Chapter 4

```
In[1]:= Needs["Graphics`PlotField`"]
```

à Question 1

(i) We can readily write this

$$\frac{dy}{dx} = \frac{y}{2x}$$
$$\frac{dy}{y} = \frac{dx}{2x}$$

Integrating both sides

$$\int \frac{dy}{y}\, dy = \int \frac{dx}{2x}\, dx$$

$$\ln(y) = \frac{1}{2}\ln(x) + c_0$$

where c_0 is the constant of integration. Then

$$y = c\sqrt{x}$$

If $x(0) = 2$ and $y(0) = 3$, then $3 = c\sqrt{2}$ and

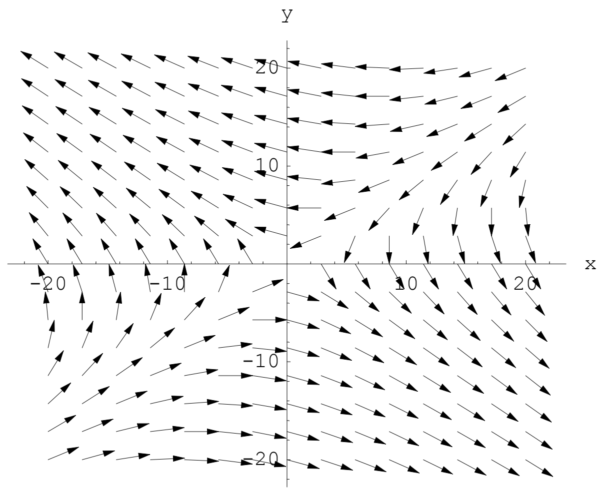
$$y = \frac{3\sqrt{x}}{\sqrt{2}}$$

(ii) To verify this result

```
In[2]:= DSolve[y'[x] == y[x]/(2 x), y[x], x]

Out[2]= {{y[x] → √x C[1]}}

In[3]:= DSolve[{y'[x] == y[x]/(2 x), y[2] == 3}, y[x], x]

Out[3]= {{y[x] → 3 √x / √2}}
```

à Question 2

```
In[4]:= arrows =
          PlotVectorField[ {x - 3 y, -2 x + y}, {x, -20, 20}, {y, -20, 20}, Axes -> True,
           AxesLabel -> {"x", "y"}, ScaleFunction -> (1 &), AspectRatio -> 0.8];
```



(a)

```
In[5]:= sol1 = NDSolve[ {x'[t] == x[t] - 3 y[t],
           y'[t] == -2 x[t] + y[t], x[0] == 4, y[0] == 2}, {x, y}, {t, 0, 10}]

Out[5]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
          y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[6]:= traj1 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol1],
           {t, 0, 1}, DisplayFunction -> Identity];
```

(b)

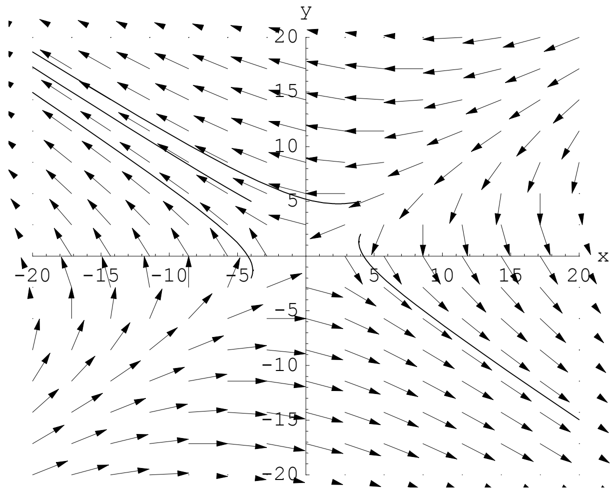
```
In[7]:= sol2 = NDSolve[ {x'[t] == x[t] - 3 y[t],
           y'[t] == -2 x[t] + y[t], x[0] == 4, y[0] == 5}, {x, y}, {t, 0, 10}]

Out[7]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
          y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[8]:= traj2 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol2],
           {t, 0, 1}, DisplayFunction -> Identity];
```

