

```

1 //
2 // コンピュータグラフィックスS
3 // OpenGLによる3次元グラフィックス演習 サンプルプログラム
4 //
5
6
7 // 基本的なヘッダファイルのインクルード
8 #ifdef _WIN32
9 #include <windows.h>
10 #endif
11 #include <stdio.h>
12 #include <math.h>
13
14 // GLUTヘッダファイルのインクルード
15 #include <GL/glut.h>
16
17 // 視点操作のための変数
18 float camera_pitch = -30.0; // X軸を軸とするカメラの回転角度
19
20 // マウスのドラッグのための変数
21 int drag_mouse_r = 0; // 右ボタンをドラッグ中かどうかのフラグ (0:非ドラッグ中, 1:ドラッグ中)
22 int last_mouse_x; // 最後に記録されたマウスカーソルのX座標
23 int last_mouse_y; // 最後に記録されたマウスカーソルのY座標
24
25 //
26 //
27 //
28 // 画面描画時に呼ばれるコールバック関数
29 //
30 void display( void )
31 {
32 // 画面をクリア (ピクセルデータとZバッファの両方をクリア)
33 glClear( GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT );
34
35 // 変換行列を設定 (ワールド座標系→カメラ座標系)
36 glMatrixMode( GL_MODELVIEW );
37 glLoadIdentity();
38 glTranslatef( 0.0, 0.0, -15.0 );
39 glRotatef( -camera_pitch, 1.0, 0.0, 0.0 );
40
41 // 光源位置を設定 (モデルビュー行列の変更にあわせて再設定)
42 float light0_position[] = { 10.0, 10.0, 10.0, 1.0 };
43 glLightfv( GL_LIGHT0, GL_POSITION, light0_position );
44
45 // 地面を描画
46 glBegin( GL_POLYGON );
47 glNormal3f( 0.0, 1.0, 0.0 );
48 glColor3f( 0.5, 0.8, 0.5 );
49
50 glVertex3f( 5.0, 0.0, 5.0 );
51 glVertex3f( 5.0, 0.0, -5.0 );
52 glVertex3f( -5.0, 0.0, -5.0 );
53 glVertex3f( -5.0, 0.0, 5.0 );
54 glEnd();
55
56 // 変換行列を設定 (物体のモデル座標系→カメラ座標系)
57 // (物体が (0.0, 1.0, 0.0) の位置にあり、静止しているとする)
58 glTranslatef( 0.0, 1.0, 0.0 );
59
60 // 物体 (1枚のポリゴン) を描画
61 glBegin( GL_TRIANGLES );
62 glColor3f( 0.0, 0.0, 1.0 );
63 glNormal3f( 0.0, 0.0, 1.0 );
64 glVertex3f( -1.0, 1.0, 0.0 );
65 glVertex3f( 0.0, -1.0, 0.0 );
66 glVertex3f( 1.0, 0.5, 0.0 );
67 glEnd();
68
69 // バックバッファに描画した画面をフロントバッファに表示
70 glutSwapBuffers();
71 }
72
73 //
74 //
75 // ウィンドウサイズ変更時に呼ばれるコールバック関数
76 //
77 void reshape( int w, int h )
78 {
79 // ウィンドウ内の描画を行う範囲を設定 (ウィンドウ全体に描画するように設定)
80 glViewport(0, 0, w, h);
81
82 // カメラ座標系→スクリーン座標系への変換行列を設定
83 glMatrixMode( GL_PROJECTION );
84 glLoadIdentity();
85 gluPerspective( 45, (double)w/h, 1, 500 );
86 }
87
88 //
89 //
90 // マウスクリック時に呼ばれるコールバック関数
91 //
92 void mouse( int button, int state, int mx, int my )
93 {
94 // 右ボタンが押されたらドラッグ開始のフラグを設定
95 if ( ( button == GLUT_RIGHT_BUTTON ) && ( state == GLUT_DOWN ) )
96 drag_mouse_r = 1;
97 // 右ボタンが離されたらドラッグ終了のフラグを設定
98 else if ( ( button == GLUT_RIGHT_BUTTON ) && ( state == GLUT_UP ) )
99 drag_mouse_r = 0;
100
101 // 現在のマウス座標を記録
102 last_mouse_x = mx;
103 last_mouse_y = my;
104 }
105
106

```

```

107 //
108 // マウスドラッグ時に呼ばれるコールバック関数
109 //
110 void motion( int mx, int my )
111 {
112     // 右ボタンのドラッグ中であれば、マウスの移動量に応じて視点を回転する
113     if ( drag_mouse_r == 1 )
114     {
115         // マウスの縦移動に応じてX軸を中心に回転
116         camera_pitch -= ( my - last_mouse_y ) * 1.0;
117         if ( camera_pitch < -90.0 )
118             camera_pitch = -90.0;
119         else if ( camera_pitch > 0.0 )
120             camera_pitch = 0.0;
121     }
122
123     // 今回のマウス座標を記録
124     last_mouse_x = mx;
125     last_mouse_y = my;
126
127     // 再描画の指示を出す (この後で再描画のコールバック関数が呼ばれる)
128     glutPostRedisplay();
129 }
130
131
132 //
133 // アイドル時に呼ばれるコールバック関数
134 //
135 void idle( void )
136 {
137     // 現在は、何も処理を行なわない
138 }
139
140
141 //
142 // 環境初期化関数
143 //
144 void initEnvironment( void )
145 {
146     // 光源を作成する
147     float light0_position[] = { 10.0, 10.0, 10.0, 1.0 };
148     float light0_diffuse[] = { 0.8, 0.8, 0.8, 1.0 };
149     float light0_specular[] = { 1.0, 1.0, 1.0, 1.0 };
150     float light0_ambient[] = { 0.1, 0.1, 0.1, 1.0 };
151     glLightfv( GL_LIGHT0, GL_POSITION, light0_position );
152     glLightfv( GL_LIGHT0, GL_DIFFUSE, light0_diffuse );
153     glLightfv( GL_LIGHT0, GL_SPECULAR, light0_specular );
154     glLightfv( GL_LIGHT0, GL_AMBIENT, light0_ambient );
155     glEnable( GL_LIGHT0 );
156
157     // 光源計算を有効にする
158     glEnable( GL_LIGHTING );
159
160     // 物体の色情報を有効にする
161     glEnable( GL_COLOR_MATERIAL );
162
163     // Zテストを有効にする
164     glEnable( GL_DEPTH_TEST );
165
166     // 背面除去を有効にする
167     glCullFace( GL_BACK );
168     glEnable( GL_CULL_FACE );
169
170     // 背景色を設定
171     glClearColor( 0.5, 0.5, 0.8, 0.0 );
172 }
173
174
175 //
176 // メイン関数 (プログラムはここから開始)
177 //
178 int main( int argc, char ** argv )
179 {
180     // GLUTの初期化
181     glutInit( &argc, argv );
182     glutInitDisplayMode( GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH );
183     glutInitWindowSize( 320, 320 );
184     glutInitWindowPosition( 0, 0 );
185     glutCreateWindow( "OpenGL & GLUT sample program" );
186
187     // コールバック関数の登録
188     glutDisplayFunc( display );
189     glutReshapeFunc( reshape );
190     glutMouseFunc( mouse );
191     glutMotionFunc( motion );
192     glutIdleFunc( idle );
193
194     // 環境初期化
195     initEnvironment();
196
197     // GLUTのメインループに処理を移す
198     glutMainLoop();
199
200     // プログラムを終了
201     return 0;
202 }
203
204

```