**Aim:** To implement Bresenham’s algorithms for drawing a line segment between two given end points.

**Objective:**

Draw a line using Bresenham's line algorithm that determines the points of an n-dimensional raster that should be selected to form a close approximation to a straight line between two points

**Theory:**

In Bresenham’s line algorithm pixel positions along the line path are obtained by determining the pixels i.e. nearer the line path at each step.

**Algorithm - {**

**x=x1;**

**y=y1;**

**dx=x2-x1;**

**dy=y2-y1;**

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

**while(x<=x2)**

**{**

**putpixel(x,y);**

**x++;**

**if(p<0)**

**{**

**p=p+2dy;**

**}**

**else**

**{**

**y=y+1;**

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

**}**

**}**

**}**

**Program -**

**#include<stdio.h>**

#include<conio.h>

#include<graphics.h>

#include<math.h>

#include<dos.h>

void main()

{

int gd=DETECT, gm;

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

clrscr();

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

printf("Enter the first co-ordinates:\n");

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

printf("Enter the second co-oedinates:\n");

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

dx=x2-x1;

dy=y2-y1;

p=2\*dy-dx;

while(x1<=x2){

putpixel(x1,y1,50);

x1++;

if(p<0){

p=p+2\*dy;

}

else

{

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

y1++;

}

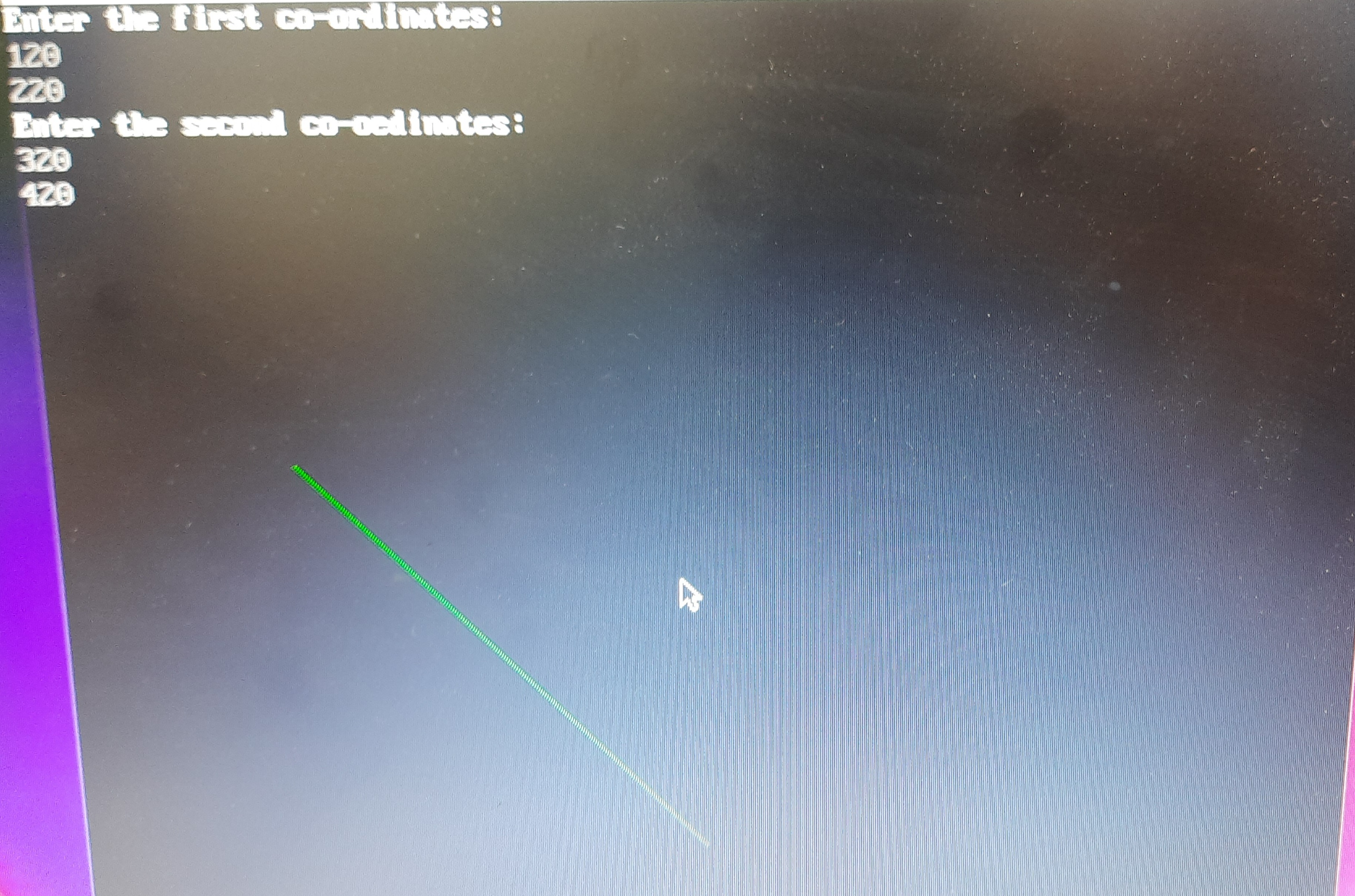
}

getch();

closegraph();

}

**Output –**



**Conclusion:** Comment on -

1. Pixel :- A pixel is the smallest unit of a digital image or display. It represents a single point and is typically square in shape. Pixels are fundamental to digital graphics and imaging, as they collectively create images by combining various colors and intensities. The quality and resolution of an image depend on the arrangement and number of pixels.
2. Equation for line :-Bresenham's Line Drawing Algorithm is a method used in computer graphics to draw a straight line between two given endpoints on a discrete grid, such as a computer screen. Unlike the more traditional slope-intercept form of a line equation (y = mx + b), Bresenham's algorithm is designed for efficient integer-only calculations, making it well-suited for raster graphics. It is widely used in graphics applications to draw lines efficiently.
3. Need of line drawing algorithm :- The need for a line drawing algorithm arises in computer graphics and image processing to efficiently and accurately render straight lines on digital displays. Without an algorithm like Bresenham's, drawing lines pixel by pixel using traditional equations (e.g., slope-intercept form) can be computationally expensive and lead to rounding errors. Bresenham's algorithm addresses these issues by providing a precise and efficient method for drawing lines, making it a fundamental tool for rendering graphics on digital screens.
4. Slow or fast :- Bresenham's Line Drawing Algorithm is known for its speed and efficiency in drawing lines on a digital grid. It utilizes integer calculations and avoids floating-point operations, which can be slower on many computer systems. As a result, it is considered a fast algorithm for line drawing and is widely used in computer graphics applications to achieve real-time rendering of lines on screens.