/\* Program for implementation of DDA ALOGORITHM

\* Name - Ashish Doneriya

\* Scholar No. 101112011

\*/

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

int mod(int);

int sign(int);

void main()

{

int x1,y1,x2,y2,nod\_x,nod\_y,length,i=1;

float ru\_x,ru\_y,x,y;

/\* Here x1,y1 and x2,y2 are the coordinates of line

\* nod denotes number of difflections

\* length denotes length of the line

\* ru denotes raster unit or difflection voltage

\* x any y are plotting pixels

\*/

int gd=DETECT,gm;

initgraph(&gd,&gm,"C:/BGI");

/\* Here we give path as C:/BGI because

\* we use dosbox here and mounting tc folder

\* as c:

\*/

clrscr();

printf("Enter the coordinates of point 1 (x1,y1) : ");

scanf("%d %d",&x1,&y1);

printf("Enter the coordinates of point 2 (x2,y2) : ");

scanf("%d %d",&x2,&y2);

nod\_x=mod(x2-x1);

nod\_y=mod(y2-y1);

if (nod\_x>=nod\_y)

length=nod\_x;

else

length=nod\_y;

ru\_x=(float)(x2-x1)/length;

ru\_y=(float)(y2-y1)/length;

x=x1+0.5\*sign(ru\_x);

y=y1+0.5\*sign(ru\_y);

while(i<=length)

{

putpixel((int)x,(int)y,BLACK);

x=x+ru\_x;

y=y+ru\_y;

i++;

}

getch();

}

int mod(int num)

{

if(num<0)

return (-num);

else

return num;

}

int sign(int num)

{

if(num<0)

return -1;

else

return 1;

}

OUTPUT