(c)

```
In[9]:= sol3 = NDSolve[ {x'[t] == x[t] - 3 y[t],
           y'[t] == -2 x[t] + y[t], x[0] == -4, y[0] == -2}, {x, y}, {t, 0, 10}]

Out[9]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
          y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[10]:= traj3 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol3],
            {t, 0, 1}, DisplayFunction -> Identity];
```

(d)

```
In[11]:= sol4 = NDSolve[ {x'[t] == x[t] - 3 y[t],
            y'[t] == -2 x[t] + y[t], x[0] == -4, y[0] == 5}, {x, y}, {t, 0, 10}]

Out[11]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[12]:= traj4 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol4],
            {t, 0, 1}, DisplayFunction -> Identity];

In[13]:= Show[{arrows, traj1, traj2, traj3, traj4},
           PlotRange -> {{-20, 20}, {-20, 20}}, AspectRatio -> 0.8];
```

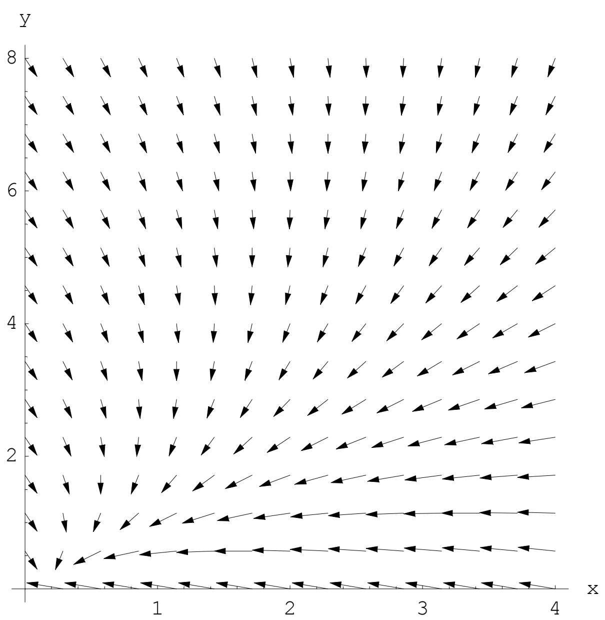


à Question 3

```
In[14]:= Clear[x, y, sol1, sol2, sol3, sol4, traj1, traj2, traj3, traj4, arrows]
```

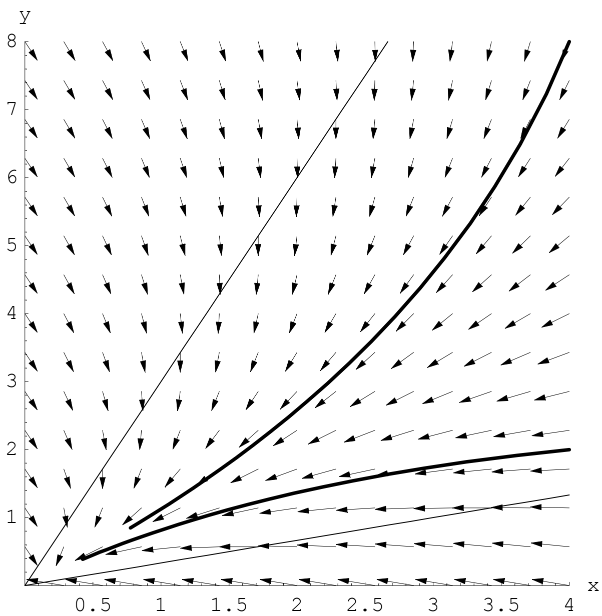
(i)

```
In[15]:= arrows1 = PlotVectorField[ {-3 x + y, x - 3 y}, {x, 0, 4}, {y, 0, 8}, Axes -> True,
            AxesLabel -> {"x", "y"}, ScaleFunction -> (1 &), AspectRatio -> 1];
```



