|  |
| --- |
| #include<stdlib.h> |
|  | #include<stdio.h> |
|  | #include<math.h> |
|  | #include<Gl/glut.h> |
|  | static GLfloat spin=360.0;/\*fan rotation variable\*/ |
|  | static GLfloat u=0.45; |
|  | static GLfloat v=0.45; |
|  | static GLfloat w=0.45; |
|  | static GLfloat b=0.45; |
|  | static GLfloat c=0.00; |
|  | static GLfloat d=0.00; |
|  | static GLfloat e=0.00; |
|  | static GLfloat a=-40; /\*clouds translation variable\*/ |
|  | static int z=0; |
|  | GLfloat x=0; |
|  | GLfloat y=0; |
|  | int m,n; |
|  | void output(char \*string) |
|  | { |
|  | while(\*string) |
|  | glutBitmapCharacter(GLUT\_BITMAP\_8\_BY\_13, \*string++); |
|  | } |
|  | void init(void) |
|  | { |
|  | glClearColor(1.0,1.0,1.0,1.0); |
|  | glShadeModel(GL\_FLAT); |
|  | } |
|  | void text() |
|  | { |
|  | glColor3f(u,v,w); |
|  | glRasterPos2f(0,13); |
|  | output("POWER HOUSE"); |
|  | glRasterPos2f(20,13); |
|  | output("STREET LIGHT"); |
|  | } |
|  | void front() |
|  | { |
|  | glColor3f(0.0,0.0,1.0); |
|  | glRasterPos2f(-22,25); |
|  | output("ELECTRIC POWER GENERATION THROUGH WIND MILL"); |
|  | glColor3f(0.3,0.1,0.4); |
|  | glRasterPos2f(-15,0.0); |
|  | output("Your Name Here"); |
|  | } |
|  | void streetlight() |
|  | { |
|  | glPushMatrix();/\*1st street light\*/ |
|  | glLoadIdentity(); |
|  | glColor3f(0.2,0.2,0.2); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex3f(28.0,-20.0,2.0); |
|  | glVertex3f(29.0,-20.0,3.0); |
|  | glVertex3f(29.0,10.0,4.0); |
|  | glVertex3f(28.0,10.0,2.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | glPushMatrix(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex3f(26.0,6.0,2.0); |
|  | glVertex3f(31.0,7.0,3.0); |
|  | glVertex3f(31.0,6.0,4.0); |
|  | glVertex3f(26.0,7.0,2.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | glPushMatrix(); |
|  | glColor3f(b,b,b); |
|  | glTranslatef(24.5,4.0,1.0); |
|  | glRotatef(260,0,0,1); |
|  | glScalef(1,3.5,1); |
|  | glutSolidCube(2); |
|  | glPopMatrix(); |
|  | glPushMatrix();/\*2nd street light\*/ |
|  | glLoadIdentity(); |
|  | glColor3f(0.2,0.2,0.2); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex3f(16.1,-10.0,2.0); |
|  | glVertex3f(16.9,-10.0,3.0); |
|  | glVertex3f(16.9,14.0,4.0); |
|  | glVertex3f(16.1,14.0,2.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | glPushMatrix(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex3f(14.5,12.0,2.0); |
|  | glVertex3f(18.5,13.0,3.0); |
|  | glVertex3f(18.5,12.0,4.0); |
|  | glVertex3f(14.5,13.0,2.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | glPushMatrix(); |
|  | glColor3f(b,b,b); |
|  | glTranslatef(13.5,10.5,1.0); |
|  | glRotatef(260,0,0,1); |
|  | glScalef(1,3.5,1); |
|  | glutSolidCube(1.5); |
