```
> restart;
```

x(t) = #prey (rabbits)
y(t) = #predators (wolves)

```
> de1 := diff( x(t),t ) = alpha1*x(t) - beta1*x(t)*y(t);
```

$$de1 := \frac{\mathrm{d}}{\mathrm{d}t}\, x(t) = \alpha 1\, x(t) - \beta 1\, x(t)\, y(t)$$

```
> de2 := diff( y(t),t) = +beta2*x(t)*y(t) - alpha2*y(t);
```

$$de2 := \frac{\mathrm{d}}{\mathrm{d}t}\, y(t) = \beta 2\, x(t)\, y(t) - \alpha 2\, y(t)$$

```
> popequil := { rhs( de1)=0, rhs(de2)=0 } ;
```

$$popequil := \{\alpha 1\, x(t) - \beta 1\, x(t)\, y(t) = 0,\ \beta 2\, x(t)\, y(t) - \alpha 2\, y(t) = 0\}$$

```
> solve( popequil, {x(t),y(t)} );
```

$$\{x(t) = 0,\ y(t) = 0\},\ \left\{x(t) = \frac{\alpha 2}{\beta 2},\ y(t) = \frac{\alpha 1}{\beta 1}\right\}$$

```
> alpha1 := 0.1; beta1 := 0.1; beta2 := 0.02; alpha2 := 0.05;
```

$$\alpha 1 := 0.1$$

$$\beta 1 := 0.1$$

$$\beta 2 := 0.02$$

$$\alpha 2 := 0.05$$

Here's how we get a numerical approximation to the solutions for x(t) and y(t).

```
> F := dsolve( {de1,de2,x(0)=1,y(0)=0.2}, {x(t),y(t)}, numeric );
```

$$F := \mathbf{proc}(x\_rkf45)\ \ldots\ \mathbf{end\ proc}$$

```
> F(0);
```

$$[t = 0.,\ x(t) = 1.,\ y(t) = 0.200000000000000]$$

```
> F(1);
```

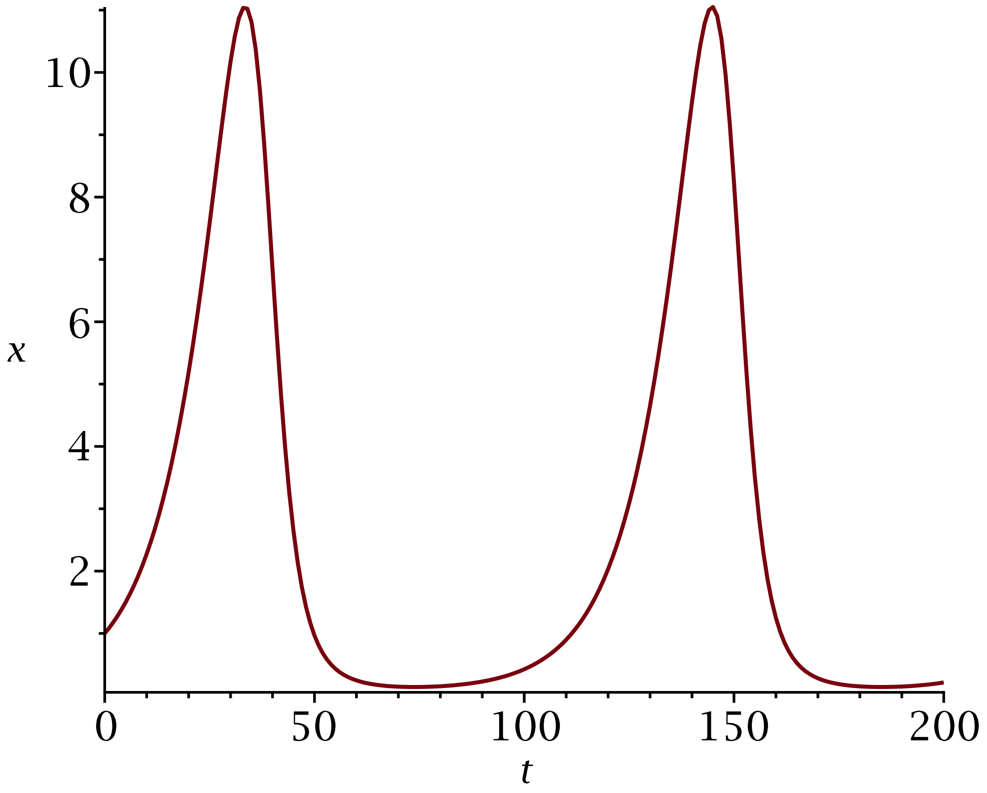
$$[t = 1.,\ x(t) = 1.08360311769698,\ y(t) = 0.194249073977779]$$

```
> F(2);
```