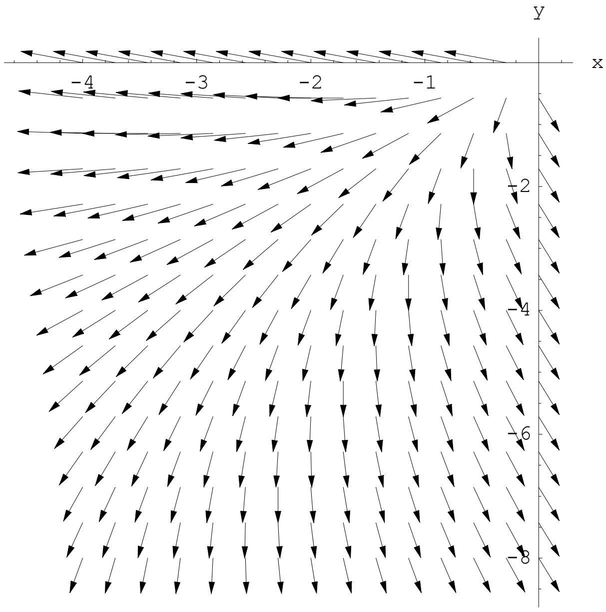
(i)

```
In[16]:= sol11 = NDSolve[ {x'[t] == -3 x[t] + y[t],
           y'[t] == x[t] - 3 y[t], x[0] == 4, y[0] == 8}, {x, y}, {t, 0, 10}]

Out[16]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[17]:= traj11 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol11], {t, 0, 1},
           PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];

In[18]:= sol12 = NDSolve[ {x'[t] == -3 x[t] + y[t],
           y'[t] == x[t] - 3 y[t], x[0] == 4, y[0] == 2}, {x, y}, {t, 0, 10}]

         General::spell1 :
          Possible spelling error: new symbol name "sol12" is similar to existing symbol "sol2".

Out[18]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[19]:= traj12 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol12], {t, 0, 1},
           PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];

         General::spell1 :
          Possible spelling error: new symbol name "traj12" is similar to existing symbol "traj2".

In[20]:= lines1 = Plot[{3 x, (1 / 3) x}, {x, 0, 4}, DisplayFunction -> Identity];

In[21]:= Show[{arrows1, traj11, traj12, lines1}, PlotRange -> {{0, 4}, {0, 8}}];
```



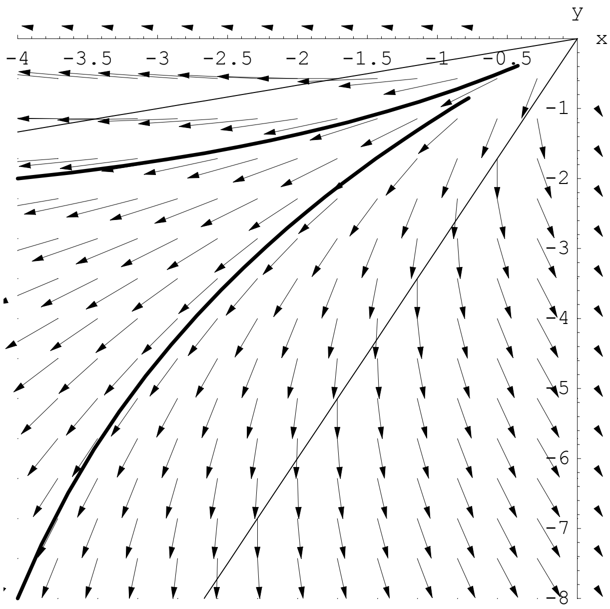
(ii)

```
In[22]:= arrows2 =
          PlotVectorField[ {-3 x + y, x - 3 y}, {x, 0, -4}, {y, 0, -8}, Axes -> True,
           AxesLabel -> {"x", "y"}, ScaleFunction -> (1 &), AspectRatio -> 1];
```



