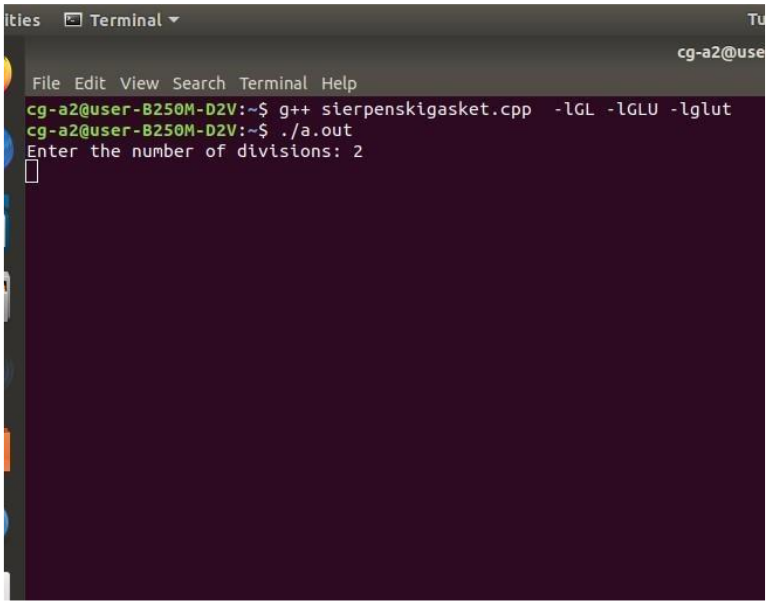
7. Program to recursively subdivide a tetrahedron to from 3D Sierpinski gasket.The number of recursive steps is to be specified by the user.

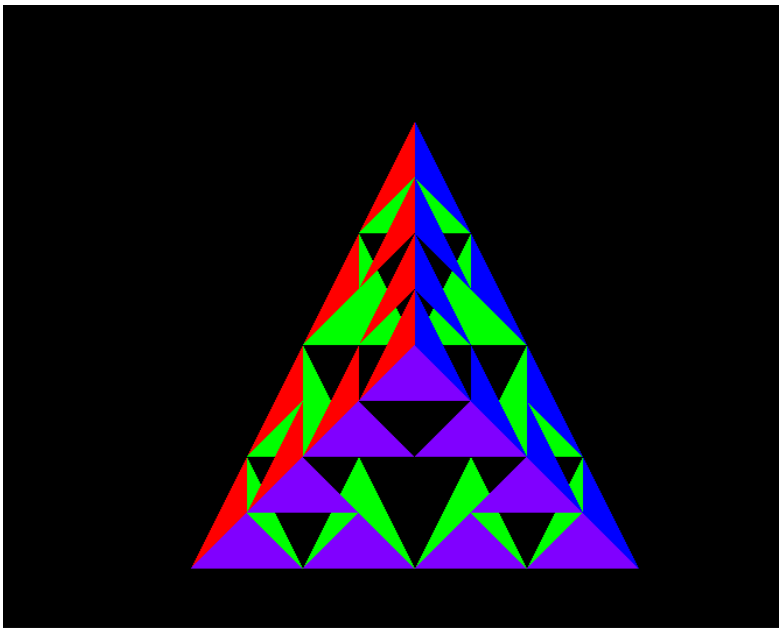
.

