Chapter Eight

Section 8.1

2. The Euler formula for this problem is

$$y_{n+1} = y_n + h\big(5\,t_n - 3\sqrt{y_n}\,\big),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 5nh^2 - 3h\sqrt{y_n}\,,$$

with $y_0 = 2$.

(a). Euler method with $h = 0.05$:

	$n = 2$	$n = 4$	$n = 6$	$n = 8$
t_n	0.1	0.2	0.3	0.4
y_n	1.59980	1.29288	1.07242	0.930175

(b). Euler method with $h = 0.025$:

	$n = 4$	$n = 8$	$n = 12$	$n = 16$
t_n	0.1	0.2	0.3	0.4
y_n	1.61124	1.31361	1.10012	0.962552

The *backward* Euler formula is

$$y_{n+1} = y_n + h\big(5\,t_{n+1} - 3\sqrt{y_{n+1}}\,\big),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 5(n+1)h^2 - 3h\sqrt{y_{n+1}}\,,$$

with $y_0 = 2$. Solving for y_{n+1}, and choosing the *positive* root, we find that

$$y_{n+1} = \left[-\frac{3}{2}h + \frac{1}{2}\sqrt{(20n+29)h^2 + 4y_n}\right]^2.$$

(c). Backward Euler method with $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.64337	1.37164	1.17763	1.05334

(d). Backward Euler method with $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.63301	1.35295	1.15267	1.02407

3. The Euler formula for this problem is

$$y_{n+1} = y_n + h(2\,y_n - 3\,t_n\,),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$,

$$y_{n+1} = y_n + 2hy_n - 3nh^2\,,$$

with $y_0 = 1$.

(a). Euler method with $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.2025	1.41603	1.64289	1.88590

(b). Euler method with $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.20388	1.41936	1.64896	1.89572

The *backward* Euler formula is

$$y_{n+1} = y_n + h(2\,y_{n+1} - 3\,t_{n+1}\,),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 2h\,y_{n+1} - 3(n+1)h^2\,,$$

with $y_0 = 1$. Solving for y_{n+1}, we find that

$$y_{n+1} = \frac{y_n - 3(n+1)h^2}{1 - 2h}\,.$$

(c). Backward Euler method with $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.20864	1.43104	1.67042	1.93076

(d). Backward Euler method with $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.20693	1.42683	1.66265	1.91802

4. The Euler formula is

$$y_{n+1} = y_n + h[2\,t_n + exp(-t_n\,y_n)].$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 2nh^2 + h\,exp(-nh\,y_n),$$

with $y_0 = 1$.

(a). Euler method with $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.10244	1.21426	1.33484	1.46399

(b). Euler method with $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.10365	1.21656	1.33817	1.46832

The *backward* Euler formula is

$$y_{n+1} = y_n + h[2\,t_{n+1} + exp(-t_{n+1}\,y_{n+1})].$$

Since $t_0 = 0$ and $t_{n+1} = (n+1)h$, we can also write

$$y_{n+1} = y_n + 2h^2(n+1) + h\,exp[-(n+1)h\,y_{n+1}],$$

with $y_0 = 1$. This equation cannot be solved *explicitly* for y_{n+1}. At each step, given the current value of y_n, the equation must be solved *numerically* for y_{n+1}.

(c). Backward Euler method with $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.10720	1.22333	1.34797	1.48110

(d). Backward Euler method with $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.10603	1.22110	1.34473	1.47688

6. The Euler formula for this problem is

$$y_{n+1} = y_n + h\left(t_n^2 - y_n^2\right) sin\, y_n .$$

Here $t_0 = 0$ and $t_n = nh$. So that

$$y_{n+1} = y_n + h\left(n^2 h^2 - y_n^2\right) sin\, y_n ,$$

with $y_0 = -1$.

(a). Euler method with $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	-0.920498	-0.857538	-0.808030	-0.770038

(b). Euler method with $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	-0.922575	-0.860923	-0.82300	-0.774965

The *backward* Euler formula is

$$y_{n+1} = y_n + h\left(t_{n+1}^2 - y_{n+1}^2\right) sin\, y_{n+1}.$$

Since $t_0 = 0$ and $t_{n+1} = (n+1)h$, we can also write

$$y_{n+1} = y_n + h\left[(n+1)^2 h^2 - y_{n+1}^2\right] sin\, y_{n+1}\,,$$

with $y_0 = -1$. Note that this equation cannot be solved *explicitly* for y_{n+1}. Given y_n, the transcendental equation

$$y_{n+1} + h\, y_{n+1}^2 sin\, y_{n+1} = y_n + h(n+1)^2 h^2$$

must be solved *numerically* for y_{n+1}.

(c). Backward Euler method with $h = 0.05$:

	$n = 2$	$n = 4$	$n = 6$	$n = 8$
t_n	0.1	0.2	0.3	0.4
y_n	-0.928059	-0.870054	-0.824021	-0.788686

(d). Backward Euler method with $h = 0.025$:

	$n = 4$	$n = 8$	$n = 12$	$n = 16$
t_n	0.1	0.2	0.3	0.4
y_n	-0.926341	-0.867163	-0.820279	-0.784275

8. The Euler formula

$$y_{n+1} = y_n + h\left(5\,t_n - 3\sqrt{y_n}\,\right),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 5nh^2 - 3h\sqrt{y_n}\,,$$

with $y_0 = 2$.

