

END SEM EXAMINATION

NAME → ANSHUL UNJHAL

COURSE → B.C.A.

SEM → 6th

SEC → A

SUBJECT → COMPUTER GRAPHICS &
ANIMATION

UNIVERSITY ROLL. NO → 1121020

PAPER CODE → TBC-602

Q → 1.7 Ans →

```
#include <stdio.h>
#include <graphics.h>
int main()
{
    int erou(float num)
    {
        return num < 0 ? num - 0.5 : num + 0.5;
    }
    int x1 = 100, x2 = 250, y1 = 100, y2 = 250, step;
    int gd = DETECT, gm;
    float x, y, m;
    int dx = x2 - x1;
    int dy = y2 - y1;
    m = dy/dx;
    if (dx > dy)
        step = dx;
    else
        step = dy;
    initgraph(&gd, &gm, "");
    outtextxy(x1, y1, "A");
    outtextxy(x2, y2, "B");
    putpixel(x1, y1, RED);
    x = x1, y = y1;
    while (step > 0)
    {
        if (m < 1)
```



```

    {
        x = x + 1;
        y = y + m;
    }
    if (m > 1)
    {
        x = x + 1/m;
        y = y + 1;
    }
    putpixel(xou(x), xou(y), RED);
    Step--;
}
getch();
return 0;
}

```

ALGORITHM

- Sol → • Starting Coordinates = (x_0, y_0)
 • Ending coordinates = (x_n, y_n)

The points generation using DDA algorithm involves the following steps:

Steps → 1. Calculate Δx , Δy and M from the given input. We know that the slope of a straight line M is given as.

These parameters are calculated as

- $\Delta x = x_n - x_0$
- $\Delta y = y_n - y_0$
- $M = \Delta y / \Delta x \Rightarrow M = \frac{y_n - y_0}{x_n - x_0}$

Step-2 → Find the number of steps or points in between the starting and ending coordinates.

if $(\text{absolute}(\Delta x) > \text{absolute}(\Delta y))$

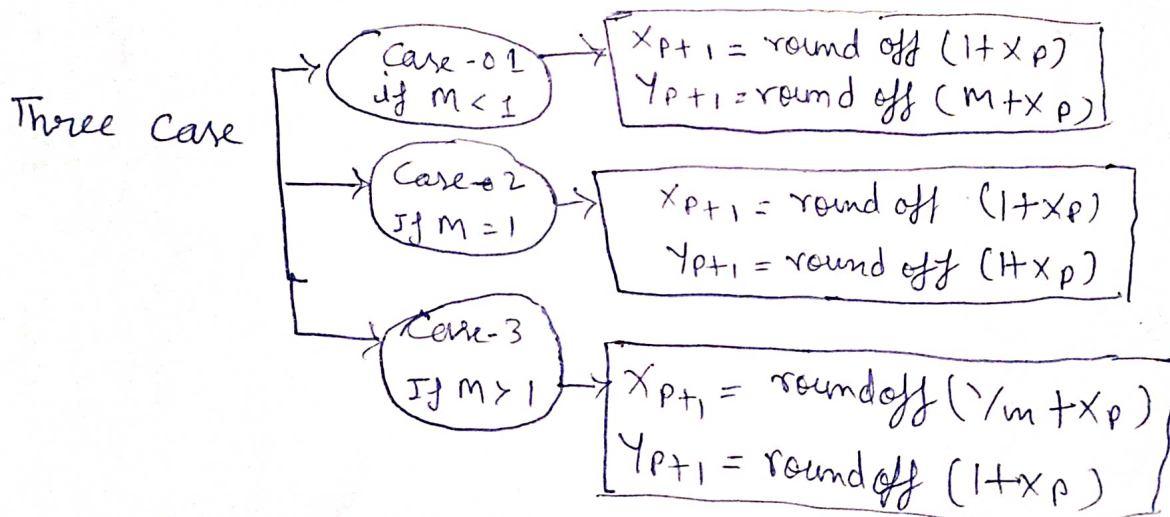
Steps = $\text{absolute}(\Delta x)$;

else

Steps = $\text{absolute}(\Delta y)$;

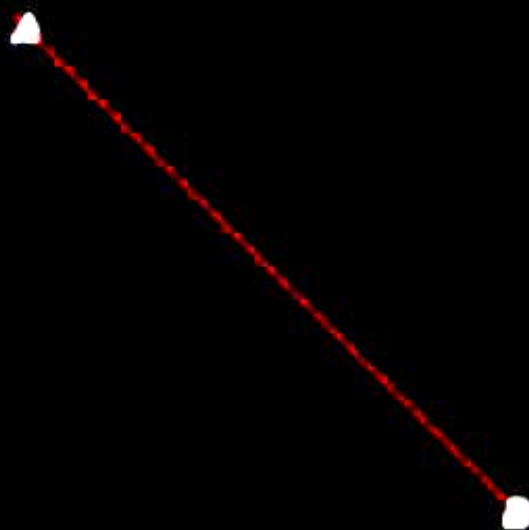
Step-3 → Suppose the current point is (x_p, y_p) and the next point is (x_{p+1}, y_{p+1})

find the next by following the below three cases:



Step-4 → Keep repeating Step-3 untill the end points (including the starting and ending points) equals to the steps count.

structure



has no

threads.