Name pranav Narayan Pisal roll no 63 div 2

Program2 Bresonhans line drawing algorithm

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 -

```
–(x1,y1,x0,y0)
dx=x1-x0
dy=y1-y0
p0=2dy-dx
for k=0 to dx do
if pk<0 then
putpixel(xi+1,yi)
pn=pk+2dy
else
putpixel(xi+1,yi+1)
pn=pk+(2dy-2dx)
end
end
```

Program

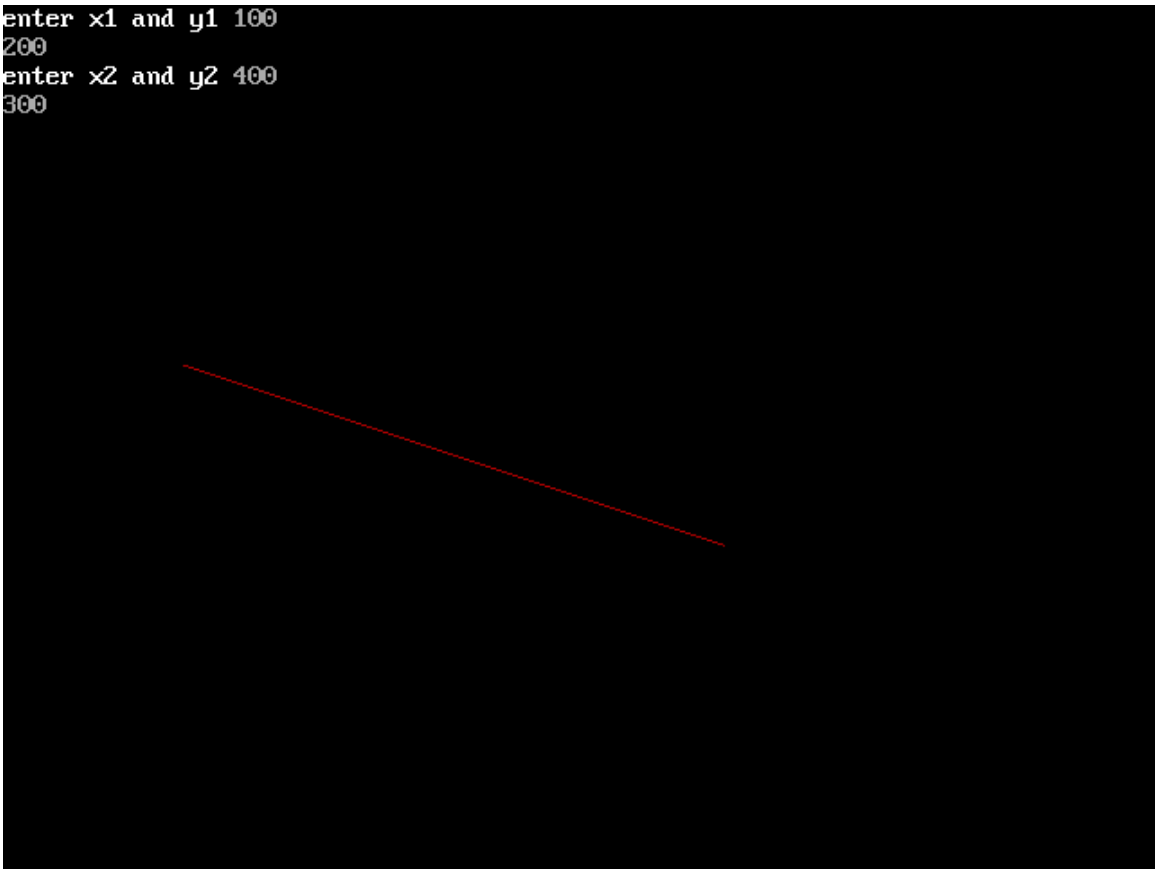
```
#include<stdio.h>
#include<conio.h>
#include<graphics.h
>

void main(){

int x1,y1,x2,y2,dx,dy,p,x,y;int
gd=DETECT,gm;

initgraph(&gd,&gm,"");
print("enter x1 and y1 ");
scanf("%d %d",&x1,&y1);
print("enter x2 and y2 ")
;scanf("%d %d",&x2,&y2);
x=x1;
y=y1;
dx=x2-x1;
dy=y2-y1;


p=2*dy-2*dx;
while(x<=x2)
{putpixel(x,y,RED);x++;
if(p<0)
{p=p+2*dy;
}
else
{p=p+2*dy-2*dx;
y++;
}
}
getch();
closegraph();
}
```



Conclusion: Comment on -
Pixel:-Bresenham's algorithm does not perform any rounding operation
Equation for line:-y=mx+c
Need of line drawing algorithm:-Involves cheaper operation like addition and subtraction
4.Slow or fast:-It is faster than DDA