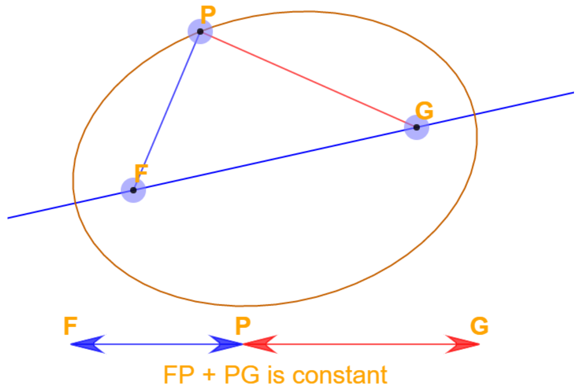
ELLIPSE-GENERATING ALGORITHMS

An ellipse is an elongated circle. Therefore, elliptical ***curves*** can be generated by modifying circle-drawing procedures to take into account the different dimensions of an ellipse along the major and minor axes.

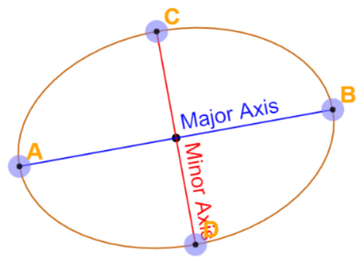


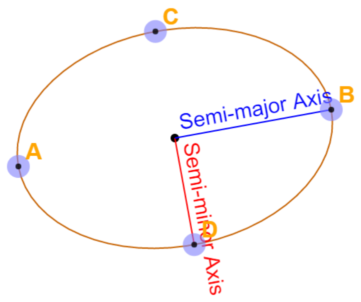
An ellipse is the **set of all points** on a plane whose distance from two fixed points F and G add up to a constant.

Major and Minor Axes

The **Major Axis** is the longest diameter. It goes from one side of the ellipse, through the center, to the other side, at the widest part of the ellipse. And the **Minor Axis** is the shortest diameter (at the narrowest part of the ellipse).

The **Semi-major Axis** is half of the Major Axis, and the **Semi-minor Axis** is half of the Minor Axis.

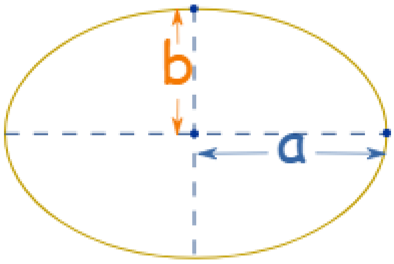




Eccentricity

The eccentricity is a measure of how "un-round" the ellipse is. The formula (using semi-major and semi-minor axis) is:

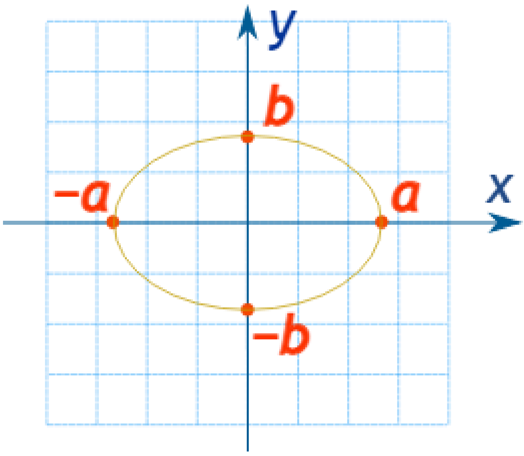
$$\frac{\sqrt{(a^2 - b^2)}}{a}$$



Equation

By placing an ellipse on an x-y graph (with its major axis on the x-axis and minor axis on the y-axis), the equation of the curve is:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$



Or we can use "parametric equations", where we have another variable "t" and we calculate x and y from it, like this:

$x = a\cos(t)$

$y = b\sin(t)$

Midpoint Ellipse Algorithm

Our approach her is similar *to* that used in displaying a raster circle. Given parameters r_x, r_y and (x_c, y_c), we determine points (***x***, ***y***) for an ellipse in standard position centered on the origin, and then we shift the points so the ellipse is centered at (x_c, y_c).

The midpoint ellipse method is applied throughout the first quadrant in two parts. Figure ***b*** shows the division of the first quadrant according to the slope of an ellipse with ***rx*** < ***ry*.** We process this quadrant by taking unit steps in the ***x*** direction where the slope of the curve has a magnitude less than 1, and taking unit steps in the ***y*** direction where the slop has a magnitude greater than 1.

Regions 1 and 2 (Fig. ***b*)**, can be processed in various ways. We can start at position (***0***, ***ry***) and step clockwise along the elliptical path in the first quadrant, shifting from unit steps in ***x*** to unit steps in ***y*** when the slope becomes < **-1**.

