Name. Trinetra Joshi Rollno. 112177

Course. BE AEC Sig. Joshi Set - C

1. Bresenham Line Drawing Algorithm

Also

Step 1. Start

Step 2. Declare required variable x1, x2, y1, y2, i1, i2, d, dx, dy

Step 3. Assign values to x1, x2, y1, y2

Step 4. Calculate: dx = x2 - x1

Calculate dy = y2 - y1

Calculate i1 = 2 * dy

Calculate i2 = 2 x (dy - dx)

Calculate d = i1 - dx

Step 5. Consider (x, y) as starting points and x end as maximum value possible for x

If dx < 0

Then Assign: x = x2

y = y2

x end = x1

If dx > 0

Then x = x1

y = y1

x end = x2

Step 6: Generate (x, y)

Step 7: Check if generation is complete

If x >= x end

then Stop

Step 8: Calculate: If d < 0 : d = d + i1 ; If d = d + i2 : y ++

Step 9: x++

Step 10: Draw new values of (x, y)

Step 11: Go to Step 7

Step 12: Stop

Name. Trinetra Joshi
Course. BCA 6c

Roll no. 1121177
sig. tjoshi
Set - C

P1. Brosenhom LDA code

```
#include <stdio.h>
#include <graphics.h>

void draw (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;
      }
      else{
            putpixel (x, y, 7);
            p = p + 2*dy;
            x = x+1;
         }

}

int main () {

   int gd = DETECT, gm, er, x0, y0, x1, y1;
   initgraph (&gd, &gm, "C:\\Programs\\Turboc3\\bgi");
   printf (" Enter coordinates of first point ");
   scanf ("%d %d", &x0, &y0);
   printf (" Enter final point coordinates");
   scanf ("%d %d", & x1, & y1);
   draw (x0, y0, x1, y1);
   return 0;

}
```