```c
#include<stdio.h>
#include<GL/glut.h>
#include <GL/gl.h>
#include <stdlib.h>
#include<math.h>
#include<string.h>
GLfloat a,b,d,y,p,z,c;



float m=0.0;
float n=0.0;     //movement of plane along x-axis
float o=0.0;     //  and y-axis
float plane=0.0;

void display3();
void display();
void display1();
void display2();



void *font = GLUT_BITMAP_TIMES_ROMAN_24;
void output(float x,float y, char *string)
{
  int len, i;

  glRasterPos2f(x, y);
  len = (int) strlen(string);
  for(i = 0; i < len; i++)
  {
    glutBitmapCharacter(font, string[i]);
  }

}



void update(int value)
{
a+=0.5;
b+=0.5;
z-=13.6;
c-=0.5;
if(c>=800)
c=0;
glutPostRedisplay();
glutTimerFunc(200,update,100);
}


void draw_pixel(GLint cx, GLint cy)
{

        glBegin(GL_POINTS);
  glVertex2i(cx,cy);
        glEnd();
}
```

```c
void plotpixels(GLint h,GLint k, GLint x,GLint y)
{
        draw_pixel(x+h,y+k);
        draw_pixel(-x+h,y+k);
        draw_pixel(x+h,-y+k);
        draw_pixel(-x+h,-y+k);
        draw_pixel(y+h,x+k);
        draw_pixel(-y+h,x+k);
        draw_pixel(y+h,-x+k);
        draw_pixel(-y+h,-x+k);
}


void draw_circle(GLint h, GLint k, GLint r)
{
        GLint d=1-r, x=0, y=r;
        while(y>x)
        {
                plotpixels(h,k,x,y);
   if(d<0) d+=2*x+3;
                else
                {
                        d+=2*(x-y)+5;
                        --y;
                }
                ++x;
        }
        plotpixels(h,k,x,y);
}


void snow(void)
{
glColor3f(1.0+rand()%10,1.0+rand()%10,1.0);
output(0.0,1050.0,"*");
output(20.0,1010.0,"*");
output(50.0,1208.0,"*");
output(350.0,1140.0,"*");
output(950.0,1460.0,"*");
output(500.0,1400.0,"*");
output(100.0,1050.0,"*");
output(120.0,1010.0,"*");
output(150.0,1208.0,"*");
output(850.0,1140.0,"*");
output(1050.0,1460.0,"*");
output(1500.0,1400.0,"*");
output(100.0,1050.0,"*");
output(520.0,1010.0,"*");
output(850.0,1208.0,"*");
output(550.0,1140.0,"*");
output(650.0,1460.0,"*");
output(800.0,1400.0,"*");
output(400.0,1050.0,"*");
output(920.0,1010.0,"*");
output(250.0,1208.0,"*");
output(650.0,1140.0,"*");
```

```
output(750.0,1460.0,"*");
output(1000.0,1400.0,"*");
output(800.0,950.0,"*");
output(1020.0,800.0,"*");
output(650.0,1008.0,"*");
output(1350.0,1000.0,"*");
output(550.0,860.0,"*");
output(700.0,800.0,"*");
output(900.0,750.0,"*");
output(820.0,1010.0,"*");
output(750.0,908.0,"*");
output(850.0,1140.0,"*");
output(1050.0,1460.0,"*");
output(1500.0,1400.0,"*");
output(100.0,750.0,"*");
output(520.0,810.0,"*");
output(850.0,608.0,"*");
output(550.0,940.0,"*");
output(650.0,1060.0,"*");
output(800.0,1100.0,"*");
output(1100.0,1350.0,"*");
output(1320.0,910.0,"*");
output(1450.0,708.0,"*");
output(1250.0,940.0,"*");
output(750.0,860.0,"*");
output(1000.0,900.0,"*");
output(1000.0,1400.0,"*");
output(1100.0,950.0,"*");
output(1020.0,800.0,"*");
output(1050.0,1008.0,"*");
output(1350.0,1000.0,"*");
output(1150.0,860.0,"*");
output(1200.0,800.0,"*");
output(1500.0,750.0,"*");
output(1420.0,1010.0,"*");
output(1190.0,908.0,"*");
output(1250.0,1140.0,"*");
output(1450.0,1460.0,"*");
output(1500.0,1400.0,"*");
output(1110.0,750.0,"*");
output(1520.0,810.0,"*");
output(1450.0,608.0,"*");
output(1550.0,940.0,"*");
output(1050.0,1060.0,"*");
output(900.0,1100.0,"*");
output(700.0,1350.0,"*");
output(970.0,1910.0,"*");
output(850.0,1508.0,"*");
output(1250.0,1140.0,"*");
output(750.0,1060.0,"*");
output(1400.0,1900.0,"*");
output(1450.0,1708.0,"*");
output(1250.0,1940.0,"*");
output(750.0,1860.0,"*");
output(1000.0,1400.0,"*");
output(1000.0,1400.0,"*");
output(1100.0,1050.0,"*");
output(1020.0,1390.0,"*");
```

```
output(1050.0,1308.0,"*");
output(1350.0,1580.0,"*");
output(1150.0,1760.0,"*");
output(1200.0,1500.0,"*");
output(1500.0,1350.0,"*");
output(1420.0,1010.0,"*");
output(1190.0,1608.0,"*");
output(1250.0,1040.0,"*");
output(1450.0,1860.0,"*");
output(1500.0,1100.0,"*");
output(1110.0,1450.0,"*");
output(1520.0,1210.0,"*");
output(1450.0,1208.0,"*");
output(1550.0,1140.0,"*");
output(1050.0,1060.0,"*");
output(900.0,1100.0,"*");
output(700.0,1050.0,"*");
output(970.0,1010.0,"*");
output(850.0,1178.0,"*");
output(1250.0,1040.0,"*");
output(750.0,1160.0,"*");
output(1400.0,1000.0,"*");
}


void water()
{
      GLint ww=0;
      glColor3f(0.0,1.0,1.0);//water//0.0,0.8,1.0
            glPushMatrix();
            glTranslated(ww,0,0.0);
            glBegin(GL_POLYGON);
            glVertex2f(0,1000);
            glVertex2f(0,900);
            glVertex2f(10,880);
            glVertex2f(20,881);
            glVertex2f(30,882);
            glVertex2f(40,883);
            glVertex2f(50,886);
            glVertex2f(60,888);
            glVertex2f(70,890);
            glVertex2f(80,894);
            glVertex2f(90,896);
            glVertex2f(100,900);
            glVertex2f(110,880);
            glVertex2f(120,882);
            glVertex2f(130,884);
            glVertex2f(140,886);
            glVertex2f(150,-888);
            glVertex2f(160,-890);
            glVertex2f(170,-891);
            glVertex2f(180,-892);
            glVertex2f(190,-893);
            glVertex2f(200,-894);
            glVertex2f(210,-880);
            glVertex2f(220,-880);
            glVertex2f(230,-888);
            glVertex2f(240,-890);
```

```
glVertex2f(250,-892);
glVertex2f(260,-894);
glVertex2f(270,-882);
glVertex2f(280,-886);
glVertex2f(290,-888);
glVertex2f(300,-892);
glVertex2f(320,-896);
glVertex2f(360,-898);
    glVertex2f(380,-892);
glVertex2f(400,-894);
glVertex2f(420,-896);
glVertex2f(440,-898);
glVertex2f(460,-900);
glVertex2f(480,-880);
glVertex2f(500,-888);
glVertex2f(520,-890);
glVertex2f(540,-892);
glVertex2f(560,-894);
glVertex2f(580,-882);
glVertex2f(600,-886);
glVertex2f(620,-888);
glVertex2f(640,-892);
glVertex2f(660,-896);
glVertex2f(680,898);
    glVertex2f(690,892);
glVertex2f(700,894);
glVertex2f(710,896);
glVertex2f(720,898);
glVertex2f(730,900);
glVertex2f(740,880);
glVertex2f(750,888);
glVertex2f(760,890);
glVertex2f(770,892);
glVertex2f(780,-894);
glVertex2f(790,  882);
glVertex2f(800,-886);
glVertex2f(820,-888);
glVertex2f(840,-892);
glVertex2f(860,-896);
glVertex2f(880,-898);
    glVertex2f(890,-892);
glVertex2f(900,-894);
glVertex2f(910,-896);
glVertex2f(920,-898);
glVertex2f(930,-900);
glVertex2f(950,-880);
glVertex2f(970,-888);
glVertex2f(990,-890);
glVertex2f(1000,-892);
glVertex2f(1020,-894);
glVertex2f(1040,-882);
glVertex2f(1060,-886);
glVertex2f(1080,-888);
glVertex2f(1100,-892);
glVertex2f(1120,-896);
glVertex2f(1130,-898);
    glVertex2f(1140,892);
