

POORNIMA

Lab - 1

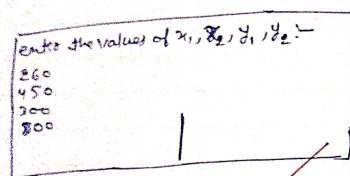
Experiment - 1

objective :- Implementation of Pixel , line , circle , ellipse and other attributes .

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

void main()
{
    int gd = DETECT, gm;
    initgraph(&gd, &gm, "path");
    outtextxy(100, 50, "circle");
    line(300, 300, 400, 300);
    line(300, 300, 350, 250);
    line(350, 250, 400, 300);
    circle(100, 100, 10);
    rectangle(100, 200, 250, 250);
    putpixel(100, 100, GREEN);
    gosc(400, 400, 150, 300, 50);
    ellipse(150, 150, 0, 350, 20, 40);
    getch();
    closegraph();
}
```

Output :-



Lab - 2
POORNIMA
Experiment - ①

Q To draw a straight line using DDA algorithm

```
#include < stdio.h >
#include < conio.h >
#include < graphics.h >
#include < stdlib.h >
void main()
{
    int gd = DETECT, gm, x1, x2, y1, y2, length, dx, dy;
    initgraph(&gd, &gm, "C:\\TURBOC2\\BGI");
    printf("enter the values of x1, x2, y1, y2 : \n");
    scanf("%d %d %d %d", &x1, &x2, &y1, &y2);
    if(abs(x2-x1) >= abs(y2-y1))
    {
        length = abs(x2-x1);
        dx = (x2-x1)/length;
        dy = (y2-y1)/length;
        x = x1 + 0.5;
        y = y1 + 0.5;
        i = 1;
        while(i <= length)
        {
            plot(x, y);
            x = x + dx;
            y = y + dy;
            i++;
        }
    }
}
```

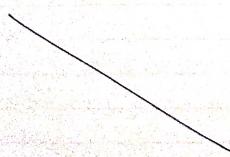
POORNIMA

```
while ( i <= length )  
{  
    putpixel ( x, y, 7 );  
    x = x + dx;  
    y = y + dy;  
    getch();  
    closegraph();  
}
```

25/9/23

O/P :-

Enter (x_1, y_1) and (x_2, y_2) coordinates :- 300 400, 500 600



POORNIMA

Lab - 3
Experiment -

Obj. \Rightarrow To draw straight line using Bresham's Algo.

```
#include < stdio.h >
#include < conio.h >
#include < graphics.h >
void main()
{
    int gd = DETECT, gm, x1, x2, y1, y2, dx, dy, p, Pk;
    initgraph(&gd, &gm, "C:\ITURBOC3\BGI");
    printf("Enter (x1,y1) and (x2,y2) coordinates :- ");
    scanf("%d %d %d %d", &x1, &y1, &x2, &y2);
    dx = x2 - x1;
    dy = y2 - y1;
    p = (2 * dy) - dx;
    Pk = p;
    while (dx >= 1)
    {
        if (Pk < 0)
            putpixel(++x1, y1, RED);
        Pk = Pk + (2 * dy);
        else
            putpixel(++x1, ++y1, RED);
        Pk = Pk + (2 * dy) - (2 * dx);
        dx--;
    }
    getch();
    closegraph();
}
```

16/10/23

Output :- Enter x and y : 100 200
Enter radius : 20



POORNIMA

Lab - 4

Experiment ①

Obj → Implementation of Mid Point Circle generating Algo.

```
#include < stdio.h>
#include < graphics.h>
#include < conio.h>
Void main ()
{
    int gd = DETECT, gm, x, y, dx, dy, p, xc, yc, r;
    initgraph ( &gd, &gm, "C:\TURBOC3\BCI" );
    printf ("Enter Xc and Yc \n");
    scanf ("%d %d", &xc, &yc);
    printf ("Enter Radius \n");
    scanf ("%d", &r);
    x = 0;
    y = r;
    p = 1 - r;
    putpixel (x + xc, y + yc, WHITE);
    putpixel (x + xc, -y + yc, WHITE);
    putpixel (-x + xc, y + yc, WHITE);
    putpixel (-x + xc, -y + yc, WHITE);
    putpixel (y + xc, x + yc, WHITE);
    putpixel (-y + xc, x + yc, WHITE);
    putpixel (-y + xc, -x + yc, WHITE);
    while (x < y)
    {
        if (p < 0)
            y--;
        else
            y--;
        x++;
        if (p < 0)
            //putpixel (x + , y, BLUE);
        p = p + (2 * r) + 1;
    }
}
```

POORNIMA

else if

