```
#include <GL/gl.h>
#include <GL/glut.h>
#include<bits/stdc++.h>

static float tx = 0.0;
static float ty = 0.0;
float angle = 0.0;

void circle(GLfloat rx, GLfloat ry, GLfloat cx, GLfloat cy, int x, int y)
{

   glBegin(GL_TRIANGLE_FAN);
   //glColor3f(1.0f, 0.0f, 0.0f);
   glVertex2f(cx, cy);

   for (int i = x;i <= y;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 triangle(double x, double y)
```

```
{
   glBegin(GL_POLYGON);
   glVertex2d(x, y);
   glVertex2d(x + 20, y + 19.7);
   glVertex2d(x + 20, y + 9.6);
   glEnd();
}

void legs(double x, double y)
{
   glColor4f(1.0f, 1.0f, 0.0f, 0.0f);
   glBegin(GL_POLYGON);
   glVertex2d(x, y);
   glVertex2d(x + 5, y);
   glVertex2d(x + 5, y - 20);
   glVertex2d(x + 0, y - 20);
   glEnd();
   glBegin(GL_POLYGON);
   glVertex2d(x - 8, y - 20);
   glVertex2d(x + 8, y - 20);
   glVertex2d(x + 8, y - 23);
   glVertex2d(x - 8, y - 23);
   glEnd();
}
void tail(double x, double y)
{
   glColor4f(1.0f, 0.0f, 0.0f, 0.0f);
   glBegin(GL_POLYGON);
   glVertex2d(x, y);
```

```
    glVertex2d(x, y + 10);
    glVertex2d(x + 20, y + 10);
    glVertex2d(x + 20, y);
    glEnd();

}

void display(void)
{
    /* clear all pixels */
    glClear(GL_COLOR_BUFFER_BIT);


    glColor3f(0.0f, 0.8f, 0.1f);
    glRectf(-100, -20, 100, -100);


    glBegin(GL_POLYGON); // tail
    glColor3f(0.9, 0.5, 0.2);
    glVertex2f(0, 0);
    glVertex2f(60, -30);
    glVertex2f(60, -35);
    glVertex2f(0, -35);
    glEnd();

    /*=============Left hand=====================*/

    glColor3f(0.1f, 0.1f, 0.5f);
    circle(20, 20, -20, 10, 0, 100);
```

```
   /*============END================*/

   /*First part of capsul*/
   glColor4f(1.0f, 0.0f, 0.0f, 0.0f);
   glRectf(-20, 60, 20, 20);
   glColor4f(1.0f, 0.0f, 0.0f, 0.0f);
   circle(20, 20, 0, 60, 0, 50);
   /*----------END---------------*/


/*Second part.............*/
   glColor4f(1.0f, 1.0f, 0.0f, 0.0f);
   glRectf(-20, 30, 20, -20);
   glColor4f(1.0f, 1.0f, 0.0f, 0.0f);
   circle(20, 20, 0, -20, 50, 100);
   /*----------END---------------*/

//===========Legs==========



//=============END==========

/* ===========Eyes==============*/

   glColor4f(1.0f, 1.0f, 1.0f, 0.0f);
   circle(10, 10, -20, 55, 0, 100);
   circle(10, 10, 2, 55, 0, 100);
```

```
    glColor3f(0.0f, 0.0f, 0.0f);
    circle(4, 4, -2, 53, 0, 100);
    circle(4, 4, -17, 53, 0, 100);

    /*============END================*/

/**********BEAK************************/

    glColor4f(1.0f, 1.0f, 0.0f, 0.0f);
    triangle(-30, 30);
    circle(5, 5, -10, 44, 75, 130);


    /*=============Right hand=====================*/

    glColor3f(0.1f, 0.1f, 0.5f);
    circle(20, 20, 10, 8, 10, 60);

    glBegin(GL_POLYGON);
    glColor3f(0.1f, 0.1f, 0.5f);
    glVertex2d(-7, -2.8);
    glVertex2d(26, 20);
    glVertex2d(40, -30);
    glVertex2d(19.5, -30);
    glEnd();
    circle(10, 10, 30, -30, 50, 100);
```

```
    /*============END================*/


/*=========LEGS===========*/
    legs(5, -33);
    legs(-10, -30);

    /*=========END============*/

    // glBegin(GL_LINES); // nasim line draw
    // glColor3f(1, 1, 1);
    // glVertex3i(-100, 0, 0);
    // glVertex3i(100, 0, 0);
    // glEnd();

    // glBegin(GL_LINES); // nasim line draw
    // glColor3f(1, 1, 1);
    // glVertex3i(0, 100, 0);
    // glVertex3i(0, -100, 0);
    // glEnd();


    //tail(0, -6);




    glFlush();
}
```

```
void init(void)
{
   /* select clearing (background) color */
   glClearColor(0.0, 0.0, 1.0, 0.0);
   /* initialize viewing values */
   glMatrixMode(GL_PROJECTION);
   glLoadIdentity();
   glOrtho(-100.0, 100.0, -100.0, 100.0, -10.0, 10.0);
   //-x,x,-y,y
}
/*
* Declare initial window size, position, and display mode
* (single buffer and RGBA). Open window with "hello"
* in its title bar. Call initialization routines.
* Register callback function to display graphics.
* Enter main loop and process events.
*/
int main(int argc, char** argv)
{
   glutInit(&argc, argv);
   glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
   glutInitWindowSize(800, 800);
   glutInitWindowPosition(100, 100);
   glutCreateWindow("Tuaha is a good boy");
   init();
   glutDisplayFunc(display);
   glutMainLoop();
   return 0; /* ISO C requires main to return int. */
}
```