glVertex2f(1160,894);
```

```
            glVertex2f(1180,896);
            glVertex2f(1200,898);
            glVertex2f(1210,900);
            glVertex2f(1220,880);
            glVertex2f(1240,888);
            glVertex2f(1260,890);
            glVertex2f(1280,892);
            glVertex2f(1300,894);
            glVertex2f(1320,882);
            glVertex2f(1340,886);
            glVertex2f(1360,888);
            glVertex2f(1380,892);
            glVertex2f(1400,896);
            glVertex2f(1420,898);
                glVertex2f(1440,892);
            glVertex2f(1460,894);
            glVertex2f(1480,896);
            glVertex2f(1500,898);
            glVertex2f(1520,900);
            glVertex2f(1540,880);
            glVertex2f(1560,888);
            glVertex2f(1580,890);
            glVertex2f(1600,892);
            glVertex2f(1600,1000);
            glEnd();
            glPopMatrix();

}
void water1()
{
        GLint ww=0;
        glColor3f(0.0,1.0,1.0);//water(0.0,0.8,1.0)
            glPushMatrix();
            glTranslated(ww,0,0.0);
            glBegin(GL_POLYGON);
            glVertex2f(0,-1000);
            glVertex2f(0,-700);
            glVertex2f(10,-680);
            glVertex2f(20,-681);
            glVertex2f(30,-682);
            glVertex2f(40,-683);
            glVertex2f(50,-686);
            glVertex2f(60,-688);
            glVertex2f(70,-690);
            glVertex2f(80,-694);
            glVertex2f(90,-696);
            glVertex2f(100,-700);
            glVertex2f(110,-680);
            glVertex2f(120,-682);
            glVertex2f(130,-684);
            glVertex2f(140,-686);
            glVertex2f(150,-688);
            glVertex2f(160,-690);
            glVertex2f(170,-691);
            glVertex2f(180,-692);
            glVertex2f(190,-693);
            glVertex2f(200,-694);
            glVertex2f(210,-680);
```

```
glVertex2f(220,-680);
glVertex2f(230,-688);
glVertex2f(240,-690);
glVertex2f(250,-692);
glVertex2f(260,-694);
glVertex2f(270,-682);
glVertex2f(280,-686);
glVertex2f(290,-688);
glVertex2f(300,-692);
glVertex2f(320,-696);
glVertex2f(360,-698);
    glVertex2f(380,-692);
glVertex2f(400,-694);
glVertex2f(420,-696);
glVertex2f(440,-698);
glVertex2f(460,-700);
glVertex2f(480,-680);
glVertex2f(500,-688);
glVertex2f(520,-690);
glVertex2f(540,-692);
glVertex2f(560,-694);
glVertex2f(580,-682);
glVertex2f(600,-686);
glVertex2f(620,-688);
glVertex2f(640,-692);
glVertex2f(660,-696);
glVertex2f(680,-698);
    glVertex2f(690,-692);
glVertex2f(700,-694);
glVertex2f(710,-696);
glVertex2f(720,-698);
glVertex2f(730,-700);
glVertex2f(740,-680);
glVertex2f(750,-688);
glVertex2f(760,-690);
glVertex2f(770,-692);
glVertex2f(780,-694);
glVertex2f(790,-682);
glVertex2f(800,-686);
glVertex2f(820,-688);
glVertex2f(840,-692);
glVertex2f(860,-696);
glVertex2f(880,-698);
    glVertex2f(890,-692);
glVertex2f(900,-694);
glVertex2f(910,-696);
glVertex2f(920,-698);
glVertex2f(930,-700);
glVertex2f(950,-680);
glVertex2f(970,-688);
glVertex2f(990,-690);
glVertex2f(1000,-692);
glVertex2f(1020,-694);
glVertex2f(1040,-682);
glVertex2f(1060,-686);
glVertex2f(1080,-688);
glVertex2f(1100,-692);
glVertex2f(1120,-696);
```

```
            glVertex2f(1130,-698);
                glVertex2f(1140,-692);
            glVertex2f(1160,-694);
            glVertex2f(1180,-696);
            glVertex2f(1200,-698);
            glVertex2f(1210,-700);
            glVertex2f(1220,-680);
            glVertex2f(1240,-688);
            glVertex2f(1260,-690);
            glVertex2f(1280,-692);
            glVertex2f(1300,-694);
            glVertex2f(1320,-682);
            glVertex2f(1340,-686);
            glVertex2f(1360,-688);
            glVertex2f(1380,-692);
            glVertex2f(1400,-696);
            glVertex2f(1420,-698);
                glVertex2f(1440,-692);
            glVertex2f(1460,-694);
            glVertex2f(1480,-696);
            glVertex2f(1500,-698);
            glVertex2f(1520,-700);
            glVertex2f(1540,-680);
            glVertex2f(1560,-688);
            glVertex2f(1580,-690);
            glVertex2f(1600,-692);
            glVertex2f(1600,-1000);
            glEnd();
            glPopMatrix();
            glScaled(1,1,0);
}


void rain()
{

glColor3f(1.0+rand()%10,1.0+rand()%10,1.0);
glBegin(GL_POLYGON);
glVertex2f(22.0,800.0);
glVertex2f(22.0,780.0);
glVertex2f(20.0,780.0);
glEnd();
glFlush();
}
void fillrain1()

{

float i=0.0,j=1.0;

for(j=0.0;j<12.0;j+=1.0)
{
glPushMatrix();
glTranslatef(0.0,-50.0*j,0.0);
for(i=0.0;i<30.0;i+=1.0)
{
glPushMatrix();
glTranslatef(50.0*i,0.0,0.0);
```

```
rain();
glPopMatrix();
}
glPopMatrix();
}
glFlush();
}
void fillrain2()
{float i=0.0,j=0.0;
for(j=0.0;j<15.0;j+=1.0)
{
glPushMatrix();
glTranslatef(0.0,-50.0*j,0.0);
for(i=0.0;i<30.0;i+=1.0)
{
glPushMatrix();
glTranslatef(50.0*i,30.0,0.0);
rain();
glPopMatrix();
}
glPopMatrix();
}
glFlush();
}


void flood()
{
glTranslated(0.0,z,0.0);
glColor3f(0.0,0.8,0.8);
glBegin(GL_POLYGON);
glVertex2f(0,-0);
glVertex2f(0,-400);
glVertex2f(0,-401);
glVertex2f(110,-646);
glVertex2f(125,-648);
glVertex2f(139,-644);
glVertex2f(150,-642);
glVertex2f(170,-648);
glVertex2f(185,-645);
glVertex2f(195,-646);
glVertex2f(210,-649);
glVertex2f(228,-647);
glVertex2f(239,-645);
glVertex2f(245,-697);
glVertex2f(255,-699);
glVertex2f(272,-695);
glVertex2f(295,-696);
glVertex2f(312,-695);
glVertex2f(354,-692);
glVertex2f(384,-796);
glVertex2f(444,-798);
glVertex2f(460,-793);
glVertex2f(490,-794);
glVertex2f(510,-799);
glVertex2f(550,-794);
glVertex2f(585,-600);
glVertex2f(602,-792);
```

```
glVertex2f(652,-782);
glVertex2f(652,-796);
glVertex2f(730,-605);
glVertex2f(780,-793);
glVertex2f(820,-797);
glVertex2f(850,-793);
glVertex2f(900,-795);
glVertex2f(850,-797);
glVertex2f(880,-808);
glVertex2f(900,-796);
glVertex2f(1020,-793);
glVertex2f(1050,-799);
glVertex2f(1080,-792);
glVertex2f(1100,-799);
glVertex2f(1120,-892);
glVertex2f(1140,-896);
glVertex2f(1250,-897);
glVertex2f(1350,-800);
glVertex2f(1450,-801);
glVertex2f(1600,-750);
glVertex2f(1700,-600);
glVertex2f(1100,0);
glEnd();
}


void cuttree()
{
int  l;
//tree1
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(180,300);
glVertex2f(200,300);
glVertex2f(200,390);
glVertex2f(180,390);
glEnd();
//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(200,370);
glVertex2f(225,410);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(180,370);
glVertex2f(150,410);
glEnd();

//tree
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(280,400);
glVertex2f(300,400);
glVertex2f(300,490);
```

