```
#include <GL/glut.h>

#include <stdlib.h>

#include <stdio.h>

#include<math.h>


void init() {

    glClearColor(1.0, 1.0, 1.0, 1);

    glOrtho(-800, 800, -800, 800, -800, 800);

}


void circle(GLfloat rx, GLfloat ry, GLfloat x, GLfloat y, GLfloat startangle = 0, GLfloat endangle = 360) {

    GLfloat PI = 3.1416;

    glBegin(GL_TRIANGLE_FAN);

    glVertex2f(x, y);

    for (GLfloat angle = startangle; angle <= endangle; angle += 2.0 * PI / 1000.0) {

        GLfloat rangle = PI * angle / 180.0;

        glVertex2f(x + (cos(rangle) * rx), y + (sin(rangle) * ry));

    }

    glEnd();

}



void display() {

    glClear(GL_COLOR_BUFFER_BIT);



    glColor3f(1, 1.0 / 255 * 182.0, 1.0 / 255 * 12.0);


    glBegin(GL_TRIANGLES);
```

```
glVertex2f(-50, 50.0f);
glVertex2f(-300.0f, -50.0f);
glVertex2f(-80.0, -50.0f);
glEnd();

glBegin(GL_TRIANGLES);
glVertex2f(50, 50.0f);
glVertex2f(300.0f, -50.0f);
glVertex2f(80.0, -50.0f);
glEnd();


glColor3f(0, 0, 0);


glBegin(GL_TRIANGLES);
glVertex2f(-150, 350.0f);
glVertex2f(-400.0f, 250.0f);
glVertex2f(-180.0, 250.0f);
glEnd();

glBegin(GL_TRIANGLES);
glVertex2f(150, 350.0f);
glVertex2f(400.0f, 250.0f);
glVertex2f(180.0, 250.0f);
glEnd();


circle(200, 150, 0, 400, 0, 360);
circle(300, 250, 0, 150, 0, 180);
circle(300, 150, 0, 150, 180, 360);


circle(20, 20, -60, 400, 0, 360);
circle(20, 20, 60, 400, 0, 360);

glColor3f(1, 1, 1);
circle(80, 80, -60, 400, 0, 360);
circle(20, 20, -60, 400, 0, 360);
circle(80, 80, 60, 400, 0, 360);
circle(250, 200, 0, 150, 0, 180);
circle(250, 120, 0, 150, 180, 360);

glColor3f(0, 0, 0);
circle(20, 20, -60, 400, 0, 360);
circle(20, 20, 60, 400, 0, 360);
circle(5, 5, -60, 400, 0, 360);
circle(5, 5, 60, 400, 0, 360);


glColor3f(1, 1, 1);
circle(5, 5, -60, 400, 0, 360);
circle(5, 5, 60, 400, 0, 360);



glColor3f(1, 1.0 / 255 * 182.0, 1.0 / 255 * 12.0);
circle(60, 60, 0, 320, 60, 120);
```

```
    glFlush();
}


int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(500, 500);
    glutInitWindowPosition(90, 0);
    glutCreateWindow("Penguine");
    init();
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```