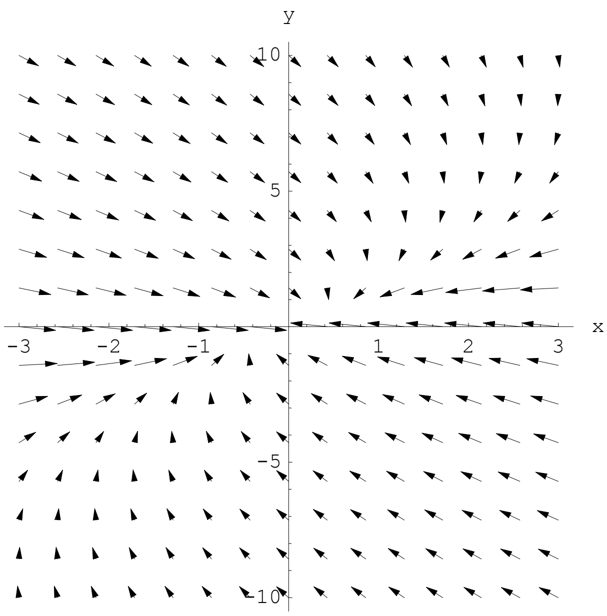

```
In[23]:= sol21 = NDSolve[ {x'[t] == -3 x[t] + y[t],
            y'[t] == x[t] - 3 y[t], x[0] == -4, y[0] == -8}, {x, y}, {t, 0, 10}]

        General::spell :
         Possible spelling error: new symbol name "sol21" is similar to existing symbols {sol1, sol12}.

Out[23]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[24]:= traj21 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol21], {t, 0, 1},
            PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];

        General::spell : Possible spelling error: new
           symbol name "traj21" is similar to existing symbols {traj1, traj12}.

In[25]:= sol22 = NDSolve[ {x'[t] == -3 x[t] + y[t],
            y'[t] == x[t] - 3 y[t], x[0] == -4, y[0] == -2}, {x, y}, {t, 0, 10}]

Out[25]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[26]:= traj22 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol22], {t, 0, 1},
            PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];

In[27]:= lines2 = Plot[{3 x, (1 / 3) x}, {x, 0, -4}, DisplayFunction -> Identity];
```

```
In[28]:= Show[{arrows2, lines2, traj21, traj22}, PlotRange -> {{0, -4}, {0, -8}}];
```



(iii)

```
In[29]:= arrows3 =
          PlotVectorField[ {-3 x + y, x - 3 y}, {x, -3, 3}, {y, -10, 10}, Axes -> True,
           AxesLabel -> {"x", "y"}, ScaleFunction -> (1 &), AspectRatio -> 1];
```




```
In[30]:= sol31 = NDSolve[ {x'[t] == -3 x[t] + y[t],
           y'[t] == x[t] - 3 y[t], x[0] == 2, y[0] == 10}, {x, y}, {t, 0, 10}]

        General::spell1 :
         Possible spelling error: new symbol name "sol31" is similar to existing symbol "sol1".

Out[30]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}
```

```
In[31]:= traj31 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol31], {t, 0, 1},
             PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];
```

```
General::spell1 :
 Possible spelling error: new symbol name "traj31" is similar to existing symbol "traj1".
```

```
In[32]:= lines3 = Plot[{3 x, (1 / 3) x}, {x, -3, 3}, DisplayFunction -> Identity];
```

```
In[33]:= sol32 = NDSolve[ {x'[t] == -3 x[t] + y[t],
             y'[t] == x[t] - 3 y[t], x[0] == -2, y[0] == -10}, {x, y}, {t, 0, 10}]
```