```
//putpixel (x++, y--, BLUE);  
P = P + (2*x) + 1 - (2*y);  
y  
putpixel (x+xc, y+yc, WHITE);  
putpixel (x+xc, -y+yc, WHITE);  
putpixel (-x+xc, -y+yc, WHITE);  
putpixel (-x+xc, y+yc, WHITE);  
putpixel (y+xc, xc+yc, WHITE);  
putpixel (-y+xc, -xc+yc, WHITE);  
putpixel (-y+xc, xc+yc, WHITE);  
}  
efetch();
```

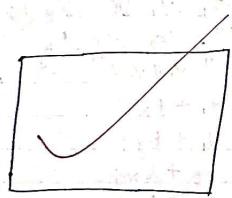
closegraph();

}

16/10/23

O/P :-

entered the coordinates x_1, y_1 , and $x_2, y_2 = 50 \ 100 \ 150 \ 200$
original figure is:-
entered the tx and ty = 2 3
after translation figure is:-



POORNIMA

Experiment No. (2)

// Scaling of Rectangle.

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

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

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

```
Void main()
```

```
{
```

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

```
int x, y, x1, y1, x2, y2, tx, ty, x3, y3, x4, y4;
```

```
scanf(
```

```
initgraph(&gd, &gm, "c:\\TURBO3\\BCI");
```

```
printf("enter the coordinates x1, y1, and x2, y2 :- ");
```

```
for scanf(" %d %d %d %d", &x1, &y1, &x2, &y2);
```

```
printf(" original figure is :- \n");
```

```
rectangle(x1, y1, x2, y2);
```

```
scanf(" %d %d ", &tx, &ty);
```

```
x3=x1+tx;
```

```
y3=y1+ty;
```

```
x4=x2+tx;
```

```
y4=y2+ty;
```

```
printf(" after translation figure is :- \n");
```

```
rectangle(x3, y3, x4, y4);
```

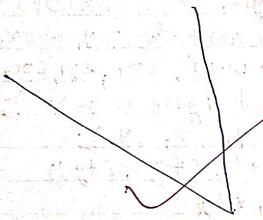
```
getch();
```

```
closegraph();
```

Q/ :-

Enter x_1 and y_1 coordinates:- 100 180
Enter x_2 and y_2 coordinates:- 200 250

line before rotation :-
Enter Angle = 60°
line after rotation :-



POORNIMA

Experiment No. ③

II Rotation of Single Line.

```
#include <stdio.h>
#include <graphics.h>
#include <stdlib.h>
#include <conio.h>
#include <math.h>
Void main()
{
    int gd=DETECT, gm, x1, x2, x3, x4, y1, y2, y3, y4, x, y, d1, d2, d3, d4;
    initgraph(&gd, &gm, "C:\TURBOC3\BGI");
    printf("Enter x1 and y1 coordinates :-");
    scanf("%d %d", &x1, &y1);
    printf("Enter x2 and y2 coordinates :-");
    scanf("%d %d", &x2, &y2);
    printf("line before rotation :-");
    line(x1, y1, x2, y2);
    printf("Enter Angle :-");
    scanf("%d", &d);
    x4 = x2 * cos((d * 3.14) / 180) - y2 * sin((d * 3.14) / 180);
    y4 = x2 * sin((d * 3.14) / 180) + y2 * cos((d * 3.14) / 180);
    printf("line after rotation :-");
    line(x1, y1, x4, y4);
    getch();
    closegraph();
```

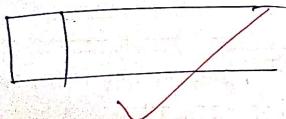
18/10/23

Q/P:-

enter the value of x_1, y_1, x_2 and y_2

50 100 150 200

about x axis 50



POORNIMA

Lab-6

Experiment No-①

11 Implementation of transformation due to shearing of rectangle.
→ At x-axis