(a). Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	0.891830	1.25225	2.37818	4.07257

(b). Euler method with $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	0.908902	1.26872	2.39336	4.08799

The *backward* Euler formula is

$$y_{n+1} = y_n + h\left(5\,t_{n+1} - 3\sqrt{y_{n+1}}\right),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 5(n+1)h^2 - 3h\sqrt{y_{n+1}}\,,$$

with $y_0 = 2$. Solving for y_{n+1}, and choosing the *positive* root, we find that

$$y_{n+1} = \left[-\frac{3}{2}h + \frac{1}{2}\sqrt{(20n+29)h^2 + 4y_n}\right]^2.$$

(c). Backward Euler method with $h = 0.025$:

	$n=20$	$n=40$	$n=60$	$n=80$
t_n	0.5	1.0	1.5	2.0
y_n	0.958565	1.31786	2.43924	4.13474

(d). Backward Euler method with $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	0.942261	1.30153	2.24389	4.11908

9. The Euler formula for this problem is

$$y_{n+1} = y_n + h\sqrt{t_n + y_n}\ .$$

Here $t_0 = 0$ and $t_n = nh$. So that

$$y_{n+1} = y_n + h\sqrt{nh + y_n}\ ,$$

with $y_0 = 3$.

10. The Euler formula is

$$y_{n+1} = y_n + h[2\,t_n + exp(-\,t_n\,y_n)].$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 2nh^2 + h\,exp(-\,nh\,y_n),$$

with $y_0 = 1$.

(a). Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	1.60729	2.46830	3.72167	5.45963

(b). Euler method with $h = 0.0125$:

	$n = 40$	$n = 80$	$n = 120$	$n = 160$
t_n	0.5	1.0	1.5	2.0
y_n	1.60996	2.47460	3.73356	5.47774

The *backward* Euler formula is

$$y_{n+1} = y_n + h[2\,t_{n+1} + exp(-\,t_{n+1}\,y_{n+1})].$$

Since $t_0 = 0$ and $t_{n+1} = (n+1)h$, we can also write

$$y_{n+1} = y_n + 2h^2(n+1) + h\,exp[-\,(n+1)h\,y_{n+1}],$$

with $y_0 = 1$. This equation cannot be solved *explicitly* for y_{n+1}. At each step, given the current value of y_n, the equation must be solved *numerically* for y_{n+1}.

(c). Backward Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	1.61792	2.49356	3.76940	5.53223

(d). Backward Euler method with $h = 0.0125$:

	$n = 40$	$n = 80$	$n = 120$	$n = 160$
t_n	0.5	1.0	1.5	2.0
y_n	1.61528	2.48723	3.75742	5.51404

11. The Euler formula is

$$y_{n+1} = y_n + h(4 - t_n\, y_n)/\left(1 + y_n^2\right).$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \left(4h - nh^2\, y_n\right)/\left(1 + y_n^2\right),$$

with $y_0 = -2$.

(a). Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	-1.45865	-0.217545	1.05715	1.41487

(b). Euler method with $h = 0.0125$:

	$n = 40$	$n = 80$	$n = 120$	$n = 160$
t_n	0.5	1.0	1.5	2.0
y_n	-1.45322	-0.180813	1.05903	1.41244

The *backward* Euler formula is

$$y_{n+1} = y_n + h(4 - t_{n+1}\, y_{n+1})/\left(1 + y_{n+1}^2\right).$$

Since $t_0 = 0$ and $t_{n+1} = (n+1)h$, we can also write

$$y_{n+1}\left(1 + y_{n+1}^2\right) = y_n\left(1 + y_{n+1}^2\right) + \left[4h - (n+1)h^2\, y_{n+1}\right],$$

with $y_0 = -2$. This equation cannot be solved *explicitly* for y_{n+1}. At each step, given the current value of y_n, the equation must be solved *numerically* for y_{n+1}.

(c). Backward Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	-1.43600	-0.0681657	1.06489	1.40575

(d). Backward Euler method with $h = 0.0125$:

	$n = 40$	$n = 80$	$n = 120$	$n = 160$
t_n	0.5	1.0	1.5	2.0
y_n	-1.44190	-0.105737	1.06290	1.40789

12. The Euler formula is

$$y_{n+1} = y_n + h\big(y_n^2 + 2\,t_n\,y_n\big)/\big(3 + t_n^2\big).$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \big(h\,y_n^2 + 2\,nh^2\,y_n\big)/\big(3 + n^2h^2\big),$$

with $y_0 = 0.5$.

(a). Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	0.587987	0.791589	1.14743	1.70973

(b). Euler method with $h = 0.0125$:

	$n = 40$	$n = 80$	$n = 120$	$n = 160$
t_n	0.5	1.0	1.5	2.0
y_n	0.589440	0.795758	1.15693	1.72955

The *backward* Euler formula is

$$y_{n+1} = y_n + h\big(y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}\big)/\big(3 + t_{n+1}^2\big).$$

Since $t_0 = 0$ and $t_{n+1} = (n+1)h$, we can also write

$$y_{n+1}\big[3 + (n+1)^2h^2\big] - h\,y_{n+1}^2 = y_n\big[3 + (n+1)^2h^2\big] + 2(n+1)h^2\,y_{n+1},$$

with $y_0 = 0.5$. Note that although this equation can be solved *explicitly* for y_{n+1}, it is also possible to use a numerical equation solver. At each time step, given the current value of y_n, the equation may be solved *numerically* for y_{n+1}.