```
glVertex2f(280,490);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(300,470);
glVertex2f(325,510);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(280,470);
glVertex2f(250,510);
glEnd();


//tree2
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(100,220);
glVertex2f(120,220);
glVertex2f(120,310);
glVertex2f(100,310);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(120,290);
glVertex2f(145,320);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(100,290);
glVertex2f(70,320);
glEnd();




//tree1
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(80,300);
glVertex2f(100,300);
glVertex2f(100,390);
glVertex2f(80,390);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
```

```
glColor3f(0.4,0.0,0.0);
glVertex2f(100,360);
glVertex2f(120,400);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(80,360);
glVertex2f(50,400);
glEnd();


//tree5
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(1030,400);
glVertex2f(1050,400);
glVertex2f(1050,490);
glVertex2f(1030,490);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(1050,460);
glVertex2f(1070,500);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(1030,460);
glVertex2f(1000,500);
glEnd();


//tree6
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(1080,300);
glVertex2f(1100,300);
glVertex2f(1100,390);
glVertex2f(1080,390);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(1100,360);
glVertex2f(1120,400);
glEnd();
//stick2
```

```
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(1080,360);
glVertex2f(1050,400);
glEnd();


//tree7
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(1030,200);
glVertex2f(1050,200);
glVertex2f(1050,290);
glVertex2f(1030,290);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(1050,260);
glVertex2f(1070,300);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(1030,260);
glVertex2f(1000,300);
glEnd();



//tree8
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,400);
glVertex2f(850,400);
glVertex2f(850,490);
glVertex2f(830,490);
glEnd();


//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(850,460);
glVertex2f(870,500);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(830,460);
glVertex2f(800,500);
glEnd();
```

```
//tree9
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,200);
glVertex2f(850,200);
glVertex2f(850,290);
glVertex2f(830,290);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(850,260);
glVertex2f(870,300);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(830,260);
glVertex2f(800,300);
glEnd();




//tree10
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(630,100);
glVertex2f(650,100);
glVertex2f(650,190);
glVertex2f(630,190);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(650,160);
glVertex2f(670,200);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(630,160);
glVertex2f(600,200);
glEnd();




//tree11
```

```
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(430,100);
glVertex2f(450,100);
glVertex2f(450,190);
glVertex2f(430,190);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(850,260);
glVertex2f(870,300);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(830,260);
glVertex2f(800,300);
glEnd();




//tree12
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(300,330);
glVertex2f(300,350);
glVertex2f(390,350);
glVertex2f(390,330);
glEnd();

 for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(300,330,l);
            draw_circle(305,350,l);
        }
        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(280,320,l);
                draw_circle(260,350,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(255,340,l);
        }


//tree13
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
```

```
glVertex2f(150,230);
glVertex2f(150,250);
glVertex2f(240,250);
glVertex2f(240,230);
glEnd();

for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(230,230,l);
            draw_circle(235,250,l);
        }
        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(270,220,l);
                draw_circle(260,250,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(275,240,l);
        }




//tree14
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(100,420);
glVertex2f(100,440);
glVertex2f(150,440);
glVertex2f(150,420);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(200,480);
glVertex2f(145,440);
glEnd();
//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(0.4,0.0,0.0);
glVertex2f(150,420);
glVertex2f(200,400);
glEnd();




//tree15
```

```
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(730,100);
glVertex2f(750,100);
glVertex2f(750,130);
glVertex2f(730,130);
glEnd();


//tree16
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,50);
glVertex2f(850,50);
glVertex2f(850,80);
glVertex2f(830,80);
glEnd();


//tree17
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(150,880);
glVertex2f(160,900);
glVertex2f(140,900);
glVertex2f(140,880);
glEnd();


for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(170,960,l);
            draw_circle(185,980,l);
        }
        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(188,990,l);
                draw_circle(186,920,l);
        }

        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(195,910,l);
        }



//tree18
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(20,190);
glVertex2f(20,210);
glVertex2f(110,220);
glVertex2f(110,190);
glEnd();
```

```
for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(70,160,l);
            draw_circle(85,180,l);
        }
        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(88,190,l);
                draw_circle(86,200,l);
        }

        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(95,210,l);
        }


}

void falltree()
{
int l;
//tree to be falled
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(980,300);
glVertex2f(1000,300);
glVertex2f(1000,390);
glVertex2f(980,390);
glEnd();

for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(960,430,l);
            draw_circle(985,400,l);
        }
        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(990,420,l);
                draw_circle(970,390,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(965,440,l);
        }


}

void road()
```

```
{
//road boundary
glPushMatrix();
glColor3f(1.0,1.0,1.0);
glBegin(GL_POLYGON);
glVertex2f(0,150);
glVertex2f(0,160);
glVertex2f(1100,160);
glVertex2f(1100,150);
glEnd();
glPopMatrix();

//road
glPushMatrix();
glColor3f(0.2,0.2,0.2);
glBegin(GL_POLYGON);
glVertex2f(0,0);
glVertex2f(0,150);
glVertex2f(1100,150);
glVertex2f(1100,0);
glEnd();