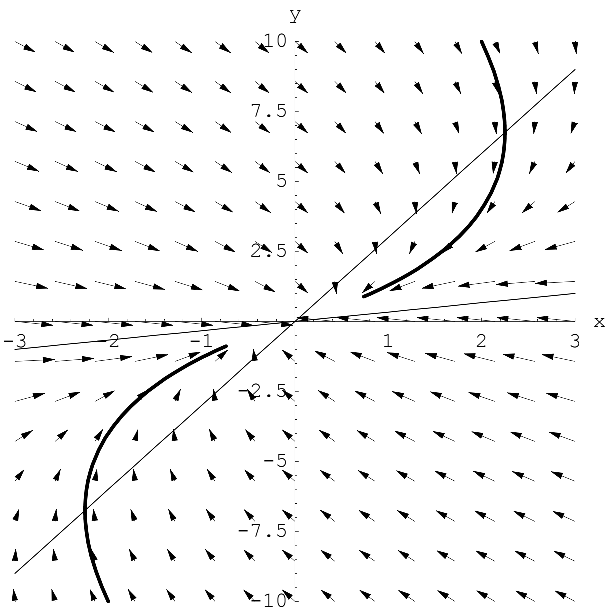
```
General::spell1 :
 Possible spelling error: new symbol name "sol32" is similar to existing symbol "sol2".
```

```
Out[33]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}
```

```
In[34]:= traj32 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol32], {t, 0, 1},
             PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];
```

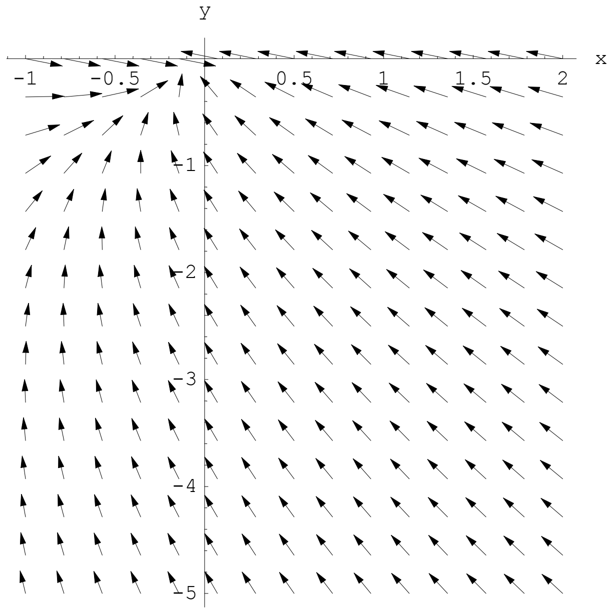
```
General::spell1 :
 Possible spelling error: new symbol name "traj32" is similar to existing symbol "traj2".
```

```
In[35]:= Show[{arrows3, lines3, traj31, traj32}, PlotRange -> {{-3, 3}, {-10, 10}}];
```