```
#include <stdio.h>
#include <graphics.h>
Void main()
{
    int gd = DETECT, gm, n, x1, y1, x2, y2, shx;
    initgraph(&gd, &gm, "C:\TURBOC3\BGI");
    scanf(" %d %d %d %d", &x1, &y1, &x2, &y2);
    printf(" enter the value of x, ,y, ,x, ,and y, :");
    printf(" about x axis");
    Scanf(" %d", &shx);
    print(" before shearing");
    rectangle(n, , y1, , x2 + y2 * shx, y2);
    getch();
    closegraph();
}
```

O/P :-

enter coordinates:
 x_1, y_1, x_2, y_2
50 100 150 200

Enter Shearing factor
20



after Shearing



POORNIMA

at y-axis :-

```
#include < stdio.h >
#include < cgraphics.h >
void main()
{
    int gd = DETECT, gm, x1, x2, y1, y2, Sx;
    initgraph(&gd, &gm, "C:\TURBOC3\BGI");
    printf("Enter the x1, y1, x2, y2\n");
    scanf("%d %d %d %d", &x1, &y1, &x2, &y2);
    rectangle(x1, y1, x2, y2);
    printf("Enter Shearing factor\n");
    scanf("%f", &Sx);
    rectangle(x1, y1, x2, y2 * Sx);
    getch();
    closegraph();
}
```

Poorva
20/11/23

POORNIMA

```

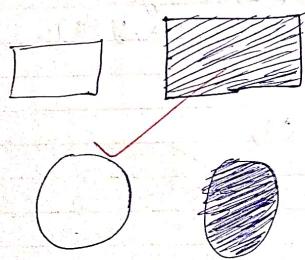
line(x1a,y1a,x1b,y1b);
line(x1b,y1b,x1c,y1c);
line(x1c,y1c,x1a,y1a);
delay(1000);
printf("Now enter the Scale factor for x and y : ");
scanf("%f %f", &x, &y);
x2a = x1a * 3x;
x2b = x1b * 3y;
y2a = y1a * 3y;
y2b = y1b * 3y;
y2c = y1c * 3y;
x2c = x1c * 3x;
printf("After Scaling");
line(x2a,y2a,x2b,y2b);
line(x2b,y2b,x2c,y2c);
line(x2c,y2c,x2a,y2a);
delay(1000);
printf("Enter the Rotational angle in degree : ");
scanf("%f", &ang);
theta = ang * (PI/100.0);
CosTheta = cos(theta);
SinTheta = sin(theta);
x1a = +x; y1a = +y;
x1b = +x; y1b = +y;
x1c = +x; y1c = +y;
x2a = x1a * CosTheta - y1a * SinTheta + +x;
y2a = x1a * SinTheta + y1a * CosTheta + +y;

```

POORNIMA

```
x2b = x1b * cosTheta - y1b * sinTheta + x;  
y2b = x1b * sinTheta + y1b * cosTheta + y;  
x2c = x1c * cosTheta - y1c * sinTheta + x;  
y2c = x1c * sinTheta + y1c * cosTheta + y;  
printf("After Rotation");  
line (x2a, y2a, x2b, y2b);  
line (x2b, y2b, x2c, y2c);  
line (x2c, y2c, x2a, y2a);  
delay(1000);  
getch();  
Closegraph();  
Y : ⑧  
11/12/23
```

Q/1-



POORNIMA

Lab - 8

Q) WAP. to fill any solid object using bgd fct and flood fill fct

