

```

#include<GL/glut.h>
#include<stdlib.h>
#include<string.h>
void controls1();
void startmenu();
void aboutf();
void lev();
void over();
int x,y,w,f=0,i,j,k,v,wwh=1024;
int
hitFlag=0,messageTrue=0,startFlag=0,backFlag=0,controlsFlag=0,abo
utFlag=0,dirFlag=0,upFlag=0,overFlag=0,levFlag;
char wel[100]="WELCOME TO 'CATCH ME' GAME...";
char start[50]="1: START GAME.";
char controls[50]="2: HOW TO PLAY.";
char about[50]="4: ABOUT GAME.";
char levels[50]="3: LEVELS.";
char control1[500]="HI...\n TO START WITH GAME, AN OBJECT WILL BE
MOVING ON THE SCREEN.";
char control2[200]="YOU HAVE TO 'CATCH' THAT OBJECT BY CLICKING
LEFT BUTTON OF THE MOUSE ON THAT OBJECT.";
char control3[200]="THERE WILL BE TEN CHANCES AND EACH CHANCE
CARRIES 10 POINTS.";
char control4[200]="AND YOU CAN CHOOSE DIFFERENT LEVELS IN THE
GAME.";
char control5[200]="press '1' to START GAME and '3' to go 'ABOUT
GAME' or LEFT CLICK to go back.";
char control6[100]="***** ALL THE BEST *****";
char about1[100]="WELL, THE GAME NAME IS 'CATCH ME IF U
CAN!!!'.";
char about2[100]="THIS GAME IS DEVELOPED BY JIVRAJANI VAIBHAV AND
DHEERAJ KUMAR 420 ";
char about3[100]="FROM 6TH SEM CS USING OPENGL AS PART OF
MINIPROJECT.";
char about4[100]="press '1' to START GAME or '2' to go 'HOW TO
PLAY GAME' or LEFT CLICK to go back.";
char lev1[100]="CHOOSE THE LEVEL U WANT TO PLAY...";
char lev2[100]="a: EASY";
char lev3[100]="b: MEDIUM";
char lev4[100]="c: HARD";
char lev5[100]="LEFT CLICK to go back.";
char over1[100]="GAME OVER!!!";
char well1[100]="WELL DONE!!!";
char yourscore[100]="YOUR SCORE IS:";
char hardluck[100]="WELL, HARD LUCK, TRY AGAIN...";
char hundred[100]="100";
char max[100]="(please maximise the window before u start the
game..)";
char key1;
char sco[10]="SCORE:";
int score=0,rem,vs1=0,vs2=0;
int yy1=660,yy2=750;

```

```

int level1=0,level2=0,level3=0;
struct lag
{
    float x1,x2,y1,y2;
    float color;
}o;

void mypos()
{
    o.x1=90;
    o.y1=100;
    o.x2=o.x1+40;
    o.y2=o.y1+40;
}

void initfun()
{
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(0,1280,0,1024);

mypos();
}

void moveRight()
{
    if(level1==0&&level2==0&&level3==0) v=5;

    if(level1==1)
        v=5;
    else if(level2==1)
        v=10;
    else if(level3==1)
        v=15;

    o.x1=o.x1+v;
    o.y1=o.y1;
    o.x2=o.x1+40;
    o.y2=o.y1+40;
    glutPostRedisplay();
}

void randomGenerate()
{
    f++;
    if(f<10)
    {

```

```

        int num=rand()%620;
        if(num<90)
            num=num+100;

        o.x1=90;
        o.y1=num;
        o.x2=o.x1+40;
        o.y2=o.y1+40;

        hitFlag=0;
        glutPostRedisplay();
    }
    else
        over();
}

void display(void)
{

    glClear(GL_COLOR_BUFFER_BIT);
    if(startFlag==0)
    {
        glColor3f(1,1,1);
        glRectf(0,0,800,600);
        glColor3f(0,0,0);
        glRectf(10,10,1270,1014);
        glColor3f(0.6,0.7,0.8);
        glRasterPos2f(300,900);
        for(w=0;w<sizeof(wel);w++)
            glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,wel[w]);

        glRasterPos2f(250,850);
        for(w=0;w<sizeof(max);w++)
            glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,max[w]);

        glColor3f(0.5,0.1,0.7);
        glRasterPos2f(200,700);
        for(w=0;w<sizeof(start);w++)

            glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,start[w]);

        glColor3f(0.7,0.8,0.9);
        glRasterPos2f(200,600);
        for(w=0;w<sizeof(controls);w++)

            glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,controls[w]);
    }
}