|  | glPopMatrix(); |
|  | } |
|  | void background() |
|  | { |
|  | glColor3f(0.0,0.1,0.0); |
|  | glBegin(GL\_POLYGON);//green ground |
|  | glVertex2i(-250.0,-250.0); |
|  | glVertex2i(250.0,-250.0); |
|  | glVertex2i(250.0,0.0); |
|  | glVertex2i(-250.0,0.0); |
|  | glEnd(); |
|  | glColor3f(0.1 ,0.1,0.1); |
|  | glBegin(GL\_POLYGON);//mid night blue sky |
|  | glVertex2i(-250.0,0.0); |
|  | glVertex2i(-250.0,250.0); |
|  | glVertex2i(250.0,250.0); |
|  | glVertex2i(250.0,0.0); |
|  | glEnd(); |
|  | } |
|  | void fan1() |
|  | { |
|  | glPushMatrix(); |
|  | glLoadIdentity(); |
|  | glColor3f(1,1,1); |
|  | glTranslatef (-8.0,20.0, 2.0);/\*rotation about fixed point\*/ |
|  | glRotatef(spin,0.0,0.0,1.0); |
|  | glTranslatef (8.0,-20.0, -2.0); |
|  | glBegin(GL\_TRIANGLES);/\*1st fan\*/ |
|  | glVertex3f(-8.0,20.0,2.0); |
|  | glVertex3f(-12.0,16.0,3.0); |
|  | glVertex3f(-12.0,18.0,4.0); |
|  | glVertex3f(-8.0,20.0,2.0); |
|  | glVertex3f(-4.0,24.0,3.0); |
|  | glVertex3f(-4.0,22.0,4.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | } |
|  | void fan2() |
|  | { |
|  | glPushMatrix(); |
|  | glLoadIdentity(); |
|  | glTranslatef (-20.0, 20.0, 2.0);/\*rotation about fixed point\*/ |
|  | glRotatef(spin,0.0,0.0,1.0); |
|  | glTranslatef (20.0, -20.0,-2.0); |
|  | glColor3f(1,1,1); |
|  | glBegin(GL\_TRIANGLES);/\*2nd fan\*/ |
|  | glVertex3f(-20.0,20.0,2.0); |
|  | glVertex3f(-25.0,17.0,3.0); |
|  | glVertex3f(-25.0,19.0,4.0); |
|  | glVertex3f(-20.0,20.0,2.0); |
|  | glVertex3f(-15.0,23.0,3.0); |
|  | glVertex3f(-15.0,21.0,4.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | } |
|  | void fan3() |
|  | { |
|  | glPushMatrix(); |
|  | glLoadIdentity(); |
|  | glTranslatef (-32.0, 20.0, 2.0);/\*rotation about fixed point\*/ |
|  | glRotatef(spin,0.0,0.0,1.0); |
|  | glTranslatef (32.0, -20.0, -2.0); |
|  | glColor3f(1,1,1); |
|  | glBegin(GL\_TRIANGLES);/\*2nd fan\*/ |
|  | glVertex3f(-32.0,20.0,2.0); |
|  | glVertex3f(-36.0,16.0,3.0); |
|  | glVertex3f(-36.0,18.0,4.0); |
|  | glVertex3f(-32.0,20.0,2.0); |
|  | glVertex3f(-28.0,24.0,3.0); |
|  | glVertex3f(-28.0,22.0,4.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | } |
|  | void fan4() |
|  | { |
|  | glPushMatrix(); |
|  | glLoadIdentity(); |
|  | glColor3f(1,1,1); |
|  | glTranslatef (28.0,25.0, 2.0);/\*rotation about fixed point\*/ |
|  | glRotatef(spin,0.0,0.0,1.0); |
|  | glTranslatef (-28.0,-25.0, -2.0); |
|  | glBegin(GL\_TRIANGLES);/\*4th fan\*/ |
|  | glVertex3f(28.0,25.0,2.0); |
|  | glVertex3f(24.0,21.0,3.0); |
|  | glVertex3f(24.0,23.0,4.0); |
|  | glVertex3f(28.0,25.0,2.0); |
|  | glVertex3f(32.0,29.0,3.0); |
|  | glVertex3f(32.0,27.0,4.0); |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | } |
|  | void wires() |
|  | { |
|  | glColor3f(.7,.5,.7); |
|  | glEnable(GL\_LINE\_STIPPLE); |
|  | glLineStipple(1,0x00FF); |
|  | glBegin(GL\_LINES); |
|  | glVertex2f(-8.0,7.0); |
|  | glVertex2f(-32.0,7.0); |
|  | glVertex2f(-8.0,10.0); |
|  | glVertex2f(1.5,10.0); |
|  | glVertex2f(26.5,7.0); |
|  | glVertex2f(14.5,12.0); |
|  | glVertex2f(31.0,7.0); |
|  | glVertex2f(18.0,12.0); |
|  | glEnd(); |
|  | glDisable(GL\_LINE\_STIPPLE); |
|  | } |
|  | void powerstation() |
|  | { |
|  | GLint ax=1.5,ay=8; |
|  | glColor3f(1.0,0.25,0.1); |
|  | glBegin(GL\_POLYGON);//from tip(anti clkwise) |
|  | glVertex2i(ax,ay);//a |
|  | glVertex2i(ax-2,ay-2);//b |
|  | glVertex2i(ax-2,ay-8);//c |
|  | glVertex2i(ax+2,ay-8);//d |
|  | glVertex2i(ax+2,ay-2);//e |
|  | glEnd(); |
|  | glColor3f(0.7,0.5,0.3); |
|  | glBegin(GL\_POLYGON);//roof (from a) |
|  | glVertex2i(ax,ay+3);//a |
|  | glVertex2i(ax-3,ay-3);//b |
|  | glVertex2i(ax+3,ay-3);//e |
|  | glEnd(); |
|  | glColor3f(v,v,w); |
|  | glBegin(GL\_POLYGON);/\* door \*/ |
|  | glVertex2i(ax-1,ay-5.0);//top left |
|  | glVertex2i(ax-1.0,ay-8.0);//bottom left |
|  | glVertex2i(ax+1.0,ay-8.0);// bottom right |
|  | glVertex2i(ax+1.0,ay-5.0);//top right |
|  | glEnd(); |
|  | } |
|  | void road() |
|  | { |
|  | glColor3f(0.0,0.0,0.0); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex2f(-1,0); |
|  | glVertex2f(4,0); |
|  | glVertex2f(43,-39); |
|  | glVertex2f(37,-40); |
|  | glEnd(); |
|  | } |
|  | void clouds() |
|  | { |
|  | glPushMatrix(); |
|  | glColor3f (0.4, 0.7,0.9); |
|  | glLoadIdentity (); /\* clear the matrix \*/ |
|  | /\* viewing transformation \*/ |
|  | glTranslatef(a+1, 40.0, -9.0); |
|  | glScalef (2.0, 1.0, 1.0); /\* modeling transformation \*/ |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a-2.0,40.0, -9.0); |
|  | glScalef (2.0, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a+7.0,40.0, -9.0); |
|  | glScalef (2, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a-7.0,40.0, -9.0); |
|  | glScalef (2, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a+18.0,40.0, -9.0); |
|  | glScalef (2, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a+25.0,40.0, -9.0); |
|  | glScalef (2, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a+36.0,40.0, -9.0); |
|  | glScalef (3.0, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a+50.0,40.0, -9.0); |
|  | glScalef (2, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glLoadIdentity(); |
|  | glTranslatef(a+56.0,40.0, -9.0); |
|  | glScalef (2, 1.0, 1.0); |
|  | glutSolidSphere (2.0,50,56); |
|  | glPopMatrix(); |
|  | } |
|  | void roof(GLint rux,GLint ruy,GLint rdx,GLint rdy) |
|  | { |
|  | glPushMatrix(); |
