#include<windows.h>

#include<stdlib.h>

#include<GL/glut.h>

#include<stdio.h>

#include<string.h>

#define maxx 81 //start of mesh variables //

#define maxy 6

#define dx 10

#define dy 19

GLfloat x[maxx],y[maxy];

GLfloat x0=100,y0=50,x3=100,y3=732;// end of mesh variables//

GLint x1=100,x2=160,y1=274,y2=274; // variables of road lines //

GLfloat m1=100,m2=150,m3=175,m4=160,m5=103; //varia

GLfloat n1=900,n2=885,n3=900,n4=840,n5=830;

GLfloat o1=100,o2=182,o3=190,o4=202,o5=100,o6=222;

GLfloat p1=885,p2=895,p3=880,p4=860;

GLfloat q1=100,q2=105,q3=180,q4=104;

GLfloat r1=900,r2=870,r3=860,r4=900;

GLint li;

GLint flag=0,f=0,g=0,d=0,s=0,b1=0,e=0,t=0,z1=0,sw=0;

GLfloat a=490;

GLfloat b=150;

void init()

{

glClearColor(0,0,0,0);

glColor3f(1,0,0);

glPointSize(1000);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(0,1000,0,1000);

glutPostRedisplay();

}

void output(int w, int e, char \*string)

{

int len, i;

glRasterPos3f(w,e,0);

len = (int) strlen(string);

for (i = 0; i < len; i++)

{

glutBitmapCharacter(GLUT\_BITMAP\_TIMES\_ROMAN\_24, string[i]);

}

}

void output1(int x, int y, char \*string)

{

int len, i;

glRasterPos3f(x, y,0);

len = (int) strlen(string);

for (i = 0; i < len; i++)

{

glutBitmapCharacter(GLUT\_BITMAP\_TIMES\_ROMAN\_10, string[i]);

}

}

void startscreen()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(0.90,0.65,0.10);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

//glColor3f(1,0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

glColor3f(0.50,0,0);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

//glColor3f(1,0,0);

glVertex2f(100,0);

glVertex2f(100,1000);

glEnd();

glColor3f(0.50,0,0);

glBegin(GL\_POLYGON);

glVertex2f(900,1000);

glVertex2f(900,0);

//glColor3f(1,0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

glColor3f(0.35,0.16,0.14);

output(230,850,"DAYANANDA SAGAR INSTITUTION");

output(420,750,"MINI PROJECT");

glLineWidth(4);

glColor3f(0.64,0.16,0.16);

output(410,700,"'HOP TO HOP'");

glColor3f(0.16,0.16,0.16);

output(230,600,"By,");

glColor3f(1,1,1);

output(260,550,"Sinduja .N 1DT11CS047");

output(260,500,"Kavitha.G.K 1DT11CS014");

glColor3f(0.1,0.3,0);

output(640,70,"press ENTER to Play");

}

void secondscreen()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(0.5,0.2,0.1);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

//glColor3f(1,0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

glColor3f(0.4,0.8,0.6);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

//glColor3f(1,0,0);

glVertex2f(100,0);

glVertex2f(100,1000);

glEnd();

glColor3f(0.4,0.8,0.6);

glBegin(GL\_POLYGON);

glVertex2f(900,1000);

glVertex2f(900,0);

//glColor3f(1,0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

output(200,800,"PRESS...");

glColor3f(0.23,0.67,0.1);

output(250,700,"1: PLAY");

output(250,600,"2: INSTRUCTIONS");

output(250,500,"q: EXIT");

}

void thirdscreen()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(0.5,0.2,0.1);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

//glColor3f(1,0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

glColor3f(0.4,0.8,0.6);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

//glColor3f(1,0,0);

glVertex2f(100,0);

glVertex2f(100,1000);

glEnd();

glColor3f(0.4,0.8,0.6);

glBegin(GL\_POLYGON);

glVertex2f(900,1000);

glVertex2f(900,0);

//glColor3f(1,0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

output(300,900,"SELECT THE CHARACTER OF YOUR CHOICE");

glColor3f(0.12,0.65,0.1);

output(250,700,"Character1: Press $ for Spider Man");

output(250,600,"Character2: Press & for Super Man");

glColor3f(0.54,0.0,0.9);

output(350,400,"Watch The RACE Of Your POWER");

}

void goldcup(void)