```
#include <stdio.h>
#include <graphics.h>
Void main()
{
    int gd = DETECT, gm;
    initgraph(&gd, &gm, "C:\\TURBO C\\BCI");
    rectangle(100, 150, 300, 350);
    bar(50, 70, 130, 180);
    circle(350, 360, 200);
    floodfill(750, 300, RED);
    getch();
    closegraph();
}
```

- bar3d (left, right, top, bottom, depth, top, flag);

bar3d() fn draws a 3d rectangular bar

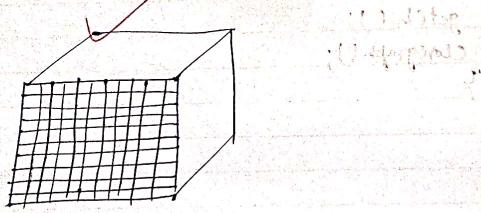
then fills it using the (white color) current fill

pattern and fill colour. The bars Depth in pixels

in pixels is given by Depth top flag.
- If top flag is non-zero value a top is put

on otherwise no top is put on the bar.

O/P:-



POORNIMA

```
#include <stdio.h>
#include <graphics.h>
void main()
{
  int gd = DETECT, gm;
  initgraph(&gd, &gm, "C:\TURBOC3\BGI");
  bar3d(120, 140, 340, 360, 50, 50);
  setfillstyle(HATCH_FILL, YELLOW);
  getch();
  Closegraph();
}
```

O/P :-

Enter translation factor

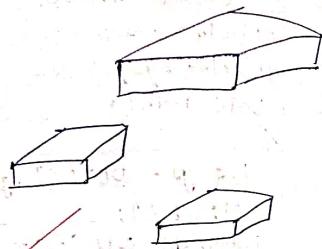
10 20

After translation:

Enter scaling factor:

2 2 2

After Scaling



POORNIMA

```
11 wap to implement 3D composite transformation.  
#include<conio.h>  
#include<stdio.h>  
int max x, max y, mid x, mid y  
Void axis()  
{  
    getch();  
    Cleargraph();  
    Line(mid x, 0, mid x, max y);  
    Line(0, mid y, max x, mid y);  
}  
Void main()  
{  
    int x, y, z, o, x1, x2, y1, y2  
    int gd = DETECT, gm;  
    detectgraph(gd, gm);  
    initgraph(gd, gm, "c:\\TURBOC3\\BG1");  
    max x = getmaxx();  
    max y = getmaxy();  
    mid x = max x / 2;  
    axis();  
    bar3d(mid x + 50, mid y - (y + 100), mid x + x + 60,  
          mid y - (y + 90), 10, 1);  
    printf("Enter translation factor");  
    scanf("%d %d", &x, &y);  
    printf("After translation:");
```

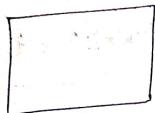
POORNIMA

```
b2d3d(midx+x+50, midy-(y+100), midx+x+50, midy-(y+50));
printf("Enter the scaling factor");
scanf("%f.%f.%f", &x, &y, &z);
printf("After Scaling");
b2d3d(midx+(x*50), midy-(y*100), midx+(x*60), midy-(y*60));
5*x, 1);
getch();
gclockgraph();
}
```

① 11/12/23

Q1:-

window :- Enter the no. of vertices of polygon



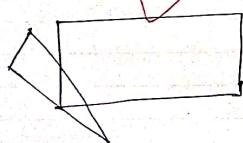
Enter the coordinates of points

(x_0, y_0) : 100 200

(x_1, y_1) : 300 400

(x_2, y_2) : 200 100

Press a button to Clip a polygon



This is Clipped Polygon



Lab-9

POORNIMA

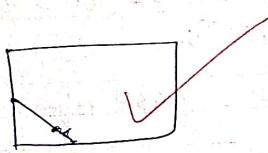
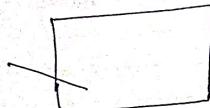
objective :- WAP in C to implement polygon clipping.