|  | glColor3f(1,0.25,0.1); |
|  | glBegin(GL\_LINE\_STRIP); |
|  | glVertex2i(rux,ruy);//roof up |
|  | glVertex2i(rdx,rdy);//roof down |
|  | glEnd(); |
|  | glPopMatrix(); |
|  | } |
|  | void hut(GLint rux,GLint ruy,GLint rdx,GLint rdy) |
|  | { |
|  | GLint blx=rdx,bly=rdy-9,brx=rdx+10,bry=rdy-9,kx=rdx-8,ky=rdy+1; |
|  | GLfloat i; |
|  | for(i=0;i<440;i=i+1) |
|  | roof(rux+i/40,ruy,rdx+i/40,rdy);/\* draw straws \*/ |
|  | glColor3f(0.3,0.25,0.1); |
|  | glBegin(GL\_POLYGON);/\* front wall \*/ |
|  | glVertex2i(rdx,rdy);//roof left |
|  | glVertex2i(rdx+10,rdy);//roof right |
|  | glVertex2i(brx,bry);//base right |
|  | glVertex2i(blx,bly);//base left |
|  | glEnd(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_POLYGON);/\* side wall \*/ |
|  | glVertex2i(rux,ruy);//roof up |
|  | glVertex2i(kx,ky);//bacK |
|  | glVertex2i(kx,ky-6);//bacK base |
|  | glVertex2i(blx,bly); |
|  | glVertex2i(rdx,rdy); |
|  | glEnd(); |
|  | glColor3f(v,v,w); |
|  | glBegin(GL\_POLYGON);/\* window \*/ |
|  | glVertex2i(kx+2,ky-2.5);//top left |
|  | glVertex2i(kx+2,ky-5.5);//bottom left |
|  | glVertex2i(blx-3,bly+3.5);//bottom right |
|  | glVertex2i(blx-3,bly+6.9);//top right |
|  | glEnd(); |
|  | glColor3f(v,v,w); |
|  | glBegin(GL\_POLYGON);/\* door \*/ |
|  | glVertex2i(blx+3.5,bly+5);//top left |
|  | glVertex2i(blx+3.5,bly);//bottom left |
|  | glVertex2i(brx-3.5,bry);//bottom right |
|  | glVertex2i(brx-3.5,bry+5);//top right |
|  | glEnd(); |
|  | } |
|  | void fanpole1() |
|  | { |
|  | glColor3f(1.0,1.0,1.0); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(-8.1,20.0); |
|  | glVertex2f(-7.9,20.0); |
|  | glVertex2f(-8.5,0.0); |
|  | glVertex2f(-7.5,0.0); |
|  | glEnd(); |
|  | } |
|  | void fanpole2() |
|  | { |
|  | glColor3f(1.0,1.0,1.0); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(-20.1,20.0); |
|  | glVertex2f(-19.9,20.0); |
|  | glVertex2f(-20.5,0.0); |
|  | glVertex2f(-19.5,0.0); |
|  | glEnd(); |
|  | } |
|  | void fanpole3() |
|  | { |
|  | glColor3f(1.0,1.0,1.0); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(-32.1,20.0); |
|  | glVertex2f(-31.9,20.0); |
|  | glVertex2f(-32.5,0.0); |
|  | glVertex2f(-31.5,0.0); |
|  | glEnd(); |
|  | } |
|  | void fanhouse() |
|  | { GLint ax=28,ay=30; |
|  | glColor3f(0.7,0.5,0.3); |
|  | glBegin(GL\_POLYGON);//from tip(anti clkwise) |
|  | glVertex2i(ax,ay);//a |
|  | glVertex2i(ax-3,ay-2);//b |
|  | glVertex2i(ax-3,ay-9);//c |
|  | glVertex2i(ax+3,ay-9);//d |
|  | glVertex2i(ax+3,ay-2);//e |
|  | glEnd(); |
|  | glColor3f(0.0,0.3,0.3); |
|  | glBegin(GL\_POLYGON);//roof (from a) |
|  | glVertex2i(ax,ay+3);//a |
|  | glVertex2i(ax-4,ay-3);//b |
|  | glVertex2i(ax+4,ay-3);//e |