```

```

        glColor3f(0.2,0.5,0.7);
        glRasterPos2f(200,500);
        for(w=0;w<sizeof(levels);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,levels[w]);


        glColor3f(1.0,0.8,0.3);
        glRasterPos2f(200,400);
        for(w=0;w<sizeof(about);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,about[w]);


        glFlush();
    }

    if(controlsFlag==1)
        //to go controls window
    {
        controls1();
    }

    if(aboutFlag==1)
        //to go about game window
    {
        aboutf();
    }

    if(levFlag==1)
    {
        lev();
    }

    if(startFlag==1) //to start
game
    {

        glColor3f(100.0/256.0, 0.0, 0.0);

        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        gluOrtho2D(0,1280,0,1024);


        glColor3f(1.0,100.0/256.0, 0.0);
        glRectf(0,0,1280,1024);
        glColor3f(220.0/256,150.0/256.0,0); //b1
        glRectf(50,50,880,700);
        glColor3f(0,100.0/256.0,0); //b2
        glRectf(70,70,860,680);
    }

```

```

        glColor3f(0,0,0);
//b3
        glRectf(90,90,840,660);
        vs2=score%10;
        vs1=score/10;

        glRasterPos2f(100,800);
        for(i=0;i<sizeof(sco);i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,sco[i]);

glRasterPos2f(170,800);
glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,vs1+48);
glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,vs2+48);


        if(hitFlag==0)
        {
            glColor3f(1.0,1.0,1.0);
//object;
            glRectf(o.x1,o.y1,o.x2,o.y2);
            glFlush();

            if(o.x2<=840)
                moveRight();

            if(o.x2>840)
                randomGenerate();

            glFlush();
        }

        glColor3f(0,0,1);
        glPointSize(5);
        glBegin(GL_POINTS);
        glVertex2f(x,y);
        glEnd();

        if(hitFlag==1)
        {
            hitFlag=0;
            glColor4f(1.0,1.0,1.0,1.0);

            char b[20]="hit...",c[20]="lost..";

            glRasterPos2f(o.x1,o.y1);
            if(messageTrue==1)
            {
                score+=10;
            }
        }
    }
}

```

```

        for(int i=0;i<7;i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,b[i]);


        /*while(score!=0)
        {
            rem=score%10;
            glColor3f(0,0,0);
            glRasterPos2f(100,800);

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,48+rem);
            score=score/10;
        }*/


        glFlush();
    }


    for(int t=0;t<1000;t++)
        for(int h=0;h<1000;h++)
            for(int y=0;y<100;y++)
                {}


        randomGenerate();
    }

}

glFlush();
}

void over()
{
    overFlag=0;
    glColor3f(0.56,0.40,0.8);
    glRectf(0,0,1280,1024);

    glColor3f(0.0,0.0,0.0);
    glRasterPos2f(500,700);
    for(w=0;w<sizeof(over1);w++)

```

```

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,over1[w]);

    if(score==100)
    {
        glRasterPos2f(400,600);
        for(i=0;i<sizeof(well);i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,well[i]);
        glRasterPos2f(400,400);
        for(i=0;i<sizeof(yourscore);i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,yourscore[i]);

        glRasterPos2f(570,400);
        for(i=0;i<sizeof(hundred);i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,hundred[i]);
    }
    else
    {
        glRasterPos2f(400,600);
        for(i=0;i<sizeof(yourscore);i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,yourscore[i]);
        glRasterPos2f(570,600);

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,vs1+48);

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,vs2+48);
        if(score==0)
        {
            glRasterPos2f(570,500);
            for(i=0;i<sizeof(hardluck);i++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,hardluck[i]);
        }
    }

glFlush();

for(i=0;i<50000;i++)
    for(j=0;j<50000;j++)
        //for(k=0;k<1000;k++)
        {}

    exit(1);
}

```

```

void controls1()
//function for controls window
{
    glColor3f(0.56,0.40,0.8);
    glRectf(0,0,1280,1024);

    glColor3f(0.3,0.45,0.76);
    glRectf(20,20,1260,1004);

    glColor3f(0.0,0.0,0.0);
    glRasterPos2f(100,800);
    for(w=0;w<sizeof(control1);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,control1[w]);

    glRasterPos2f(90,700);
    for(w=0;w<sizeof(control2);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,control2[w]);

    glRasterPos2f(100,600);
    for(w=0;w<sizeof(control3);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,control3[w]);

    glRasterPos2f(100,500);
    for(w=0;w<sizeof(control4);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,control4[w]);

    glRasterPos2f(100,400);
    for(w=0;w<sizeof(control5);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,control5[w]);

    glRasterPos2f(200,300);
    for(w=0;w<sizeof(control6);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,control6[w]);

    controlsFlag=0;

    glFlush();
}

