```
#include<stdlib.h>

#include<GL/glut.h>

#include<time.h>

//#include<dos.h>

#include<stdio.h>

//#include<conio.h>

//#include<windows.h>


float bspd=0.02; // block dx value


char name[25];


float b1x=50.0,b1y=0;//block 1 init position


float hm=0.0;//copter moving dy value


int i=0,sci=1;float scf=1; // for increment score score_int score_flag


char scs[20],slevel[20];

//to store score_string using itoa() and level as well


int level=1,lflag=1,wflag=1; //level_flag & welcome_flag init w/ 1
```

```
void init(void)
{
srand(time(0));
b1y=(rand()%45)+10;//b/w 10 to 44

glClearColor (0.0, 0.0, 0.0, 0.0);
glShadeModel (GL_SMOOTH);
glLoadIdentity ();
glOrtho(0.0, 100.0, 0.0, 100.0, -1.0 , .0);
}

void drawcopter()
{
glColor3f(0.7,1.0,1.0);
glRectf(10,49.8,19.8,44.8);//body
glRectf(2,46,10,48);//tail
glRectf(2,46,4,51);//tail up
glRectf(14,49.8,15.8,52.2);//propeller stand
glRectf(7,53.6,22.8,52.2);//propeller*/
}

void renderBitmapString(float x,float y,float z,void *font,char*string)
{

char *c;
```

```
glRasterPos3f(x, y,z);


for(c=string; *c != '\0'; c++)

{

glutBitmapCharacter(font, *c);

}

}


void display(void)

{

glClear(GL_COLOR_BUFFER_BIT);

//GameOver Checking


if(

(i==730||i==-700)

//top and bottom checking


||

( ((int)b1x==10||(int)b1x==7||(int)b1x==4||(int)b1x==1)
&&(int)b1y<53+(int)hm&&(int)b1y+35>53+(int)hm)

        // propeller front checking


||

( ((int)b1x==9||(int)b1x==3||(int)b1x==6) &&(int)b1y<45+(int)hm&&(int)b1y+35>45+(int)hm)

//lower body checking
```

```
||
( ((int)b1x==0) && (int)b1y<46+(int)hm&&(int)b1y+35>46+(int)hm))
// lower tail checking

{

glColor3f(0.0,0.0,1.0);
glRectf(0.0,0.0,100.0,100.0);
glColor3f(1.0,0.0,0.0);
renderBitmapString(40,70,0,GLUT_BITMAP_HELVETICA_18,"GAME OVER!!!");
glColor3f(1.0,1.0,1.0);
renderBitmapString(25,58,0,GLUT_BITMAP_TIMES_ROMAN_24,"You");
renderBitmapString(45,58,0,GLUT_BITMAP_TIMES_ROMAN_24,"scored:");
renderBitmapString(70,58,0,GLUT_BITMAP_TIMES_ROMAN_24,scs);
glutSwapBuffers();
glFlush();
printf("\nGAME OVER\n\n");
printf("%s\You scored  %s" ,name,scs);
printf("\n\nClose the console window to exit...\n");
//getch();
exit(0);
}

else if(wflag==1)//Welcome Screen
```

```
{

wflag=0;

glColor3f(0.0,0.5,0.7);
glRectf(0.0,0.0,100.0,10.0);//ceil
glRectf(0.0,100.0,100.0,90.0);//floor

glColor3f(1.0,1.0,1.0);
renderBitmapString(35,85,0,GLUT_BITMAP_HELVETICA_18,"CITY ENGINEERING COLLEGE");

renderBitmapString(41,80,0,GLUT_BITMAP_HELVETICA_12,"Bangalore, Karnataka-560 062");

glColor3f(1.0,1.0,0.0);
renderBitmapString(20,65,0,GLUT_BITMAP_8_BY_13,"a mini project for Computer Graphics &
Visualization Laboratery");
renderBitmapString(45.5,70,0,GLUT_BITMAP_TIMES_ROMAN_24,"Helicopter");

glColor3f(1.0,0.0,0.0);
renderBitmapString(40,45,0,GLUT_BITMAP_TIMES_ROMAN_24,"Welcome");