glPopMatrix();

glFlush();
}


void draw_tree()
{
int l;
/*glColor3f(0,0.501960,0);//leaves1
glBegin(GL_POLYGON);
draw_circle(75,750,70);
glEnd();

glColor3f(0,0.501960,0);//leaves2
glBegin(GL_POLYGON);
draw_circle(135,820,70);
glEnd();

glColor3f(0,0.501960,0);//leaves3
glBegin(GL_POLYGON);
draw_circle(240,820,70);
glEnd();

glColor3f(0,0.501960,0);//leaves4
glBegin(GL_POLYGON);
draw_circle(230,750,70);
glEnd();

glColor3f(0,0.5019060,0);//leaves5
glBegin(GL_POLYGON);
draw_circle(210,690,70);
glEnd();

glColor3f(0,0.5019060,0);//leaves6
glBegin(GL_POLYGON);
draw_circle(120,675,70);
```

```
        glEnd();

        glColor3f(0,0.5019060,0);//gap
        glBegin(GL_POLYGON);
        glVertex2f(60,650);
        glVertex2f(100,650);
        glVertex2f(150,650);
        glVertex2f(150,815);
        glVertex2f(250,815);
        glVertex2f(250,650);
        glEnd();

        glColor3f(0.4745,0.23921,0.0);//branch
        glBegin(GL_POLYGON);
        glVertex2f(110,500);
        glVertex2f(110,670);
        glVertex2f(125,690);
        glVertex2f(150,660);
        glVertex2f(175,700);
        glVertex2f(190,690);
        glVertex2f(190,500);
        glEnd();
        */

        //tree1
        glColor3f(0.4,0.0,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(80,300);
        glVertex2f(100,300);
        glVertex2f(100,390);
        glVertex2f(80,390);
        glEnd();


                for(l=0;l<=30;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(100,400,l);
                    draw_circle(120,450,l);
                }

                for(l=0;l<=25;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(90,390,l);
                        draw_circle(70,440,l);
                }

                for(l=0;l<=20;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(55,410,l);
                }


        //tree2
        glColor3f(0.4,0.0,0.0);
        glBegin(GL_POLYGON);
```

```
glVertex2f(180,300);
glVertex2f(200,300);
glVertex2f(200,390);
glVertex2f(180,390);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(200,400,l);
            draw_circle(220,450,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(190,390,l);
                draw_circle(170,440,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(155,410,l);
        }

//tree3
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(280,400);
glVertex2f(300,400);
glVertex2f(300,490);
glVertex2f(280,490);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(300,500,l);
            draw_circle(320,550,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(290,490,l);
                draw_circle(270,540,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(255,510,l);
        }
```

```
//tree1
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(100,220);
glVertex2f(120,220);
glVertex2f(120,310);
glVertex2f(100,310);
glEnd();


        for(l=0;l<=30;l++)


{
                glColor3f(0.0,0.3,0.0);
                draw_circle(120,320,l);
            draw_circle(140,370,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(110,300,l);
                draw_circle(90,360,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(75,330,l);
        }

//tree4
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(980,300);
glVertex2f(1000,300);
glVertex2f(1000,390);
glVertex2f(980,390);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(1000,400,l);
            draw_circle(1020,450,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(990,390,l);
                draw_circle(970,440,l);
        }

        for(l=0;l<=20;l++)
        {
```

```
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(965,410,l);
            }

//tree5
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(1030,400);
glVertex2f(1050,400);
glVertex2f(1050,490);
glVertex2f(1030,490);
glEnd();


            for(l=0;l<=30;l++)
            {
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(1050,500,l);
                draw_circle(1070,550,l);
            }

            for(l=0;l<=25;l++)
            {
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(1040,490,l);
                    draw_circle(1020,540,l);
            }

            for(l=0;l<=20;l++)
            {
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(1015,510,l);
            }




//tree6
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(1080,300);
glVertex2f(1100,300);
glVertex2f(1100,390);
glVertex2f(1080,390);
glEnd();


            for(l=0;l<=30;l++)
            {
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(1100,400,l);
                draw_circle(1120,450,l);
            }

            for(l=0;l<=25;l++)
            {
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(1090,390,l);
                    draw_circle(1070,440,l);
```

```
                }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(1065,410,l);
        }
//tree7
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(1030,200);
glVertex2f(1050,200);
glVertex2f(1050,290);
glVertex2f(1030,290);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(1050,300,l);
            draw_circle(1070,350,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(1040,290,l);
                draw_circle(1020,340,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
        }
//tree8
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,400);
glVertex2f(850,400);
glVertex2f(850,490);
glVertex2f(830,490);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(850,500,l);
            draw_circle(870,550,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(840,490,l);
                draw_circle(820,540,l);
        }
```

```
                for(l=0;l<=20;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(815,510,l);
                }

//tree9
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,200);
glVertex2f(850,200);
glVertex2f(850,290);
glVertex2f(830,290);
glEnd();


                for(l=0;l<=30;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(850,300,l);
                    draw_circle(870,350,l);
                }

                for(l=0;l<=25;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(840,290,l);
                        draw_circle(820,340,l);
                }

                for(l=0;l<=20;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(815,310,l);
                }

//tree10
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(630,100);
glVertex2f(650,100);
glVertex2f(650,190);
glVertex2f(630,190);
glEnd();


                for(l=0;l<=30;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(650,200,l);