```
#include <stdio.h>
#include <graphics.h>
#include <conio.h>
#include <stdlib.h>
int main()
{
    int gd, gm, n, x, y, i, k = 0;
    int w[] = {220, 140, 420, 140, 420, 340, 220, 340, 220,
               140};
    detectgraph(&gd, &gm);
    initgraph(&gd, &gm, "C:\TURBOC3\BCI");
    printf("window = ");
    setcolor(RED);
    drawpoly(s, w);
    printf("Enter the no. of vertices of polygon : ");
    scanf("%d", &n);
    printf("Enter the coordinates of points : \n");
    k = 0;
    for (i = 0; i < n * 2; i += 2)
    {
        printf("(x%d, y%d) : ", k, k);
        scanf("%d %d", &x[i], &y[i]);
        k++;
    }
    x[n * 2] = x[0];
    x[n * 2 + 1] = y[1];
```

Poornima

```
Setcolor(WHITE);
drawpoly(n+1,x);
printf("\nPress a button to lip a polygon---");
getch();
Setcolor(RED);
drawpoly(s,w);
setfillstyle(SOLID_FILL, BLACK);
floodfill(2,2,RED);
gotoxy(1,1);
printf("\n This is clipped polygon---");
getch();
Cleardevice();
Closegraph();
return 0;
```

O/P:- Enter any key to continue



Poornima

Lab - 10

Objective :- Cohen Sutherland & Line Clipping and Line Line

```
#include <Studio.h>
#include <graphics.h>
#include <stdlib.h>
#include <conio.h>

Void main()
{
    int a[4], b[4];
    float m, x-new, y-new;
    float xl=100, yl=100, xh=300, yh=300, xa=10, ya=200,
        xb=150, yb=250;
    int gd = DETECT, gm, color;
    initgraph(&gd, &gm, "C:\TURBOC3\BGI");
    SetColor(5);
    line(xa, ya, xb, yb);
    SetColor(12);
    rectangle(xl, yl, xh, yh);
    m = (yb - ya) / (xb - xa);
    if (xa < xl)
        a[3] = 1;
    else
        a[3] = 0;
    if (xa > xh)
        a[2] = 1;
    else
        a[2] = 0;
    if (ya < yl)
        a[1] = 1;
    else
        a[1] = 0;
```

Poornima

```
if (ya > yb)
    a[0] = 1;
else a[0] = 0;
if (xb < xl)
    b[3] = 1;
else b[3] = 0;
if (xb > xr)
    b[2] = 0;
if (yb < yl)
    b[1] = 1;
else b[1] = 0;
if (yb > yr)
    b[0] = 1;
else b[0] = 0;
printf("Enter any key to Continue ....");
getch();
if (a[0] == 0 && a[1] == 0 && a[2] == 0 && a[3] == 0) {
    if (b[0] == 0 && b[1] == 0 && b[2] == 0 && b[3] == 0)
        printf(" No Clipping");
    line (xa, ya, xb, yb);
    y
} else if ( a[0] && b[0] || a[1] && b[1] || a[2] && b[2] || a[3] && b[3] )
    d
    clear();
    printf("Line Discarded");
    de (tangle (xl, yl, xr, yr));
} else d
```

POORNIMA

```
if (a[3] == 1 & b[3] == 0)
{
    y-new = (m * (xL - xA)) + ya;
    SetColor(12);
    rectangle(xL, yL, xh, yh);
    SetColor(0);
    line(xA, ya, xb, ys);
    SetColor(1);
    line(xL, y-new, xb, ys);
}
else if (a[1] == 1 & b[1] == 0)
{
    x-new = -xa + (yL - ya) / m;
    SetColor(0);
    line(xA, ya, xb, yb);
    SetColor(15);
    line(x-new, yh, xb, yb);
}
else if (a[0] == 1 & b[0] == 0)
{
    x-new = xa + (yh - ya) / m;
    SetColor(0);
    line(xA, ya, xb, yb);
    SetColor(15);
    line(x-new, yh, xb, yb);
}
getch();
closegraph();
}
```

~~Q111213~~

Output:

Enter translation vector

9

9

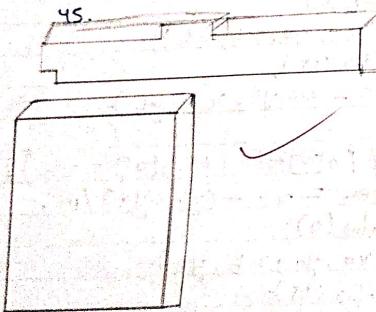
Enter scaling vector

2

1

Enter the angle

45.



POORNIMA

Experiment

#11