{

glColor3f(1,1,0); //rectangle

glBegin(GL\_POLYGON);

glVertex2f(400,750);

glVertex2f(600,750);

glVertex2f(600,400);

glVertex2f(400,400);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //rectangle border

glVertex2f(400,750);

glVertex2f(600,750);

glVertex2f(600,400);

glVertex2f(400,400);

glEnd();

glColor3f(1,1,0);

glBegin(GL\_POLYGON); //triangle

glVertex2f(400,400);

glVertex2f(600,400);

glVertex2f(500,300);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //triangle

glVertex2f(400,400);

glVertex2f(600,400);

glVertex2f(500,300);

glEnd();

glColor3f(1,1,0); //square

glBegin(GL\_POLYGON);

glVertex2f(450,350);

glVertex2f(550,350);

glVertex2f(550,250);

glVertex2f(450,250);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);//square border

glBegin(GL\_LINE\_LOOP);

glVertex2f(450,350);

glVertex2f(550,350);

glVertex2f(550,250);

glVertex2f(450,250);

glEnd();

glColor3f(1,1,0); //base

glBegin(GL\_POLYGON);

glVertex2f(450,250);

glVertex2f(550,250);

glVertex2f(600,150);

glVertex2f(400,150);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);//base border

glBegin(GL\_LINE\_LOOP);

glVertex2f(450,250);

glVertex2f(550,250);

glVertex2f(600,150);

glVertex2f(400,150);

glEnd();

glColor3f(0,0,0);

glLineWidth(10);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(600,675);

glVertex2f(650,675);

glVertex2f(650,550);

glVertex2f(600,550);

glEnd();

glColor3f(1,1,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(600,675);

glVertex2f(650,675);

glVertex2f(650,550);

glVertex2f(600,550);

glEnd();

glColor3f(0,0,0);

glLineWidth(10);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(400,675);

glVertex2f(350,675);

glVertex2f(350,550);

glVertex2f(400,550);

glEnd();

glColor3f(1,1,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(400,675);

glVertex2f(350,675);

glVertex2f(350,550);

glVertex2f(400,550);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,740);

glVertex2f(600,740);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,730);

glVertex2f(600,730);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,720);

glVertex2f(600,720);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,400);

glVertex2f(600,400);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(410,390);

glVertex2f(590,390);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(420,380);

glVertex2f(580,380);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(430,370);

glVertex2f(570,370);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(440,360);

glVertex2f(560,360);

glEnd();

glFlush();

}

void silvercup(void)

{

glColor3f(0.90,0.91,0.98); //rectangle

glBegin(GL\_POLYGON);

glVertex2f(400,750);

glVertex2f(600,750);

glVertex2f(600,400);

glVertex2f(400,400);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //rectangle border

glVertex2f(400,750);

glVertex2f(600,750);

glVertex2f(600,400);

glVertex2f(400,400);

glEnd();

glColor3f(0.90,0.91,0.98);

glBegin(GL\_POLYGON); //triangle

glVertex2f(400,400);

glVertex2f(600,400);

glVertex2f(500,300);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //triangle

glVertex2f(400,400);

glVertex2f(600,400);

glVertex2f(500,300);

glEnd();

glColor3f(0.90,0.91,0.98); //square

glBegin(GL\_POLYGON);

glVertex2f(450,350);

glVertex2f(550,350);

glVertex2f(550,250);

glVertex2f(450,250);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);//square border

glBegin(GL\_LINE\_LOOP);

glVertex2f(450,350);

glVertex2f(550,350);

glVertex2f(550,250);

glVertex2f(450,250);

glEnd();

glColor3f(0.90,0.91,0.98); //base

glBegin(GL\_POLYGON);

glVertex2f(450,250);

glVertex2f(550,250);

glVertex2f(600,150);

glVertex2f(400,150);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);//base border

glBegin(GL\_LINE\_LOOP);

glVertex2f(450,250);

glVertex2f(550,250);

glVertex2f(600,150);

glVertex2f(400,150);

glEnd();

glColor3f(0,0,0);

glLineWidth(10);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(600,675);

glVertex2f(650,675);

glVertex2f(650,550);

glVertex2f(600,550);

glEnd();

glColor3f(0.90,0.91,0.98);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(600,675);

glVertex2f(650,675);

glVertex2f(650,550);

glVertex2f(600,550);

glEnd();

glColor3f(0,0,0);

glLineWidth(10);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(400,675);

glVertex2f(350,675);

glVertex2f(350,550);

glVertex2f(400,550);

glEnd();

glColor3f(0.90,0.91,0.98);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(400,675);

glVertex2f(350,675);

glVertex2f(350,550);

glVertex2f(400,550);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,740);

glVertex2f(600,740);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,730);

glVertex2f(600,730);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,720);

glVertex2f(600,720);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,400);

glVertex2f(600,400);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(410,390);

glVertex2f(590,390);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(420,380);

glVertex2f(580,380);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(430,370);

glVertex2f(570,370);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(440,360);

glVertex2f(560,360);

glEnd();

glFlush();