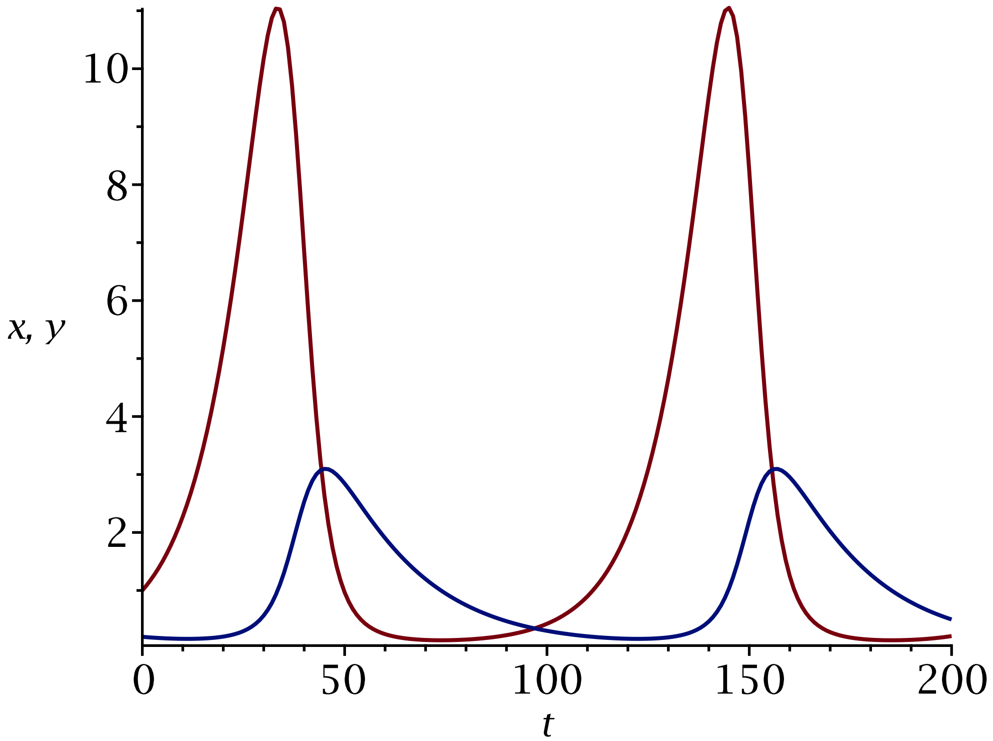
$$[t = 2.,\ x(t) = 1.17484203973331,\ y(t) = 0.188993456628175]$$

To visualize this, the odeplot in the plots package. There are several options.

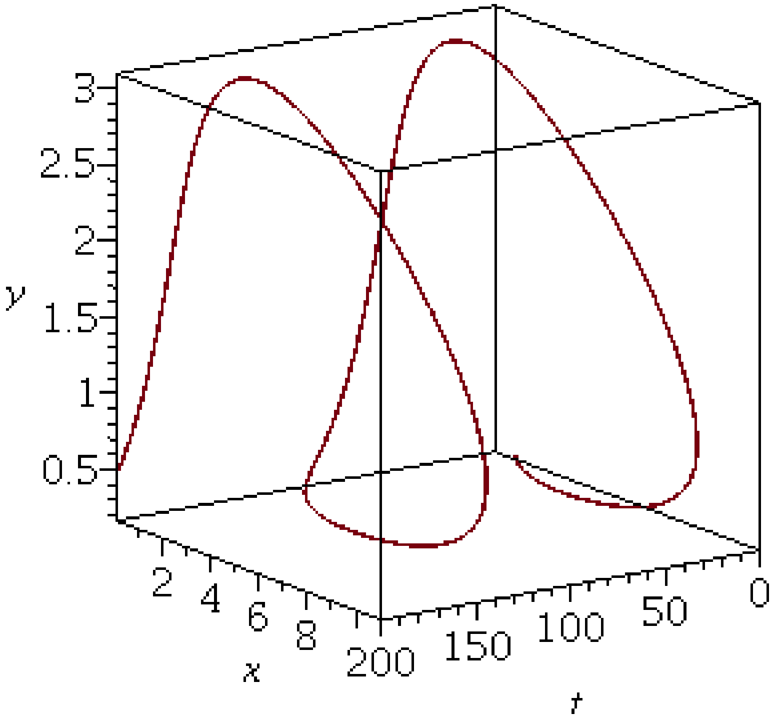
```
> with(plots):
> odeplot( F, [t,x(t)], t=0..200 );
```