```
#include <stdio.h>
#include <Conio.h>
#include <graphics.h>
#include <math.h>
#include <stdlib.h>
void main()
{
    int gd = DETECT, gm, sx, sy, sz, tx, ty, zx, zy = 50,
        y1 = 100, x2 = 150, y2 = 150, z = 20, x3, y3, x4, y4
        double a, theta;
    int x5, y5, x6, y6;
    initgraph(gd, gm, "C:\TURBOC3\BGI");
    bar3d(x1, y1, x2, y2, x6, y6);
    initgraph(gd, gm,
    printf("Enter translation vector\n");
    scanf("%d %d", &tx, &ty);
    x1 = x1 + tx;
    y1 = y1 + ty;
    x2 = x2 + tx;
    y2 = y2 + ty;
    bar3d(x1, y1, x2, y2, z, 1)
    printf("Enter scaling vector\n");
    scanf("%d %d", &sx, &sy);
    x3 = x1 * sx;
    y3 = y1 * sy;
    x4 = x2 * sx;
```

POORNIMA

```
y4 = y2 * s y9  
bar8a (x3, y3, x4, y4, z, 1);  
printf ("enter the angle\n");  
scanf ("%f", &a);  
theta = (a * 3.14) / 1800;  
x5 = x3 * (cos(theta)) - y3 * (sin(theta));  
y5 = x3 * (sin(theta)) + y3 * (cos(theta));  
x6 = x4 * (cos(theta)) - y4 * (sin(theta));  
y6 = x4 * (sin(theta)) + y4 * (cos(theta));  
bar8d (x5, y5, x6, y6, z, 1);
```

```
} getch();  
closegraph();
```

```
#include <graphics.h>
#include <stdio.h>
#define MAXCOUNT 30
void fractal (float left, float top, float xside, float yside)
{
    float xscale, yscale, zx, zy, cx, tempx, cy;
    int x, y, i, j;
    int maxx, maxy, count;
    maxx = getmaxx();
    maxy = getmaxy();
    xscale = xside / maxx;
    yscale = yside / maxy;
    rectangle (0, 0, maxx, maxy);
    for (y = 1; y <= maxy - 1; y++)
    {
        for (x = 1; x <= maxx - 1; x++)
        {
            cx = x * xscale + left;
            cy = y * yscale + top;
            zx = 0;
            zy = 0;
            count = 0;
            while ((zx * zx + zy * zy >= 4) && (count < maxcount))
            {
                tempx = zx * zx - zy * zy + cx;
                zy = 2 * zx * zy + cy;
                zx = tempx;
                count++;
            }
        }
    }
}
```

POORNIMA

count = count + 1;

putpixel (x, y, count);

}

int main()

int gd = DETECT, gm;

float left, top, xside, yside;

left = -1.75;

top = -0.25;

xside = 0.25;

yside = 0.45;

Fillgraph (&gd, &gm, "");

fractrl (left, top, xside, yside);

getch();

Closegraph ();

return 0;

POORNIMA

Lab - (13)

Q

Aim: C program to draw 3D bar graph using graphics

#include <graphics.h>

#include <conio.h>

int main () {

initgraph (&gd, &gm, "C:\TC11\BGI");

settextstyle (BOLD FONT, HORIZ-DIR, 2);

outtextxy (275, 0, "3D BAR GRAPH");

setlinestyle (SOLID-LINE, 0, 2);

/* Print X and Y Axis */

line (90, 410, 90, 50);

line (90, 410, 590, 410);

line (25, 60, 40, 50);

line (25, 60, 90, 50);

line (585, 405, 590, 410);

line (585, 415, 590, 410);

outtextxy (65, 60, "Y");

outtextxy (570, 420, "X");

outtextxy (70, 415, "O");

POORNIMA

```
setfillstyle (XHATCH_FIII, RED);
bar3d (150, 80, 100, 410, 15, 1);
bar3d (225, 100, 275, 410, 15, 1);
bar3d (300, 120, 350, 410, 15, 1);
bar3d (375, 170, 425, 410, 15, 1);
bar3d (450, 135, 500, 410, 15, 1);
```

getch();
clsgraph();
rectangle;

①
13/10/23