}

void bronzecup(void)

{

glColor3f(0.55,0.47,0.14); //rectangle

glBegin(GL\_POLYGON);

glVertex2f(400,750);

glVertex2f(600,750);

glVertex2f(600,400);

glVertex2f(400,400);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //rectangle border

glVertex2f(400,750);

glVertex2f(600,750);

glVertex2f(600,400);

glVertex2f(400,400);

glEnd();

glColor3f(0.55,0.47,0.14);

glBegin(GL\_POLYGON); //triangle

glVertex2f(400,400);

glVertex2f(600,400);

glVertex2f(500,300);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //triangle

glVertex2f(400,400);

glVertex2f(600,400);

glVertex2f(500,300);

glEnd();

glColor3f(0.55,0.47,0.14); //square

glBegin(GL\_POLYGON);

glVertex2f(450,350);

glVertex2f(550,350);

glVertex2f(550,250);

glVertex2f(450,250);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);//square border

glBegin(GL\_LINE\_LOOP);

glVertex2f(450,350);

glVertex2f(550,350);

glVertex2f(550,250);

glVertex2f(450,250);

glEnd();

glColor3f(0.55,0.47,0.14); //base

glBegin(GL\_POLYGON);

glVertex2f(450,250);

glVertex2f(550,250);

glVertex2f(600,150);

glVertex2f(400,150);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);//base border

glBegin(GL\_LINE\_LOOP);

glVertex2f(450,250);

glVertex2f(550,250);

glVertex2f(600,150);

glVertex2f(400,150);

glEnd();

glColor3f(0,0,0);

glLineWidth(10);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(600,675);

glVertex2f(650,675);

glVertex2f(650,550);

glVertex2f(600,550);

glEnd();

glColor3f(0.55,0.47,0.14);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(600,675);

glVertex2f(650,675);

glVertex2f(650,550);

glVertex2f(600,550);

glEnd();

glColor3f(0,0,0);

glLineWidth(10);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(400,675);

glVertex2f(350,675);

glVertex2f(350,550);

glVertex2f(400,550);

glEnd();

glColor3f(0.55,0.47,0.14);

glLineWidth(3);

glBegin(GL\_LINE\_LOOP); //side

glVertex2f(400,675);

glVertex2f(350,675);

glVertex2f(350,550);

glVertex2f(400,550);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,740);

glVertex2f(600,740);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,730);

glVertex2f(600,730);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,720);

glVertex2f(600,720);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(400,400);

glVertex2f(600,400);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(410,390);

glVertex2f(590,390);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(420,380);

glVertex2f(580,380);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(430,370);

glVertex2f(570,370);

glEnd();

glColor3f(0,0,0);

glLineWidth(3);

glBegin(GL\_LINES);

glVertex2f(440,360);

glVertex2f(560,360);

glEnd();

glFlush();

}

void background(void)

{

int i,j;

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);// start of my reshape manually //

glVertex2f(0,0);

glVertex2f(100,0);

glVertex2f(100,1000);

glVertex2f(0,1000);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);

glVertex2f(0,950);

glVertex2f(1000,950);

glVertex2f(1000,1000);

glVertex2f(0,1000);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);

glVertex2f(0,0);

glVertex2f(1000,0);

glVertex2f(1000,50);

glVertex2f(0,50);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);

glVertex2f(900,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glVertex2f(900,1000);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);

glVertex2f(0,0);

glVertex2f(100,0);

glVertex2f(100,1000);

glVertex2f(0,1000);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);

glVertex2f(0,950);

glVertex2f(1000,950);

glVertex2f(1000,1000);

glVertex2f(0,1000);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.62,0.35,0.62);

glVertex2f(0,0);

glVertex2f(1000,0);

glVertex2f(1000,50);

glVertex2f(0,50);

glEnd();

glLineWidth(4);

glBegin(GL\_LINE\_LOOP);

glColor3f(0.57,0.85,0.43);

glVertex2f(100,50);

glVertex2f(900,50);

glVertex2f(900,950);

glVertex2f(100,950);

glEnd(); // end of my reshape manually //

glBegin(GL\_POLYGON);

glColor3f(0.75,0.75,0.75); // start of bottom plat form //

glVertex2f(100,50);

glVertex2f(900,50);

glVertex2f(900,162);

glVertex2f(100,162);

glEnd();

for(i=0;i<maxx;i++) // start of bottom platform mesh //

x[i]=x0+i\*dx;

for(j=0;j<maxy;j++)

y[j]=y0+j\*dy;

for(i=0;i<maxx-1;i++)