|  | glEnd(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(30.1,21.0); |
|  | glVertex2f(29.9,21.0); |
|  | glVertex2f(30.5,0.0); |
|  | glVertex2f(29.5,0.0); |
|  | glEnd(); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(26.1,21.0); |
|  | glVertex2f(25.9,21.0); |
|  | glVertex2f(26.5,0.0); |
|  | glVertex2f(25.5,0.0); |
|  | glEnd(); |
|  | glColor3f(0.0,0.3,0.3); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(30.0,22.0); |
|  | glVertex2f(29.5,22.0); |
|  | glVertex2f(29.0,19.0); |
|  | glVertex2f(28.5,19.0); |
|  | glEnd(); |
|  | glBegin(GL\_TRIANGLE\_STRIP); |
|  | glVertex2f(26.5,22.0); |
|  | glVertex2f(26.0,22.0); |
|  | glVertex2f(25.5,19.0); |
|  | glVertex2f(25.0,19.0); |
|  | glEnd(); |
|  | glColor3f(0,0,d); |
|  | glEnable(GL\_LINE\_STIPPLE); |
|  | glLineStipple(1,0x00FF); |
|  | glBegin(GL\_LINES); |
|  | glVertex2f(25.5,19.0); |
|  | glVertex2f(25.5,-1.0); |
|  | glVertex2f(25.0,19.0); |
|  | glVertex2f(25.0,-1.0); |
|  | glVertex2f(25.25,19.0); |
|  | glVertex2f(25.25,-1.0); |
|  | glEnd(); |
|  | glColor3f(0,0,e); |
|  | glBegin(GL\_LINES); |
|  | glVertex2f(28.5,19.0); |
|  | glVertex2f(28.5,-1.0); |
|  | glVertex2f(29.0,19.0); |
|  | glVertex2f(29.0,-1.0); |
|  | glVertex2f(28.75,19.0); |
|  | glVertex2f(28.75,-1.0); |
|  | glEnd(); |
|  | glDisable(GL\_LINE\_STIPPLE); |
|  | glColor3f(0,0,1); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex2f(25.0,-1.0); |
|  | glVertex2f(29.5,-1.0); |
|  | glVertex2f(29.5,-2.0); |
|  | glVertex2f(25.0,-2.0); |
|  | glEnd(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex2f(24.5,-1.0); |
|  | glVertex2f(25.0,-1.0); |
|  | glVertex2f(25.0,-2.0); |
|  | glVertex2f(24.5,-2.0); |
|  | glEnd(); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex2f(24.5,-2.0); |
|  | glVertex2f(30.0,-2.0); |
|  | glVertex2f(30.0,-3.0); |
|  | glVertex2f(24.5,-3.0); |
|  | glEnd(); |
|  | glBegin(GL\_POLYGON); |
|  | glVertex2f(29.5,-1.0); |
|  | glVertex2f(30.0,-1.0); |
|  | glVertex2f(30.0,-2.0); |
|  | glVertex2f(29.5,-2.0); |
|  | glEnd(); |
|  | } |
|  | void woman() |
|  | { |
|  | glClearColor(0.48,0.5,0.5,0.0); |
|  | glColor3f(1.0,0.0,0.0); |
|  | glBegin(GL\_POLYGON);//veil |
|  | glVertex2f(21.0+x,-17.0+y); |
|  | glVertex2f(22.0+x,-17.0+y); |
|  | glVertex2f(22.5+x,-20.0+y); |
|  | glVertex2f(20.5+x,-20.0+y); |
|  | glEnd(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_POLYGON);//face |
|  | glVertex2f(21.0+x,-18.0+y); |
|  | glVertex2f(21.0+x,-17.0+y); |
|  | glVertex2f(22.0+x,-17.0+y); |
|  | glVertex2f(22.0+x,-18.0+y); |
|  | glEnd(); |
|  | glBegin(GL\_POLYGON);//neck |
|  | glVertex2f(21.5+x,-17.0+y); |
|  | glVertex2f(21.6+x,-17.0+y); |
|  | glVertex2f(21.6+x,-18.5+y); |
|  | glVertex2f(21.5+x,-18.5+y); |
|  | glEnd(); |