                    draw_circle(670,250,l);
                }

                for(l=0;l<=25;l++)
                {
                        glColor3f(0.0,0.3,0.0);
                        draw_circle(640,190,l);
```

```
                        draw_circle(620,240,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(615,210,l);
        }

//tree11
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(430,100);
glVertex2f(450,100);
glVertex2f(450,190);
glVertex2f(430,190);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(450,200,l);
             draw_circle(470,250,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(440,190,l);
                draw_circle(420,240,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(415,210,l);
        }

//tree12
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(330,100);
glVertex2f(350,100);
glVertex2f(350,190);
glVertex2f(330,190);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(350,200,l);
             draw_circle(370,250,l);
        }

        for(l=0;l<=25;l++)
        {
```

```
                glColor3f(0.0,0.3,0.0);
                draw_circle(340,190,l);
                draw_circle(320,240,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(315,210,l);
        }

//tree13
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(230,100);
glVertex2f(250,100);
glVertex2f(250,190);
glVertex2f(230,190);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(250,200,l);
            draw_circle(270,250,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(240,190,l);
                draw_circle(220,240,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(215,210,l);

        }
//tree14
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(430,100);
glVertex2f(450,100);
glVertex2f(450,140);
glVertex2f(430,140);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(450,150,l);
            draw_circle(470,200,l);
        }
```

```
        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(440,140,l);
                draw_circle(420,190,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(415,170,l);
        }

//tree15
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,100);
glVertex2f(850,100);
glVertex2f(850,190);
glVertex2f(830,190);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(850,200,l);
            draw_circle(870,250,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(840,190,l);
                draw_circle(820,240,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(815,210,l);
        }

//tree16
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(830,50);
glVertex2f(850,50);
glVertex2f(850,140);
glVertex2f(830,140);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(850,150,l);
```

```
                draw_circle(870,200,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(840,140,l);
                draw_circle(820,190,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(815,160,l);
        }

//tree17
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(980,50);
glVertex2f(1000,50);
glVertex2f(1000,140);
glVertex2f(980,140);
glEnd();


        for(l=0;l<=30;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(1000,150,l);
            draw_circle(1020,200,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(990,140,l);
                draw_circle(970,190,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(965,160,l);
        }

//tree18
glColor3f(0.4,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(530,20);
glVertex2f(550,20);
glVertex2f(550,110);
glVertex2f(530,110);
glEnd();


        for(l=0;l<=30;l++)
        {
```

```
                    glColor3f(0.0,0.3,0.0);
                    draw_circle(550,120,l);
                draw_circle(570,170,l);
        }

        for(l=0;l<=25;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(540,110,l);
                draw_circle(520,160,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(0.0,0.3,0.0);
                draw_circle(515,130,l);
        }




}

void pond()
{
//pond
glColor3f(0.0,0.8,0.8);
glBegin(GL_POLYGON);
glVertex2f(355,350);
glVertex2f(355,375);
glVertex2f(380,400);
glVertex2f(505,410);
glVertex2f(530,420);
glVertex2f(630,420);
glVertex2f(655,410);
glVertex2f(680,405);
glVertex2f(705,390);
glVertex2f(730,375);
glVertex2f(740,350);
glVertex2f(730,320);
glVertex2f(705,300);
glVertex2f(680,295);
glVertex2f(655,290);
glVertex2f(630,285);
glVertex2f(605,280);
glVertex2f(580,280);
glVertex2f(555,280);
glVertex2f(530,290);
glVertex2f(505,300);
glVertex2f(380,310);
glEnd();
}




void house()
{
int l;
```

```
//small house
glColor3f(1.0,0.5,0.0);
glBegin(GL_POLYGON);
glVertex2f(200,275);
glVertex2f(200,350);
glVertex2f(280,350);
glVertex2f(280,275);
glEnd();

//window
glColor3f(0.5,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(725,400);
glVertex2f(725,420);
glVertex2f(740,420);
glVertex2f(740,400);
glEnd();



//house
glBegin(GL_POLYGON);
glColor3f(1.0,0.5,0.0);
glVertex2f(100,275);
glColor3f(0.0,0.5,0.0);
glVertex2f(100,350);
glColor3f(0.0,0.5,1.0);
glVertex2f(150,425);
glColor3f(0.0,0.5,0.0);
glVertex2f(200,350);
glColor3f(1.0,0.5,0.0);
glVertex2f(200,275);
glEnd();

//roof
glBegin(GL_POLYGON);
glColor3f(0.5,0.0,0.0);
glVertex2f(200,350);
glColor3f(0.8,0.0,0.0);
glVertex2f(150,425);
glColor3f(0.8,0.0,0.0);
glVertex2f(250,425);
glColor3f(0.5,0.0,0.0);
glVertex2f(280,350);
glEnd();

//door
glColor3f(0.5,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(140,275);
glVertex2f(140,310);
glVertex2f(160,310);
glVertex2f(160,275);
glEnd();

//house2 pink blue
glBegin(GL_POLYGON);
glColor3f(0.8,0.0,0.0);
```

```
glVertex2f(700,375);
glColor3f(0.2,0.0,1.0);
glVertex2f(700,530);
glColor3f(0.2,0.0,1.0);
glVertex2f(780,530);
glColor3f(0.8,0.0,0.0);
glVertex2f(780,375);
glEnd();

//window1
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(715,400);
glVertex2f(715,420);
glVertex2f(730,420);
glVertex2f(730,400);
glEnd();

//door
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(735,375);
glVertex2f(735,410);
glVertex2f(755,410);
glVertex2f(755,375);
glEnd();


//window2
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(760,400);
glVertex2f(760,420);
glVertex2f(775,420);
glVertex2f(775,400);
glEnd();

//window3
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(715,430);
glVertex2f(715,450);
glVertex2f(730,450);
glVertex2f(730,430);
glEnd();

//window4
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(760,430);
glVertex2f(760,450);
glVertex2f(775,450);
glVertex2f(775,430);
glEnd();

//window5
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