renderBitmapString(53,45,0,GLUT_BITMAP_TIMES_ROMAN_24,name);
renderBitmapString(43,30,0,GLUT_BITMAP_TIMES_ROMAN_24,"Click To Start");
renderBitmapString(17,24,0,GLUT_BITMAP_9_BY_15,"CLICK AND HOLD LEFT MOUSE BUTTON TO GO UP
RELEASE TO GO DOWN");

glColor3f(0.0,0.0,0.0);
```

```
drawcopter();

glutSwapBuffers();
glFlush();

}

else
{

//on every increase by 50 in score in each level
if(sci%50==0&&lflag==1)
{
lflag=0; //make level_flag=0
level++;//increase level by 1
bspd+=0.01;//increase block_dx_speed by 0.01
}

//within every level make level_flag=1
else if(sci%50!=0&&lflag!=1)
{
lflag=1;
}

glPushMatrix();
```

```
glColor3f(0.0,0.5,0.7);
glRectf(0.0,0.0,100.0,10.0);     //ceil
glRectf(0.0,100.0,100.0,90.0);    //floor

glColor3f(0.0,0.0,0.0);   //score
renderBitmapString(1,3,0,GLUT_BITMAP_TIMES_ROMAN_24,"Distance:");
//glColor3f(0.7,0.7,0.7);

sprintf(slevel,"%d",level);   //level
renderBitmapString(80,3,0,GLUT_BITMAP_TIMES_ROMAN_24,"Level:");
renderBitmapString(93,3,0,GLUT_BITMAP_TIMES_ROMAN_24,slevel);

scf+=0.025;            //so less as program run very fast
sci=(int)scf;
sprintf(scs,"%d",sci);
//from int to char convertion to display score

renderBitmapString(20,3,0,GLUT_BITMAP_TIMES_ROMAN_24,scs);

glTranslatef(0.0,hm,0.0);
// hm(=dy) changes occur by mouse func

drawcopter();
//code for helicopter
```

```
//if wall move towards left & get out of projection volume
if(b1x<-10)
{

b1x=50;          //total width is 50
b1y=(rand()%25)+20;
//10 for selling+10 for giving enough space
// block bottom limit 0+20 & top limit 24+20=44

}

else
b1x-=bspd;
//within the projection volume dec its x value by block_speed

glTranslatef(b1x,-hm,0.0);

glColor3f(1.0,0.0,0.0);
glRectf(b1x,b1y,b1x+5,b1y+35);//block 1

glPopMatrix();

glutSwapBuffers();
glFlush();
```

```
}

}

void moveHeliU(void)
{

hm+=0.05;
i++;
glutPostRedisplay();

}

void moveHeliD()
{

hm-=0.05;
i--;
glutPostRedisplay();

}

void mouse(int button, int state, int x, int y)
```

```
{

switch (button)
{

case GLUT_LEFT_BUTTON:

if (state == GLUT_DOWN)
glutIdleFunc(moveHeliU);

else if (state == GLUT_UP)
glutIdleFunc(moveHeliD);
break;

default: break;

}

}

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

if(key=='w') glutIdleFunc(moveHeliU);
```

```
if(key=='m') glutIdleFunc(moveHeliD);


}


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


printf("enter your name to play: ");
scanf("%s",name);
glutInit(&argc, argv);
glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
glutInitWindowSize (800, 600);
glutInitWindowPosition (200,200);
glutCreateWindow ("2D Copter Game");
init();
glutDisplayFunc(display);
        glutMouseFunc(mouse);
        glutKeyboardFunc(keys);
        glutMainLoop();
        return 0;
        }
```

helicopter