```

```

void aboutf()
//function for about game

```



```

{
    glColor3f(1.0,0.40,1.0);
    glRectf(0,0,1280,1024);
    glColor3f(0.7,0.45,0.36);
    glRectf(20,20,1260,1004);

    glColor3f(0.0,0.0,0.0);
    glRasterPos2f(100,800);
    for(w=0;w<sizeof(about1);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,about1[w]);

    glRasterPos2f(100,700);
    for(w=0;w<sizeof(about2);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,about2[w]);

    glRasterPos2f(100,600);
    for(w=0;w<sizeof(about3);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,about3[w]);

    glRasterPos2f(100,500);
    for(w=0;w<sizeof(about4);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,about4[w]);

    aboutFlag=0;
    glFlush();
}

void startmenu()
{
    if(dirFlag==1)
    {
        dirFlag=0;upFlag=0;

        yy1-=100;
        yy2-=100;
        glColor3f(0.6,0.7,0.8);
        glRasterPos2f(300,900);
        for(w=0;w<sizeof(wel);w++)
            glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,wel[w]);

        glRasterPos2f(250,850);
        for(w=0;w<sizeof(max);w++)
            glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,max[w]);

        glColor3f(0.5,0.1,0.7);
        glRasterPos2f(200,700);
        for(w=0;w<sizeof(start);w++)

```

```

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,start[w]);

    glColor3f(0.7,0.8,0.9);
    glRasterPos2f(200,600);
    for(w=0;w<sizeof(controls);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,controls[w]);

    glColor3f(0.2,0.5,0.7);
    glRasterPos2f(200,500);
    for(w=0;w<sizeof(levels);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,levels[w]);

    glColor3f(1.0,0.8,0.3);
    glRasterPos2f(200,400);
    for(w=0;w<sizeof(about);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,about[w]);

    glFlush();
}

if(upFlag==1)
{
    upFlag=0;dirFlag=0;
    yy1+=100;
    yy2+=100;

    glColor3f(0.6,0.7,0.8);
    glRasterPos2f(300,900);
    for(w=0;w<sizeof(wel);w++)
        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,wel[w]);

    glColor3f(0.5,0.1,0.7);
    glRasterPos2f(200,700);
    for(w=0;w<sizeof(start);w++)

glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18,start[w]);

    glColor3f(0.7,0.8,0.9);
    glRasterPos2f(200,600);
    for(w=0;w<sizeof(controls);w++)

```

```

        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, controls[w]);

        glColor3f(0.2, 0.5, 0.7);
        glRasterPos2f(200, 500);
        for(w=0; w<sizeof(levels); w++)

        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, levels[w]);

        glColor3f(1.0, 0.8, 0.3);
        glRasterPos2f(200, 400);
        for(w=0; w<sizeof(about); w++)

        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, about[w]);

        glFlush();
    }
}

void lev()
{
    glColor3f(0.5, 0.5, 0.5);
    glRectf(0, 0, 1280, 1024);

    glColor3f(0, 0, 0);
    glRasterPos2f(300, 800);
    for(w=0; w<sizeof(lev1); w++)
        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, lev1[w]);

    glRasterPos2f(300, 700);
    for(w=0; w<sizeof(lev2); w++)
        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, lev2[w]);

    glRasterPos2f(300, 600);
    for(w=0; w<sizeof(lev3); w++)
        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, lev3[w]);

    glRasterPos2f(300, 400);
    for(w=0; w<sizeof(lev5); w++)
        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, lev5[w]);

    glRasterPos2f(300, 500);
    for(w=0; w<sizeof(lev4); w++)
        glutBitmapCharacter(GLUT_BITMAP_HELVETICA_18, lev4[w]);

    levFlag=0;
    glFlush();
}

```

```

void keyboard(unsigned char key,int mx,int my)
{
    if(key=='1')
        startFlag=1;
    if(key=='2')
        controlsFlag=1;
    if(key=='4')
        aboutFlag=1;
    if(key=='3')
        levFlag=1;
    if(key=='a')
        level1=1;
    if(key=='b')
        level2=1;
    if(key=='c')
        level3=1;

    glutPostRedisplay();
}

void mouse(int b,int s,int mx,int my)
{
    x=mx;
    y=wwh-my;

    if(x>=o.x1 && x<o.x1+40 && y>=o.y1 && y<=o.y1+40 )
    {
        hitFlag=1; messageTrue=1;
    }

    glutPostRedisplay();
}

void reshape(int ww,int wh)
{
    wwh=wh;
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(0,ww,0,wh);

    mypos();
}

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

```

```
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(1280,1024);
glutInitWindowPosition(0.0,0.0);
glutCreateWindow("CATCH ME IF U CAN!!!");

initfun();
glutDisplayFunc(display);

glutKeyboardFunc(keyboard);
glutMouseFunc(mouse);
glutReshapeFunc(reshape);

glutMainLoop();
return 0;
}
```