(c). Backward Euler method with $h = 0.025$:

	$n = 20$	$n = 40$	$n = 60$	$n = 80$
t_n	0.5	1.0	1.5	2.0
y_n	0.593901	0.808716	1.18687	1.79291

(d). Backward Euler method with $h = 0.0125$:

	$n = 40$	$n = 80$	$n = 120$	$n = 160$
t_n	0.5	1.0	1.5	2.0
y_n	0.592396	0.804319	1.17664	1.77111

13. The Euler formula for this problem is

$$y_{n+1} = y_n + h(1 - t_n + 4\,y_n\,),$$

in which $t_n = t_0 + nh$. Since $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h - nh^2 + 4h\,y_n\,,$$

with $y_0 = 1$. With $h = 0.01$, a total number of 200 iterations is needed to reach $\bar{t} = 2$. With $h = 0.001$, a total of 2000 iterations are necessary.

14. The *backward* Euler formula is

$$y_{n+1} = y_n + h(1 - t_{n+1} + 4\,y_{n+1}\,).$$

Since the equation is linear, we can solve for y_{n+1} in terms of y_n :

$$y_{n+1} = \frac{y_n + h - h\,t_{n+1}}{1 - 4h}\,.$$

Here $t_0 = 0$ and $y_0 = 1$. With $h = 0.01$, a total number of 200 iterations is needed to reach $\bar{t} = 2$. With $h = 0.001$, a total of 2000 iterations are necessary.

18. Let $\phi(t)$ be a solution of the initial value problem. The *local* truncation error for the Euler method, on the interval $t_n \le t \le t_{n+1}$, is given by

$$e_{n+1} = \frac{1}{2}\phi''(\,\bar{t}_n)h^2,$$

where $t_n < \bar{t}_n < t_{n+1}$. Since $\phi'(t) = t^2 + [\phi(t)]^2$, it follows that

$$\begin{aligned}\phi''(t) &= 2t + 2\,\phi(t)\phi'(t)\\ &= 2t + 2t^2\phi(t) + 2[\phi(t)]^3.\end{aligned}$$

Hence

$$|e_{n+1}| \leq \left[t_{n+1} + t_{n+1}^2 M_{n+1} + M_{n+1}^3\right]h^2,$$

in which $M_{n+1} = max\{\phi(t)\,|\,t_n \leq t \leq t_{n+1}\}$.

20. Given that $\phi(t)$ is a solution of the initial value problem, the *local* truncation error for the Euler method, on the interval $t_n \leq t \leq t_{n+1}$, is

$$e_{n+1} = \frac{1}{2}\phi''(\bar{t}_n)h^2,$$

where $t_n < \bar{t}_n < t_{n+1}$. Based on the ODE, $\phi'(t) = \sqrt{t + \phi(t)}$, and hence

$$\begin{aligned}\phi''(t) &= \frac{1 + \phi'(t)}{2\sqrt{t + \phi(t)}} \\ &= \frac{1}{2\sqrt{t + \phi(t)}} + \frac{1}{2}\,.\end{aligned}$$

Therefore

$$|e_{n+1}| \leq \frac{1}{4}\left[1 + \frac{1}{\sqrt{\bar{t}_n + \phi(\bar{t}_n)}}\right]h^2.$$

21. Let $\phi(t)$ be a solution of the initial value problem. The *local* truncation error for the Euler method, on the interval $t_n \leq t \leq t_{n+1}$, is given by

$$e_{n+1} = \frac{1}{2}\phi''(\bar{t}_n)h^2,$$

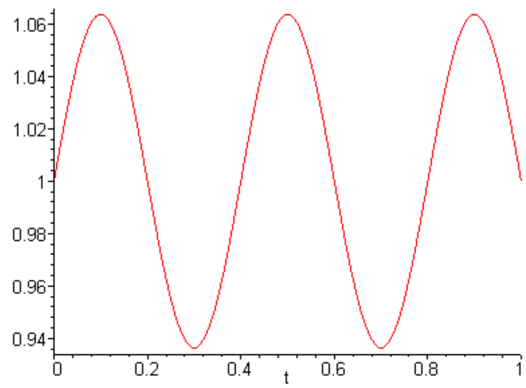
where $t_n < \bar{t}_n < t_{n+1}$. Since $\phi'(t) = 2t + exp[-t\,\phi(t)]$, it follows that

$$\begin{aligned}\phi''(t) &= 2 - 2[\phi(t) + t\,\phi'(t)] \cdot exp[-t\,\phi(t)] \\ &= 2 - \left\{\phi(t) + 2t^2 + t\,exp[-t\,\phi(t)]\right\} \cdot exp[-t\,\phi(t)].\end{aligned}$$

Hence

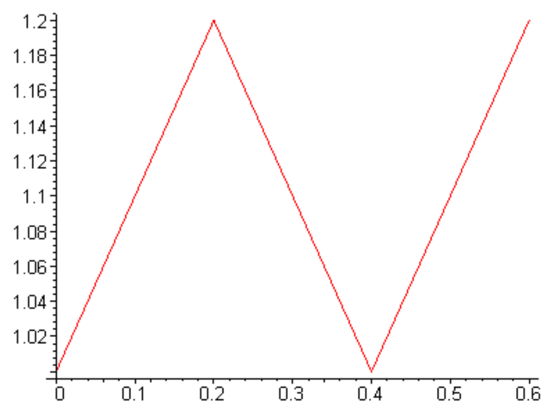
$$e_{n+1} = h^2 - \frac{h^2}{2}\left\{\phi(\bar{t}_n) + 2\,\bar{t}_n^2 + \bar{t}_n exp[-\bar{t}_n\phi(\bar{t}_n)]\right\} \cdot exp[-\,\bar{t}_n\phi(\bar{t}_n)].$$

22(a). Direct integration yields $\phi(t) = \frac{1}{5\pi} sin\,5\pi t + 1$.