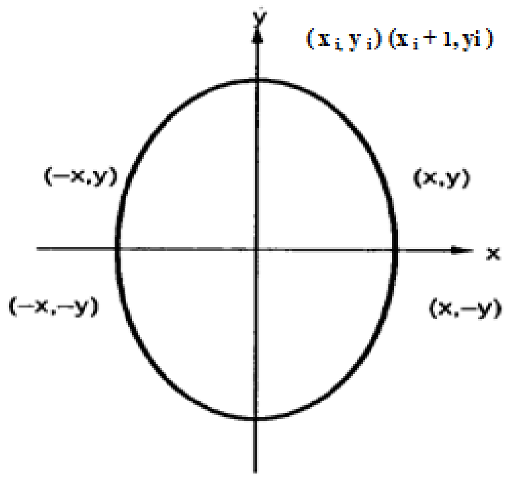
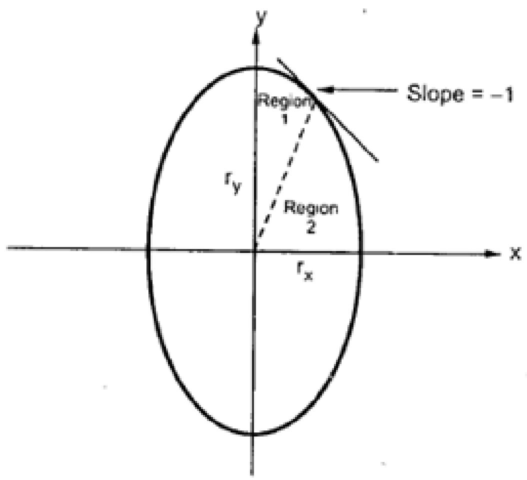


Fig (a). Four way symmetry of Ellipse



Fig(b). Ellipse processing regions

Alternatively, we could start at (***rx***, ***0***) and select points in a counter clockwise order, shifting from unit steps in ***y*** to unit steps in ***x*** when the slope becomes greater than **-1.** With parallel processors, we could calculate pixel positions in the two regions simultaneously. As an example of a sequential implementation of the midpoint algorithm, we take the start position at ***(0, ry***) and step along the ellipse path in clockwise order throughout the first quadrant.

Function of ellipse:

$f_{ellipse}(x, y) = r_y^2x^2 + r_x^2y^2 - r_x^2r_y^2$
$f_{ellipse}(x, y) < 0$ then (x, y) is inside the ellipse.
$f_{ellipse}(x, y) > 0$ then (x, y) is outside the ellipse.
$f_{ellipse}(x, y) = 0$ then (x, y) is on the ellipse.

Decision parameter:

Initially, we have two decision parameters $p1_0$ in region **1** and $p2_0$ in region ***2***. These parameters are defined as:
$p1_0$ in region ***1*** is given as: $p1_0 = r_y^2 + (1/4)r_x^2 - r_x^2r_y$
$p2_0$ in region ***2*** is given as: $p2_0 = r_y^2(x_0 + 1/2)^2 + r_x^2\ (y_0 - 1)^2 - r_x^2r_y^2$

Mid-Point Ellipse Algorithm:

1. Take input radius along ***x*** axis and ***y*** axis and obtain center of ellipse.
2. Initially, we assume ellipse to be centered at origin and the first point as: $(x, y_0) = (0, r_y)$.
3. Obtain the initial decision parameter for region ***1*** as: $p1_0 = r_y^2 + (1/4)r_x^2 - r_x^{\ 2}r_y$
4. For every x_k position in region ***1***:
   If $p1_k < 0$ then
      the next point along the is (x_{k+1}, y_k) and $p1_{k+1} = p1_k + 2r_y^2x_{k+1} + r_y^2$
   Else,
      the next point is (x_{k+1}, y_{k-1}), and $p1_{k+1} = p1_k + 2r_y^2x_{k+1} - 2r_x^2y_{k+1} + r_y^2$
5. Obtain the initial value in region ***2*** using the last point (x_0, y_0) of region ***1*** as: $p2_0 = r_y^2(x_0 + 1/2)^2 + r_x^2\ (y_0 - 1)^2 - r_x^2r_y^2$
6. At each y_k in region ***2*** starting at $k = 0$ perform the following task.
   If $p2_k > 0$ then
      The next point is (x_k, y_{k-1}) and $p2_{k+1} = p2_k - 2r_x^2y_{k+1} + r_x^2$
   Else,
      the next point is (x_{k+1}, y_{k-1}) and $p2_{k+1} = p2_k + 2r_y^2x_{k+1} - 2r_x^2y_{k+1} + r_x^2$
7. Now obtain the symmetric points in the three quadrants and plot the coordinate value as: $x = x + xc$, $y = y + yc$
8. Repeat the steps for region ***1*** (step 4), until $2r_y^2x >= 2r_x^2y$