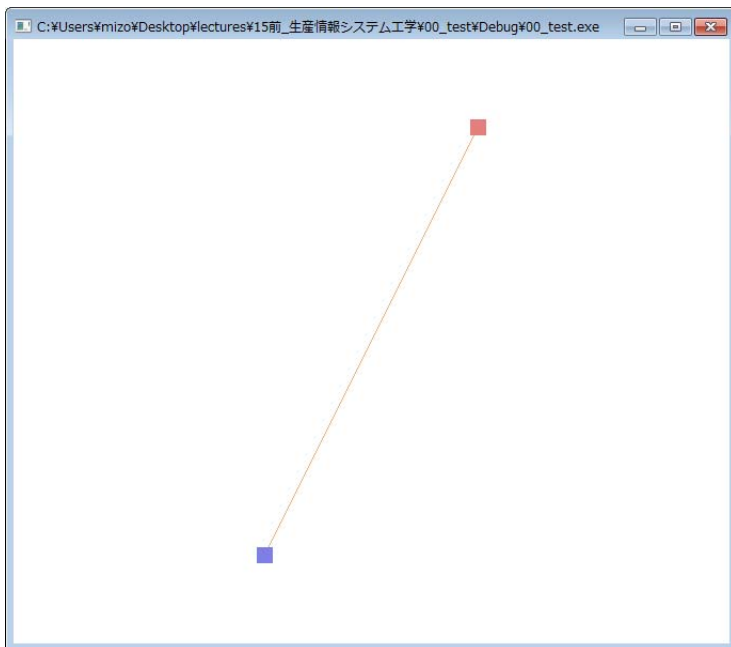


グラフィックスプログラミング

■ “OpenGL” を使用する

- 標準のグラフィックスライブラリ(表示のためのプログラム)
- コンピュータのグラフィックスハードウェアを制御する
- 2次元(2D), 3次元(3D)の図形の表示に使える
- ゲームなどの3D-CG映像のリアルタイム生成にも使用される

OpenGLを使ったプログラムの例



```
void display()
{
    glClear( GL_COLOR_BUFFER_BIT );

    GLdouble p[2] = { 200.0, 400.0 };
    GLdouble q[2] = { -200.0, -400.0 };

    // 線分
    glLineWidth( 1.0 );
    glColor3d( 0.9, 0.7, 0.5 );
    glBegin( GL_LINES );
        glVertex2d( p[0], p[1] );
        glVertex2d( q[0], q[1] );
    glEnd();

    // 点
    glPointSize( 15.0 );
    glBegin( GL_POINTS );
        glColor3d( 0.9, 0.5, 0.5 );
        glVertex2d( p[0], p[1] );
        glColor3d( 0.5, 0.5, 0.9 );
        glVertex2d( q[0], q[1] );
    glEnd();

    glFlush();
}
```

2D図形

■ 図形を描く

- glBegin()からglEnd()の間に、その図形の各頂点の座標を設定する

`void glBegin(GLnum mode)`

`void glEnd()`

■ 2次元の座標値を設定する

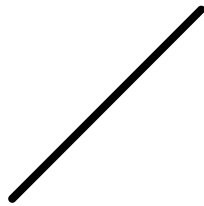
`void glVertex2d(GLdouble x, GLdouble y)`