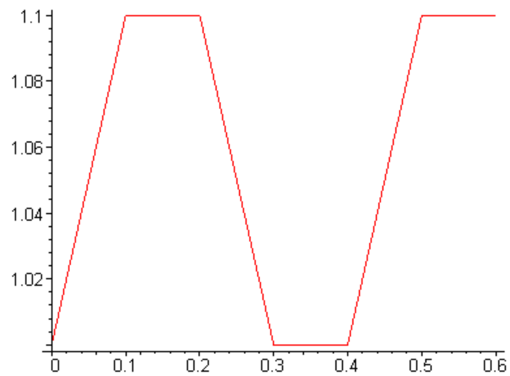
(b).

	$n=0$	$n=1$	$n=2$	$n=3$
t_n	0.0	0.2	0.4	0.6
y_n	1.0	1.2	1.0	1.2



(c).

	$n=0$	$n=2$	$n=4$	$n=6$
t_n	0.0	0.2	0.4	0.6
y_n	1.0	1.1	1.0	1.1



(d). Since $\phi''(t) = -5\pi \, sin\, 5\pi t$, the *local* truncation error for the Euler method, on the interval $t_n \le t \le t_{n+1}$, is given by

$$e_{n+1} = -\frac{5\pi h^2}{2} sin\, 5\pi\, \bar{t}_n \,.$$

In order to satisfy

$$|e_{n+1}| \le \frac{5\pi h^2}{2} < 0.05\,,$$

it is necessary that

$$h < \frac{1}{\sqrt{50\pi}} \approx 0.08\,.$$

$25(a)$. The Euler formula is

$$y_{n+1} = y_n + h(1 - t_n + 4\,y_n\,).$$

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.55	2.34	3.46	5.07

(b). The Euler formula for this problem is

$$y_{n+1} = y_n + h(3 + t_n - y_n\,).$$

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.20	1.39	1.57	1.74

(c). The Euler formula is

$$y_{n+1} = y_n + h(2\,y_n - 3\,t_n).$$

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.20	1.42	1.65	1.90

26(a).

$$1000\cdot\begin{vmatrix} 6.0 & 18 \\ 2.0 & 6.0 \end{vmatrix} = 1000\cdot(0) = 0\,.$$

(b).

$$1000\cdot\begin{vmatrix} 6.01 & 18.0 \\ 2.00 & 6.00 \end{vmatrix} = 1000(0.06) = 60\,.$$

(c).

$$1000\cdot\begin{vmatrix} 6.010 & 18.04 \\ 2.004 & 6.000 \end{vmatrix} = 1000(\,-0.09216) = \,-92.16\,.$$

27. Rounding to *three* digits, $a(b-c) \approx 0.224$. Likewise, to *three* digits, $ab \approx 0.702$ and $ac \approx 0.477$. It follows that $ab - ac \approx 0.225$.

Section 8.2

1. The improved Euler formula for this problem is

$$y_{n+1} = y_n + h\left(3 + \frac{1}{2}t_n + \frac{1}{2}t_{n+1} - y_n\right) - \frac{h^2}{2}(3 + t_n - y_n).$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h(3 - y_n) + \frac{h^2}{2}(y_n - 2 + 2n) - \frac{nh^3}{2} \ ,$$

with $y_0 = 1$.

(a). $h = 0.05$:

	$n = 2$	$n = 4$	$n = 6$	$n = 8$
t_n	0.1	0.2	0.3	0.4
y_n	1.19512	1.38120	1.55909	1.72956

(b). $h = 0.025$:

	$n = 4$	$n = 8$	$n = 12$	$n = 16$
t_n	0.1	0.2	0.3	0.4
y_n	1.19515	1.38125	1.55916	1.72965

(c). $h = 0.0125$:

	$n = 8$	$n = 16$	$n = 24$	$n = 32$
t_n	0.1	0.2	0.3	0.4
y_n	1.19516	1.38126	1.55918	1.72967

2. The improved Euler formula is

$$y_{n+1} = y_n + \frac{h}{2}\big(5\,t_n - 3\sqrt{y_n}\,\big) + \frac{h}{2}\Big(5\,t_{n+1} - 3\sqrt{K_n}\,\Big),$$

in which $K_n = y_n + h\big(5\,t_n - 3\sqrt{y_n}\,\big)$. Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \frac{h}{2}\big(5\,nh - 3\sqrt{y_n}\,\big) + \frac{h}{2}\Big[5\,(n+1)h - 3\sqrt{K_n}\Big],$$

with $y_0 = 2$.

(a). $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.62283	1.33460	1.12820	0.995445

(b). $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.62243	1.33386	1.12718	0.994215

(c). $h = 0.0125$:

	$n=8$	$n=16$	$n=24$	$n=32$
t_n	0.1	0.2	0.3	0.4
y_n	1.62234	1.33368	1.12693	0.993921

3. The improved Euler formula for this problem is

$$y_{n+1} = y_n + \frac{h}{2}(4\,y_n - 3\,t_n - 3\,t_{n+1}) + h^2(2\,y_n - 3\,t_n).$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + 2h\,y_n + \frac{h^2}{2}(4\,y_n - 3 - 6n) - 3nh^3\ ,$$

with $y_0 = 1$.

