**Write a program to implement Bresenham's Line Drawing Algrorithm.**

**ALGORITHM:**

**Step1:** Start Algorithm

**Step2:** Declare variable x1,x2,y1,y2,d,i1,i2,dx,dy

**Step3:** Enter value of x1,y1,x2,y2  
                Where x1,y1are coordinates of starting point  
                And x2,y2 are coordinates of Ending point

**Step4:** Calculate dx = x2-x1  
                Calculate dy = y2-y1  
                Calculate i1=2\*dy  
                Calculate i2=2\*(dy-dx)  
                Calculate d=i1-dx

**Step5:** Consider (x, y) as starting point and xendas maximum possible value of x.  
                If dx < 0  
                        Then x = x2  
                        y = y2  
                          xend=x1  
                If dx > 0  
                    Then x = x1  
                y = y1  
                        xend=x2

**Step6:** Generate point at (x,y)coordinates.

**Step7:** Check if whole line is generated.  
                If x > = xend  
                Stop.

**Step8:** Calculate co-ordinates of the next pixel  
                If d < 0  
                    Then d = d + i1  
                If d ≥ 0  
          Then d = d + i2  
                Increment y = y + 1

**Step9:** Increment x = x + 1

**Step10:** Draw a point of latest (x, y) coordinates

**Step11:** Go to step 7

**Step12:** End of Algorithm

**PROGRAM:**

#include<stdio.h>

#include<dos.h>

#include<graphics.h>

void drawline(int x0, int y0, int x1, int y1)

{

int dx, dy, p, x, y;

dx=x1-x0;

dy=y1-y0;

x=x0;

y=y0;

p=2\*dy-dx;

while(x<x1)

{

if(p>=0)

{

putpixel(x,y,7);

y=y+1;

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

delay(200);

}

else

{

putpixel(x,y,7);

p=p+2\*dy;}

x=x+1;

delay(200);

}

}

int main()

{

int gdriver=DETECT, gmode, error, x0, y0, x1, y1;

initgraph(&gdriver, &gmode, "c:\\turboc3\\bgi");

printf("Enter co-ordinates of first point: ");

scanf("%d%d", &x0, &y0);

printf("Enter co-ordinates of second point: ");

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

drawline(x0, y0, x1, y1);

return 0;

}

**OUTPUT:**

