Name : : SAMRUDDHI NEVAGI

Roll no : 23540

**CG ASSIGNMENT – 7.2**

**CODE:**

#include <iostream>

#include <math.h>

#include <time.h>

#include <GL/glut.h>

using namespace std;

int xx[4],yy[4];

int i = 0;

int ch;

void init(){

glClearColor(1.0,1.0,1.0,0.0);

glMatrixMode(GL\_PROJECTION);

gluOrtho2D(0,600,600,0);

glClear(GL\_COLOR\_BUFFER\_BIT);

}

void putpixel(double xt,double yt )

{

glColor3f(1,0,0);

glBegin(GL\_POINTS);

glVertex2d(xt,yt);

glEnd();

glFlush();

}

void Algorithm(){

glColor3f(0,0,0);

glBegin(GL\_LINES);

glVertex2i(xx[0],yy[0]);

glVertex2i(xx[1],yy[1]);

glVertex2i(xx[1],yy[1]);

glVertex2i(xx[2],yy[2]);

glVertex2i(xx[2],yy[2]);

glVertex2i(xx[3],yy[3]);

glEnd();

glFlush();

double t;

for (t = 0.0; t < 1.0; t += 0.0005)

{

double xt = pow(1-t, 3) \* xx[0] + 3 \* t \* pow(1-t, 2) \* xx[1] + 3 \* pow(t, 2) \* (1-t) \* xx[2] + pow(t, 3) \* xx[3];

double yt = pow(1-t, 3) \* yy[0] + 3 \* t \* pow(1-t, 2) \* yy[1] + 3 \* pow(t, 2) \* (1-t) \* yy[2] + pow(t, 3) \* yy[3];

glColor3f(0,1,1);

glPointSize(2);

glBegin(GL\_POINTS);

glVertex2i(xt,yt);

glEnd();

glFlush();

}

}

void mouse(int button, int state, int x, int y)

{

if (button == GLUT\_LEFT\_BUTTON && state == GLUT\_DOWN)

{

if (ch == 1)

{

xx[i] = x;

yy[i] = y;

glColor3f(1,0,0);

glPointSize(5);

glBegin(GL\_POINTS);

glVertex2i(xx[i],yy[i]);

glEnd();

glFlush();

i++;

}

}

}

void keyboard(unsigned char key, int x, int y)

{

switch (key)

{

case 'c':

{

ch = 1;

cout << "Choose points for polygon" << endl;

glutMouseFunc(mouse);

break;

}

case 'd':

{

ch = 2;

cout << "Display the Bezier curve" << endl;

if (i >= 3){

cout << "Yes" << endl;

Algorithm();

}

break;

}

case 'x':

{

for(int j=0; x <= i; j++){

xx[j] = 0;

yy[j] = 0;

}

cout << "Queue is cleared" << endl;

break;

}

case 'X':

{

glClearColor(1.0, 1.0, 1.0, 1.0);

glClear(GL\_COLOR\_BUFFER\_BIT);

cout << "Screen is cleared" << endl;

break;

}

}

glutPostRedisplay();

}

void P\_C() {

glColor3f(0,0,0);

glBegin(GL\_POINTS);

glVertex2i(0,0);

glEnd();

glFlush();

glutKeyboardFunc(keyboard);

}

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

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

glutInitWindowSize(600,600);

glutInitWindowPosition(200,200);

glutCreateWindow("Bezier 4 point");

init();

glutDisplayFunc(P\_C);

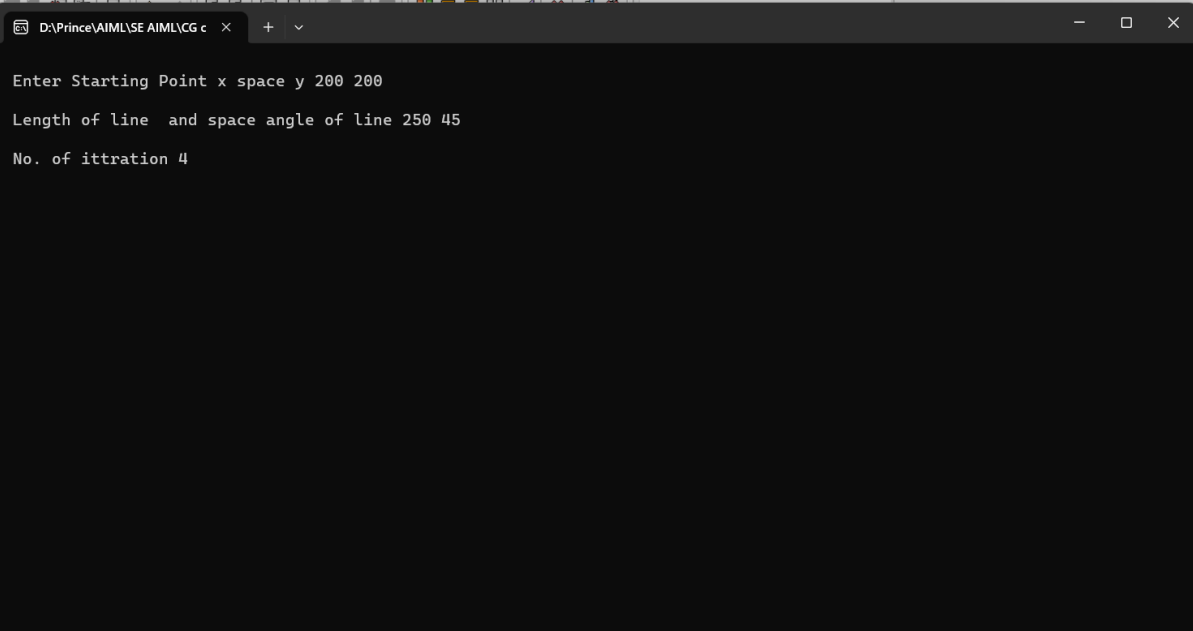
glutMainLoop();

return 0;

}

// c then d

**OUTPUT:**

****

****