Some setup: commands for plotting the dynamical system (ignore this.)

In[275]:=
```
iterates[x10_, x20_, L_, S_] :=
 Partition[Flatten[(x[0] := {{x10}, {x20}};
    x[n_] := S.x[n - 1]; Table[x[n], {n, 0, L}])], 2]
```

In[276]:=
```
multiiterates[x0s_, L_, S_] := Flatten[
  Table[iterates[x0s[[i, 1]], x0s[[i, 2]], L, S], {i, 1, Length[x0s]}], 1]
```

In[277]:=
```
plotmultiiter[x0s_, L_, S_] := Show[
  ListPlot[multiiterates[x0s, L, S], PlotStyle → PointSize[Large],
   AspectRatio → 1, PlotRange → {{-5, 5}, {-5, 5}}], ListPlot[
   x0s, PlotStyle → {Red, PointSize[Large]}, AspectRatio → 1]]
```

Constructing a diagonal matrix B. (In lecture this matrix was called D, but *Mathematica* reserves the name "D" for the differentiation operator.)

In[278]:=
```
λ1 = 1.14; λ2 = 0.87;
```

In[279]:=
```
B = {{λ1, 0}, {0, λ2}};
```

In[280]:=
```
B // MatrixForm
```

Out[280]//MatrixForm=

$$\begin{pmatrix} 1.14 & 0 \\ 0 & 0.87 \end{pmatrix}$$

A matrix A which is similar to B; the two are related by the change-of-basis matrix P.

In[281]:=
```
P = {{1, 1}, {-1, 1}};
```

In[282]:=
```
A = P.B.Inverse[P];
```

In[283]:=
```
A // MatrixForm
```

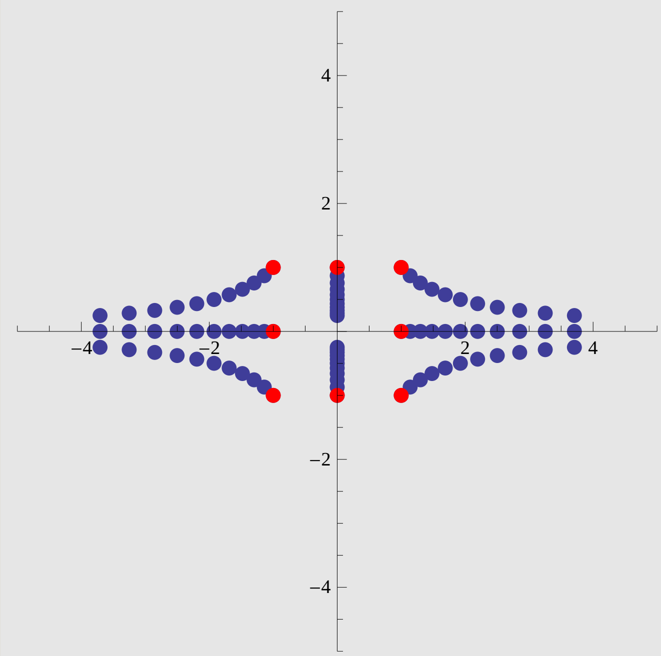
Out[283]//MatrixForm=

$$\begin{pmatrix} 1.005 & -0.135 \\ -0.135 & 1.005 \end{pmatrix}$$

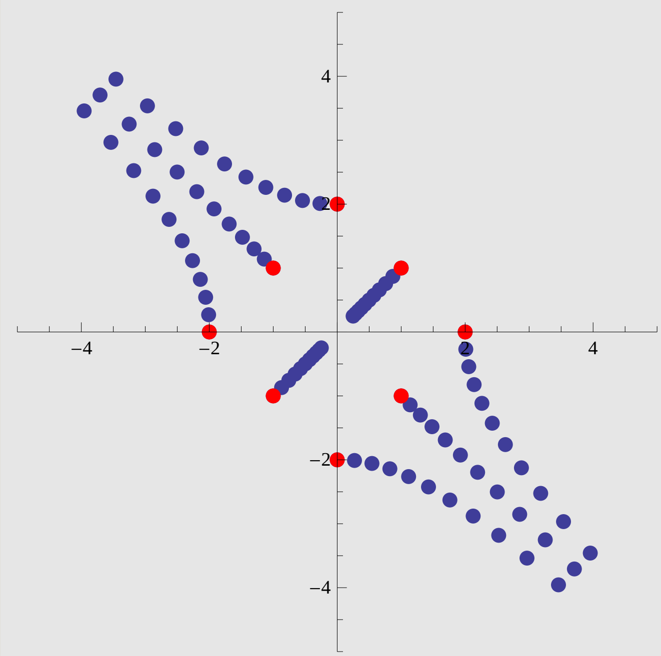
Iterating the dynamical systems defined by B (first) and A (second) for 10 time steps, with a few different choices of initial condition. The initial points are shown in red. The two pictures are related to one another by the linear transformation P.

In[284]:=
```
startingpoints = {{1, 0}, {-1, 0},
   {0, 1}, {0, -1}, {1, 1}, {1, -1}, {-1, -1}, {-1, 1}};
```

In[285]:=
```
plotmultiiter[startingpoints, 10, B]
```

Out[285]=


In[286]:=
```
plotmultiiter[Transpose[P.Transpose[startingpoints]], 10, A]
```

Out[286]=


The eigenvalues of A. Note they are the same as the eigenvalues of B, as they must be since the two matrices are similar.

In[287]:=
```
Eigenvalues[A]
```

Out[287]= {1.14, 0.87}

In[288]:=
```
Eigenvalues[B]
```

Out[288]= {1.14, 0.87}

The absolute values of the eigenvalues. These predict whether the dynamical system has an attracting, repelling or saddle point.

In[289]:=
```
Abs[Eigenvalues[A]]
```

Out[289]= {1.14, 0.87}