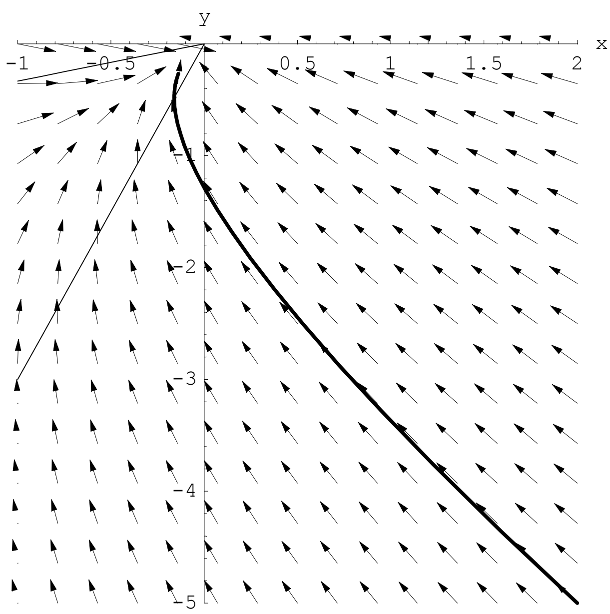


(iv)

```
In[36]:= arrows4 =
           PlotVectorField[ {-3 x + y, x - 3 y},  {x, -1, 2}, {y, -5, 0},  Axes -> True,
            AxesLabel -> {"x", "y"}, ScaleFunction -> (1 &), AspectRatio -> 1];
```




```
In[37]:= sol4 = NDSolve[ {x'[t] == -3 x[t] + y[t],
            y'[t] == x[t] - 3 y[t],  x[0] == 2,  y[0] == -5}, {x, y}, {t, 0, 10}]

Out[37]= {{x → InterpolatingFunction[{{0., 10.}}, <>],
           y → InterpolatingFunction[{{0., 10.}}, <>]}}

In[38]:= traj4 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol4], {t, 0, 1},
            PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];

In[39]:= lines4 = Plot[{3 x, (1 / 3) x}, {x, -1, 2}, DisplayFunction -> Identity];

In[40]:= Show[{arrows4, lines4, traj4}, PlotRange -> {{-1, 2}, {-5, 0}}];
```



As can be seen from the resulting figure, the trajectory passes into another quadrant before converging on the fixed point.

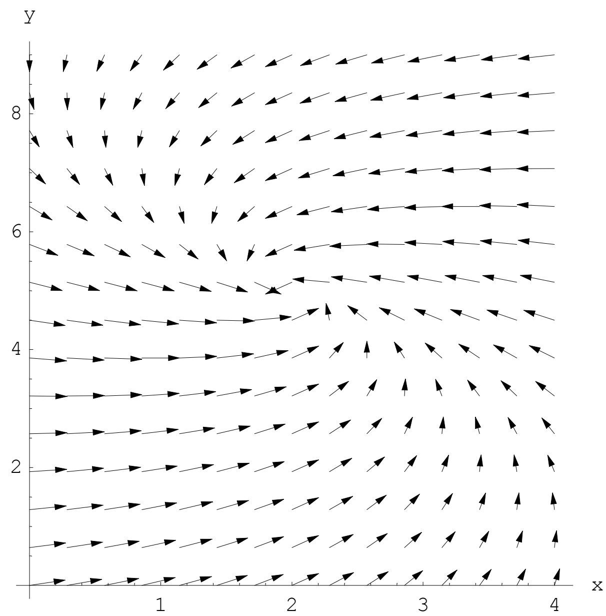
à Question 4

```
In[41]:= Clear[x, y, sol1, sol2, sol3, sol4, sol5, traj1,
          traj2, traj3, traj4, traj5, lines1, lines2, lines3, lines4,
          lines5, arrows1, arrows2, arrows3, arrows4, arrows5]
```

```
In[42]:= Solve[{-2 x - y + 9 == 0, -y + x + 3 == 0}, {x, y}]
```

```
Out[42]= {{x → 2, y → 5}}
```

```
In[43]:= arrows1 = PlotVectorField[{-2 x - y + 9, -y + x + 3},
            {x, 0, 4}, {y, 0, 9}, AspectRatio -> 1, Axes -> True,
            AxesLabel -> {"x", "y"}, ScaleFunction -> (1 &)];
```



(i)

```
In[44]:= sol1 = NDSolve[ {x'[t] == -2 x[t] - y[t] + 9,
            y'[t] == -y[t] + x[t] + 3, x[0] == 0, y[0] == 2}, {x, y}, {t, 0, 20}]
```

```
Out[44]= {{x → InterpolatingFunction[{{0., 20.}}, <>],
           y → InterpolatingFunction[{{0., 20.}}, <>]}}
```

```
In[45]:= traj1 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol1], {t, 0, 5},
            PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];
```