(a). $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.20526	1.42273	1.65511	1.90570

(b). $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	1.20533	1.42290	1.65542	1.90621

(c). $h = 0.0125$:

	$n=8$	$n=16$	$n=24$	$n=32$
t_n	0.1	0.2	0.3	0.4
y_n	1.20534	1.42294	1.65550	1.90634

5. The improved Euler formula is

$$y_{n+1} = y_n + h\,\frac{y_n^2 + 2\,t_n\,y_n}{2(3+t_n^2)} + h\,\frac{K_n^2 + 2\,t_{n+1}K_n}{2\big(3+t_{n+1}^2\big)},$$

in which

$$K_n = y_n + h\,\frac{y_n^2 + 2\,t_n\,y_n}{3+t_n^2}.$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h\,\frac{y_n^2 + 2nh\,y_n}{2(3+n^2h^2)} + h\,\frac{K_n^2 + 2(n+1)hK_n}{2\big[3+(n+1)^2h^2\big]},$$

with $y_0 = 0.5$.

(a). $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	0.510164	0.524126	0.54083	0.564251

(b). $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	0.510168	0.524135	0.542100	0.564277

(c). $h = 0.0125$:

	$n=8$	$n=16$	$n=24$	$n=32$
t_n	0.1	0.2	0.3	0.4
y_n	0.51069	0.524137	0.542104	0.564284

6. The improved Euler formula for this problem is

$$y_{n+1} = y_n + \frac{h}{2}\big(t_n^2 - y_n^2\big)sin\,y_n + \frac{h}{2}\big(t_{n+1}^2 - K_n^2\big)sin\,K_n,$$

in which

$$K_n = y_n + h\left(t_n^2 - y_n^2\right) sin\, y_n \,.$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \frac{h}{2}\left(n^2h^2 - y_n^2\right) sin\, y_n + \frac{h}{2}\left[(n+1)^2h^2 - K_n^2\right] sin\, K_n,$$

with $y_0 = -1$.

(a). $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	-0.924650	-0.864338	-0.816642	-0.780008

(b). $h = 0.025$:

	$n=4$	$n=8$	$n=12$	$n=16$
t_n	0.1	0.2	0.3	0.4
y_n	-0.924550	-0.864177	-0.816442	-0.779781

(c). $h = 0.0125$:

	$n=8$	$n=16$	$n=24$	$n=32$
t_n	0.1	0.2	0.3	0.4
y_n	-0.924525	-0.864138	-0.816393	-0.779725

7. The improved Euler formula for this problem is

$$y_{n+1} = y_n + \frac{h}{2}(4\,y_n - t_n - t_{n+1} + 1) + h^2(2\,y_n - t_n + 0.5).$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h(2\,y_n + 0.5) + h^2(2\,y_n - n) - nh^3\,,$$

with $y_0 = 1$.

(a). $h = 0.025$:

	$n=20$	$n=40$	$n=60$	$n=80$
t_n	0.5	1.0	1.5	2.0
y_n	2.96719	7.88313	20.8114	55.5106

(b). $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	2.96800	7.88755	20.8294	55.5758

8. The improved Euler formula is

$$y_{n+1} = y_n + \frac{h}{2}\left(5\,t_n - 3\sqrt{y_n}\,\right) + \frac{h}{2}\left(5\,t_{n+1} - 3\sqrt{K_n}\,\right),$$

in which $K_n = y_n + h\left(5\,t_n - 3\sqrt{y_n}\,\right)$. Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \frac{h}{2}\left(5\,nh - 3\sqrt{y_n}\,\right) + \frac{h}{2}\left[5\,(n+1)h - 3\sqrt{K_n}\right],$$

with $y_0 = 2$.

(a). $h = 0.025$:

	$n=20$	$n=40$	$n=60$	$n=80$
t_n	0.5	1.0	1.5	2.0
y_n	0.926139	1.28558	2.40898	4.10386

(b). $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	0.925815	1.28525	2.40869	4.10359

9. The improved Euler formula for this problem is

$$y_{n+1} = y_n + \frac{h}{2}\sqrt{t_n + y_n} + \frac{h}{2}\sqrt{t_{n+1} + K_n}\,,$$

in which $K_n = y_n + h\sqrt{t_n + y_n}$. Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \frac{h}{2}\sqrt{nh + y_n} + \frac{h}{2}\sqrt{(n+1)h + K_n}\,,$$

with $y_0 = 3$.

(a). $h = 0.025$:

	$n=20$	$n=40$	$n=60$	$n=80$
t_n	0.5	1.0	1.5	2.0
y_n	3.96217	5.10887	6.43134	7.92332

(b). $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	3.96218	5.10889	6.43138	7.92337

10. The improved Euler formula is

$$y_{n+1} = y_n + \frac{h}{2}[2\,t_n + exp(-\,t_n\,y_n)] + \frac{h}{2}[2\,t_{n+1} + exp(-\,t_{n+1}K_n)],$$

in which $K_n = y_n + h[2\,t_n + exp(-\,t_n\,y_n)]$. Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + \frac{h}{2}[2\,nh + exp(-\,nh\,y_n)] + \frac{h}{2}\{2(n+1)h + exp[-\,(n+1)hK_n]\},$$

with $y_0 = 1$.

(a). $h = 0.025$:

	$n=20$	$n=40$	$n=60$	$n=80$
t_n	0.5	1.0	1.5	2.0
y_n	1.61263	2.48097	3.74556	5.49595

(b). $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	1.61263	2.48092	3.74550	5.49589

12. The improved Euler formula is

$$y_{n+1} = y_n + h\,\frac{y_n^2 + 2\,t_n\,y_n}{2(3 + t_n^2)} + h\,\frac{K_n^2 + 2\,t_{n+1}K_n}{2\big(3 + t_{n+1}^2\big)},$$

in which

$$K_n = y_n + h\,\frac{y_n^2 + 2\,t_n\,y_n}{3 + t_n^2}.$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h\,\frac{y_n^2 + 2nh\,y_n}{2(3+n^2h^2)} + h\,\frac{K_n^2 + 2(n+1)hK_n}{2\big[3+(n+1)^2h^2\big]},$$

with $y_0 = 0.5$.