2D図形

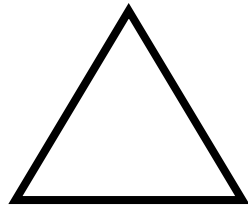
■ glBegin()の引数「mode」に指定できるタイプ



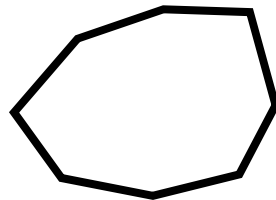
GL_POINTS
点



GL_LINES
2点を結ぶ直線



GL_TRIANGLES
3点を結ぶ三角形



GL_POLYGON
3点以上を結ぶ凸多角形

2D図形

■ 色をつける

```
void glColor3d( GLdouble r, GLdouble g, GLdouble b );
```

■ 点の大きさ

```
void glPointSize( GLdouble size );
```

■ 線の太さ

```
void glLineWidth( GLdouble width );
```

2D図形

■ 関数の命名法

```
void glColor3d( GLdouble r, GLdouble g, GLdouble b );
```

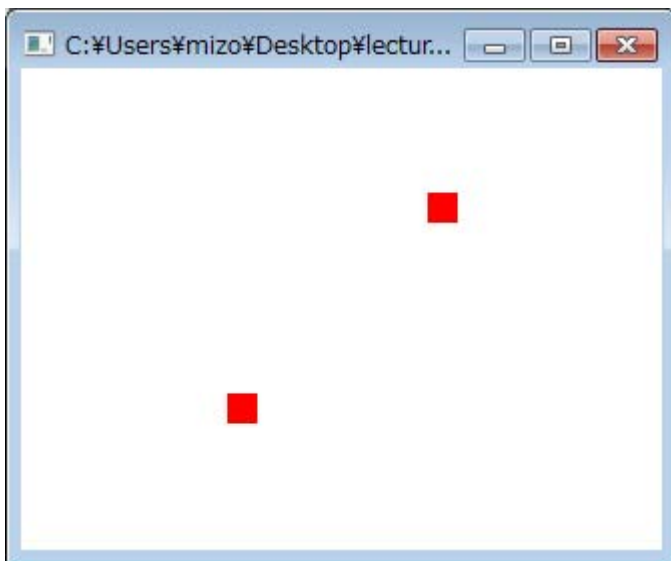
↑ ↑
引数の型

d: double, f: float, i: int, s: short

↑
引数の個数

2D図形の描画の例(1)

■ 点の描画



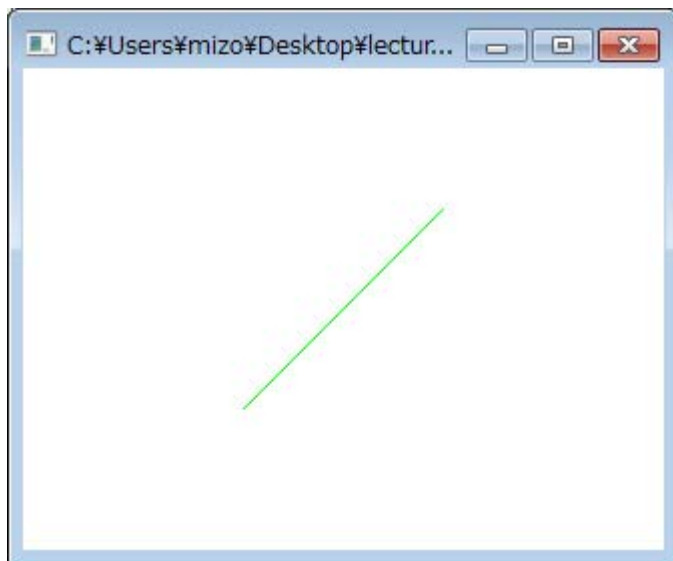
```
void display()
{
    glClearColor( GL_COLOR_BUFFER_BIT );

    // 点
    glPointSize( 15.0 );
    glColor3d( 1.0, 0.0, 0.0 );
    glBegin( GL_POINTS );
    glVertex2d( 100.0, 100.0 );
    glVertex2d( -100.0, -100.0 );
    glEnd();

    glFlush();
}
```

2D図形の描画の例(2)

■ 線の描画



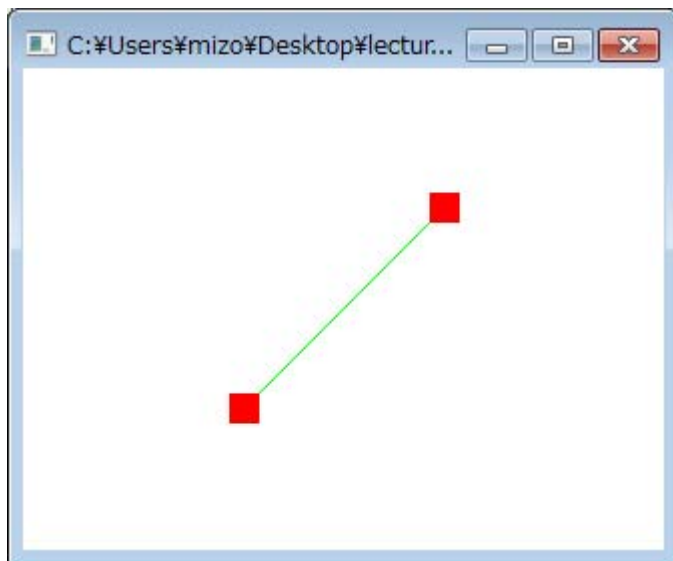
```
void display()
{
    glClearColor( GL_COLOR_BUFFER_BIT );

    // 線
    glLineWidth( 3.0 );
    glColor3d( 0.0, 1.0, 0.0 );
    glBegin( GL_LINES );
    glVertex2d( 100.0, 100.0 );
    glVertex2d( -100.0, -100.0 );
    glEnd();

    glFlush();
}
```