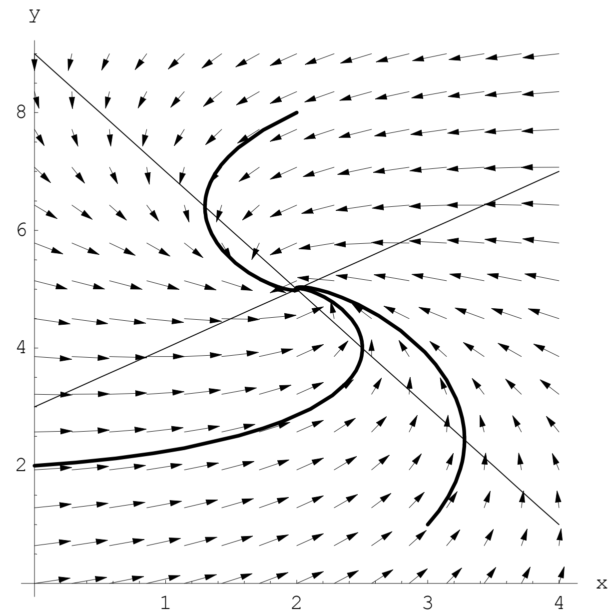
```
In[46]:= lines1 = Plot[{-2 x + 9, x + 3}, {x, 0, 4}, DisplayFunction -> Identity];
```

(ii)

```
In[47]:= sol2 = NDSolve[ {x'[t] == -2 x[t] - y[t] + 9,
            y'[t] == -y[t] + x[t] + 3, x[0] == 2, y[0] == 8}, {x, y}, {t, 0, 20}]

Out[47]= {{x → InterpolatingFunction[{{0., 20.}}, <>],
           y → InterpolatingFunction[{{0., 20.}}, <>]}}

In[48]:= traj2 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol2], {t, 0, 5},
            PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];
```

(iii)

```
In[49]:= sol3 = NDSolve[ {x'[t] == -2 x[t] - y[t] + 9,
            y'[t] == -y[t] + x[t] + 3, x[0] == 3, y[0] == 1}, {x, y}, {t, 0, 20}]

Out[49]= {{x → InterpolatingFunction[{{0., 20.}}, <>],
           y → InterpolatingFunction[{{0., 20.}}, <>]}}

In[50]:= traj3 = ParametricPlot[Evaluate[ {x[t], y[t]} /. sol3], {t, 0, 5},
            PlotStyle -> {Thickness[0.007]}, DisplayFunction -> Identity];

In[51]:= Show[{arrows1, lines1, traj1, traj2, traj3}];
```



The resulting diagram shows quite clearly that all trajectories follow a counter-clockwise spiral towards the fixed point. However, it also shows that the spiral motion involves a direct convergence on the fixed point. In other words, there is no *repeated* over- and under-shooting.

à Question 5

```
In[52]:= Clear[x, y]
```

(i) Characteristic roots obtained by defining the matrix of the system and using the Eigenvalues command.

```
In[53]:= a = {{2, 3}, {3, 2}} ; MatrixForm[a]
```

Out[53]//MatrixForm=

$$\begin{pmatrix} 2 & 3 \\ 3 & 2 \end{pmatrix}$$

```
In[54]:= Eigenvalues[a]

Out[54]= {-1, 5}
```

(ii) Now employ the Eigenvectors command.

```
In[55]:= Eigenvectors[a]

Out[55]= {{-1, 1}, {1, 1}}
```

(iii) Solving directly we have:

```
In[56]:= DSolve[{x'[t] == 2 x[t] + 3 y[t], y'[t] == 3 x[t] + 2 y[t]}, {x[t], y[t]}, t]
```

$$\text{Out[56]= } \left\{\left\{x[t] \to \frac{1}{2} e^{-t} \left(C[1] + e^{6t} C[1] - C[2] + e^{6t} C[2]\right), y[t] \to \frac{1}{2} e^{-t} \left(-C[1] + e^{6t} C[1] + C[2] + e^{6t} C[2]\right)\right\}\right\}$$

