INSTITUTE OF ENGINEERING

ADVANCED COLLEGE OF ENGINEERING AND MANAGEMENT
KALANKI, KATHMANDU
(AFFILIATED TO TRIBHUVAN UNIVERSITY)

ADVANCED COLLEGE
OF ENGINEERING & MANAGEMENT

LAB REPORT

SUBJECT : Computer Graphics

LAB NO : 3

SUBMITTED BY:

NAME : Aayush Basnet
ROLL NO: 004
DATE :

SUBMITTED TO:

Department of Computer
and Electronics

Title: Introduction to Bresenham's Algorithm

bjective:-

o learn Bresenham's algorithm and implement it using C-programming

heory:-

It is an accurate and efficient raster line generation of algorithm eveloped by Bresenham.

Assuming the pixel at (x_k, y_k) to be displayed nd is determined, we need to decide which pixel at position (x_k+1, y_k) d (x_k+1, y_k+1).

we have,

For slope $(m < 1)$ and positive value.

At (x, y)

$y = mx + c$ ··· (i)

t (x_k+1, y)

$y = m(x_k+1) + c$ ··· (ii)

ow,

${}_1 = y - y_k$

${}_2 = y_{k+1} - y$

${}_1 - d_2 = y - y_k - y_{k+1} + y$

$= 2y - 2y_k - 1$

$= 2\left(\frac{\Delta y}{\Delta x}(x_m + 1) + c\right) - 2y_k - 1$

$(d_1 - d_2) = 2\Delta y\, x_k + \Delta y + 2\Delta y - 2\Delta x\, y_k - \Delta x$

$P_k = 2\Delta y\, x_k - 2\Delta x\, y_k + b$ ···· (iii)

ere,

$P_k = \Delta x(d_1 - d_2)$ = Decision parameter

$b = \Delta y - \Delta x - 2\Delta x\, y_k$

ext decision parameter is,

$P_{k+1} = 2\Delta y\, x_{k+1} - 2\Delta x\, y_{k+1} + b$ ····· (iv)

Applying (iv) - (iii)

$P_{k+1} = P_k + 2\Delta y(x_{k+1} - x_k) - 2\Delta x(y_{k+1} - y_k)$

If $P_k < 0$

$x_{k+1} = x_k + 1$

$y_{k+1} = y_k$

$P_{k+1} = P_k + 2\Delta y$

If $P_k \geq 0$

$x_{k+1} = x_k + 1$

$y_{k+1} = y_k + 1$

$P_{k+1} = P_k + 2\Delta y - 2\Delta x$

Now, The initial decision parameter is,

$P_0 = 2\Delta y - \Delta x$

For slope, $|m| > 1$, we interchange x and y.

Algorithm

```
Input two points $(x_1, y_1)$ and $(x_2, y_2)$.
Compute $\Delta x = |x_2 - x_1|$ and $\Delta y = |y_2 - y_1|$
If $(x_2 > x_1)$
      $dx = 1$
   else
      $dx = -1$

If $(y_2 > y_1)$
     $dy = 1$
  else
     $dy = -1$

Plot $(x_1, y_1)$
If $\Delta x > \Delta y$ (i.e. $|m| < 1$)
{
   6.1 calculate $P_{0K} = 2\Delta y - \Delta x$
   6.2 Starting at $K = 0$ to $\Delta x$ times, repeat
          if $(P_K < 0)$
               $x_1 = x_1 + dx$
               $y_1 = y_1$
               $P_K = P_K + 2\Delta y$
          else
               $x_1 = x_1 + dx$
               $y = y_1 + dy$
}              $P_K = P_K + 2\Delta y - 2\Delta x$
```

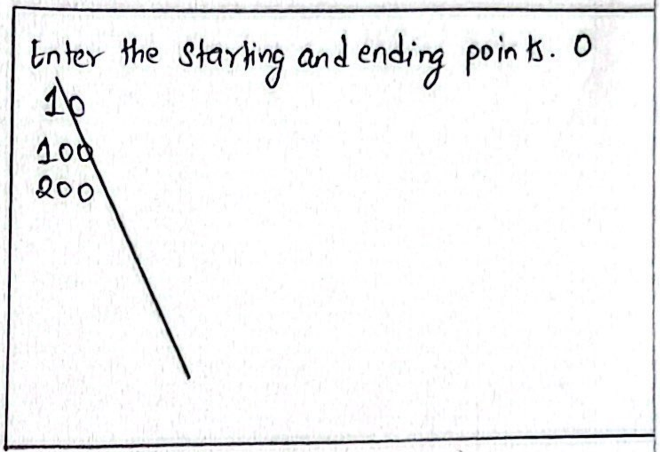
```
else (i.e. $|m| > 1$)
{
    6.1 Calculate  $P_K = 2\Delta x - \Delta y$
    6.2 Starting at $k=0$ to $\Delta y$ times, repeat
        if ($P_K < 0$)
            $x_1 = x_1$
            $y_1 = y_1 + dy$
            $P_K = P_K + 2\Delta x$
        else
            $x_1 = x_1 + dx$
            $y_1 = y_1 + dy$
            $P_K = P_K + 2\Delta x - 2\Delta y$
}

End
```

WAP to implement [redacted] algorithm.

```
#include <stdio.h>
#include <conio.h>
#include < graphics.h>
#include< dos.h>
int main()
{
 int gd = DETECT, gm, i;
 int x1,x2,y1,y2, lx, ly, po,dx,dy;
 initgraph(&gd, &gm, "C:\\TURBOC3\\BGI");
 printf("Enter the starting and ending points.");
 scanf("%d %d %d %d",&x1, &y1, &x2, &y2);
 dx= x2-x1;
 dy=y2-y1;
 lx= x2>x1 ? 1:-1;
 ly = y2>y1? 1:-1;
 if (dx>dy){
  po = 2*dy-dx;
  for (i=0; i<=dx ; i++){
   if (po<0){
   x1 = x1+lx;
   y1 = y1;
   po = po+ 2*dy; }
  else{
   x1 = x1+lx;
   y1 =y1+ly;
   po= po+2*dy -2*dx; }
   putpixel (x1,y1, RED);}}
 else{
  po= 2*dx-dy;
  for(i=0 ; i<=dy ;i++){
  if (po<0){
   x1=x1;
   y1=y1+ly;
   po = po+2*dx; }
else{
x1=x1+dx;
y1=y1+dx;
p0=p0+2*dx-2*dy;}
putpixel(x1,y1,RED);}}

getch();
closegraph();
return 0;
}
```



```
Enter the starting and ending points. 0
10
100
200
```

iscussion and Conclusion

n this lab we discussed about Bresenham's algorithm and used it o draw a line. Bresenham's algorithm helped us to draw a more accurate line compared to DDA algorithm as increment in either x or y co-ordinate was decided by the decision parameter which helped to draw a more accurate line.

Thus our objective to learn and implement Bresenham's algorithm was fulfilled.