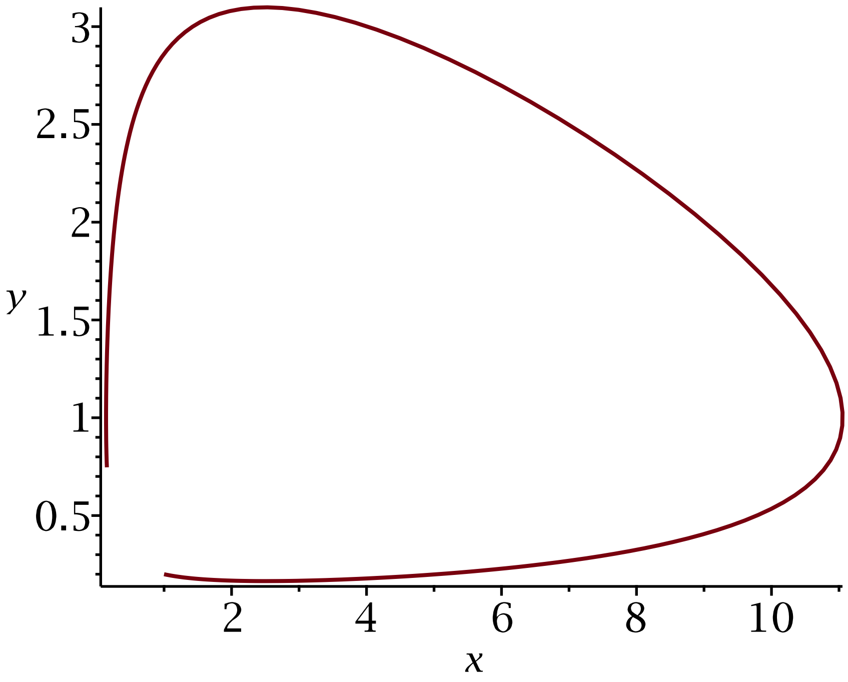
```
> odeplot( F, [[t,x(t)],[t,y(t)]], t=0..200 );
```



```
> odeplot( F, [[t,x(t),y(t)]], t=0..200 );
```



```
> odeplot( F, [[x(t),y(t)]], t=0..80 );
```