(iv) The initial value problem is:

```
In[57]:= DSolve[{x'[t] == 2 x[t] + 3 y[t],
           y'[t] == 3 x[t] + 2 y[t], x[0] == 1, y[0] == 0}, {x[t], y[t]}, t]
```

$$\text{Out[57]= } \left\{\left\{x[t] \to \frac{1}{2} e^{-t} \left(1 + e^{6t}\right), y[t] \to \frac{1}{2} e^{-t} \left(-1 + e^{6t}\right)\right\}\right\}$$

à Question 6

```
In[58]:= Clear[x, y, a]
```

(i)

```
In[59]:= a = {{1, 3}, {5, 3}}; MatrixForm[a]
```

Out[59]//MatrixForm=

$$\begin{pmatrix} 1 & 3 \\ 5 & 3 \end{pmatrix}$$

```
In[60]:= Eigenvalues[a]

Out[60]= {-2, 6}
```

(ii)

```
In[61]:= Eigenvectors[a]

Out[61]= {{-1, 1}, {3, 5}}
```

(iii)

```
In[62]:= DSolve[{x'[t] == x[t] + 3 y[t],
           y'[t] == 5 x[t] + 3 y[t], x[0] == 1, y[0] == 3}, {x[t], y[t]}, t]
```

Out[62]= $\left\{\left\{x[t] \to \frac{1}{2} e^{-2t} (-1 + 3 e^{8t}),\ y[t] \to \frac{1}{2} e^{-2t} (1 + 5 e^{8t})\right\}\right\}$

à Question 7

```
In[63]:= Clear[x, y, a]

In[64]:= a = {{1, 1}, {-2, 4}}; MatrixForm[a]
```

Out[64]//MatrixForm=

$$\begin{pmatrix} 1 & 1 \\ -2 & 4 \end{pmatrix}$$

```
In[65]:= Eigenvalues[a]

Out[65]= {2, 3}

In[66]:= Eigenvectors[a]

Out[66]= {{1, 1}, {1, 2}}
```

The Wronksian matrix, here denoted w, is formed from the eigenvectors of the system. If Det[w] is nonzero then the eigenvectors are linearly independent.

```
In[67]:= w = Eigenvectors[a]; MatrixForm[w]
```

Out[67]//MatrixForm=

$$\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}$$

```
In[68]:= Det[w]

Out[68]= 1
```

à Question 8

```
In[69]:= Clear[x, y, a, w]

In[70]:= a = {{1, 0, 0}, {2, 3, 1}, {0, 2, 4}}; MatrixForm[a]
```

Out[70]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 2 & 3 & 1 \\ 0 & 2 & 4 \end{pmatrix}$$

(i) The eigenvalues and eigenvectors are as follows:

```
In[71]:= Eigenvalues[a]

Out[71]= {1, 2, 5}

In[72]:= Eigenvectors[a]

Out[72]= {{2, -3, 2}, {0, -1, 1}, {0, 1, 2}}
```

(ii)The general solution is derived by solving the following differential equation system:

```
In[73]:= DSolve[ {x'[t] == x[t], y'[t] == 2 x[t] + 3 y[t] + z[t], z'[t] == 2 y[t] + 4 z[t]},
          {x[t], y[t], z[t]}, t]
```

Out[73]= $\left\{\left\{x[t] \to e^{t} C[1],\ y[t] \to \frac{1}{6} e^{t} (-9 C[1] + 8 e^{t} C[1] + e^{4t} C[1] + 4 e^{t} C[2] + 2 e^{4t} C[2] - 2 e^{t} C[3] + 2 e^{4t} C[3]),\ z[t] \to \frac{1}{3} e^{t} (3 C[1] - 4 e^{t} C[1] + e^{4t} C[1] - 2 e^{t} C[2] + 2 e^{4t} C[2] + e^{t} C[3] + 2 e^{4t} C[3])\right\}\right\}$

(iii) The Wronksian is given by:

```
In[74]:= w = Eigenvectors[a]; MatrixForm[w]
```

Out[74]//MatrixForm=

$$\begin{pmatrix} 2 & -3 & 2 \\ 0 & -1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$$

```
In[75]:= Det[w]
```

Out[75]= -6