(a). $h = 0.025$:

	$n=20$	$n=40$	$n=60$	$n=80$
t_n	0.5	1.0	1.5	2.0
y_n	0.590897	0.799950	1.16653	1.74969

(b). $h = 0.0125$:

	$n=40$	$n=80$	$n=120$	$n=160$
t_n	0.5	1.0	1.5	2.0
y_n	0.590906	0.799988	1.16663	1.74992

16. The exact solution of the initial value problem is $\phi(t) = \frac{1}{2} + \frac{1}{2}e^{2t}$. Based on the result in Prob. $14(c)$, the local truncation error for a *linear* differential equation is

$$e_{n+1} = \frac{1}{6}\phi'''(\bar{t}_n)h^3,$$

where $t_n < \bar{t}_n < t_{n+1}$. Since $\phi'''(t) = 4\,e^{2t}$, the local truncation error is

$$e_{n+1} = \frac{2}{3}exp(2\,\bar{t}_n)h^3.$$

Furthermore, with $0 \le \bar{t}_n \le 1$,

$$|e_{n+1}| \le \frac{2}{3}\,e^2h^3.$$

It also follows that for $h = 0.1$,

$$|e_1| \le \frac{2}{3}\,e^{0.2}(0.1)^3 = \frac{1}{1500}\,e^{0.2}.$$

Using the improved Euler method, with $h = 0.1$, we have $y_1 \approx 1.11000$. The exact value is given by $\phi(0.1) = 1.1107014$.

17. The exact solution of the initial value problem is given by $\phi(t) = \frac{1}{2}t + e^{2t}$. Using the modified Euler method, the local truncation error for a *linear* differential equation is

$$e_{n+1} = \frac{1}{6}\phi'''(\bar{t}_n)h^3,$$

where $t_n < \bar{t}_n < t_{n+1}$. Since $\phi'''(t) = 8\,e^{2t}$, the local truncation error is

$$e_{n+1} = \frac{4}{3} exp(2\,\bar{t}_n)h^3.$$

Furthermore, with $0 \leq \bar{t}_n \leq 1$, the *local* error is bounded by

$$|e_{n+1}| \leq \frac{4}{3}\,e^2 h^3.$$

It also follows that for $h = 0.1$,

$$|e_1| \leq \frac{4}{3}\,e^{0.2}(0.1)^3 = \frac{1}{750}\,e^{0.2}.$$

Using the improved Euler method, with $h = 0.1$, we have $y_1 \approx 1.27000$. The exact value is given by $\phi(0.1) = 1.271403$.

18. Using the *Euler method*,

$$\begin{aligned} y_1 &= 1 + 0.1(0.5 - 0 + 2 \cdot 1) \\ &= 1.25\,. \end{aligned}$$

Using the *improved Euler method*,

$$\begin{aligned} y_1 &= 1 + 0.05(0.5 - 0 + 2 \cdot 1) + 0.05(0.5 - 0.1 + 2 \cdot 1.25) \\ &= 1.27\,. \end{aligned}$$

The estimated error is $e_1 \approx 1.27 - 1.25 = 0.02$. The step size should be adjusted by a factor of $\sqrt{0.0025/0.02} \approx 0.354$. Hence the required step size is estimated as

$$h \approx (0.1)(0.36) = 0.036\,.$$

20. Using the *Euler method*,

$$\begin{aligned} y_1 &= 3 + 0.1\sqrt{0+3} \\ &= 3.173205\,. \end{aligned}$$

Using the *improved Euler method*,

$$\begin{aligned} y_1 &= 3 + 0.05\sqrt{0+3} + 0.05\sqrt{0.1 + 3.173205} \\ &= 3.177063\,. \end{aligned}$$

The estimated error is $e_1 \approx 3.177063 - 3.173205 = 0.003858$. The step size should be adjusted by a factor of $\sqrt{0.0025/0.003858} \approx 0.805$. Hence the required step size is estimated as

$$h \approx (0.1)(0.805) = 0.0805\,.$$

21. Using the *Euler method*,

$$\begin{aligned} y_1 &= 0.5 + 0.1\frac{(0.5)^2 + 0}{3+0} \\ &= 0.508334 \end{aligned}$$

Using the *improved Euler method*,

$$\begin{aligned} y_1 &= 0.5 + 0.05\frac{(0.5)^2 + 0}{3+0} + 0.05\frac{(0.508334)^2 + 2(0.1)(0.508334)}{3 + (0.1)^2} \\ &= 0.510148\,. \end{aligned}$$

The estimated error is $e_1 \approx 0.510148 - 0.508334 = 0.0018$. The local truncation error is *less* than the given tolerance. The step size can be adjusted by a factor of $\sqrt{0.0025/0.0018} \approx 1.1785$. Hence it is possible to use a step size of

$$h \approx (0.1)(1.1785) \approx 0.117\,.$$

22. Assuming that the solution has continuous derivatives at least to the third order,

$$\phi(t_{n+1}) = \phi(t_n) + \phi'(t_n)h + \frac{\phi''(t_n)}{2!}h^2 + \frac{\phi'''(\bar{t}_n)}{3!}h^3,$$

where $t_n < \bar{t}_n < t_{n+1}$. Suppose that $y_n = \phi(t_n)$.

