

Name - Deepak Jantwal
Course - BCA Sec. - A
Subject - Computer graphics.
Roll No. - 1121039

Ans \Rightarrow

Step 1 - Start

Step 2 - Read 2 end points of line as $P1(x1, y1)$
& $P2(x2, y2)$

Step 3 - Read 2 corner points of the Clipping window
(left - top & right - bottom) as $(wx1, wy1)$
& $(wx2, wy2)$

Step 4 - Assign the region codes for 2 endpoints $P1$ &
 $P2$ using following steps:

initialize code with 0000

Set bit 1 if $x < wx1$

Set bit 2 if $x > wx2$

Set bit 3 if $y < wy1$

Set bit 4 if $y > wy2$

Step 5 - Check for visibility of line

a. If region codes for both endpoints are zero then
line is completely visible. Draw the line go to step 9.

b. If region codes for endpoints are not zero &
logical ANDing of them is also non zero then line
is invisible. Discard the line & move to step 9.

c. If it does not satisfy 4.a & 4.b then line is
partially visible.

Step 6: Determines the intersecting edge of clipping window as follows:

- a. If region codes for both endpoints are non-zero find intersection points P_1 & P_2 with boundary edges.
- b. If region code for any one end point is non-zero then find intersection point P_1 or P_2 .

Step 7: Divide the line segments considering intersection points.

Step 8: Reject line segment if any end point of line appears outside of any boundary.

Step 9: Draw the clipped line segment.

Step 10: Stop.

Program :

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <graphics.h>
#include <dos.h>
```

```
typedef struct Coordinate
```

```
{
```

```
int x, y;
```

```
char Code[4];
```

```
} PT;
```

```
void drawwindow();
```

```
void drawline (PT P1, PT P2);
```

```
PT SetCode (PT P);
```

```
int Visibility (PT P1, PT P2);
```

```
PT reetendpt (PT P1, PT P2);
```

```
void main()
```

```
{ int gd = DETECT, v, gm;
```

```
PT P1, P2, P3, P4, Ptemp;
```

```
printf ("\n Enter x1 & y1 \n");
```

```
scanf ("%d %d", &P1.x, &P1.y);
```

```
printf ("\n Enter x2 & y2 \n");
```

```
scanf ("%d %d", &P2.x, &P2.y);
```

```
initgraph (&gd, &gm, "C:\\turbo C3\\bgi");
```

```
drawwindow();
```

```
delay (500);
```

```
drawline (P1, P2);
```

```
delay (500);
```

```

ClearDevice();
delay(500);
P1 = SetCode(P1);
P2 = SetCode(P2);
V = visibility(P1, P2);
delay(500);
Switch (V)
{
Case 0: drawwindow();
delay(500);
drawline(P1, P2);
break;
Case 1: drawwindow();
delay(500);
break;
Case 2: P3 = rotatept(P1, P2);
P4 = rotatept(P2, P1);
drawwindow();
delay(500);
drawline(P3, P4);
break;
}
delay(5000);
CloseGraph();
}

void drawwindow()
{
line(150, 100, 450, 100);
line(450, 100, 450, 350);
line(450, 350, 150, 350);
line(150, 350, 150, 100);
}

```



```

3
void drawline (PT p1, PT p2)
{
    line (p1.x, p1.y, p2.x, p2.y);
}

PT ptemp;
if (p.y < 100)
    ptemp.Code[0] = '1';
else
    ptemp.Code[0] = '0';
if (p.y > 350)
    ptemp.Code[1] = '1';
else
    ptemp.Code[1] = '0';
if (p.x > 450)
    ptemp.Code[2] = '1';
else
    ptemp.Code[2] = '0';
if (p.x < 150)
    ptemp.Code[3] = '1';
else
    ptemp.Code[3] = '0';
ptemp.x = p.x;
ptemp.y = p.y;
return (ptemp);
}

int visibility (PT p1, PT p2)
{
    int i, flag = 0;

```

```
for (i=0; i<4; i++)
```

```
{
```

```
if ((P1.Code[i] != '0') || (P2.Code[i] != '0'))
```

```
flag = 1;
```

```
}
```

```
if (flag == 0)
```

```
return 0;
```

```
for (i=0; i<4; i++)
```

```
{
```

```
if ((P1.Code[i] == P2.Code[i]) && (P1.Code[i] == '1'))
```

```
flag = '0';
```

```
}
```

```
if (flag == 0)
```

```
return 1;
```

```
return 2;
```

```
}
```

```
PT resetendpt (PT P1, PT P2)
```

```
{
```

```
PT temp;
```

```
int x, y, i;
```

```
float m, R;
```

```
if (P1.Code[3] == '1')
```

```
x = 150;
```

```
if (P1.Code[2] == '1')
```

```
x = 450;
```

```
if ((P1.Code[3] == '1') || (P1.Code[2] == '1'))
```

(4)

$$m = (\text{float})(P2.y - P1.y) / (P2.x - P1.x);$$

$$R = (P1.y + (m * (x - P1.x)))$$

$$\text{temp}.y = R;$$

$$\text{temp}.x = x;$$

$$\text{for } (i = 0; i < 4; i++)$$

$$\text{temp}.Code[i] = P1.Code[i];$$

$$\text{if } (\text{temp}.y < 350 \text{ \&\& } \text{temp}.y > 100)$$

$$\text{return } (\text{temp});$$

3

$$\text{if } (P1.Code[0] == '1')$$

$$y = 100;$$

$$\text{if } (P1.Code[0] == '1')$$

$$y = 350;$$

$$\text{if } ((P1.Code[0] == '1') \parallel (P1.Code[1] == '1'))$$

{

$$m = (\text{float})(P2.y - P1.y) / (P2.x - P1.x);$$

$$R = (\text{float})P1.x + (\text{float})(y - P1.y) / m;$$

$$\text{temp}.x = R;$$

$$\text{temp}.y = y;$$

$$\text{for } (i = 0; i < 4; i++)$$

$$\text{temp}.Code[i] = P1.Code[i];$$

$$\text{return } (\text{temp});$$

3

$$\text{else return } (P1);$$

3

(5)

Enter x1 and y1

100

100

Enter x2 and y2

200

200_

Before Clipping