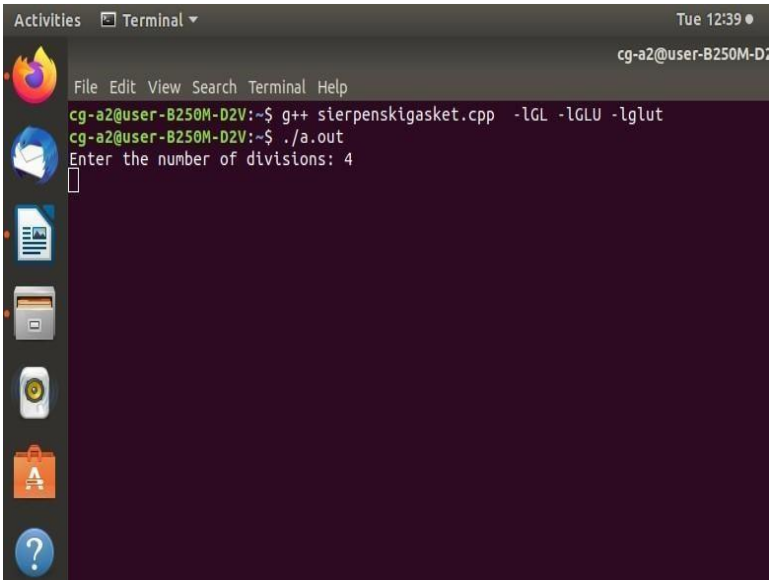
```
#include <stdlib.h>
#include <stdio.h>
#include <GL/glut.h>

typedef float point[3];
point v[]={{0.0, 0.0, 0.0}, {0.0, 1.0, -1.0},{-1.0, -1.0, -1.0},{1.0,
-1.0, -1.0}};

int n;
void triangle(point a, point b, point c)
{
      glBegin(GL_POLYGON);
      glVertex3fv(a);
      glVertex3fv(b);
      glVertex3fv(c);
      glEnd();
}
void divide_triangle(point a, point b, point c, int m)
{
      point v1, v2, v3;
      if (m > 0)
      {
            for (int i = 0; i < 3; i++)
            {
                  v1[i] = (a[i] + b[i]) / 2;
                  v2[i] = (a[i] + c[i]) / 2;
                  v3[i] = (b[i] + c[i]) / 2;
            }
      divide_triangle(a, v1, v2, m - 1);
      divide_triangle(c, v2, v3, m - 1);
      divide_triangle(b, v3, v1, m - 1);
      }
else
{
      triangle(a, b, c);
/* draw triangle at end of recursion */
}
}
void tetrahedron(int m)
{
      glColor3f(1.0, 0.0, 0.0);
      divide_triangle(v[0], v[1], v[2], m);
      glColor3f(0.0, 1.0, 0.0);
      divide_triangle(v[3], v[2], v[1], m);
      glColor3f(0.0, 0.0, 1.0);
      divide_triangle(v[0], v[3], v[1], m);
      glColor3f(0.0, 0.0, 0.0);
```

```
      divide_triangle(v[0], v[2], v[3], m);
}
void display()
{
      glClearColor(1.0, 1.0, 1.0, 1.0);
      glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
      glLoadIdentity();
      tetrahedron(n);
      glFlush();
}
void myReshape(int w, int h)
{
      glViewport(0, 0, w, h);
      glMatrixMode(GL_PROJECTION);
      glLoadIdentity();
      if (w <= h)
      glOrtho(-2.0, 2.0, -2.0 * (GLfloat) h / (GLfloat) w, 2.0 *
      (GLfloat) h / (GLfloat) w, -10.0, 10.0);
else
      glOrtho(-2.0 * (GLfloat) w / (GLfloat) h, 2.0 * (GLfloat) w /
      (GLfloat) h, -2.0, 2.0, -10.0, 10.0);
      glMatrixMode(GL_MODELVIEW);
}
int main(int argc, char **argv)
{
      printf("Enter number of recursive steps \n");
      scanf("%d",&n);
      glutInit(&argc, argv);
      glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB | GLUT_DEPTH);
      glutInitWindowSize(500,500);
      glutInitWindowPosition(0, 0);
      glutCreateWindow("3D Sierpinski Gasket");
      glutFullScreen();
      glutReshapeFunc(myReshape);
      glutDisplayFunc(display);
      glEnable(GL_DEPTH_TEST);
      glutMainLoop();
}
```

OUTPUT:


ities Terminal Tu
cg-a2@use
File Edit View Search Terminal Help
cg-a2@user-B250M-D2V:~$ g++ sierpenskigasket.cpp -lGL -lGLU -lglut
cg-a2@user-B250M-D2V:~$ ./a.out
Enter the number of divisions: 2




Activities Terminal Tue 12:39
cg-a2@user-B250M-D
File Edit View Search Terminal Help
cg-a2@user-B250M-D2V:~$ g++ sierpenskigasket.cpp -lGL -lGLU -lglut
cg-a2@user-B250M-D2V:~$ ./a.out
Enter the number of divisions: 4