**Name:** Nishan Paul

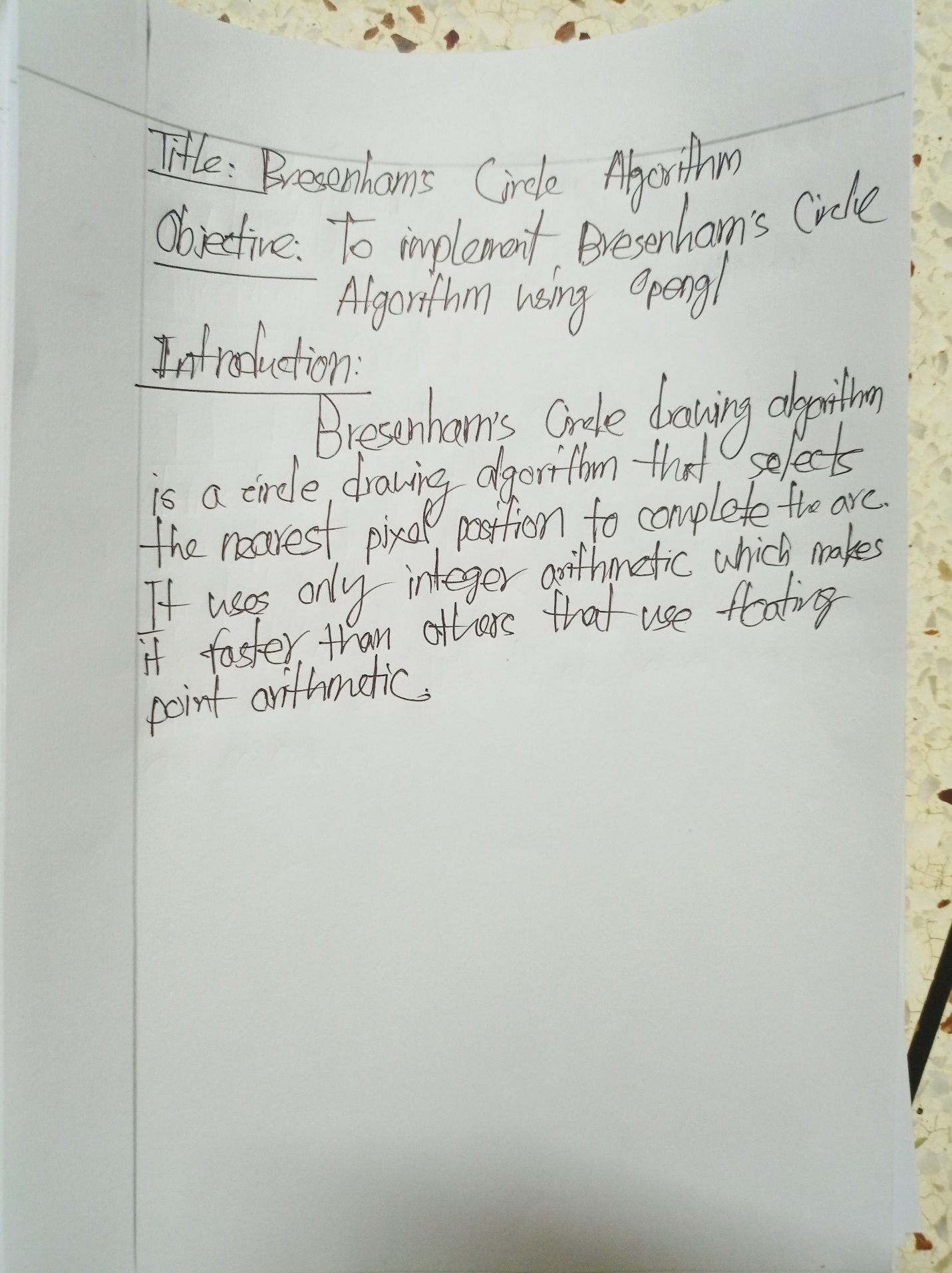
**ID:** 1604085

**Course Title:** Computer Graphics (Sessional)

**Course No:** CSE-458

**Level 4, Term 1**

**Assignment 02**



**Code:**

#include<windows.h>

#include<GL/glut.h>

#include<stdlib.h>

#include<stdio.h>

void \_display(int X1, int Y1, int r) {

glBegin(GL\_POINTS);

int x = 0, y = r;

float decision = 3 - 2\*r;

while(y > x) {

if(decision < 0) {

x++;

decision += 4\*x+6;

}

else {

y--;

x++;

decision += 4\*(x-y)+10;

}

glVertex2i(x+X1, y+Y1);

glVertex2i(x+X1, -y+Y1);

glVertex2i(-x+X1, y+Y1);

glVertex2i(-x+X1, -y+Y1);

glVertex2i(y+X1, x+Y1);

glVertex2i(-y+X1, x+Y1);

glVertex2i(y+X1, -x+Y1);

glVertex2i(-y+X1, -x+Y1);

}

glEnd();

glFlush();

}

void display(float x1, float y1, float x2, float y2) {

float x = x1;

float y = y1;

float dx = x2-x1;

float dy = y2-y1;

float m = dy/dx;

int limit = m<=1? abs(dx) : abs(dy);

if(m<=1)

dx = 1;

else

dy = 1;

for(int i=0; i<limit; i++){

x += dx;

y += dy;

glBegin(GL\_POINTS);

glVertex2f(x,y);

}

glEnd();

glFlush();

}

void init(void) {

glClear(GL\_COLOR\_BUFFER\_BIT);

glClearColor(0,0,0,0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(0, 640, 0, 480);

}

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

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(500,500);

glutInitWindowPosition(100,100);

glutCreateWindow("");

init();

int x1 = 100, y1 = 100, x2 = 400, y2 = 400;

int X1 = (x1+x2)/2, Y1 = (y1+y2)/2, r = (x2-x1)/2;

glutDisplayFunc([]() {

display(100, 100, 400, 100);

display(400, 100, 400, 400);

display(100, 400, 400, 400);

display(100, 100, 100, 400);

\_display(250, 250, 150);

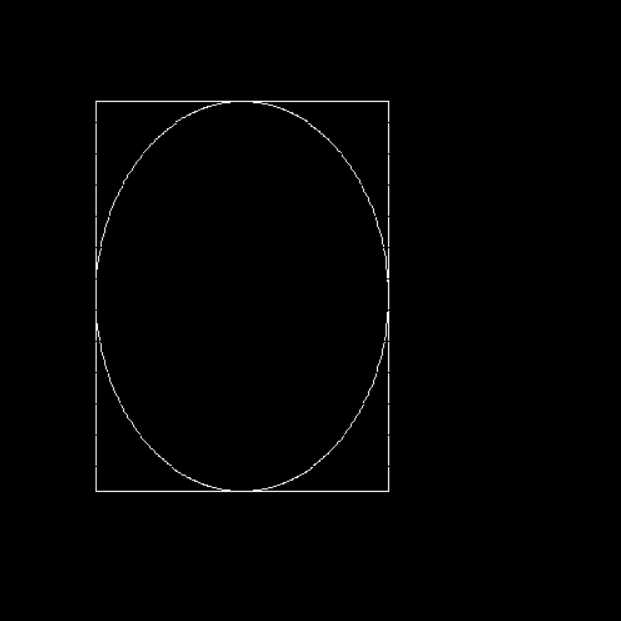
});

glutMainLoop();

return 0;

}

**Output:**

****