```

```
glVertex2f(715,460);
glVertex2f(715,490);
glVertex2f(730,490);
glVertex2f(730,460);
glEnd();

//window6
glColor3f(0.2,0.0,0.0);
glBegin(GL_POLYGON);
glVertex2f(760,460);
glVertex2f(760,490);
glVertex2f(775,490);
glVertex2f(775,460);
glEnd();

//house green
glBegin(GL_POLYGON);
glColor3f(0.0,0.3,0.0);
glVertex2f(350,225);
glColor3f(0.0,0.0,0.7);
glVertex2f(350,430);
glColor3f(0.0,0.0,0.7);
glVertex2f(430,430);
glColor3f(0.0,0.3,0.0);
glVertex2f(430,225);
glEnd();

//door
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(405,255);
glVertex2f(405,290);
glVertex2f(420,290);
glVertex2f(420,255);
glEnd();

//window1
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(405,300);
glVertex2f(405,335);
glVertex2f(420,335);
glVertex2f(420,300);
glEnd();

//window2
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(405,350);
glVertex2f(405,385);
glVertex2f(420,385);
glVertex2f(420,350);
glEnd();

//window3
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(360,255);
```

```
glVertex2f(360,290);
glVertex2f(375,290);
glVertex2f(375,255);
glEnd();

//window4
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(360,300);
glVertex2f(360,335);
glVertex2f(375,335);
glVertex2f(375,300);
glEnd();

//window5
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(360,350);
glVertex2f(360,385);
glVertex2f(375,385);
glVertex2f(375,350);
glEnd();

//window6
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(560,460);
glVertex2f(560,480);
glVertex2f(575,480);
glVertex2f(575,460);
glEnd();




//housepink blue
glBegin(GL_POLYGON);
glColor3f(0.0,0.0,1.0);
glVertex2f(500,375);
glColor3f(1.0,0.0,0.5);
glVertex2f(500,530);
glColor3f(1.0,0.0,0.5);
glVertex2f(580,530);
glColor3f(0.0,0.0,1.0);
glVertex2f(580,375);
glEnd();

//door
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(535,375);
glVertex2f(535,410);
glVertex2f(550,410);
glVertex2f(550,375);
glEnd();

//window1
glColor3f(1.0,0.9,0.0);
```

```
glBegin(GL_POLYGON);
glVertex2f(515,400);
glVertex2f(515,420);
glVertex2f(530,420);
glVertex2f(530,400);
glEnd();

//window2
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(560,400);
glVertex2f(560,420);
glVertex2f(575,420);
glVertex2f(575,400);
glEnd();

//window3
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(515,430);
glVertex2f(515,450);
glVertex2f(530,450);
glVertex2f(530,430);
glEnd();

//window4
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(560,430);
glVertex2f(560,450);
glVertex2f(575,450);
glVertex2f(575,430);
glEnd();

//window5
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(515,460);
glVertex2f(515,480);
glVertex2f(530,480);
glVertex2f(530,460);
glEnd();

//window6
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(560,460);
glVertex2f(560,480);
glVertex2f(575,480);
glVertex2f(575,460);
glEnd();

//houselight blue and dark blue
glBegin(GL_POLYGON);
glColor3f(0.0,0.0,1.0);
glVertex2f(800,175);
glColor3f(0.0,0.0,0.2);
glVertex2f(800,330);
```

```
glColor3f(0.0,0.0,0.2);
glVertex2f(880,330);
glColor3f(0.0,0.0,1.0);
glVertex2f(880,175);
glEnd();

//door
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(835,175);
glVertex2f(835,210);
glVertex2f(850,210);
glVertex2f(850,175);
glEnd();

//window1
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(815,200);
glVertex2f(815,220);
glVertex2f(830,220);
glVertex2f(830,200);
glEnd();

//window2
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(860,200);
glVertex2f(860,220);
glVertex2f(875,220);
glVertex2f(875,200);
glEnd();

//window3
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(815,230);
glVertex2f(815,250);
glVertex2f(830,250);
glVertex2f(830,230);
glEnd();

//window4
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(860,230);
glVertex2f(860,250);
glVertex2f(875,250);
glVertex2f(875,230);
glEnd();

//window5
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(815,260);
glVertex2f(815,280);
glVertex2f(830,280);
glVertex2f(830,260);
```

```
        glEnd();

        //window6
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(860,260);
        glVertex2f(860,280);
        glVertex2f(875,280);
        glVertex2f(875,260);
        glEnd();

        //building dark blue and blue
        glBegin(GL_POLYGON);
        glColor3f(0.0,0.0,0.0);
        glVertex2f(900,275);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(900,430);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(980,430);
        glColor3f(0.0,0.0,0.0);
        glVertex2f(980,275);
        glEnd();

        //doorwhite and black
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(935,345);
        glVertex2f(935,275);
        glVertex2f(950,275);
        glVertex2f(950,345);
        glEnd();

        //window1
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(915,400);
        glVertex2f(915,420);
        glVertex2f(930,420);
        glVertex2f(930,400);
        glEnd();


        //window2
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(960,400);
        glVertex2f(960,420);
        glVertex2f(975,420);
        glVertex2f(975,400);
        glEnd();

        //window3
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(915,330);
        glVertex2f(915,350);
        glVertex2f(930,350);
        glVertex2f(930,330);
```

```
        glEnd();


        //window4
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(960,330);
        glVertex2f(960,350);
        glVertex2f(975,350);
        glVertex2f(975,330);
        glEnd();

        //window5
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(915,360);
        glVertex2f(915,380);
        glVertex2f(930,380);
        glVertex2f(930,360);
        glEnd();

        //window6
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(960,360);
        glVertex2f(960,380);
        glVertex2f(975,380);
        glVertex2f(975,360);
        glEnd();



        //window2 blue and dark blue
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(860,200);
        glVertex2f(860,220);
        glVertex2f(875,220);
        glVertex2f(875,200);
        glEnd();

        //window3
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(815,230);
        glVertex2f(815,250);
        glVertex2f(830,250);
        glVertex2f(830,230);
        glEnd();


        //window4
        glColor3f(1.0,0.9,0.0);
        glBegin(GL_POLYGON);
        glVertex2f(860,230);
        glVertex2f(860,250);
        glVertex2f(875,250);
```

```
glVertex2f(875,230);
glEnd();

//window5
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(815,260);
glVertex2f(815,280);
glVertex2f(830,280);
glVertex2f(830,260);
glEnd();

