# SAGARMATHA ENGINEERING COLLEGE

*(TU Affiliated)*

**Sanepa, Lalitpur**

LAB NO: 4

A LAB REPORT ON

MID POINT ELLIPSE ALGORITHM


**Submitted By:**

**Name:…….**

**Faculty/Year:……**

**Roll No:……..**

**Date:……**

**Submitted To:**

**Department of electronics and Computer Engineering**

**Signature:……..**

**Date:……**

# COMPUTER GRAPHICS LAB-04

## TITLE

MID POINT ELLIPSE ALGORITHM

## OBJECTIVES

- ✓ To be familiar with fundamental knowledge of Mid-point ellipse drawing algorithm and its implementation.

## HARDWARE/SOFTWARE REQUIRED

- ✓ C Compiler

## RELATED THEROY

### Mid-point Ellipse Algorithm

An ellipse is defined as the set of points such that the sum of the distances from two fixed point/ positions (foci) is same for all points.

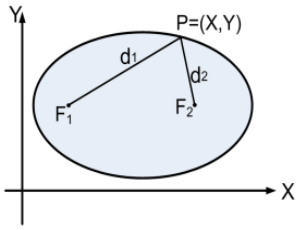
**Elongated circle**

**Equation of ellipse:**

$$d_1 + d_2 = constant$$

**F1→($x_1$,$y_1$), F2→($x_2$,$y_2$)**

$$\sqrt{(x-x_1)^2+(y-y_1)^2}+\sqrt{(x-x_2)^2+(y-y_2)^2} = constant$$

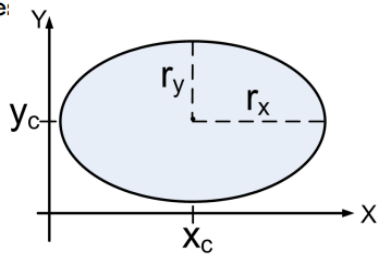



**General Equation**

$$Ax^2 + By^2 + Cxy + Dx + Ey + F = 0$$

**In terms of ellipse center coordinate:**

$$\left(\frac{x-x_c}{r_x}\right)^2 + \left(\frac{y-y_c}{r_y}\right)^2 = 1$$




**In polar co-ordinate**

$$x = x_c + r_x \cos\theta$$
$$y = y_c + r_y \sin\theta$$

$$f_{ellipse}(x,y) = r_y^2 x^2 + r_x^2 y^2 - r_x^2 r_y^2$$

$$f_{ellipse}(x,y)\begin{cases} <0, & if\ (x,y)\ is\ inside\ the\ ellipse\ boundary \\ =0, & if\ (x,y)\ is\ on\ the\ ellipse\ boundary \\ >0, & if\ (x,y)\ is\ outside\ the\ ellipse\ boundary \end{cases}$$

# COMPUTER GRAPHICS LAB-04

## ALGORITHM

1. Input $r_x$,$r_y$, and the ellipse center ($x_c$, $y_c$) and obtain the first point on an ellipse centered on the origin as

$$(x_0,y_0) = (0,r_y)$$

2. Calculate the initial value of the decision parameter in region 1 as

$$p1_0 = r_y^2 - r_x^2 r_y + \frac{1}{4} r_x^2$$

3. At each $x_k$ position in region 1, starting at k = 0, perform the following test:

If $p1_k < 0$ /* next point $(x_{k+1},y_k)$

$$x_{k+1} = x_k+1,\ y_{k+1} = y_k$$

$$p1_{k+1} = p1_k + 2r_y^2 x_{k+1} + r_y^2$$

Else /* next point $(x_k+1,y_k-1)$ */

$$x_{k+1} = x_k+1,\ y_{k+1} = y_k -1$$

$$p1_{k+1} = p1_k + 2r_y^2 x_{k+1} - 2r_x^2 y_{k+1} + r_y^2$$

With

$$2r_y^2 x_{k+1} = 2r_y^2 x_k + 2r_y^2, \qquad 2r_x^2 y_{k+1} = 2r_x^2 y_k - 2r_x^2$$

and continue until $2r_y^2 x \geq 2r_x^2 y$

4. Calculate the initial value of decision parameter in region 2 using the last point $(x_0,y_0)$ calculated in region 1 as