|  | glColor3f(0.97,0.45,0.84); |
|  | glBegin(GL\_POLYGON);//body |
|  | glVertex2f(21.0+x,-18.5+y); |
|  | glVertex2f(22.1+x,-18.5+y); |
|  | glVertex2f(22.1+x,-20.0+y); |
|  | glVertex2f(21.0+x,-20.0+y); |
|  | glEnd(); |
|  | glColor3f(0.59,0.137,0.985); |
|  | glBegin(GL\_POLYGON);//skirt |
|  | glVertex2f(21.0+x,-20.0+y); |
|  | glVertex2f(22.1+x,-20.0+y); |
|  | glVertex2f(22.7+x,-21.0+y); |
|  | glVertex2f(20.5+x,-21.0+y); |
|  | glEnd(); |
|  | glColor3f(0.0,0.0,0.0); |
|  | glPointSize(1.4); |
|  | glBegin(GL\_POINTS);//eyes |
|  | glVertex2f(21.3+x,-17.30+y); |
|  | glVertex2f(21.8+x,-17.30+y); |
|  | glEnd(); |
|  | glBegin(GL\_LINES);//nose |
|  | glVertex2f(21.6+x,-17.6+y); |
|  | glVertex2f(21.6+x,-17.3+y); |
|  | glEnd(); |
|  | glBegin(GL\_LINES);//smile |
|  | glVertex2f(21.5+x,-17.8+y); |
|  | glVertex2f(21.8+x,-17.8+y); |
|  | glEnd(); |
|  | glColor3f(0.3,0.15,0.1); |
|  | glBegin(GL\_POLYGON);//hand 1 |
|  | glVertex2f(21.0+x,-18.5+y); |
|  | glVertex2f(20.5+x,-20.0+y); |
|  | glVertex2f(21.0+x,-19.0+y); |
|  | glEnd(); |
|  | glBegin(GL\_POLYGON);//hand 2 |
|  | glVertex2f(22.1+x,-18.5+y); |
|  | glVertex2f(22.7+x,-19.0+y); |
|  | glVertex2f(22.1+x,-19.0+y); |
|  | glEnd(); |
|  | glColor3f(1,1,0.4); |
|  | glBegin(GL\_POLYGON);//torch |
|  | glVertex2f(22.7+x,-19.0+y); |
|  | glVertex2f(22.4+x,-19.0+y); |
|  | glVertex2f(22.5+x,-19.7+y); |
|  | glVertex2f(23.0+x,-19.7+y); |
|  | glEnd(); |
|  | glColor3f(c,c,c); |
|  | glBegin(GL\_POLYGON);//torch light |
|  | glVertex2f(22.7+x,-19.2+y); |
|  | glVertex2f(22.4+x,-19.2+y); |
|  | glVertex2f(22.5+x,-19.7+y); |
|  | glVertex2f(23.0+x,-19.7+y); |
|  | glEnd(); |
|  | glColor3f(c,c,c); |
|  | glEnable(GL\_LINE\_STIPPLE); |
|  | glLineStipple(1,0x00FF); |
|  | glBegin(GL\_LINES); |
|  | glVertex2f(22.5+x,-19.2+y); |
|  | glVertex2f(24.0+x,-25.0+y); |
|  | glVertex2f(22.6+x,-19.2+y); |
|  | glVertex2f(25.5+x,-25.0+y); |
|  | glVertex2f(22.7+x,-19.2+y); |
|  | glVertex2f(27.0+x,-25.0+y); |
|  | glEnd(); |
|  | } |
|  | void mykey(unsigned char key,int m,int n) |
|  | { |
|  | if(key=='w') y+=.1,x-=0.1; |
|  | if(key=='s') y-=.1,x+=0.1; |
|  | glutPostRedisplay(); |
|  | } |
|  | void display(void) |
|  | { |
|  | int b=0; |
|  | glClear(GL\_COLOR\_BUFFER\_BIT|GL\_DEPTH\_BUFFER\_BIT); |
|  | if(z<50) |
|  | { |
|  | for(z=0;z<=1500;z++) |
|  | { |
|  | front(); |
|  | glutPostRedisplay(); |
|  | glutSwapBuffers(); |
|  | glFlush(); |
|  | } |
|  | } |
|  | else |
|  | { |
|  | background(); |
|  | fanhouse(); |
|  | text(); |
|  | road(); |
|  | hut(-29,-23,-24,-33); |
|  | hut(0,-11,5,-21); |
|  | hut(-21,-1,-14,-11); |
|  | clouds(); |
|  | powerstation(); |
|  | wires(); |
|  | streetlight(); |