{

for(j=0;j<maxy-1;j++)

{

glColor3f(0.4,0.4,0.4);

glBegin(GL\_LINE\_LOOP);

glVertex2f(x[i],y[j]);

glVertex2f(x[i],y[j+1]);

glColor3f(0.2,0.2,0.2);

glVertex2f(x[i+1],y[j+1]);

glVertex2f(x[i+1],y[j]);

glEnd();

}

glFlush();

} // end of bottom plat form mesh //

glBegin(GL\_POLYGON); // start of road //

glColor3f(0.5,0.5,0.5);

glVertex2f(100,162);

glVertex2f(900,162);

glVertex2f(900,722);

glVertex2f(100,722);

glEnd(); // end of road //

glFlush();

glBegin(GL\_POLYGON); // start of top platform

glColor3f(0.75,0.75,0.75);

glVertex2f(100,722);

glVertex2f(900,722);

glVertex2f(900,834);

glVertex2f(100,834);

glEnd(); // end of top platform //

for(i=0;i<maxx;i++) // start of top platform mesh //

x[i]=x3+i\*dx;

for(j=0;j<maxy;j++)

y[j]=y3+j\*dy;

for(i=0;i<maxx-1;i++)

{

for(j=0;j<maxy-1;j++)

{

glColor3f(0.4,0.4,0.4);

glBegin(GL\_LINE\_LOOP);

glVertex2f(x[i],y[j]);

glVertex2f(x[i],y[j+1]);

glColor3f(0.2,0.2,0.2);

glVertex2f(x[i+1],y[j+1]

);

glVertex2f(x[i+1],y[j]);

glEnd();

}

glFlush();

} //end of top platform mesh//

glBegin(GL\_POLYGON);

glColor3f(0.19,0.85,0.19); //start of destination and green polygon//

glVertex2f(100,834);

glVertex2f(900,834);

glVertex2f(900,950);

glVertex2f(100,950);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.55,0.47,0.14);

glVertex2f(150,834);

glVertex2f(240,834);

glVertex2f(240,930);

glVertex2f(150,930);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.90,0.91,0.98);

glVertex2f(460,834);

glVertex2f(550,834);

glVertex2f(550,930);

glVertex2f(460,930);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(1,0.93,0.67); //bronze plat form//

glVertex2f(760,834);

glVertex2f(850,834);

glVertex2f(850,930);

glVertex2f(760,930);

glEnd();

glBegin(GL\_LINE\_LOOP);

glColor3f(0.64,0.16,0.16);

glVertex2f(150,834);

glVertex2f(240,834);

glVertex2f(240,930);

glVertex2f(150,930);

glEnd();

glBegin(GL\_LINE\_LOOP);

glColor3f(0.64,0.16,0.16);

glVertex2f(460,834);

glVertex2f(550,834);

glVertex2f(550,930);

glVertex2f(460,930);

glEnd();

glBegin(GL\_LINE\_LOOP);

glColor3f(0.64,0.16,0.16);

glVertex2f(760,834);

glVertex2f(850,834);

glVertex2f(850,930);

glVertex2f(760,930);

glEnd();

glBegin(GL\_LINE\_LOOP);

glColor3f(0.5,0.1,0.4);

glVertex2f(100,50);

glVertex2f(900,50);

glVertex2f(900,950);

glVertex2f(100,950);

glEnd();

glFlush(); // end of destination decorations and end of back ground //

}

void lines()

{

int i1,j1;

int k=0;

for(i1=0;i1<=3;i1++)//road lines

{

for(j1=0;j1<=23;j1++)

{

glLineWidth(3);

if(k%2==0)

{

glColor3f(0.5,0.5,0.5);

glBegin(GL\_LINES);

glVertex2f(x1,y1);

glVertex2f(x2,y2);

glEnd();

x1=x1+25;

x2=x2+25;

k++;

}

else

{

glColor3f(0.74,0.74,0.74);

glBegin(GL\_LINES);

glVertex2f(x1,y1);

glVertex2f(x2,y2);

glEnd();

x1=x1+40;

x2=x2+40;

k++;

}

}

x1=100;

x2=160;

y1=y1+112;

y2=y2+112;

}

y1=274;

y2=274;

}

void car1()

{

//glClear(GL\_COLOR\_BUFFER\_BIT);

glBegin(GL\_POLYGON);

glColor3f(0.89,0.47,0.29);

glVertex2f(m1,177);

glVertex2f(m1+80,177);

