

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement Digital Differential Analyzer Line Algorithm
=====*/

```

```

#include<graphics.h>
#include<stdio.h>
#include<conio.h>
#include<math.h>

voidmain()
{
    intgdriver =DETECT,gmode,steps,dx,dy,k,xa,ya,xb,yb;
    floatx_incr,y_incr,x,y;
    initgraph(&gdriver, &gmode, "C:\\\\TC\\\\BGI");

    printf("Enter the starting point (xa,ya) of the line\n");
    scanf("%d%d",&xa,&ya);
    printf("Enterthe ending point (xb,yb) of the line\n");
    scanf("%d%d",&xb,&yb);
    dx=xb-xa;
    dy=yb-ya;

    if(abs(dx)>abs(dy))
        steps=abs(dx);
    else
    {
        steps=abs(dy);
        x_incr=dx/steps;
        y_incr=dy/steps;
        x=xa;
        y=ya;
    }
    putpixel(x,y,White);
    for(k=0;k<steps;k++){
        x+=x_incr;
        y+=y_incr;
        putpixel(x,y,White);
    }
    getch();
}

```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement Bresenham Line Algorithm
=====*/

```

```

#include <graphics.h>
#include <stdio.h>
#include <conio.h>
#include <math.h>

void main()
{
    int gdriver=DETECT,gmode,xa,ya,xb,yb,dx,dy,x,y,xend,p;

    initgraph(&gdriver,&gmode,"C:\\TC\\BGI");

    printf("Enter the starting coordinates of the line \n");
    scanf("%d%d",&xa,&ya);
    printf("Enter the ending coordinates of the line \n");
    scanf("%d%d",&xb,&yb);

    dx=abs(xa-xb);
    dy=abs(ya-yb);
    p=2*dy-dx;

    if(xa>xb)
    {
        x=xb;
        y=yb;
        xend=xa;
    }
    else
    {
        x=xa;
        y=ya;
        xend=xb;
    }
    putpixel(x,y,1);

```

```

while(x<xend)
{
    x=x+1;
    if(p<0)
    {
        p=p+2*dy;
    }
    else
    {
        y=y+1;
        p=p+2*(dy-dx);
    }
    putpixel(x,y,1);
}
getch();
}

```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement Mid-Point Circle Algorithm
=====*/

```

```

#include<graphics.h>
#include<stdio.h>
#include<math.h>
#include<conio.h>

void plotpoints(int xcenter, int ycenter, int x, int y)
{
    putpixel(xcenter+x, ycenter+y, 15);
    putpixel(xcenter-x, ycenter+y, 14);
    putpixel(xcenter+x, ycenter-y, 13);
    putpixel(xcenter-x, ycenter-y, 12);
    putpixel(xcenter+y, ycenter+x, 11);
    putpixel(xcenter-y, ycenter+x, 10);
    putpixel(xcenter+y, ycenter-x, 9);
    putpixel(xcenter-y, ycenter-x, 8);
}

void main()
{
    int gdriver=DETECT, gmode, xcenter, ycenter, radius, p, x, y;
    initgraph(&gdriver, &gmode, "..\\BGI");

    printf("Enter the centre coordinates of the circle\n");
    scanf("%d%d", &xcenter, &ycenter);
    printf("Enter the radius of the circle\n");
    scanf("%d", &radius);

    x=0;
    y=radius;
    plotpoints(xcenter, ycenter, x, y);
    p=1-radius;

    while (x<y)
    {
        if(p<0)
        {
            x=x+1;

```

```
        p=p+2*x+1;
    }
    else
    {
        x=x+1;
        y=y-1;
        p=p+2*(x-y)+1;
    }

    plotpoints(xcenter,ycenter,x,y);
}
getch();
}
```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement Mid-Point Ellipse Algorithm
=====*/

```

```

#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<math.h>

void disp();

float x,y;
int xc,yc;
void main()
{
    int gdriver=DETECT, gmode;
    int a,b;
    float p1,p2;

    initgraph(&gdriver, &gmode, "C:\\\\TC\\\\bgi");
    clrscr();

    printf("Enter center");
    scanf("%d%d",&xc,&yc);
    printf("Enter radius x-axis,y-axis");
    scanf("%d%d",&a,&b);
    x=0;y=b;
    disp();
    p1=(b*b)-(a*a*b)+(a*a)/4;
    while((2.0*b*b*x)<=(2.0*a*a*y))
    {
        x++;
        if(p1<=0)
            p1=p1+(2.0*b*b*x)+(b*b);
        else
        {
            y--;
            p1=p1+(2.0*b*b*x)+(b*b)-(2.0*a*a*y);
        }
        disp();
        x=-x;
        disp();
    }
}

