Name⇒ Saurabh Bhardwaj

Course⇒ BCA-6-C

Rollno⇒ 1121146 (31)

Sub⇒ Computer graphics & animation

Date⇒ 16/06/21

Section⇒ C

Ans 1) Algorithm for Bresenham's line drawing algo.

Step 1) Start

Step 2) Declare x_1, y_1, x_2, y_2

Step 3) Calculate $dx = x_2 - x_1$

$dy = y_2 - y_1$

Step 4) Calculate slope $m = dy/dx$

Step 5) For $m < 1$: calculate initial decision variable $P = 2dy - dx$.

Step 6) While $(x_1 <= x_2)$

if $(P < 0)$:

$x_k = x_{k+1}$

$P = P + 2dy$

$y_k = y_k$

else

$x_k = x_{k+1}$

$P = P + 2dy - 2dx$

$y_k = y_{k+1}$

Step 7) Plot (x_k, y_k)

Step 8) Stop

Scanned with CamScanner

Code

```
#include <stdio.h>
#include <graphics.h>
int main()
{
  int gd = DETECT, gm, x0, y0, x1, y1, dx, dy, p, x, y;
  printf("Co-ordinate of first point:");
  printf("\n Enter the value of x1:");
  scanf("%d", &x0);
  printf("Enter the value of y1:");
  scanf("%d", &y0);
  printf("Co-ordinate of second point:");
  printf("Enter the value of x2:");
  scanf("%d", &x1);
  printf("Enter the value of y2:");
  scanf("%d", &y1);
  initgraph(&gd, &gm, "");
    dx = x1 - x0;
    dy = y1 - y0;
    x = x0;
    y = y0;
    p = 2*dy - dx;
    while (x < x1)
  {
    if (p >= 0)
    {
      putpixel(x, y, 4);
```

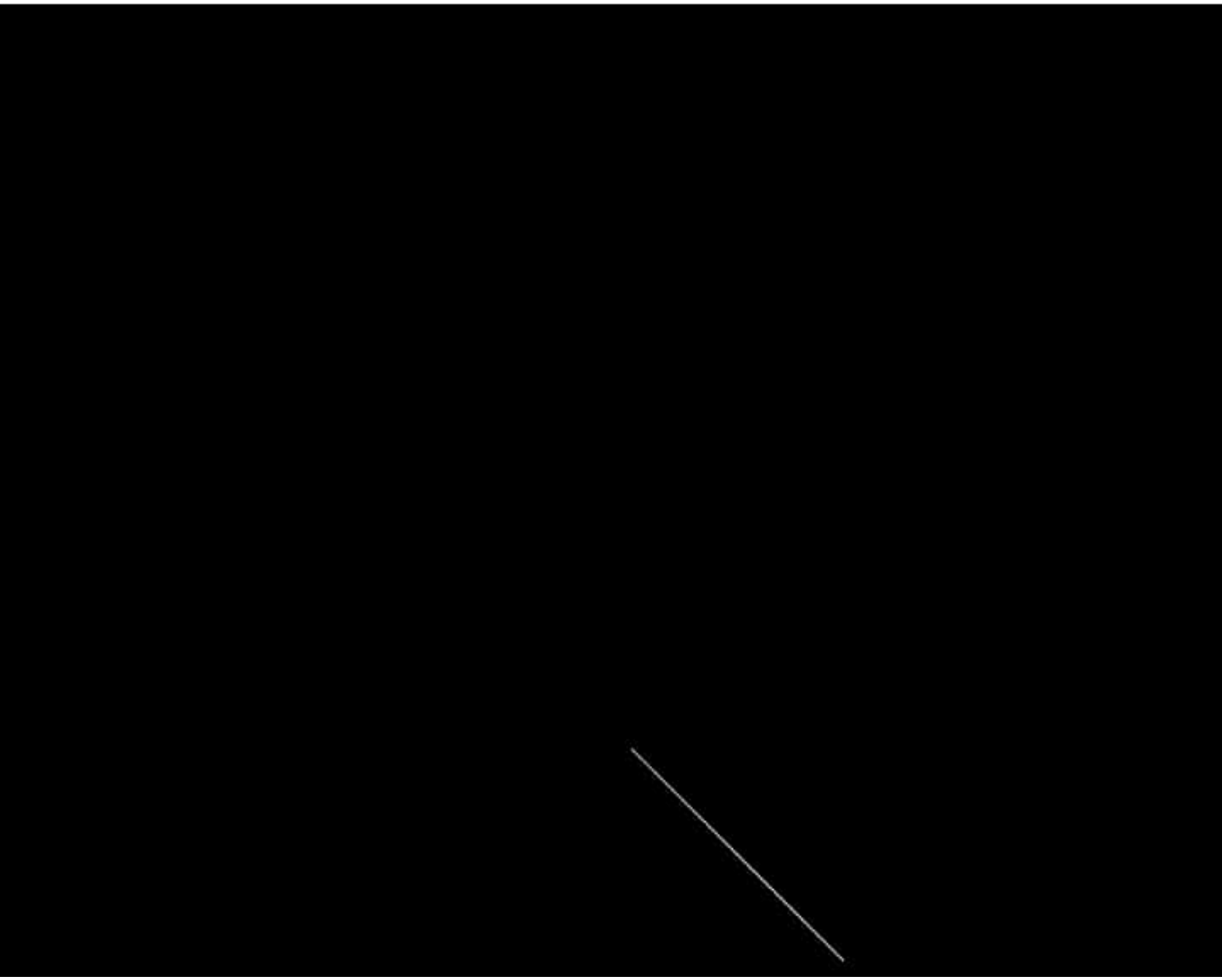
Scanned with CamScanner

```
        y = y+1;
        p = p+2*dy - 2*dx
    }
    else
    {
        putpixel (x, y, 4)
        p = p + 2*dy
    }
    x = x+1;
}
    getch ();
    return 0;
}
```

Scanned with CamScanner

```
nter co-ordinates of first point: 300 350
nter co-ordinates of second point: 400 500
```

Windows BGI



Scanned with CamScanner