**Name: Khushi Nitinkumar Patel**

**PRN: 2020BTECS00037**

**Batch: T5**

**Experiment 3: Implementation line algorithms**

**CODE:**

#include <iostream>

#include <graphics.h>//Uncomment the graphics library functions if you are using it

using namespace std;

void plotPixel(int x1, int y1, int x2, int y2, int dx, int dy, int decide)

{

int pk = 2 \* dy - dx;

for (int i = 0; i <= dx; i++)

{

cout << x1 << "," << y1 << endl;

x1 < x2 ? x1++ : x1--;

if (pk < 0)

{

if (decide == 0)

{

putpixel(x1, y1, RED);

pk = pk + 2 \* dy;

}

else

{

// (y1,x1) is passed in xt

putpixel(y1, x1, RED);

pk = pk + 2 \* dy;

}

}

else

{

y1 < y2 ? y1++ : y1--;

if (decide == 0)

{

putpixel(x1, y1, RED);

}

else

{

putpixel(y1, x1, RED);

}

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

}

}

}

int main()

{

int gd = DETECT, gm;

initgraph(&gd, &gm, "xxx");

int x1 = 100, y1 = 110, x2 = 500, y2 = 500, dx, dy, pk;

dx = abs(x2 - x1);

dy = abs(y2 - y1);

if (dx > dy)

{

plotPixel(x1, y1, x2, y2, dx, dy, 0);

}

else

{

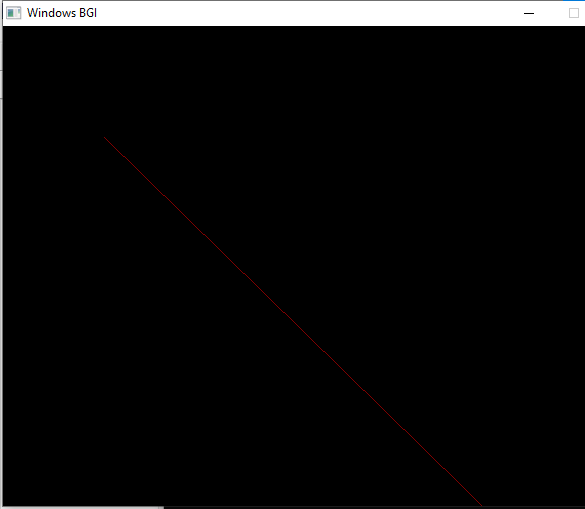
plotPixel(y1, x1, y2, x2, dy, dx, 1);

}

getch();

}

**OUTPUT:**

****