```

```

x=-x;
}
x=a;
y=0;
disp()

p2=(a*a)+2.0*(b*b*a)+(b*b)/4;
while((2.0*b*b*x)>(2.0*a*a*y))
{
    y++;
    if(p2>0)
        p2=p2+(a*a)-(2.0*a*a*y);
    else
    {
        x--;
        p2=p2+(2.0*b*b*x)-(2.0*a*a*y)+(a*a);
    }
    disp();
    y=-y;
    disp();
    y=-y;
}

getch();
closegraph();
}

voiddisp()
{
    putpixel(xc+x,yc+y,4);
    putpixel(xc-x,yc+y,4);
    putpixel(xc+x,yc-y,4);
    putpixel(xc-x,yc-y,4);
}

```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement Scan Line Polygon Fill Algorithm
=====*/

```

```

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

void main()
{
    int n,i,j,k,gd,gm,dy,dx;
    int x,y,temp;
    int a[20][2],xi[20];
    float slope[20];

    clrscr();
    printf("\n\n\tEnter the no. of edges of polygon : ");
    scanf("%d",&n);
    printf("\n\n\tEnter the coordinates of polygon : \n\n\n");

    for(i=0;i<n;i++)
    {
        printf("\tX%dY%d :",i,i);
        scanf("%d %d",&a[i][0],&a[i][1]);
    }

    a[n][0]=a[0][0];
    a[n][1]=a[0][1];

    detectgraph(&gd,&gm);
    initgraph(&gd,&gm,"c:\\tc\\bgi");

    /*- draw polygon -*/

    for(i=0;i<n;i++)
    {
        line(a[i][0],a[i][1],a[i+1][0],a[i+1][1]);
    }
    getch();
    for(i=0;i<n;i++)
    {

```



```

        dy=a[i+1][1]-a[i][1];
        dx=a[i+1][0]-a[i][0];
    if(dy==0) slope[i]=1.0;
    if(dx==0) slope[i]=0.0;
        if((dy!=0)&&(dx!=0))
            {
                slope[i]=(float)dx/dy;
            }
    }

    for(y=0;y<480;y++)
    {
        k=0;
        for(i=0;i<n;i++)
        {
            if( ((a[i][1]<=y)&&(a[i+1][1]>y)) || ((a[i][1]>y)&&(a[i+1][1]<=y)))
            {
                xi[k]=(int)(a[i][0]+slope[i]*(y-a[i][1]));
                k++;
            }
        }
        for(j=0;j<k-1;j++)
        {
            for(i=0;i<k-1;i++)
            {
                if(xi[i]>xi[i+1]){
                    temp=xi[i];
                    xi[i]=xi[i+1];
                    xi[i+1]=temp;
                }
            }
        }
        setcolor(35);

        for(i=0;i<k;i+=2)
        {
            line(xi[i],y,xi[i+1]+1,y);
        }

    getch();
}

```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement Ellipse with flood fill Algorithm
=====*/

```

```

#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<dos.h>
#include<snap.h>
void drawellipse(){
    float x,y;
    int xc,yc;
    int a,b;
    float p1,p2;
    printf("enter center");
    scanf("%d%d",&xc,&yc);
    printf("enter radius x-axis,y-axis");
    scanf("%d%d",&a,&b);
    x=0;y=b;
    disp();
    p1=(b*b)-(a*a*b)+(a*a)/4;
    while((2.0*b*b*x)<=(2.0*a*a*y)){
        x++;
    if(p1<=0)
        p1=p1+(2.0*b*b*x)+(b*b);
    else{
        y--;
        p1=p1+(2.0*b*b*x)+(b*b)-(2.0*a*a*y);
    }
    disp();
    x=-x;
    disp();
    x=-x;
    }
    x=a;
    y=0;
    disp();
    p2=(a*a)+2.0*(b*b*a)+(b*b)/4;
    while((2.0*b*b*x)>(2.0*a*a*y)){
        y++;
    if(p2>0)
        p2=p2+(a*a)-(2.0*a*a*y);