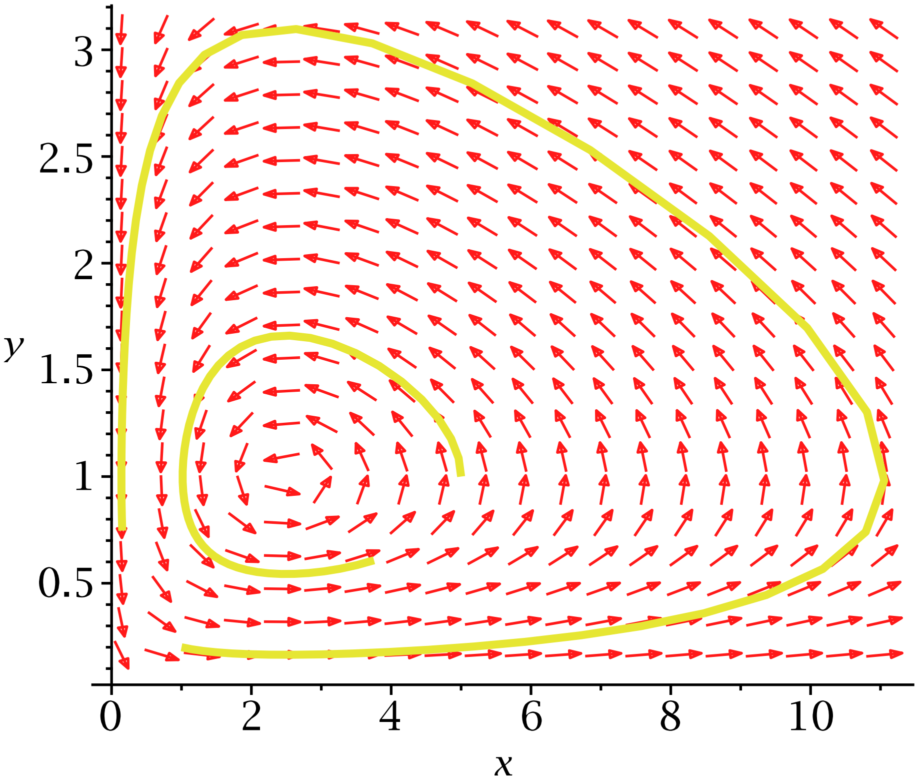


The above representation, a plot of x verses y, is what DEplot shows.

```
> with(DEtools):
> initvals := [[x(0)=1,y(0)=0.2], [x(0)=5,y(0)=1]];
```

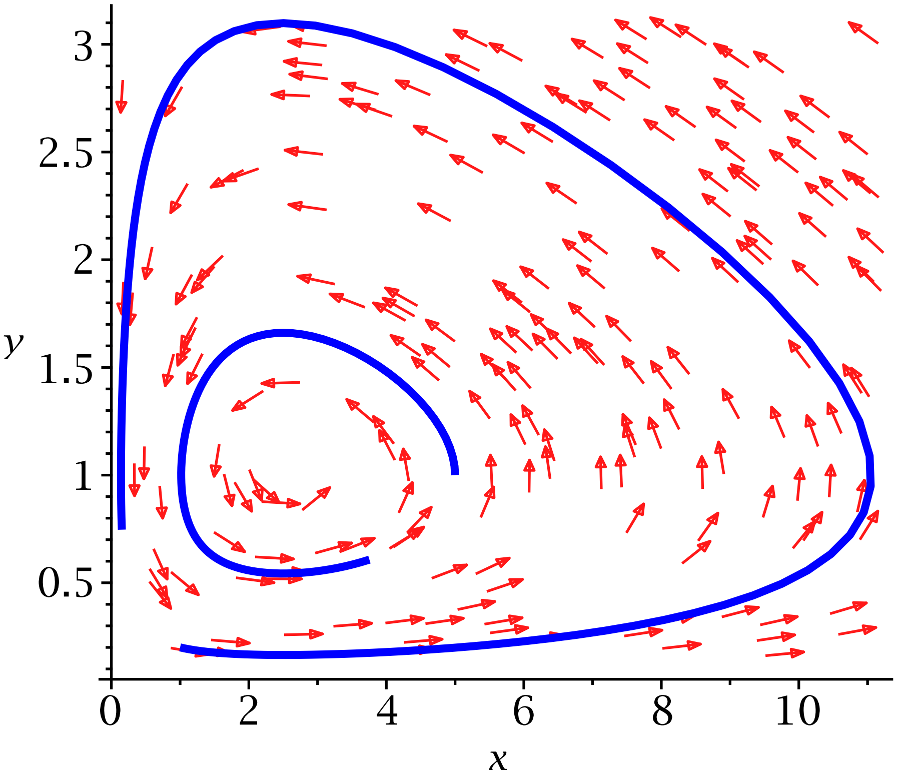
$$initvals := [[x(0) = 1, y(0) = 0.2], [x(0) = 5, y(0) = 1]]$$

```
> DEplot( {de1,de2}, {x(t),y(t)}, t=0..80, initvals, arrows=medium
);
```



The 20 by 20 grid of arrows shows the direction of motion. Random arrow placement is visually much better. A MACM 204 student implemented this.

```
> DEplot( {de1,de2}, {x(t),y(t)}, t=0..80, initvals, arrows=medium,
  numpoints=100, linecolour=blue, dirfield=200 );
```


3
2.5
2
y
1.5
1
0.5
0
2
4
6
8
10
x