**Rohan Nyati**

**500075940**

**R177219148**

**B-5 AI&ML SEM-5**

**Experiment-2**

**DDA Line Algorithm**

#include<windows.h>

#include<GL/glu.h>

#include<GL/glut.h>

#include<stdlib.h>

#include<stdio.h>

float x1,x2,y1,y2;

void display(void)

{

float dy,dx,step,x,y,k,m;

dx=x2-x1;

dy=y2-y1;

m=dy/dx;

if(abs(dx)> abs(dy))

{

step = abs(dx);

}

else

step = abs(dy);

x=x1;

y=y1;

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

for (k=1 ;k<=step;k++)

{

if(m<1){

x= 1 + x;

y= m + y;

}

if(m==1){

x= 1 + x;

y= 1 + y;

}

if(m>1){

x= (1/m) + x;

y= 1 + y;

}

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

}

glFlush();

}

void init(void)

{

glClearColor(0.7,0.7,0.7,0.7);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-100,100,-100,100);

}

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

printf("Enter the value of x1 : ");

scanf("%f",&x1);

printf("Enter the value of y1 : ");

scanf("%f",&y1);

printf("Enter the value of x2 : ");

scanf("%f",&x2);

printf("Enter the value of y2 : ");

scanf("%f",&y2);

glutInit(&argc, argv);

glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize (500, 500);

glutInitWindowPosition (100,100);

glutCreateWindow ("DDA Line Algo");

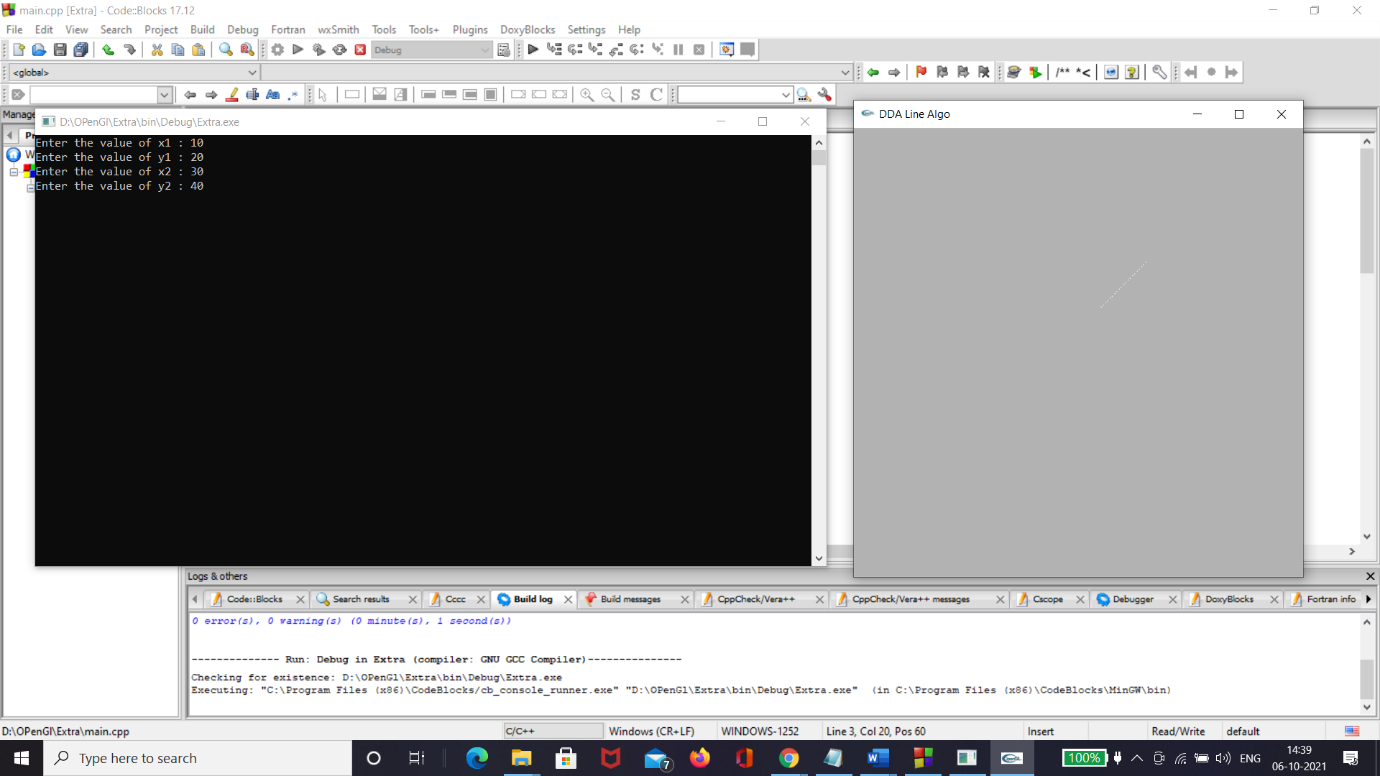
init();

glutDisplayFunc(display);

glutMainLoop();

return 0;

}



**Bresenham Line Algorithm**

#include<windows.h>

#include<GL/glu.h>

#include<GL/glut.h>

#include<stdlib.h>

#include<stdio.h>

float x1,x2,y1,y2;

void display(void)

{

float dy,dx,step,x,y,pk;

dx=x2-x1;

dy=y2-y1;

pk = 2\*(dy) - dx ;

step = dx-1 ;

x=x1;

y=y1;

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

for (int k=1 ;k<=step;k++)

{

if(pk<0){

pk=pk + 2\*(dy) ;

x = x + 1;

y = y;

}

if(pk>=0){

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

x= x + 1;

y= y + 1;

}

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

}

glFlush();