$$p2_0 = r_y^2\left(x_0+\frac{1}{2}\right)^2 + r_x^2(y_0-1)^2 - r_x^2 r_y^2$$

5. At each $y_k$ position in region 2, starting at k = 0, perform the following test:

If $p2k \leq 0$ /* next point $(x_k+1,y_k-1)$ */

$x_{k+1} = x_k+1$

$y_{k+1} = y_k-1$

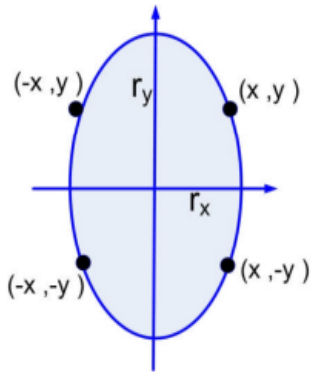
$$p2_{k+1} = p2_k + 2r_y^2 x_{k+1} - 2r_x^2 y_{k+1} + r_x^2$$

Else /*next point $(x_k,y_k-1)$*/

$x_{k+1} = x_k$

$y_{k+1} = y_k-1$

$$p2_{k+1} = p2_k - 2r_x^2 y_{k+1} + r_x^2$$




4. Using the same incremental calculations for x and y as in region 1. continue until y=0.
6. Determine the symmetry points in the other three quadrants.
7. Move each calculated pixel position (x,y) onto the elliptical path centered on $(x_c,y_c)$ and plot the co-ordinate values:

$$x = x + x_c,\ y = y + y_c$$

# COMPUTER GRAPHICS LAB-04

## IMPLEMENTATION OF MID-POINT ELLIPSE ALGORITHM

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

//Function to plot the ellipse
void pixel (int xc, int yc, int x,int y)
{
    putpixel(320+(xc+x),240-(yc+y),14);
    putpixel(320+(xc+x),240-(yc-y),14);
    putpixel(320+(xc-x),240-(yc-y),14);
    putpixel(320+(xc-x),240-(yc+y),14);
}

int main()
{
    int xc, yc, x, y, p, rx, ry;

    int gdriver=DETECT, gmode; //Detects the graphics drivers automatically
    initgraph(&gdriver, &gmode,(char*)""); //Initialize to graphics mode

    printf("Enter the co-ordinates of the center of the ellipse : ");
    scanf("%d %d",&xc,&yc);
    printf("Enter the value of rx and ry of the ellipse : ");
    scanf("%d %d",&rx,&ry);


//Take start position as(0,ry)
x=0;
y=ry;

p=(ry*ry)-(rx*rx*ry)+((rx*rx)/4); //Finding decision parameter p in region 1

pixel (xc, yc, x, y);

while((2*x*ry*ry)<(2*y*rx*rx)) //At each x position in region 1
{
    if(p<0) //If decision parameter is less than 0
    {
        x++; //Increment x
        p=p+(2*ry*ry*x)+(ry*ry); //Calculate the new decision parameter
    }
    else //If decision parameter is greater than 0
    {
        x++; //Increment x
        y--; //Decrement y
        p=p+(2*ry*ry*x)+(ry*ry)-(2*rx*rx*y); //Calculate the new decision parameter
    }
    pixel (xc, yc, x, y);
}
```

# COMPUTER GRAPHICS LAB-04

```
    p=ry*ry*pow((float)x+0.5,2) + rx*rx*pow(y-1,2)-rx*rx*ry*ry; //Finding decision parame
    pixel (xc, yc, x, y);

    while(y>=0) //At each y position in region 2
    {
        if(p>0) //If decision parameter is greater than 0
        {
            y--; //Decrement y
            p=p-(2*y*rx*rx)+(rx*rx); //Calculate the new decision parameter
        }
        else //If decision parameter is less than 0
        {
            y--; //Decrement y
            x++; //Increment x
            p=p+(2*ry*ry*x)-(2*y*rx*rx)+(rx+rx); //Calculate the new decision parameter
        }
        pixel (xc, yc, x, y);
    }

    getch(); //Pauses the Output Console until a key is pressed
    closegraph(); //Closes the graphics mode
    return 0;
}
```

## OUTPUT

## RESULT:

Program is compiled, Mid-point Ellipse algorithm implementation was done.