// glColor3f(0,1,0);

glVertex2f(m1+80,259);

glVertex2f(m1,259);

glEnd();

glFlush();

glBegin(GL\_POLYGON);

glColor3f(0,0.54,0);

glVertex2f(m2,182);

glVertex2f(m2+20,182);

//glColor3f(1,0,0.76);

glVertex2f(m2+20,254);

glVertex2f(m2,254);

glEnd();

glLineWidth(9);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(m4,261);

glVertex2f(m4+18,261);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(m4,177);

glVertex2f(m4+18,177);

glEnd();

glLineWidth(9);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(m5,261);

glVertex2f(m5+20,261);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(m5,177);

glVertex2f(m5+20,177);

glEnd();

glLineWidth(9);

glBegin(GL\_LINES);

glColor3f(1,0,1);

glVertex2f(m3,256);

glVertex2f(m3,248);

glEnd();

glBegin(GL\_LINES);

glColor3f(1,0,1);

glVertex2f(m3,179);

glVertex2f(m3,187);

glEnd();

glutPostRedisplay();

}

void racecar()

{

//glClear(GL\_COLOR\_BUFFER\_BIT);

glBegin(GL\_POLYGON);

glColor3f(0.98,0,0);

glVertex2f(n1,289);

glVertex2f(n1,371);

// glColor3f(0,1,0);

glVertex2f(n1-80,330);

//glVertex2f(m1,259);

glEnd();

glFlush();

glBegin(GL\_POLYGON);

glColor3f(0,0.54,0);

glVertex2f(n2,304);

glVertex2f(n2,356);

//glColor3f(1,0,0.76);

glVertex2f(n2-15,356);

glVertex2f(n2-15,304);

glEnd();

glLineWidth(9);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(n3,371);

glVertex2f(n3-18,371);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(n3,289);

glVertex2f(n3-18,289);

glEnd();

glLineWidth(9);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(n4,371);

glVertex2f(n4-18,371);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(n4,289);

glVertex2f(n4-20,289);

glEnd();

glLineWidth(4);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(n5,330);

glVertex2f(n5,289);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(n5,330);

glVertex2f(n5,371);

glEnd();

glutPostRedisplay();

}

void lorry()

{

//glClear(GL\_COLOR\_BUFFER\_BIT);

glBegin(GL\_POLYGON);

glColor3f(0,1,0);

glVertex2f(o2,483);

glVertex2f(o2,401);

glVertex2f(o2+30,401);

glVertex2f(o2+45,421);

glVertex2f(o2+45,463);

glVertex2f(o2+30,483);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.2,0,1);

glVertex2f(o1,401);

glVertex2f(o1,483);

glVertex2f(o1+80,483);

glVertex2f(o1+80,401);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.2,0,1);

glVertex2f(o3,473);

glVertex2f(o3,411);

glVertex2f(o3+15,411);

glVertex2f(o3+15,473);

glEnd();

glLineWidth(8);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o4,483);

glVertex2f(o4+15,483);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o4,401);

glVertex2f(o4+15,401);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5,401);

glVertex2f(o5+10,401);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5,483);

glVertex2f(o5+10,483);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5+12,483);

glVertex2f(o5+22,483);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5+12,401);

glVertex2f(o5+22,401);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5+55,483);

glVertex2f(o5+65,483);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5+55,401);

glVertex2f(o5+65,401);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5+67,483);

glVertex2f(o5+77,483);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(o5+67,401);

glVertex2f(o5+77,401);

glEnd();

glBegin(GL\_LINES);

glColor3f(1,0,0);

glVertex2f(o6,421);

glVertex2f(o6+5,421);

glEnd();

glBegin(GL\_LINES);

glColor3f(1,0,0);

glVertex2f(o6,463);

glVertex2f(o6+5,463);

glEnd();

glutPostRedisplay();

}

void tanker()

{

glBegin(GL\_POLYGON);

glColor3f(0.4,1,0.6);

glVertex2f(p1,528);

glColor3f(0,0.4,0);

glVertex2f(p1,580);

glVertex2f(p1-40,580);

glVertex2f(p1-40,528);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0,0,0);

glVertex2f(p2,598);

glVertex2f(p2,580);

glVertex2f(p2-70,580);

glVertex2f(p2-70,598);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0,0,0);

glVertex2f(p2,513);

glVertex2f(p2,528);

glVertex2f(p2-70,528);

glVertex2f(p2-70,513);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0,0,0);

glVertex2f(p3,570);

glVertex2f(p3,538);

glVertex2f(p3-20,538);

glVertex2f(p3-20,570);

glEnd();

glLineWidth(6);

glBegin(GL\_LINES);

glVertex2f(p4,555);

glVertex2f(p4-60,555);

glEnd();

glLineWidth(9);

glBegin(GL\_LINES);

glVertex2f(p4-60,556);

glVertex2f(p4-70,556);

glEnd();

glutPostRedisplay();

