Practical -4

```
// Boundry fill
#include <iostream>
#include <math.h>
//#include <time.h>
#include <GL/glut.h>

using namespace std;

void init()
      {
            glClearColor(0.0,0.0,0.0,1.0);
            glMatrixMode(GL_PROJECTION);
            gluOrtho2D(0,640,0,480);
      }

void flood_it(int x, int y, float* bc)
      {
            float color[3];

            //to read the current pixel information
            glReadPixels(x,y,1.0,1.0,GL_RGB,GL_FLOAT,color);

                  //checking current pixel color is not equal to boundary
color
if(color[0]!=bc[0] || color[1]!=bc[1] ||
color[2]!=bc[2])

            {

            //to fill the pixel by new color
              glColor3f(bc[0],bc[1],bc[2]);

              glBegin(GL_POINTS);
            glVertex2i(x,y);
              glEnd();
              glFlush();

            //recursive call to the function

              flood_it(x+1,y,bc);
              flood_it(x-2,y,bc);
            flood_it(x,y+1,bc);
              flood_it(x,y-2,bc);


            }
      }
//mouse callback function
void mouse(int btn, int state, int x, int y)
      {
            y = 480-y;
            if(btn==GLUT_LEFT_BUTTON)

            {
                  if(state==GLUT_DOWN)
      {
```

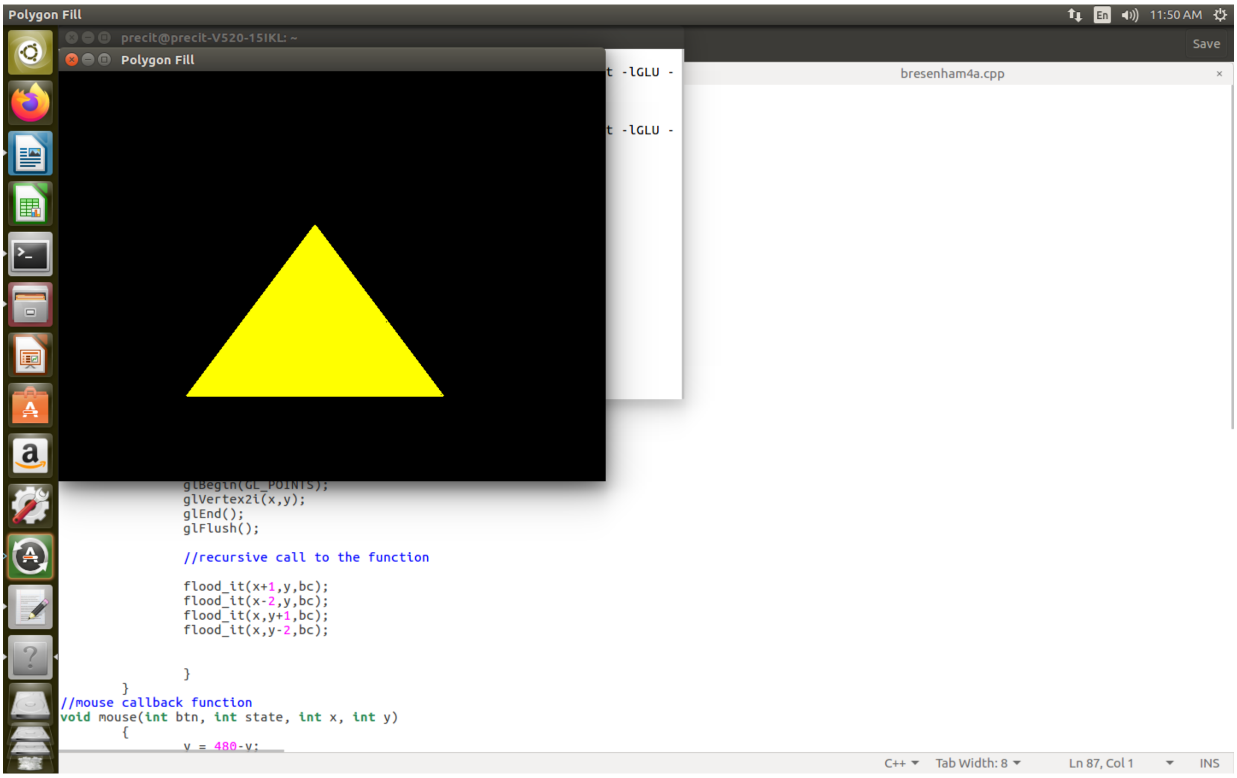
```
                              float bCol[] = {1,1,0};

                           flood_it(x,y,bCol);
                              }
             }
        }
//to draw the object
void world()
        {
         glLineWidth(2);
         glPointSize(2);
         glClear(GL_COLOR_BUFFER_BIT);
         glColor3f(1,1,0);
         glBegin(GL_LINE_LOOP);
           glVertex2i(150,100);
           glVertex2i(300,300);
           glVertex2i(450,100);
         glEnd();
         glFlush();
        }

int main(int argc, char** argv)
        {
        glutInit(&argc, argv);                  //initialization of the
GLUT
        glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);   //to intialize the
display mode
        glutInitWindowSize(640,480);       //to set the size of the window
        glutInitWindowPosition(50,50);     //to set the position of the
window
        glutCreateWindow("Rameshwari Shirsath Roll No:70");//to give user
defined name to the window
        init();
        glutDisplayFunc(world);        //to creat the object
        glutMouseFunc(mouse);  //glutMouseFunc sets the mouse callback for
the current window.


        glutMainLoop();
        return 0;
        }
```

OUTPUT:



```
t -lGLU -

t -lGLU -
```

```
                glBegin(GL_POINTS);
                glVertex2i(x,y);
                glEnd();
                glFlush();

                //recursive call to the function

                flood_it(x+1,y,bc);
                flood_it(x-2,y,bc);
                flood_it(x,y+1,bc);
                flood_it(x,y-2,bc);


                }
        }
//mouse callback function
void mouse(int btn, int state, int x, int y)
        {
                y = 480-y;
```