

Name - Vikas Bisht

Vikas Bisht

~~Course~~ Sub - Computer Graphics & Animation.

Roll no - 1121164 (49)

Date - 16-06-2021

Paper code - PBC-602

Course - BCA 6-C

1 Ans)

Source Code

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

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

```
int main ()
```

```
{  
    int gd = DETECT, gm, x0, y0, x1, y1, dx, dy,  
        p, x, y;
```

```
printf("Co-ordinates of first point: ");
```

```
printf("\n Enter the value of x1: ");
```

```
scanf("%d", &x0);
```

```
printf("Enter the value of y1: ");
```

```
scanf("%d", &y0);
```

```
printf("Co-ordinates of Second point: ");
```

```
printf("\n Enter the value of x2: ");
```

```
scanf("%d", &y1);
```

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

```
dx = x1 - x0;
```


$dy = y_1 - y_0;$

$x = x_0;$

$y = y_0;$

$P = 2 * dy - dx;$

while ($x < x_1$)

{

if ($P \geq 0$)

{

putpixel ($x, y, 4$);

$y = y + 1;$

$P = P + 2 * dy - 2 * dx;$

}

else {

putpixel ($x, y, 4$);

~~$P = P + 2 * dx$~~ $P = P + 2 * dy;$

}

$x = x + 1;$

}

getch();

return 0;

}

Algorithm

Step 1: Start Algorithm

Step 2: Declare variable $x_1, x_2, y_1, y_2, d, i_1, i_2, dx, dy$

Step 3: Enter value of x_1, y_1, x_2, y_2

Where x_1, y_1 are coordinates of starting point
And x_2, y_2 are coordinates of ending point.

Step 4 - Calculate $dx = x_2 - x_1$
Calculate $dy = y_2 - y_1$
Calculate $i_1 = 2 * dy$
Calculate $i_2 = 2 * (dy - dx)$
Calculate $d = i_1 - dx$

Step 5 - Consider (x, y) as starting point and x_{end} as maximum possible value of x .

if $dx < 0$

Then $x = x_2$

$y = y_2$

$x_{end} = x_1$

if $dx > 0$

Then $x = x_1$

$y = y_1$

$x_{end} = x_2$

Step 6 - Generate point at (x, y) coordinates.

Step 7 - Check if whole line is generated

if $x \geq x_{end}$

Stop.

Step 8 - Calculate co-ordinates of the next pixel

if $d < 0$

Then $d = d + i_1$

if $d \geq 0$

Then $d = d + 1$

Increment $y = y + 1$

Step 9:- Increment $x = x + 1$

Step 10:- Draw a point of latest (x, y) coordinates

Step 11:- Go to Step 7

Step 12:- End of Algorithm.

```
pc-102@gehu-HP-EliteDesk-800-G2-SFF:~/Desktop/Vatsal_G$ gcc bres_line.c -o bress -lgraph
pc-102@gehu-HP-EliteDesk-800-G2-SFF:~/Desktop/Vatsal_G$ ./bress
Co-ordinates of first point:
Enter the value of x1: 100
Enter the value of y1: 100
Co-ordinates of second point:
Enter the value of x2: 265
Enter the value of y2: 275
[xcb] Un
[xcb] Mo
called
[xcb] Ab
bress: .
ence_lo

```

SDL-libgraph -- Graphics on GNU/Linux