```

```

else{
    x--;
    p2=p2+(2.0*b*b*x)-(2.0*a*a*y)+(a*a);
}
disp();
y=-y;
disp();
y=-y;
}
}
void disp(){
    putpixel(xc+x,yc+y,4);
    putpixel(xc-x,yc+y,4);
    putpixel(xc+x,yc-y,4);
    putpixel(xc-x,yc-y,4);
}
void floodfill4(intx,inty,intfillcolor,intoldcolor){
    if(getpixel(x,y)==oldcolor){
        putpixel(x,y,fillcolor);
        floodfill4(x+1,y,fillcolor,oldcolor);
        floodfill4(x-1,y,fillcolor,oldcolor);
        floodfill4(x,y+1,fillcolor,oldcolor);
        floodfill4(x,y-1,fillcolor,oldcolor);
    }
}

void boundryfill4(intx,inty,intfillcolor,intboundrycolor){
    int current;
    current=getpixel(x,y);
    if((current!=boundrycolor)&&(current!=fillcolor)){
        putpixel(x,y,fillcolor);
        boundryfill4(x+1,y,fillcolor,boundrycolor);
        boundryfill4(x-1,y,fillcolor,boundrycolor);
        boundryfill4(x,y+1,fillcolor,boundrycolor);
        boundryfill4(x,y-1,fillcolor,boundrycolor);
    }
}

void main(){
    intgdriver=DETECT,gmode;
    inta,b;
    initgraph(&gdriver,&gmode,"C:\\TC\\BGI");
    circle(xc,yc,radius);
    printf("\nBoundry fill:");
    printf("\nenter interior points (a,b):");
    scanf("%d%d",&a,&b);
    boundryfill4(a,b,BLUE,WHITE);
    floodfill4(a,b,RED,BLUE);
    getch();
}

```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to implement flood fill Algorithm
=====*/

```

```

#include<stdio.h>
#include<conio.h>
#include<graphics.h>
#include<dos.h>
#include<snap.h>
void drawcircle(){
    int i,r,x,y,xc,yc;
    float d;
    printf("Enter Radius\n");
    scanf("%d",&r);
    printf("Enter Center of circle\n");
    scanf("%d",&xc);
    scanf("%d",&yc);
    d=1.25-r;
    x=0;
    y=r;
    do{
        if(d<0){
            x=x+1;
            d=d+2*x+1;
        }
        else{
            x=x+1;
            y=y-1;
            d=d+2*x-2*y+10;
        }
        putpixel(xc+x,yc+y,5);
        putpixel(xc-y,yc-x,5);
        putpixel(xc+y,yc-x,5);
        putpixel(xc-y,yc+x,5);
        putpixel(xc+y,yc+x,5);
        putpixel(xc-x,yc-y,5);
        putpixel(xc+x,yc-y,5);
        putpixel(xc-x,yc+y,5);
    }
    while(x<y);
}

void boundryfill4(intx,inty,intfillcolor,intboundrycolor){
    int current;

```

```

        current=getpixel(x,y);
        if((current!=boundrycolor)&&(current!=fillcolor)){
            putpixel(x,y,fillcolor);
            boundryfill4(x+1,y,fillcolor,boundrycolor);
            boundryfill4(x-1,y,fillcolor,boundrycolor);
            boundryfill4(x,y+1,fillcolor,boundrycolor);
            boundryfill4(x,y-1,fillcolor,boundrycolor);
        }
    }
}

void main()
{
    intgdriver=DETECT,gmode;
    inta,b;
    initgraph(&gdriver,&gmode,"C:\\TC\\BGI");
    drawcircle(xc,yc,radius);
    printf("\nBoundry fill:");
    printf("\nenter interior points (a,b):");
    scanf("%d%d",&a,&b);
    boundryfill4(a,b,BLUE,5);
    getch();
}

```

```

/*=====
    Name      Teja Ram
    Roll No   13EEBCS861
    Batch     B2
    Branch    C.S.E (VI Sem)
    Write a Program to draw a rectangle with diagonal
=====*/

```

```

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

voidddaline(int x1,int y1,int x2,int y2)
{
    floatx,y,dx,dy,steps,k,xincrement,yincrement;
    dx=x2-x1;dy=y2-y1;
    x=x1;y=y1;
    if(abs(dx)>abs(dy))
        steps=dx;
    else
    {
        steps=dy;
        xincrement=dx/steps;
        yincrement=dy/steps;
    }
    putpixel(x,y,1);
    for(k=0;k<steps;k++)
    {
        x+=xincrement;
        y+=yincrement;
        putpixel(x,y,WHITE);
    }

}

void main()
{
    intgdriver = DETECT,gmode;
    int x1,x2,x3,x4,y1,y2,y3,y4;
    initgraph(&gdriver,&gmode,"C:\\TC\\BGI");
    printf("\nenter points in clockwise direction:");
    printf("\nenter points");

```

```
printf("\nenter (x1,x2,x3,x4):");
scanf("%d%d%d%d",&x1,&x2,&x3,&x4);
printf("\nenter (y1,y2,y3,y4):");
scanf("%d%d%d%d",&y1,&y2,&y3,&y4);

ddaline(x1,y1,x2,y2);
ddaline(x2,y2,x3,y3);
ddaline(x3,y3,x4,y4);
ddaline(x1,y1,x4,y4);
ddaline(x1,y1,x3,y3);
getch();
}
```