2D図形の描画の例(3)

■ 点と線の描画



```
void display()
{
    glClear( GL_COLOR_BUFFER_BIT );

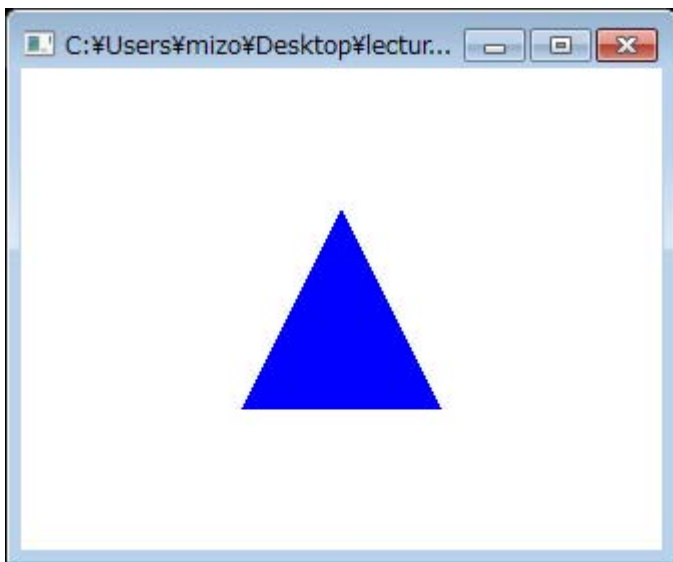
    // 線
    glColor3d( 0.0, 1.0, 0.0 );
    glBegin( GL_LINES );
    glVertex2d( 100.0, 100.0 );
    glVertex2d( -100.0, -100.0 );
    glEnd();

    // 点
    glColor3d( 0.0, 1.0, 0.0 );
    glBegin( GL_POINTS );
    glVertex2d( 100.0, 100.0 );
    glVertex2d( -100.0, -100.0 );
    glEnd();

    glFlush();
}
```

2D図形の描画の例(4)

■ 三角形の描画



```
void display()
{
    glClear( GL_COLOR_BUFFER_BIT );

    // 三角形
    glColor3d( 0.0, 0.0, 1.0 );
    glBegin( GL_TRIANGLES );
    glVertex2d( 100.0, -100.0 );
    glVertex2d( -100.0, -100.0 );
    glVertex2d( 0.0, 100.0 );
    glEnd();

    glFlush();
}
```