//window6
glColor3f(1.0,0.9,0.0);
glBegin(GL_POLYGON);
glVertex2f(860,260);
glVertex2f(860,280);
glVertex2f(875,280);
glVertex2f(875,260);
glEnd();



glFlush();
}



void cloud()
{
int l;
//cloud1


        for(l=0;l<=20;l++)
        {
  glColor3f(1.0,1.0,1.0);
                draw_circle(160+m,625,l);

        }


        for(l=0;l<=35;l++)
        {
                glColor3f(1.0,1.0,1.0);
                draw_circle(200+m,625,l);
                draw_circle(225+m,625,l);
        }

        for(l=0;l<=20;l++)
        {
                glColor3f(1.0,1.0,1.0);
                draw_circle(265+m,625,l);
        }

//cloud2
```

```
for(l=0;l<=20;l++)
        {
              glColor3f(1.0,1.0,1.0);
                draw_circle(370+m,615,l);
        }
}



        for(l=0;l<=35;l++)
        {

              glColor3f(1.0,1.0,1.0);
              draw_circle(410+m,615,l);
              draw_circle(435+m,615,l);
              draw_circle(470+m,615,l);
        }

for(l=0;l<=20;l++)
        {
              glColor3f(1.0,1.0,1.0);
              draw_circle(500+m,615,l);
}
}

void draw_object(void)
{
int l;
glColor3f(0.0,0.3,0.0);//grass1
glBegin(GL_POLYGON);
glVertex2i(250,165);
glVertex2i(235,200);
glVertex2i(235,200);
glVertex2i(255,165);

glBegin(GL_POLYGON);
glVertex2i(260,165);
glVertex2i(275,200);
glVertex2i(275,200);
glVertex2i(255,165);
glBegin(GL_POLYGON);
glVertex2i(255,165);
glVertex2i(240,200);
glVertex2i(240,200);
glVertex2i(260,165);

glBegin(GL_POLYGON);
glVertex2i(265,165);
glVertex2i(280,200);
glVertex2i(280,200);
glVertex2i(260,165);
glBegin(GL_POLYGON);
glVertex2i(252,165);
glVertex2i(257,200);
glVertex2i(257,200);
glVertex2i(262,165);
glEnd();
```

```
glColor3f(0.0,0.3,0.0);///grass2
glBegin(GL_POLYGON);
glVertex2i(430,340);
glVertex2i(415,375);
glVertex2i(415,375);
glVertex2i(435,340);
glBegin(GL_POLYGON);
glVertex2i(440,340);
glVertex2i(455,375);
glVertex2i(455,375);
glVertex2i(435,340);
glBegin(GL_POLYGON);
glVertex2i(435,340);
glVertex2i(420,375);
glVertex2i(420,375);
glVertex2i(430,340);
glBegin(GL_POLYGON);
glVertex2i(445,340);
glVertex2i(460,375);
glVertex2i(460,375);
glVertex2i(440,340);
glBegin(GL_POLYGON);
glVertex2i(432,340);
glVertex2i(437,375);
glVertex2i(437,375);
glVertex2i(442,340);
glEnd();

glColor3f(0.0,0.3,0.0);///grass3
glBegin(GL_POLYGON);
glVertex2i(730,340);
glVertex2i(715,375);
glVertex2i(715,375);
glVertex2i(735,340);
glBegin(GL_POLYGON);
glVertex2i(740,340);
glVertex2i(755,375);
glVertex2i(755,375);
glVertex2i(735,340);
glBegin(GL_POLYGON);
glVertex2i(735,340);
glVertex2i(720,375);
glVertex2i(720,375);
glVertex2i(730,340);
glBegin(GL_POLYGON);
glVertex2i(745,340);
glVertex2i(760,375);
glVertex2i(760,375);
glVertex2i(740,340);
glBegin(GL_POLYGON);
glVertex2i(732,340);
glVertex2i(737,375);
glVertex2i(737,375);
glVertex2i(742,340);
glEnd();

glColor3f(0.0,0.3,0.0);///grass4
glBegin(GL_POLYGON);
```

```
glVertex2i(1030,340);
glVertex2i(1015,375);
glVertex2i(1015,375);
glVertex2i(1035,340);
glBegin(GL_POLYGON);
glVertex2i(1040,340);
glVertex2i(1055,375);
glVertex2i(1055,375);
glVertex2i(1035,340);
glBegin(GL_POLYGON);
glVertex2i(1035,340);
glVertex2i(1020,375);
glVertex2i(1020,375);
glVertex2i(1030,340);
glBegin(GL_POLYGON);
glVertex2i(1045,340);
glVertex2i(1060,375);
glVertex2i(1060,375);
glVertex2i(1040,340);
glBegin(GL_POLYGON);
glVertex2i(1032,340);
glVertex2i(1037,375);
glVertex2i(1037,375);
glVertex2i(1042,340);
glEnd();

glColor3f(0.0,0.3,0.0);///grass5
glBegin(GL_POLYGON);
glVertex2i(630,240);
glVertex2i(615,275);
glVertex2i(615,275);
glVertex2i(635,240);
glBegin(GL_POLYGON);
glVertex2i(640,240);
glVertex2i(655,275);
glVertex2i(655,275);
glVertex2i(635,240);
glBegin(GL_POLYGON);
glVertex2i(635,240);
glVertex2i(620,275);
glVertex2i(620,275);
glVertex2i(630,240);
glBegin(GL_POLYGON);
glVertex2i(645,240);
glVertex2i(660,275);
glVertex2i(660,275);
glVertex2i(640,240);
glBegin(GL_POLYGON);
glVertex2i(632,240);
glVertex2i(637,275);
glVertex2i(637,275);
glVertex2i(642,240);
glEnd();

//sky
glBegin(GL_POLYGON);
glColor3f(0.0,0.8,0.8);
glVertex2f(0,450);
```

```
glColor3f(0.0,0.0,0.4);
glVertex2f(0,700);
glColor3f(0.0,0.0,0.4);
glVertex2f(1100,700);
glColor3f(0.0,0.8,0.8);
glVertex2f(1100,450);
glEnd();

//sun


        for(l=0;l<=35;l++)
{
                glColor3f(1.0,0.9,0.0);
                draw_circle(100,625,l);
}

//plane
if(plane==1)
{
        glColor3f(1.0,1.0,1.0);
        glBegin(GL_POLYGON);
        glVertex2f(925+n,625+o);
glVertex2f(950+n,640+o);
        glVertex2f(1015+n,640+o);
 glVertex2f(1030+n,650+o);
        glVertex2f(1050+n,650+o);
        glVertex2f(1010+n,625+o);
glEnd();

        glColor3f(0.8,0.8,0.8);
        glBegin(GL_LINE_LOOP);
        glVertex2f(925+n,625+o);
glVertex2f(950+n,640+o);
        glVertex2f(1015+n,640+o);
        glVertex2f(1030+n,650+o);
        glVertex2f(1050+n,650+o);
        glVertex2f(1010+n,625+o);
glEnd();

}


}


void grassfull()
{
//grass
glBegin(GL_POLYGON);
glColor3f(0.0,0.3,0.0);
glVertex2f(0,0);
glColor3f(0.0,1.0,0.0);
glVertex2f(0,500);
glColor3f(0.0,1.0,0.0);
glVertex2f(1100,500);
glColor3f(0.0,0.3,0.0);
glVertex2f(1100,0);
```

```
glEnd();
}


void jcb()
{
int l;
glPushMatrix();
//jcb outline
 glColor3f(1.0,0.9,0.0);
    glBegin(GL_POLYGON);
    glVertex2f(770,330);
    glVertex2f(890,330);
    glVertex2f(890,390);
    glVertex2f(770,390);
    glVertex2f(800,330);
    glVertex2f(860,330);
    glVertex2f(860,480);
    glVertex2f(800,480);
glEnd();

//stick1
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(1.0,0.9,0.0);
glVertex2f(770,350);
glVertex2f(740,480);
glEnd();

//stick2
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(1.0,0.9,0.0);
glVertex2f(740,480);
glVertex2f(740,370);
glEnd();

//stick3
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(1.0,0.9,0.0);
glVertex2f(890,390);
glVertex2f(920,350);
glEnd();

//stick4
glLineWidth(15.0);
glBegin(GL_LINES);
glColor3f(1.0,0.9,0.0);
glVertex2f(920,350);
glVertex2f(935,330);
glEnd();

glColor3f(0.9,0.0,0.0);//body
glBegin(GL_POLYGON);
glVertex2f(935,335);
glVertex2f(945,305);
glVertex2f(975,315);
```

```
glVertex2f(935,330);
glEnd();


glColor3f(0.9,0.0,0.0);//body1
glBegin(GL_POLYGON);
glVertex2f(725,350);
glVertex2f(740,340);
glVertex2f(745,375);
glVertex2f(720,367);
glEnd();

//tire
        for(l=0;l<=30;l++)
{
                glColor3f(0.0,0.0,0.0);
                draw_circle(780,325,l);
}

 for(l=0;l<=18;l++)
{
                glColor3f(0.0,0.0,0.0);
                draw_circle(870,325,l);
}

glPopMatrix();
glVertex2f(770,390);
