**EXPERIMENT NO.1(a)**

**AIM-**To draw a line using Simple DDA Algorithmfor positive line slope.

**Source Code:**

#include<iostream.h>

#include<conio.h>

#include<graphics.h>

#include<dos.h>

#include<math.h>

void main()

{

float x1,x2,y1,y2,xinc,yinc,l,dx,dy,x,y;

float temp;

int gd=DETECT,gm;

initgraph(&gd,&gm,"C:\\turboc3\\bgi ");

cout<<"Enter the coordinates of first point"<<endl;

cin>>x1>>y1;

cout<<"Enter the coordinates of second point"<<endl;

cin>>x2>>y2;

dx=x2-x1;

dy=y2-y1;

if(dx>=dy)

l=dx;

else

l=dy;

xinc=dx/l;

yinc=dy/l;

x=abs(x1+0.5);

y=abs(y1+0.5);

for(int i=1;i<=l;i++)

{

putpixel(x,y,WHITE);

x=x+xinc;

y=y+yinc;

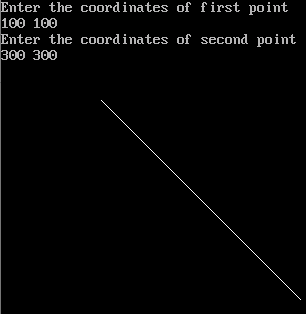
}

getch();

closegraph();

}

**OUTPUT:-**



**EXPERIMENT NO. 1(b)**

**AIM-**To draw a line using Symmetrical DDAAlgorithm for positive line slope.

**Source Code:**

#include<iostream.h>

#include<conio.h>

#include<graphics.h>

#include<dos.h>

#include<math.h>

void main()

{

intgd=DETECT,gm;

initgraph(&gd,&gd,"C:\\turboc3\\bgi");

int x1,y1,x2,y2,i,dx,dy;

floatx,y,xInc,yInc,length;

cout<<"Ente the co-odinates of first point"<<endl;

cin>>x1>>y1;

cout<<"Ente the co-odinates of second point"<<endl;

cin>>x2>>y2;

dx=x2-x1;

dy=y2-y1;

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

length=abs(dx);

else

length=abs(dy);

float n=log10(length)/log10(2);

cout<<"value of n="<<n;

xInc=dx/(pow(2,n));

yInc=dy/(pow(2,n));

x=x1+0.5;

y=y1+0.5;

putpixel(x,y,WHITE);

for(i=1;i<=length;i++)

{

x=x+xInc;

y=y+yInc;

putpixel(x,y,WHITE);

delay(100);

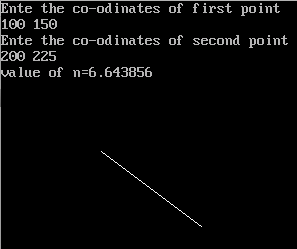
}

getch();

closegraph();

}

**OUTPUT:-**



**EXPERIMENT NO. 1(c)**

**AIM-**To draw a line using Bresenham’s Algorithmfor positive line slope.

**Source Code:**

#include<iostream.h>

#include<conio.h>

#include<graphics.h>

#include<dos.h>

void main()

{

intgd=DETECT,gm;

initgraph(&gd,&gd,"C:\\turboc3\\bgi");

int x1,x2,y1,y2,dx,dy,x,y,p,ymax,xmax;

cout<<"Ente the co-odinates of first point"<<endl;

cin>>x1>>y1;

cout<<"Ente the co-odinates of second point"<<endl;

cin>>x2>>y2;

dx=x2-x1;

dy=y2-y1;

if(x1>x2)

{

x=x2;

y=y2;

}

else

{

x=x1;

y=y1;

}

p=2\*dy-dx;

ymax=(y1>=y2)?y1:y2;

xmax=(x1>=x2)?x1:x2;

// while(x<=x2 && y<=y2)

while(x<=ymax || y<=xmax)

{

if(p<0)

{

x=x+1;

p=p+2\*dy;

}

else if(p>=0)

{

x=x+1;

y=y+1;

p=p+2\*dy-2\*dx;

}

putpixel(x,y,WHITE);

delay(50);

}

getch();

closegraph();

}

**OUTPUT:-**

