```cpp
#include <GL/gl.h>
#include <GL/glut.h>
#include<bits/stdc++.h>
void blackthoth(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)
{
glBegin(GL_TRIANGLE_FAN);
glColor3f(0.0f, 0.0f, 0.0f);
glVertex2f(cx, cy);
for (int i = 17;i <= 30;i++)
{
float angle = 2.0f * 3.1416f * i / 100;
float x = rx * cosf(angle);
float y = ry * sinf(angle);
glVertex2f((x + cx), (y + cy));
}
glEnd();
}
void thoth(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)
{
glBegin(GL_TRIANGLE_FAN);
glColor3f(1,0.8,0.1);
glVertex2f(cx, cy);
for (int i = 17;i <= 30;i++)
{
float angle = 2.0f * 3.1416f * i / 100;
float x = rx * cosf(angle);
float y = ry * sinf(angle);
glVertex2f((x + cx), (y + cy));
}
```

```
glEnd();

}


void nicherthoth(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)

{

glBegin(GL_TRIANGLE_FAN);

glColor3f(1,0.0,0.0);

glVertex2f(cx, cy);

for (int i = 19;i <= 30;i++)

{

float angle = 2.0f * 3.1416f * i / 100;

float x = rx * cosf(angle);

float y = ry * sinf(angle);

glVertex2f((x + cx), (y + cy));

}

glEnd();

}


void body(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)

{

glBegin(GL_TRIANGLE_FAN);

glColor3f(1,1,0.2);

glVertex2f(cx, cy);

for (int i = 0;i <= 100;i++)

{

float angle = 2.0f * 3.1416f * i / 100;

float x = rx * cosf(angle);

float y = ry * sinf(angle);

glVertex2f((x + cx), (y + cy));
```

```
}

glEnd();

}

void tikli(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)

{

glBegin(GL_TRIANGLE_FAN);

glColor3f(1,0,0.0);

glVertex2f(cx, cy);

for (int i = 0;i <= 100;i++)

{

float angle = 2.0f * 3.1416f * i / 100;

float x = rx * cosf(angle);

float y = ry * sinf(angle);

glVertex2f((x + cx), (y + cy));

}

glEnd();

}

void black(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)

{

glBegin(GL_TRIANGLE_FAN);

glColor3f(0.0,0.0,0.0);

glVertex2f(cx, cy);

for (int i = 0;i <= 100;i++)

{

float angle = 2.0f * 3.1416f * i / 100;

float x = rx * cosf(angle);

float y = ry * sinf(angle);

glVertex2f((x + cx), (y + cy));

}
```

```
glEnd();

}


void eye(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy)

{

glBegin(GL_TRIANGLE_FAN);

glColor3f(1.0,1.0,1.0);

glVertex2f(cx, cy);

for (int i = 0;i <= 100;i++)

{

float angle = 2.0f * 3.1416f * i / 100;

float x = rx * cosf(angle);

float y = ry * sinf(angle);

glVertex2f((x + cx), (y + cy));

}

glEnd();

}



void display(void)/////////////////////////////////////////////display ////////////

{

/* clear all pixels */

glClear(GL_COLOR_BUFFER_BIT);




/////////////////////////////////leg1
```

```
glBegin(GL_POLYGON);
glColor3f (1,0.6,0.2);
glVertex2f(-20.0f, -32.0f );
glVertex2f(-16.0f, -32.0f );
glVertex2f(-16.0f, -45.0f );
glVertex2f(-20.0f, -45.0f );
 glEnd();


 glBegin(GL_POLYGON);
glColor3f (1,0.6,0.2);
glVertex2f(-30.0f, -45.0f );
glVertex2f(-8.0f, -38.0f );
glVertex2f(-8.0f, -43.0f );
glVertex2f(-30.0f, -50.0f );
 glEnd();


 glBegin(GL_POLYGON);
glColor3f (1,0.6,0.2);
glVertex2f(-20.0f, -45.0f );
glVertex2f(-16.0f, -45.0f );
glVertex2f(-16.0f, -55.0f );
glVertex2f(-20.0f, -55.0f );
 glEnd();




 ///////////////////////////////\eg2


glBegin(GL_POLYGON);
```

```
glColor3f (1,0.6,0.2);
glVertex2f(20.0f, -32.0f );
glVertex2f(16.0f, -32.0f );
glVertex2f(16.0f, -45.0f );
glVertex2f(20.0f, -45.0f );
 glEnd();


 glBegin(GL_POLYGON);
glColor3f (1,0.6,0.2);
glVertex2f(30.0f, -45.0f );
glVertex2f(8.0f, -38.0f );
glVertex2f(8.0f, -43.0f );
glVertex2f(30.0f, -50.0f );
 glEnd();


 glBegin(GL_POLYGON);
glColor3f (1,0.6,0.2);
glVertex2f(20.0f, -45.0f );
glVertex2f(16.0f, -45.0f );
glVertex2f(16.0f, -55.0f );
glVertex2f(20.0f, -55.0f );
 glEnd();




black(36, 51, 0, 10);
body(35, 50, 0, 10);


black(26, 31, 0, 37);//head
```

```
body(25, 30, 0, 37);//head

black(8, 11, 9, 38);

black(8, 11, -9, 38);

eye(7, 10, 9, 38);

eye(7, 10, -9, 38);

black(2, 3, 8.5, 38);////eye ball

black(2, 3, -8.5, 38);////eye ball




nicherthoth(13, 17, 2, 10);

blackthoth(14, 18, 0, 10);

thoth(13, 17, 0, 10);

nicherthoth(13, 17, 0, 55);

tikli(5, 8, 0.3, 75);////tikli

/////////////////////////////

//rx,ry,cx,cy

glFlush();

}

void init(void)

{

glClearColor(1.0, 1.0, 1.0, 0.0);

glMatrixMode(GL_PROJECTION);

glLoadIdentity();

glOrtho(-100, 100, -100, 100, -15, 15);

//-x,x,-y,y

}
```

```c
int main(int argc, char** argv)
{
glutInit(&argc, argv);
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize(1000, 600);
glutInitWindowPosition(100, 100);
glutCreateWindow("Circle 192-15-13126");
init();
glutDisplayFunc(display);
glutMainLoop();
return 0; /* ISO C requires main to return int. */
}
```