

**Computer Graphics Lab**

**Name:** Pratham Kandari

**Sap Id:** 500097663

**Batch:** 7

**Experiment -4**

**Q1) Write a program for the Bresenhams line algorithm.**

**Ans:**

#include <GL/glut.h>

#include <stdio.h>

GLint x0=-20,y3=-30,xEnd=30,yEnd=40;

void init()

{

glClearColor(1.0,1.0,1.0,0.0);

glColor3f(1.0f,0.0f,0.0f);

glPointSize(1.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-50.0,50.0,-50.0,50.0);

}

void setPixel(GLint xcoordinate, GLint ycoordinate)

{

glBegin(GL\_POINTS);

glVertex2i(xcoordinate,ycoordinate);

glEnd();

glFlush();

}

void lineBA(GLint x0,GLint y3,GLint xEnd,GLint yEnd)

{

GLint dx = xEnd-x0;

GLint dy = yEnd-y3;

GLint steps,k;

steps=dx;

GLint x,y,p0=(2\*dy)-dx;

setPixel(x0,y3);

x=x0;

y=y3;

for(k=0;k<steps;k++)

{

if(p0<0)

{

p0=p0+(2\*dy);

x+=1;

}

else

{

p0=p0+(2\*dy)-(2\*dx);

x+=1;

y+=1;

}

setPixel(x,y);

}

}

void Display(void)

{

//static int i=1;

glClear(GL\_COLOR\_BUFFER\_BIT);

lineBA(x0,y3,xEnd,yEnd);

}

int main(int argc,char\*\* argv)

{

glutInit(&argc,argv);

glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

glutInitWindowSize(600,600);

glutInitWindowPosition(0,0);

glutCreateWindow("Breshnam's Line Drawing Algorithm");

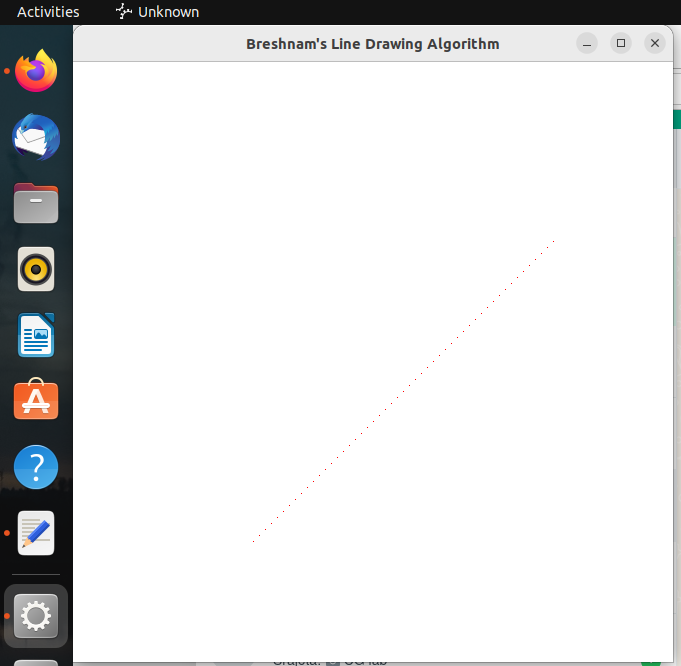
glutDisplayFunc(Display);

init();

glutMainLoop();

return 0;

}



**Q2) Write a program for the Circle algorithm.**

**Ans:**

#include <stdio.h>

#include <math.h>

#include <GL/glut.h>

int radius1 = 20;

int xc = 320, yc = 240;

void plot\_point(int x, int y)

{

glBegin(GL\_POINTS);

glVertex2i(xc+x, yc+y);

glVertex2i(xc+x, yc-y);

glVertex2i(xc+y, yc+x);

glVertex2i(xc+y, yc-x);

glVertex2i(xc-x, yc-y);

glVertex2i(xc-y, yc-x);

glVertex2i(xc-x, yc+y);

glVertex2i(xc-y, yc+x);

glEnd();

}

void bresenham\_circle(int r)

{

int x=0,y=r;

float pk=(5.0/4.0)-r;

plot\_point(x,y);

int k;

while(x < y)

{

x = x + 1;

if(pk < 0)

pk = pk + 2\*x+1;

else

{

y = y - 1;

pk = pk + 2\*(x - y) + 1;

}

plot\_point(x,y);

}

glFlush();

}

void concentric\_circles(void)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

bresenham\_circle(radius1);

}

void Init()

{

glClearColor(1.0,1.0,1.0,0);

glColor3f(0.0,0.0,0.0);

gluOrtho2D(0 , 640 , 0 , 480);

}

int main(int argc, char \*\*argv)

{

glutInit(&argc,argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowPosition(0,0);

glutInitWindowSize(640,480);

glutCreateWindow("bresenham\_circle");

Init();

glutDisplayFunc(concentric\_circles);

glutMainLoop();

return 0;

}