|  | woman(); |
|  | fanpole1(); |
|  | fanpole2(); |
|  | fanpole3(); |
|  | fan1(); |
|  | fan2(); |
|  | fan3(); |
|  | fan4(); |
|  | glutSwapBuffers(); |
|  | glFlush(); |
|  | } |
|  | } |
|  | void spinclockwise(void) |
|  | { |
|  | w=0.3;u=0;v=1;b=0.5;c=1;d=1;e=0; |
|  | a=a+0.1; |
|  | if(a>40) |
|  | a-=100.0; |
|  | spin=spin-1.0; |
|  | if(spin<0) |
|  | spin=spin+360.0; |
|  | glutPostRedisplay(); |
|  | } |
|  | void anticlockwise(void ) |
|  | { |
|  | u=0;w=.3;v=1;b=0.5;c=1;d=1,e=0; |
|  | if(a==40) |
|  | a=40; |
|  | a=a-0.1; |
|  | if(a<-100) |
|  | a+=100.0; |
|  | if(spin==360.0) |
|  | spin=spin-360; |
|  | spin=spin+1.0; |
|  | if(spin>360.0) |
|  | spin=spin-360.0; |
|  | glutPostRedisplay(); |
|  | } |
|  | void spinclockwise1(void) |
|  | { |
|  | u=0;w=.4;v=1;b=1;c=0.0;d=e=1; |
|  | a=a+0.3; |
|  | if(a>40) |
|  | a-=100.0; |
|  | spin=spin-10.0; |
|  | if(spin<0) |
|  | spin=spin+360.0; |
|  | glutPostRedisplay(); |
|  | } |
|  | void anticlockwise1(void ) |
|  | { |
|  | u=0;w=.4;v=1;b=1;c=0.0;d=e=1; |
|  | if(a==40) |
|  | a=40; |
|  | a=a-0.3; |
|  | if(a<-100) |
|  | a+=100.0; |
|  | if(spin==360.0) |
|  | spin=spin-360; |
|  | spin=spin+10.0; |
|  | if(spin>360.0) |
|  | spin=spin-360.0; |
|  | glutPostRedisplay(); |
|  | } |
|  | void reshape(int w, int h) |
|  | { |
|  | glViewport(0, 0, (GLsizei) w, (GLsizei) h); |
|  | glMatrixMode(GL\_PROJECTION); |
|  | glLoadIdentity(); |
|  | glOrtho(-35.0, 35.0, -45.0, 45.0, -20.0, 20.0); |
|  | glMatrixMode(GL\_MODELVIEW); |
|  | glLoadIdentity(); |
|  | } |
|  | void menu(int id ) |
|  | { |
|  | switch(id) |
|  | { |
|  | case 1: u=v=w=b=0.45;c=d=e=1; |
|  | glutIdleFunc(display); |
|  | break; |
|  | case 2: glutIdleFunc(spinclockwise); |
|  | break; |
|  | case 3: glutIdleFunc(anticlockwise); |
|  | break; |
|  | case 4: glutIdleFunc(spinclockwise1); |
|  | break; |
|  | case 5: glutIdleFunc(anticlockwise1); |
|  | break; |
|  | case 6:exit(0); |
|  | } |
|  | } |
|  | int main(int argc,char \*\*argv) |
|  | { |
|  | glutInit(&argc,argv); |
|  | glutInitDisplayMode(GLUT\_DOUBLE|GLUT\_RGB|GLUT\_DEPTH); |
|  | glutInitWindowSize(500,500); |
|  | glutInitWindowPosition(100,100); |
|  | glutCreateWindow("WIND ENERGY"); |
|  | init(); |
|  | glutDisplayFunc(display); |
|  | glutReshapeFunc(reshape); |
|  | glutKeyboardFunc(mykey); |
|  | glutCreateMenu(menu); |
|  | glutAddMenuEntry("No Wind",1); |
|  | glutAddMenuEntry("Wind CW",2); |
|  | glutAddMenuEntry("Wind ACW",3); |
|  | glutAddMenuEntry ("Fast Wind CW",4); |
|  | glutAddMenuEntry("Fast Wind ACW",5); |
|  | glutAddMenuEntry("Quit",6); |
|  | glutAttachMenu(GLUT\_RIGHT\_BUTTON); |
|  | glutMainLoop(); |
|  | return 0; |
|  | } |

a