(a). The local truncation error is given by

$$e_{n+1} = \phi(t_{n+1}) - y_{n+1}.$$

The *modified Euler formula* is defined as

$$y_{n+1} = y_n + h\,f\left[t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}h\,f(t_n, y_n)\right].$$

Observe that $\phi'(t_n) = f(t_n\,, \phi(t_n)) = f(t_n\,, y_n)$. It follows that

$$\begin{aligned} e_{n+1} &= \phi(t_{n+1}) - y_{n+1} \\ &= h\,f(t_n\,, y_n) + \frac{\phi''(t_n)}{2!}h^2 + \frac{\phi'''(\bar{t}_n)}{3!}h^3 - \\ &- h\,f\left[t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}h\,f(t_n, y_n)\right]. \end{aligned}$$

(b). As shown in Prob. $14(b)$,

$$\phi''(t_n) = f_t(t_n\,, y_n) + f_y(t_n\,, y_n)f(t_n\,, y_n)\,.$$

Furthermore,

$$\begin{aligned} f\left[t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}h\,f(t_n, y_n)\right] &= f(t_n\,, y_n) + f_t(t_n\,, y_n)\frac{h}{2} + f_y(t_n\,, y_n)\,k + \\ &+ \frac{1}{2!}\left[\frac{h^2}{4}f_{tt} + hk\,f_{ty} + k^2 f_{yy}\right]_{t=\xi\,, y=\eta}, \end{aligned}$$

in which $k = \frac{1}{2}h\,f(t_n, y_n)$ and $t_n < \xi < t_n + h/2$, $y_n < \eta < y_n + k$. Therefore

$$e_{n+1} = \frac{\phi'''(\bar{t}_n)}{3!}h^3 - \frac{h}{2!}\left[\frac{h^2}{4}f_{tt} + hk\,f_{ty} + k^2 f_{yy}\right]_{t=\xi\,, y=\eta}.$$

Note that each term in the brackets has a factor of h^2. Hence the local truncation error is *proportional* to h^3.

(c). If $f(t, y)$ is *linear*, then $f_{tt} = f_{ty} = f_{yy} = 0$, and

$$e_{n+1} = \frac{\phi'''(\bar{t}_n)}{3!}h^3.$$

23. The *modified* Euler formula for this problem is

$$\begin{aligned} y_{n+1} &= y_n + h\left\{3 + t_n + \frac{1}{2}h - \left[y_n + \frac{1}{2}h(3 + t_n - y_n)\right]\right\} \\ &= y_n + h(3 + t_n - y_n) + \frac{h^2}{2}(y_n - t_n - 2). \end{aligned}$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h(3 + nh - y_n) + \frac{h^2}{2}(y_n - nh - 2)\;,$$

with $y_0 = 1$. Setting $h = 0.1$, we obtain the following values :

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	1.19500	1.38098	1.55878	1.72920

25. The *modified* Euler formula is

$$\begin{aligned} y_{n+1} &= y_n + h\left[2y_n - 3t_n - \frac{3}{2}h + h(2y_n - 3t_n)\right] \\ &= y_n + h(2y_n - 3t_n) + \frac{h^2}{2}(4y_n - 6t_n - 3). \end{aligned}$$

Since $t_n = t_0 + nh$ and $t_0 = 0$, we can also write

$$y_{n+1} = y_n + h(2y_n - 3nh) + \frac{h^2}{2}(4y_n - 6\,nh - 3)\ ,$$

with $y_0 = 1$. Setting $h = 0.1$, we obtain :

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	1.20500	1.42210	1.65396	1.90383

26. The *modified* Euler formula for this problem is

$$y_{n+1} = y_n + h\left\{2t_n + h + exp\left[-\left(t_n + \frac{h}{2}\right)K_n\right]\right\},$$

in which $K_n = y_n + \frac{h}{2}[2t_n + exp(-t_n y_n)]$. Now $t_n = t_0 + nh$, with $t_0 = 0$ and $y_0 = 1$. Setting $h = 0.1$, we obtain the following values :

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	1.104885	1.21892	1.34157	1.472724

27. Let $f(t, y)$ be *linear* in both variables. The *improved* Euler formula is

$$\begin{aligned} y_{n+1} &= y_n + \frac{1}{2}h[f(t_n, y_n) + f(t_n + h, y_n + hf(t_n, y_n))] \\ &= y_n + \frac{1}{2}hf(t_n, y_n) + \frac{1}{2}hf(t_n, y_n) + \frac{1}{2}hf[h, hf(t_n, y_n)] \\ &= hf(h, y_n) + \frac{1}{2}hf[h, hf(t_n, y_n)]. \end{aligned}$$

The *modified* Euler formula is

$$\begin{aligned} y_{n+1} &= y_n + hf\left[t_n + \frac{1}{2}h, y_n + \frac{1}{2}hf(t_n, y_n)\right] \\ &= y_n + hf(t_n, y_n) + hf\left[\frac{1}{2}h, \frac{1}{2}hf(t_n, y_n)\right]. \end{aligned}$$

Since $f(t, y)$ is *linear* in both variables,

$$f\left[\frac{1}{2}h, \frac{1}{2}hf(t_n, y_n)\right] = \frac{1}{2}f[h, hf(t_n, y_n)].$$

Section 8.3

1. The ODE is linear, with $f(t,y) = 3 + t - y$. The Runge-Kutta algorithm requires the evaluations

$$\begin{aligned}
k_{n1} &= f(t_n, y_n) \\
k_{n2} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n1}\right) \\
k_{n3} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n2}\right) \\
k_{n4} &= f(t_n + h\,, y_n + hk_{n3}).
\end{aligned}$$

The next estimate is given as the weighted average

$$y_{n+1} = y_n + \frac{h}{6}(k_{n1} + 2\,k_{n2} + 2\,k_{n3} + k_{n4}).$$

(a). For $h = 0.1$:

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	1.19516	1.38127	1.55918	1.72968

(b). For $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.19516	1.38127	1.55918	1.72968

The exact solution of the IVP is $y(t) = 2 + t - e^{-t}$.