}

void bus()

{

glBegin(GL\_POLYGON);

glColor3f(1,0.49,0);

glVertex2f(q1,630);

//glColor3f(0,0.4,0);

glVertex2f(q1,703);

glVertex2f(q1+85,703);

glVertex2f(q1+85,630);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0.92,0.92,0.98);

glVertex2f(q2,635);

//glColor3f(0,0.4,0);

glVertex2f(q2,697);

glVertex2f(q2+65,697);

glVertex2f(q2+65,635);

glEnd();

glColor3f(0,1,1);

glLineWidth(8);

glBegin(GL\_LINES);

glVertex2f(q3,695);

glVertex2f(q3+5,695);

glEnd();

glBegin(GL\_LINES);

glVertex2f(q3,635);

glVertex2f(q3+5,635);

glEnd();

glLineWidth(4);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(q4,703);

glVertex2f(q4+15,703);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(q4,630);

glVertex2f(q4+15,630);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(q4+60,703);

glVertex2f(q4+75,703);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(q4+60,630);

glVertex2f(q4+75,630);

glEnd();

glutPostRedisplay();

}

void bike()

{

glBegin(GL\_POLYGON);

glColor3f(0.8,0.2,0.4);

glVertex2f(r1,762);

//glColor3f(0,0.4,0);

glVertex2f(r1,794);

glVertex2f(r1-40,814);

glVertex2f(r1-65,794);

glVertex2f(r1-65,762);

glVertex2f(r1-40,742);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0,1,1);

glVertex2f(r2,794);

glVertex2f(r2,762);

glVertex2f(r2-10,752);

glVertex2f(r2-10,804);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(r3,814);

glVertex2f(r3-15,814);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(r3,742);

glVertex2f(r3-15,742);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(r4,798);

glVertex2f(r4-10,798);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(r4,757);

glVertex2f(r4-10,757);

glEnd();

glutPostRedisplay();

}

void spiderman()

{

glBegin(GL\_POLYGON);

glColor3f(1,0,0);

glVertex2f(a+5,b);

glVertex2f(a,b-10);

glVertex2f(a+10,b-30);

glVertex2f(a+15,b-30);

glVertex2f(a+25,b-10);

glVertex2f(a+20,b);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(1,0,0);

glVertex2f(a-1,b-30);

glColor3f(1,0,0);

glVertex2f(a-1,b-80);

glColor3f(1,0,0);

glVertex2f(a+26,b-80);

glColor3f(1,0,0);

glVertex2f(a+26,b-30);

glEnd();

glLineWidth(8);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a+5,b-80);

glVertex2f(a+5,b-100);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a+20,b-80);

glVertex2f(a+20,b-100);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a-1,b-36);

glVertex2f(a-18,b-46);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a+25,b-36);

glVertex2f(a+42,b-46);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(1,0,0);

glVertex2f(a+5,b-40);

glVertex2f(a+5,b-70);

glVertex2f(a+20,b-70);

glVertex2f(a+20,b-40);

glEnd();

}

void superman()

{

glBegin(GL\_POLYGON);

glColor3f(0,0,1);

glVertex2f(a+5,b);

glVertex2f(a,b-10);

glVertex2f(a+10,b-30);

glVertex2f(a+15,b-30);

glVertex2f(a+25,b-10);

glVertex2f(a+20,b);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0,0,1);

glVertex2f(a-1,b-30);

glColor3f(0,1,0);

glVertex2f(a-1,b-80);

glColor3f(0,0,1);

glVertex2f(a+26,b-80);

glColor3f(0,0,1);

glVertex2f(a+26,b-30);

glEnd();

glLineWidth(8);

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a+5,b-80);

glVertex2f(a+5,b-100);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a+20,b-80);

glVertex2f(a+20,b-100);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a-1,b-36);

glVertex2f(a-18,b-46);

glEnd();

glBegin(GL\_LINES);

glColor3f(0,0,0);

glVertex2f(a+25,b-36);

glVertex2f(a+42,b-46);

glEnd();

glBegin(GL\_POLYGON);

glColor3f(0,0,1);

glVertex2f(a+5,b-40);

glVertex2f(a+5,b-70);

glVertex2f(a+20,b-70);

glVertex2f(a+20,b-40);

glEnd();

}

void dead()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

if(d==1)