レポート課題#1

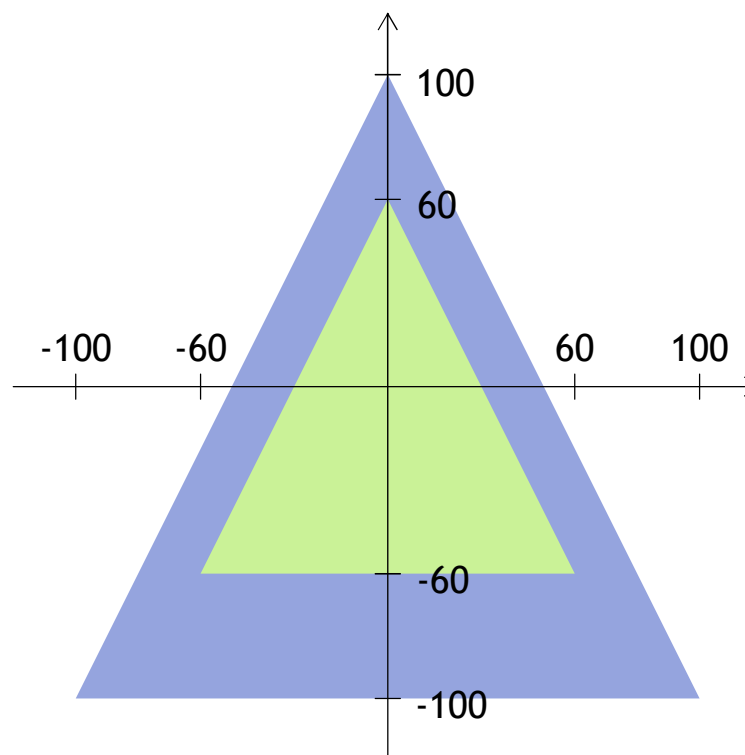
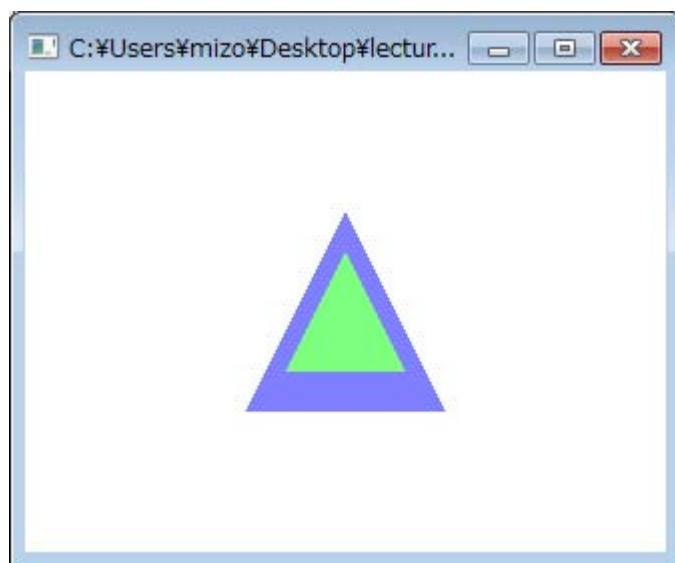
■ 次ページにある3つのプログラムを作成し、レポートにまとめて提出しなさい

- レポートに記述する項目
 - 表紙
 - ソースコード（全問題）
 - 実行結果（全問題）
 - 感想
- 表紙はポータルサイトにあります
- 提出期限：4月28日(火) 18:00まで
- 提出場所：61-408室前のレポートボックス

演習問題(1)

■ 2つの三角形を重ねて描くプログラム

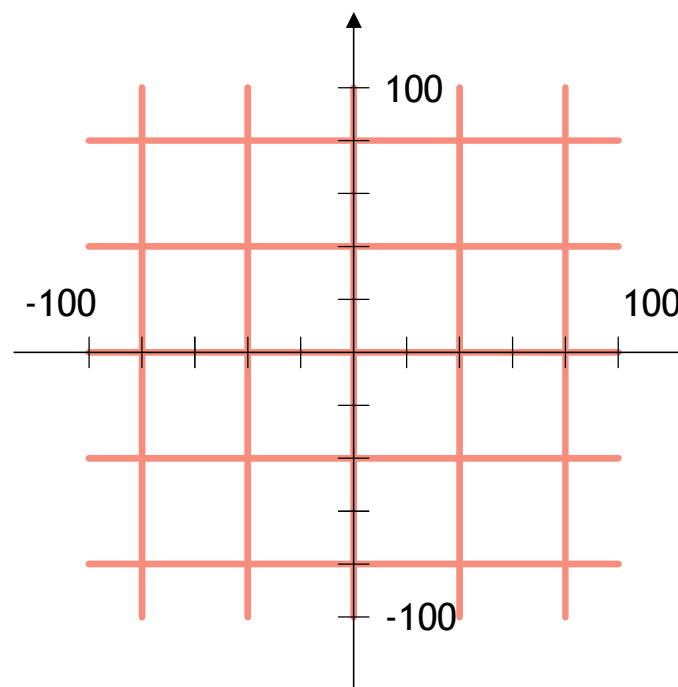
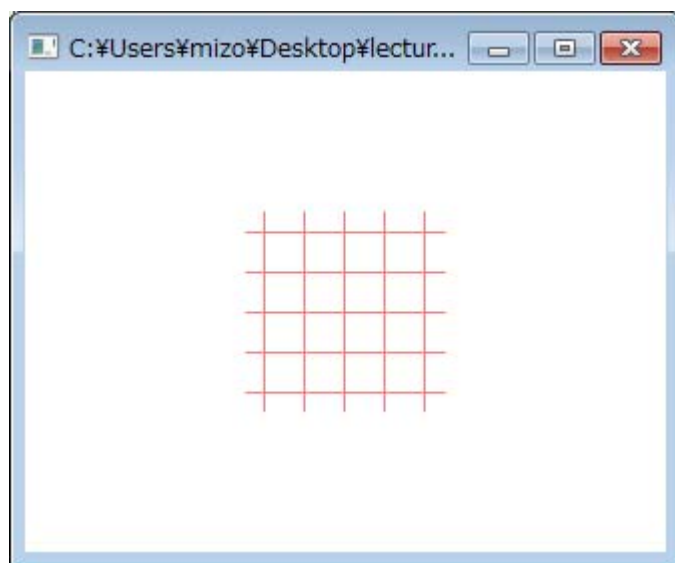
- ヒント：大きい方を先に、小さい方を後に描くとよい