glColor3f(1.0,1.0,1.0);

glFlush();
}

void rock()
{
glColor3f(0.47,0.46,0.49);//stone1
glBegin(GL_POLYGON);
glVertex2f(0,0);
glVertex2f(100,0);
glVertex2f(120,30);
glVertex2f(125,50);
glVertex2f(120,70);
glVertex2f(100,100);
glVertex2f(50,50);
glVertex2f(0,50);
glEnd();
glColor3f(0.34,0.31,0.45);//stone2
glBegin(GL_POLYGON);
glVertex2f(0,50);
glVertex2f(50,50);
glVertex2f(100,100);
glVertex2f(100,125);
glVertex2f(75,150);
glVertex2f(50,170);
glVertex2f(0,175);
glVertex2f(0,50);
glEnd();
glColor3f(0.44,0.4,0.50);//stone3
glBegin(GL_POLYGON);
```

```
glVertex2f(100,100);
glVertex2f(120,65);
glVertex2f(150,100);
glVertex2f(165,150);
glVertex2f(145,175);
glVertex2f(100,175);
glVertex2f(75,150);
glVertex2f(100,125);
glVertex2f(100,100);
glEnd();
glColor3f(0.32,0.28,0.41);//stone4
glBegin(GL_POLYGON);
glVertex2f(150,100);
glVertex2f(180,115);
glVertex2f(200,110);
glVertex2f(215,100);
glVertex2f(230,75);
glVertex2f(235,40);
glVertex2f(225,0);
glVertex2f(100,0);
glVertex2f(120,30);
glVertex2f(125,50);
glVertex2f(120,70);
glVertex2f(150,100);
glEnd();
glColor3f(0.32,0.28,0.41);//stone5
glBegin(GL_POLYGON);
glVertex2f(1200,15);
glVertex2f(1175,20);
glVertex2f(1145,35);
glVertex2f(1125,65);
glVertex2f(1125,100);
glVertex2f(1135,125);
glVertex2f(1150,150);
glVertex2f(1175,165);
glVertex2f(1200,170);
glVertex2f(1200,15);
glEnd();
glColor3f(0.44,0.4,0.50);//stone6
glBegin(GL_POLYGON);
glVertex2f(1085,0);
glVertex2f(1090,25);
glVertex2f(1105,50);
glVertex2f(1125,65);
glVertex2f(1145,35);
glVertex2f(1175,20);
glVertex2f(1200,15);
glVertex2f(1150,0);
glVertex2f(1200,0);
glVertex2f(1200,15);
glVertex2f(1085,0);
glEnd();
glColor3f(0.47,0.46,0.49);//stone7
glBegin(GL_POLYGON);
glVertex2f(1000,0);
glVertex2f(1005,30);
glVertex2f(1010,50);
glVertex2f(1025,85);
```

```
glVertex2f(1050,115);
glVertex2f(1080,135);
glVertex2f(1105,150);
glVertex2f(1140,160);
glVertex2f(1175,165);
glVertex2f(1150,150);
glVertex2f(1135,125);
glVertex2f(1125,100);
glVertex2f(1125,65);
glVertex2f(1105,50);
glVertex2f(1090,25);
glVertex2f(1085,0);
glVertex2f(1000,0);
glEnd();

glColor3f(0.0,0.5,0.0);//grass
glBegin(GL_POLYGON);
glVertex2f(1175,165);
glVertex2f(1135,180);
glVertex2f(1100,190);
glVertex2f(1050,185);
glVertex2f(1000,175);
glVertex2f(1050,205);
glVertex2f(1095,215);
glVertex2f(1050,250);
glVertex2f(985,280);
glVertex2f(1005,290);
glVertex2f(1050,285);
glVertex2f(1110,240);
glVertex2f(1085,300);
glVertex2f(1050,340);
glVertex2f(1100,325);
glVertex2f(1140,265);
glVertex2f(1140,300);
glVertex2f(1160,380);
glVertex2f(1155,275);
glVertex2f(1200,320);
glVertex2f(1200,165);
glVertex2f(1175,165);
glEnd();
glColor3f(0.0,0.5,0.0);//grass 2
glBegin(GL_POLYGON);
glVertex2f(225,0);
glVertex2f(235,40);
glVertex2f(250,75);
glVertex2f(260,155);
glVertex2f(270,75);
glVertex2f(330,135);
glVertex2f(280,65);
glVertex2f(385,85);
glVertex2f(290,35);
glVertex2f(360,25);
glVertex2f(300,0);
glVertex2f(225,0);
glEnd();
glColor3f(0.0,0.5,0.0);//leaf1
glBegin(GL_POLYGON);
glVertex2f(805,60);
```

```
glVertex2f(775,20);
glVertex2f(760,10);
glVertex2f(740,15);
glVertex2f(730,25);
glVertex2f(715,50);
glVertex2f(750,85);
glVertex2f(805,60);
glEnd();
glColor3f(0.0,0.5,0.0);//leaf2
glBegin(GL_POLYGON);
glVertex2f(805,60);
glVertex2f(795,20);
glVertex2f(805,5);
glVertex2f(840,10);
glVertex2f(860,30);
glVertex2f(805,60);
glEnd();
glColor3f(0.0,0.5,0.0);//leaf3
glBegin(GL_POLYGON);
glVertex2f(805,60);
glVertex2f(840,105);
glVertex2f(870,115);
glVertex2f(890,85);
glVertex2f(895,50);
glVertex2f(845,45);
glVertex2f(805,60);
glEnd();

}


void mouse(int btn,int state,int x,int y)
{
        if(btn==GLUT_LEFT_BUTTON && state==GLUT_DOWN)
exit(0);
}


void keys(unsigned char key,int x,int y)
{
      GLint i,j,k,m;
      switch(key)
      {
            case 'P':
            case 'p':
                  for(j=0;j<800;j++)
                        display();//FRAME1
            case 'h':
            case 'H':
                  for(k=0;k<800;k++)
                        display1();//FRAME2
            case 'c':
            case 'C':
                  for(i=0;i<800;i++)
                        display2();//FRAME3

            case 'q':
            case 'Q':if(key=='q')
                        exit(1);
```

```
                    break;
            default:display3();
        }
}



void myinit()
{
glClearColor(1.0,1.0,1.0,1.0);
glColor3f(0.0,0.0,1.0);
glPointSize(2.0);
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
gluOrtho2D(0.0,1100.0,0.0,700.0);
}

void details()
 {
 glClear(GL_COLOR_BUFFER_BIT);
glColor3f(0.0,0.0,0.0);
glBegin(GL_POLYGON);
glColor3f(0.0,0.0,0.1);
glVertex2f(0,0);
glColor3f(0.0,0.0,0.1);
glVertex2f(1200,0);
glColor3f(0.0,0.0,1.0);
glVertex2f(1200,800);
glColor3f(0.0,0.0,0.1);
glVertex2f(0,800);
glEnd();
glColor3f(1.0,1.0,0.2);

output(470,650,"Welcome to my project");
glEnd();

  }


void display3()
 {

    a+=0.7;
    glClear(GL_COLOR_BUFFER_BIT);
    //drawtocover();
    details();
    if(a>5)
     {
       display1();
     }
    glFlush();
    glutSwapBuffers();
   }


void display()
```

```
{
glClear(GL_COLOR_BUFFER_BIT);
{
grassfull();
draw_object();
pond();
draw_tree();
rock();
glPushMatrix();
glTranslated(a,0,0);
a+=0.5;
cloud();
glPopMatrix();
if(a>10)
{
display1();
}
glutPostRedisplay();
glutSwapBuffers();
glFlush();
}
}

void display1()
{

glClear(GL_COLOR_BUFFER_BIT);
{
grassfull();
draw_object();
pond();
rock();
cuttree();
glPushMatrix();
glTranslated(b,0,0);
b+=1.0;
cloud();
glPopMatrix();

glPushMatrix();
glTranslated(b,0,0);
jcb();
glPopMatrix();
glPushMatrix();
glTranslated(0,b,0);
if(b>-0.5)
{
falltree();
}
glPopMatrix();

if(b>28)
{
display2();
}
glutPostRedisplay();
glutSwapBuffers();
```

```c
glFlush();

}
}

void display2()
{
glClear(GL_COLOR_BUFFER_BIT);
{
grassfull();
draw_object();
house();
road();
cloud();


glPushMatrix();
glTranslated(0,z,0);
z-=-40.0;
flood();
fillrain1();
glPopMatrix();

glutPostRedisplay();
glutSwapBuffers();

glFlush();
}
}



int main(int argc,char** argv)
{
int c_menu;

        printf("Press RIGHT MOUSE BUTTON to display menu \n");
        printf("Press LEFT MOUSE BUTTON to quit the program \n");

        glutInit(&argc,argv);
        glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);
        glutInitWindowSize(1100.0,700.0);
        glutInitWindowPosition(0,0);
        glutCreateWindow("Lake denotification");
        glutKeyboardFunc(keys);
        glutDisplayFunc(display3);
        glutMouseFunc(mouse);
         myinit();
        glutAttachMenu(GLUT_RIGHT_BUTTON);


        glutMainLoop();
        return 0;
}
```