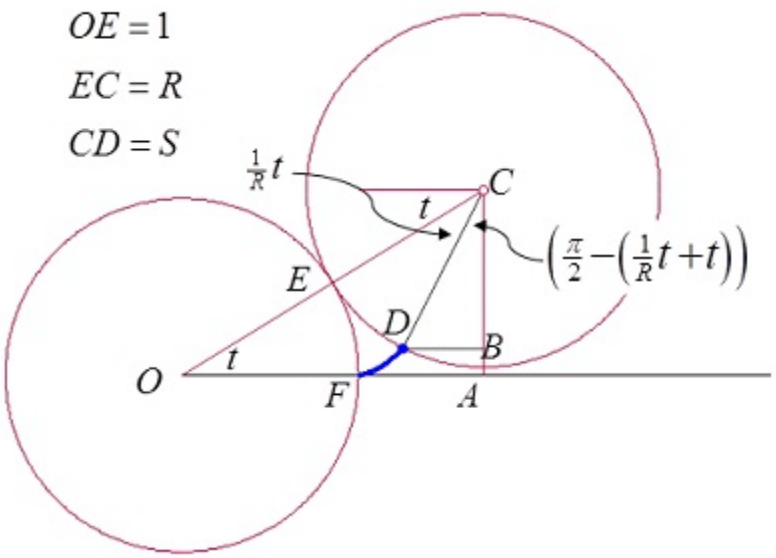
**Notes on the Roulette Generator**

As mentioned in the posts this type of program will generate many kinds of roulettes – curves traced by a point on a moving circle.

The full Winplot Inventory is below. I'll explain this line-by-line in hope that you can adapt it to whatever program you are using.




1. **(x,y) = (cos(t),sin(t)); 0.0 <= t <= 2pi**

   The parametric equations of the fixed circle with center at the origin, *O*. You will not need to change the radius of this circle.

2. **(x,y) = (((1+R)cos(A)+Rcos(T)),((1+R)sin(A)+Rsin(t))); 0.0 <= t <= 2pi**

   The parametric equations of the moving circle of radius R. Set up a slider for A and R. The "A" slider is the one that animates the circle and makes it roll around the fixed circle. The "R" slider allows you to change the radius of the moving circle.

3. **(x,y) = ((1+R)cos(A),(1+R)sin(A)).**

   The point at the center of the moving circle. The "A" and "R" sliders are as above in 2.

4. **(x,y) = (((1+R)cos(t)-S*cos((1/R)t+t))*chi(0,A,t),((1+R)sin(t)-S*sin((1/R)t+t))*chi(0,A,t)); 0.0 <= t <= 30pi.**

   These are the parametric equations of the path the point follows. The **chi(0,A,t)** multiplied by everything else is Winplot's "Characteristic function." It makes the path grow as A's value increases, instead of being graphed all at once. The way to make this happen may be different in other programs. The S is the distance from the center of the moving circle to the moving point *D*. This starts equal to R, but in later posts changes. Set up an S slider.

   The range from 0 to 30pi is to allow the circle to go around many times. You may to adjust the 30pi.

5. **segment ((1+R)cos(A),(1+R)sin(A))--(((1+R)cos(A)-S*cos((1/R)*A+A)),((1+R)sin(A)-S*sin((1/R)*A+A)))**

   The segment from *C* to the moving point to *D*.

6. **segment (0,0)--((1+R)cos(A),(1+R)sin(A))**

   The segment between the centers of the two circles

7. **(x,y) = (((1+R)cos(A)-S*cos((1/R)*A+A)),((1+R)sin(A)-S*sin((1/R)*A+A)))**

   The moving point *D*. The path is the locus of this point.

8. **(x,y) = (((1+R)cos(t)-S*cos((1/R)t+t))*chi(0,A,t),((1+R)sin(t)-S*sin((1/R)t+t))*chi(0,A,t)); 0.0 <= t <= 2pi**

   The first revolution graphed in Orange. Shows where the first revolution of the moving circle ends when there is more than one revolution.

9. **(x,y) = ((-(1+R)sin(t)+S(1/R+1)sin(t/R+t))*chi(0,A,t),((1+R)cos(t)-S(1/R+1)cos(t/R+t))*chi(0,A,t)); 0.0 <= t <= 2pi**

   The path of the parametric derivative x = dx/dy and y = dy/dt

10. **segment (0,0)-->((-(1+R)sin(A)+S(1/R+1)sin(A/R+A)),((1+R)cos(A)-S(1/R+1)cos(A/R+A)))**

    The velocity vector plotted as a position vector.

11. **segment (((1+R)cos(A)-S*cos((1/R)A+A)),((1+R)sin(A)-S*sin((1/R)A+A)))-->((-(1+R)sin(A)+S(1/R+1)sin(A/R+A))+((1+R)cos(A)-S*cos((1/R)A+A)),((1+R)cos(A)-S(1/R+1)cos(A/R+A))+((1+R)sin(A)-S*sin((1/R)A+A)))**

    The velocity vector with its tail on the moving point shows the direction of motion and its length is the speed.