演習問題(2)

■ 格子状の図形を描くプログラム

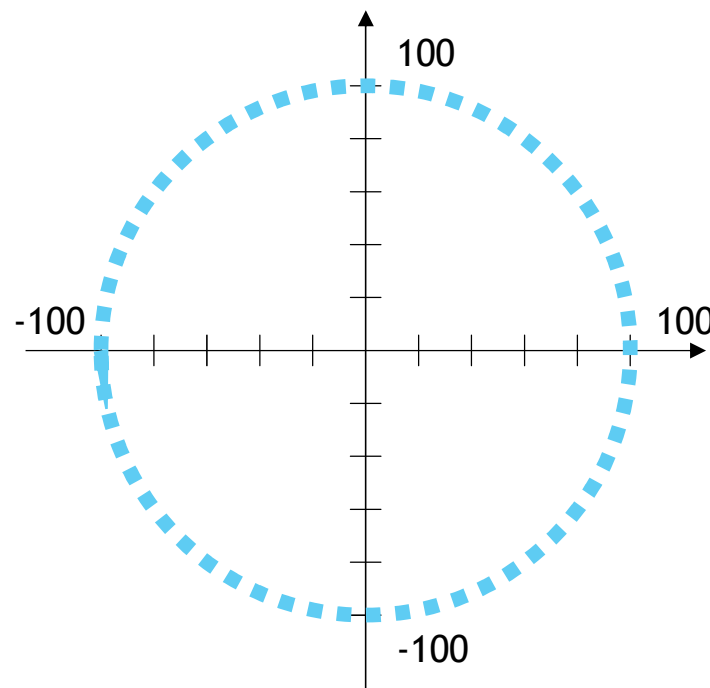
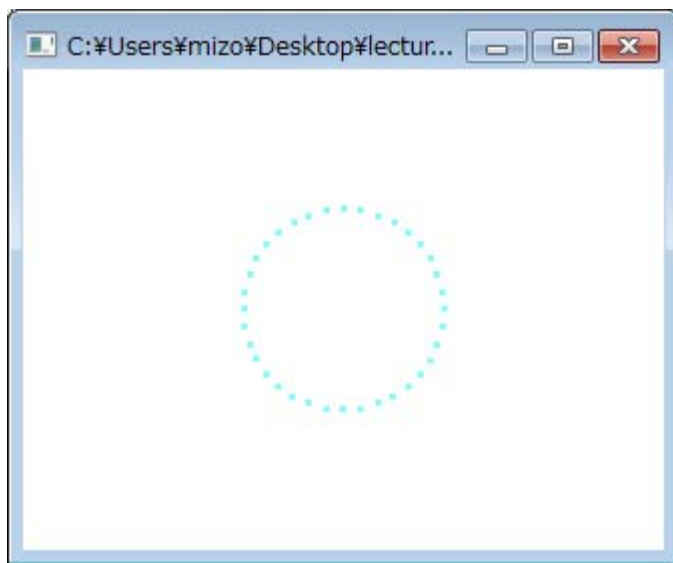
- ヒント：for文を2回使用すると簡単に書ける



演習問題(3)

■ 円周上に点を等間隔で描画するプログラム

- `math.h`をインクルードして, `sin`関数, `cos`関数を使用する



参考書

- GLUTによるOpenGL入門, 床井浩平著, 工学社
- はじめてのOpenGL—ライブラリを使った「3D-CGプログラミング」の基本, MICC著, 海事国際協力センター著, I・O BOOKS