{

glColor3f(0.5,0.5,0.5);

glBegin(GL\_POLYGON);

glVertex2f(0,0);

glVertex2f(0,1000);

glVertex2f(1000,1000);

glVertex2f(1000,0);

glEnd();

glColor3f(1,0,0);

glBegin(GL\_POLYGON);

glVertex2f(100,700);

glVertex2f(100,300);

glVertex2f(580,300);

glVertex2f(580,700);

glEnd();

glColor3f(0,1,0);

glBegin(GL\_POLYGON);

glVertex2f(450,650);

glVertex2f(450,350);

glVertex2f(550,450);

glVertex2f(550,550);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(480,750);

glVertex2f(480,700);

glVertex2f(570,700);

glVertex2f(570,750);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(480,250);

glVertex2f(480,300);

glVertex2f(570,300);

glVertex2f(570,250);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(120,750);

glVertex2f(120,700);

glVertex2f(210,700);

glVertex2f(210,750);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(120,250);

glVertex2f(120,300);

glVertex2f(210,300);

glVertex2f(210,250);

glEnd();

glColor3f(0,1,0);

glBegin(GL\_POLYGON);

glVertex2f(150,650);

glVertex2f(150,350);

glVertex2f(400,350);

glVertex2f(400,650);

glEnd();

glColor3f(1,0,0);

glBegin(GL\_POLYGON);

glVertex2f(600,650);

glVertex2f(660,600);

glVertex2f(690,600);

glVertex2f(750,650);

glVertex2f(700,700);

glVertex2f(650,700);

glEnd();

glColor3f(0,1,0);

glBegin(GL\_POLYGON);

glColor3f(0,1,1);

glVertex2f(600,600);

glColor3f(0,1,0);

glVertex2f(600,350);

glColor3f(1,0,0);

glVertex2f(750,350);

glColor3f(1,0,1);

glVertex2f(750,600);

glEnd();

glColor3f(0,0,1);

glBegin(GL\_POLYGON);

glVertex2f(625,550);

glVertex2f(625,400);

glVertex2f(725,400);

glVertex2f(725,550);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(710,350);

glVertex2f(710,200);

glVertex2f(740,200);

glVertex2f(740,350);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(610,350);

glVertex2f(610,200);

glVertex2f(640,200);

glVertex2f(640,350);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(600,600);

glVertex2f(600,550);

glVertex2f(500,450);

glVertex2f(500,500);

glEnd();

glColor3f(0,0,0);

glBegin(GL\_POLYGON);

glVertex2f(750,600);

glVertex2f(750,550);

glVertex2f(840,450);

glVertex2f(840,500);

glEnd();

glColor3f(0.64,0.16,0.16);

output(450,800,"CRASHED");

output(100,10,"Press 'r' to replay at any time");

output(700,100,"Press 'q' to exit");

}

if(g==1)

{

glColor3f(0,0,1);

glColor3f(0.90,0.91,0.98);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

silvercup();

glColor3f(0.64,0.16,0.16);

output(450,800,"SILVERWINNER");

output(700,100,"Press 'q' to exit");

}

if(s==1)

{

glColor3f(1,1,1);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

bronzecup();

glColor3f(0.64,0.16,0.16);

output(450,800,"BRONZEWINNER");

output(700,100,"Press 'q' to exit");

}

if(b1==1)

{

glColor3f(1,1,1);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

goldcup();

glColor3f(0.64,0.16,0.16);

output(450,800,"GOLDWINNER");

output(700,100,"Press 'q' to exit");

}

if(t==1)

{

glColor3f(0,1,0);

glBegin(GL\_POLYGON);

glVertex2f(0,1000);

glVertex2f(0,0);

glVertex2f(1000,0);

glVertex2f(1000,1000);

glEnd();

glColor3f(0.64,0.16,0.16);

output(450,800,"you didn't get anything");

output(700,100,"Press 'q' to exit");

glFlush();

}

}

void keys(unsigned char key,int x,int y)

{

switch(key)

{

case 13:e=1;

glutPostRedisplay();

break;

case 'd':if(a!=850)

{

a=a+40;

}

glutPostRedisplay();

break;

case 'a':if(a!=130)

{

a=a-40;

}

glutPostRedisplay();

break;

case 'w':if(b!=934)

{

b=b+112;

}

if(b==934)

{

//printf("%f",a);

if(a+20>=490&&a+20<=560)

{

f=1;

g=1;

sw=1;

}

else if(a>=150&&a<=262)

{

f=1;

s=1;

sw=1;

}

else if(a+20>=760&&a+20<=850)

{

f=1;

b1=1;

sw=1;

