PROGRAM -2

2. Create and rotate a triangle about the origin and a fixed point.

```
#include<GL/glut.h>
#define NULL 0
static GLfloat angle=90;
int sb,db;
void dd()
{
glClear(GL_COLOR_BUFFER_BIT);
glColor3f(1,0,0);

glBegin(GL_LINES);
glVertex2f(-2,0);
glVertex2f(2,0);
glVertex2f(0,2);
glVertex2f(0,-2);
glEnd();

glColor3f(1,0,1);
glBegin(GL_TRIANGLES);
glVertex2f(0.3,0.2);
glVertex2f(0,0);
glVertex2f(0.2,0.3);
glEnd();

glColor3f(0,1,0);
glRotatef(90,0,0,1);
glBegin(GL_TRIANGLES);
glVertex2f(0.3,0.2);
glVertex2f(0,0);
glVertex2f(0.2,0.3);
glEnd();
glutSwapBuffers();
}

void ds()
{
glClear(GL_COLOR_BUFFER_BIT);
glColor3f(1,0,0);
```

```
glBegin(GL_LINES);
glVertex2f(-2,0);
glVertex2f(2,0);
glVertex2f(0,2);
glVertex2f(0,-2);
glEnd();

glColor3f(1,0,1);
glBegin(GL_TRIANGLES);
glVertex2f(0.3,0.2);
glVertex2f(0.6,0.2);
glVertex2f(0.6,0.6);
glEnd();

glPushMatrix();
glTranslatef(0.3,0.2,0.0);
glRotatef(90,0,0,1);
glTranslatef(-0.3,-0.2,0.0);

glColor3f(0,1,0);
glBegin(GL_TRIANGLES);
glVertex2f(0.3,0.2);
glVertex2f(0.6,0.2);
glVertex2f(0.6,0.6);
glEnd();

glPopMatrix();
}

void sd()
{
glutSetWindow(sb);
glLoadIdentity();
glutSetWindow(db);
glLoadIdentity();
glutPostRedisplay();
}

void minit()
{
```

```
glClearColor(1,1,1,1);
glColor3f(0,1,1);
glShadeModel(GL_FLAT);
}

void myres(int w, int h)
{
glViewport(0,0,w,h);
glMatrixMode(GL_PROJECTION);
glLoadIdentity();
if(w<=h)
glOrtho(-1,1,-1*(GLfloat)h/(GLfloat)w,1*(GLfloat)h/(GLfloat)w,-1,1);
else
glOrtho(-1*(GLfloat)w/(GLfloat)h,1*(GLfloat)w/(GLfloat)h,
-1,1,-1,1);
glMatrixMode(GL_MODELVIEW);
glLoadIdentity();
}

int main(int argc, char **argv)
{
glutInit(&argc,argv);
glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);
glutInitWindowSize(300,350);
glutInitWindowPosition(700,300);
sb=glutCreateWindow("FPR");
minit();
glutDisplayFunc(ds);
glutReshapeFunc(myres);
glutIdleFunc(sd);
glutInitDisplayMode(GLUT_DOUBLE|GLUT_RGB);
glutInitWindowSize(300,350);
glutInitWindowPosition(400,0);
db=glutCreateWindow("OR");
minit();
glutDisplayFunc(dd);
glutReshapeFunc(myres);
glutIdleFunc(sd);
glutMainLoop();
return 0;
```

}

Output command

To create file - gedit filename.c

To compile file - gcc filename.c -lGL -lGLU -lglut

To execute - ./a.out