}

void init(void)

{

glColor3f(1.0,0.0,0.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-100,100,-100,100);

}

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

printf("Enter the value of x1 : ");

scanf("%f",&x1);

printf("Enter the value of y1 : ");

scanf("%f",&y1);

printf("Enter the value of x2 : ");

scanf("%f",&x2);

printf("Enter the value of y2 : ");

scanf("%f",&y2);

glutInit(&argc, argv);

glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize (500, 500);

glutInitWindowPosition (100,100);

glutCreateWindow ("Bresenham Line Algo");

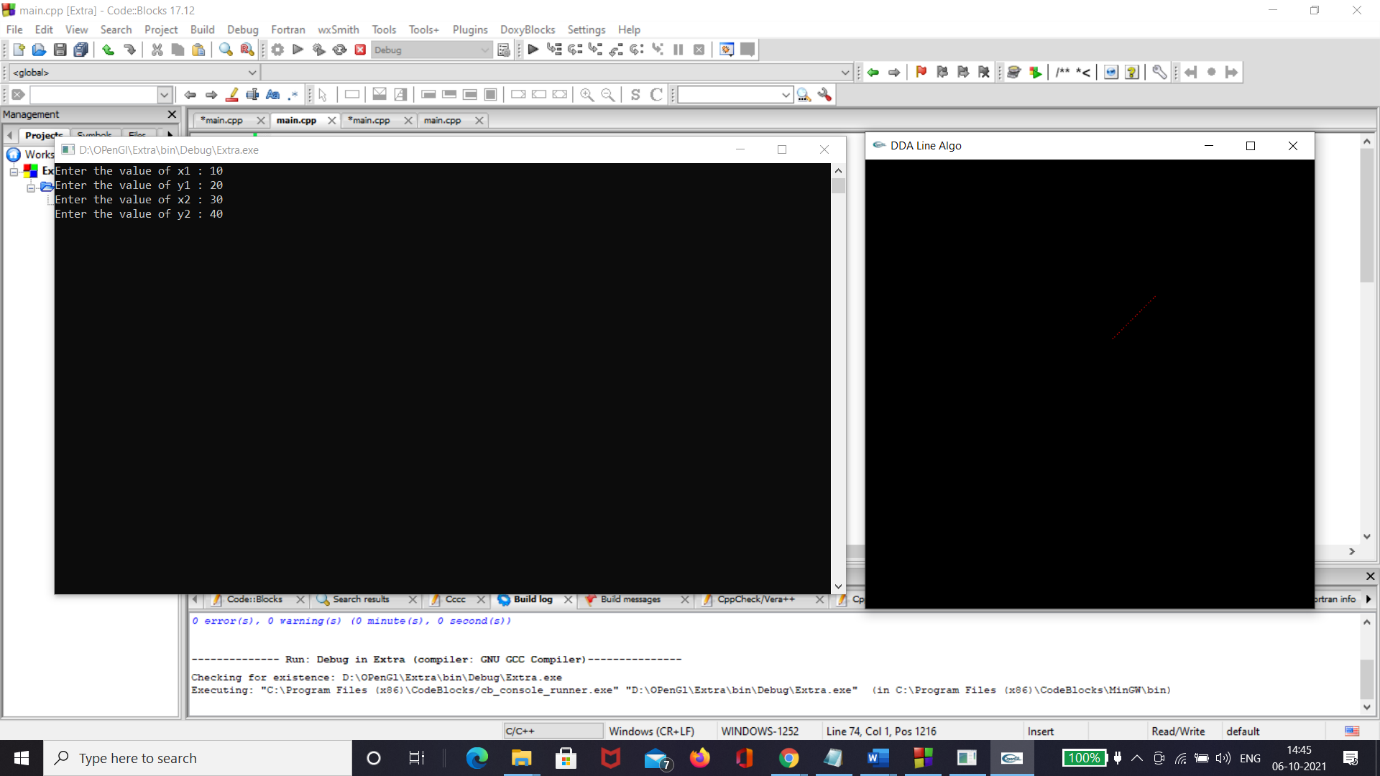
init();

glutDisplayFunc(display);

glutMainLoop();

return 0;

}



**Mid Point Line Algorithm**

#include<windows.h>

#include<GL/glu.h>

#include<GL/glut.h>

#include<stdlib.h>

#include<stdio.h>

float x1,x2,y1,y2;

void display(void)

{

float dy,dx,x,y,D\_initial,dD,D\_new;

dx=x2-x1;

dy=y2-y1;

D\_initial = 2\*(dy) - dx ;

dD = 2\*(dy) - 2\*(dx);

x=x1;

y=y1;

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

for (int x=x1 ; x<=x2 ; x++)

{

if(D\_initial<0){

y = y;

D\_new = D\_initial + 2\*(dy) ;

D\_initial = D\_new;

}

if(D\_initial>=0){

y= y + 1;

D\_new = D\_initial + dD ;

D\_initial = D\_new;

}

glBegin(GL\_POINTS);

glVertex2i(x,y);

glEnd();

}

glFlush();

}

void init(void)

{

glColor3f(1.0,0.0,0.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(-100,100,-100,100);

}

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

printf("Enter the value of x1 : ");

scanf("%f",&x1);

printf("Enter the value of y1 : ");

scanf("%f",&y1);

printf("Enter the value of x2 : ");

scanf("%f",&x2);

printf("Enter the value of y2 : ");

scanf("%f",&y2);

glutInit(&argc, argv);

glutInitDisplayMode (GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize (500, 500);

glutInitWindowPosition (100,100);

glutCreateWindow ("Mid Point Line Algo");

init();

glutDisplayFunc(display);

glutMainLoop();

return 0;

}