}

else

{

f=1;

t=1;

sw=1;

}

}

glutPostRedisplay();

break;

case 's':if(b!=150)

{

b=b-112;

}

glutPostRedisplay();

break;

case '1':e=5;

//printf("%d\n",e);

glutPostRedisplay();

break;

case '2':e=3;

glutPostRedisplay();

break;

case '3':e=5;

glutPostRedisplay();

break;

case '$':e=2;

li=0;

printf("%d\n",e);

glutPostRedisplay();

break;

case '&':e=2;

li=1;

printf("%d\n",e);

glutPostRedisplay();

break;

case 'q':e=4;

glutPostRedisplay();

break;

case 'r': e=1;

glutPostRedisplay();

break;

}

}

void display(void)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

if(e==0)

{

startscreen();

}

else if(e==1)

{

secondscreen();

}

else if(e==5)

{

thirdscreen();

}

else if(e==2)

{

if(f!=1)

{

background();

lines();

car1();

m1=m1+1;

m2=m2+1;

m3=m3+1;

m4=m4+1;

m5=m5+1;

if(b==262)

{

if(a>=m1-20&&a<=m1+80)

{

f=1;

d=1;

sw=1;

}

}

if(m1==820)

{

m1=100;m2=150;m3=175;m4=160;m5=103;

}

racecar();

n1=n1-3;

n2=n2-3;

n3=n3-3;

n4=n4-3;

n5=n5-3;

if(b==374)

{

if(a<=n1+20&&a>=n1-90)

{

f=1;

d=1;

sw=1;

}

}

if((int)n1==150)

{

n1=900;n2=885;n3=900;n4=840;n5=830;

}

lorry();

o1+=2;

o2+=2;

o3+=2;

o4+=2;

o5+=2;

o6+=2;

if(b==486)

{

if(a>=o1-20&&a<=o1+120)

{

f=1;

d=1;

sw=1;

}

}

if(o1==780)

{

o1=100;o2=182;o3=190;o4=202;o5=100;o6=222;

}

tanker();

if(b==598)

{

if(a<=p1+10&&a>=p1-100)

{

f=1;

d=1;

sw=1;

}

}

p1-=1;

p2-=1;

p3-=1;

p4-=1;

if(p1==180)

{

p1=885;p2=895;p3=880;p4=860;

}

bus();

q1+=2;

q2+=2;

q3+=2;

q4+=2;

if(b==710)

{

if(a>=q1-20&&a<=q1+120)

{

sw=1;

f=1;

d=1;

}

}

if(q1==820)

{

q1=100;q2=105;q3=180;q4=104;

}

bike();

r1-=5;

r2-=5;

r3-=5;

r4-=5;

printf("%f\n",r1);

if(b==822)

{

if(a<=r1+10&&a>=r1-100)

{

f=1;

d=1;

sw=1;

}

}

if(r1==130)

{

r1=900;r2=870;r3=860;r4=900;

}

if(li==0)

{

spiderman();

}

else if(li==1) superman();

glColor3f(0.9,0.2,0);

output(700,10,"Press 'q' to exit at any time");

glColor3f(0.9,0.2,0);

output(100,10,"Press 'r' to replay at any time");

}

}

else if(e==3)

{

glColor3f(0.3,0.5,0.4);

output(600,100,"press enter to go back to MAIN MENU");

glColor3f(0,0.6,0.9);

output(250,700,"In this game you have to move man across the road and get settle in one of the GRID");

glColor3f(1,0.93,0.67);

output(250,600,"If get settled at GOLD platform then you win a GOLD cup");

glColor3f(0.91,0.91,0.98);

output(250,500,"If get settled at SILVER platform then you win a SILVER cup");

glColor3f(0.55,0.47,0.14);

output(250,400,"If get settled at BRONZE platform then you win a BRONZE cup");

glColor3f(0,0.6,0.9);

output(250,300,"press w->move UP");

output(250,250,"press s->move DOWN");

output(250,200,"press a->move LEFT");

output(250,150,"press d->move RIGHT");

output(700,50,"Press 'q' to exit");

}

else if(e==4)

{

exit(0);

}

if(f==1)

{

dead();

}

glutSwapBuffers();

}

int main(int argc,char \*\*argv)

{

glutInit(&argc,argv);

glutInitDisplayMode(GLUT\_DOUBLE|GLUT\_RGB);

glutInitWindowSize(1350,690);

glutInitWindowPosition(0,0);

glutCreateWindow("hop to hop");

// glutReshapeFunc(myreshape);

glutDisplayFunc(display);

glutKeyboardFunc(keys);

init();

glutMainLoop();

}