2. In this problem, $f(t,y) = 5t - 3\sqrt{y}$. At each time step, the Runge-Kutta algorithm requires the evaluations

$$\begin{aligned}
k_{n1} &= f(t_n, y_n) \\
k_{n2} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n1}\right) \\
k_{n3} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n2}\right) \\
k_{n4} &= f(t_n + h\,, y_n + hk_{n3}).
\end{aligned}$$

The next estimate is given as the weighted average

$$y_{n+1} = y_n + \frac{h}{6}(k_{n1} + 2\,k_{n2} + 2\,k_{n3} + k_{n4}).$$

(a). For $h = 0.1$:

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	1.62231	1.33362	1.12686	0.993839

(b). For $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.62230	1.33362	1.12685	0.993826

The exact solution of the IVP is given *implicitly* as

$$\frac{1}{\left(2\sqrt{y}+5t\right)^5\left(t-\sqrt{y}\right)^2} = \frac{\sqrt{2}}{512}\,.$$

3. The ODE is linear, with $f(t, y) = 2y - 3t$. The Runge-Kutta algorithm requires the evaluations

$$\begin{aligned}
k_{n1} &= f(t_n, y_n) \\
k_{n2} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n1}\right) \\
k_{n3} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n2}\right) \\
k_{n4} &= f(t_n + h\,, y_n + hk_{n3}).
\end{aligned}$$

The next estimate is given as the weighted average

$$y_{n+1} = y_n + \frac{h}{6}(k_{n1} + 2\,k_{n2} + 2\,k_{n3} + k_{n4}).$$

(a). For $h = 0.1$:

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	1.20535	1.42295	1.65553	1.90638

(b). For $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	1.20535	1.42296	1.65553	1.90638

The exact solution of the IVP is $y(t) = e^{2t}/4 + 3t/2 + 3/4$.

5. In this problem, $f(t, y) = (y^2 + 2ty)/(3 + t^2)$. The Runge-Kutta algorithm

requires the evaluations

$$\begin{aligned}
k_{n1} &= f(t_n, y_n) \\
k_{n2} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n1}\right) \\
k_{n3} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n2}\right) \\
k_{n4} &= f(t_n + h\,, y_n + hk_{n3}).
\end{aligned}$$

The next estimate is given as the weighted average

$$y_{n+1} = y_n + \frac{h}{6}(k_{n1} + 2\,k_{n2} + 2\,k_{n3} + k_{n4}).$$

(a). For $h = 0.1$:

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	0.510170	0.524138	0.542105	0.564286

(b). For $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	0.520169	0.524138	0.542105	0.564286

The exact solution of the IVP is $y(t) = (3 + t^2)/(6 - t)$.

6. In this problem, $f(t, y) = (t^2 - y^2) sin\, y$. At each time step, the Runge-Kutta algorithm requires the evaluations

$$\begin{aligned}
k_{n1} &= f(t_n, y_n) \\
k_{n2} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n1}\right) \\
k_{n3} &= f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n2}\right) \\
k_{n4} &= f(t_n + h\,, y_n + hk_{n3}).
\end{aligned}$$

The next estimate is given as the weighted average

$$y_{n+1} = y_n + \frac{h}{6}(k_{n1} + 2\,k_{n2} + 2\,k_{n3} + k_{n4}).$$

(a). For $h = 0.1$:

	$n=1$	$n=2$	$n=3$	$n=4$
t_n	0.1	0.2	0.3	0.4
y_n	-0.924517	-0.864125	-0.816377	-0.779706

(b). For $h = 0.05$:

	$n=2$	$n=4$	$n=6$	$n=8$
t_n	0.1	0.2	0.3	0.4
y_n	-0.924517	-0.864125	-0.816377	-0.779706

7. (a). For $h = 0.1$:

	$n=5$	$n=10$	$n=15$	$n=20$
t_n	0.5	1.0	1.5	2.0
y_n	2.96825	7.88889	20.8349	55.5957

(b). For $h = 0.05$:

	$n=10$	$n=20$	$n=30$	$n=40$
t_n	0.5	1.0	1.5	2.0
y_n	2.96828	7.88904	20.8355	55.5980

The exact solution of the IVP is $y(t) = e^{2t} + t/2$.

8. See Prob. 2 . for the *exact* solution.

(a). For $h = 0.1$:

	$n=5$	$n=10$	$n=15$	$n=20$
t_n	0.5	1.0	1.5	2.0
y_n	0.925725	1.28516	2.40860	4.10350

(b). For $h = 0.05$:

	$n=10$	$n=20$	$n=30$	$n=40$
t_n	0.5	1.0	1.5	2.0
y_n	0.925711	1.28515	2.40860	4.10350

9(a). For $h = 0.1$:

	$n = 5$	$n = 10$	$n = 15$	$n = 20$
t_n	0.5	1.0	1.5	2.0
y_n	3.96219	5.10890	6.43139	7.92338

(b). For $h = 0.05$:

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	3.96219	5.10890	6.43139	7.92338

The exact solution is given *implicitly* as

$$ln\left[\frac{2}{y+t-1}\right] + 2\sqrt{t+y} - 2\,arctanh\,\sqrt{t+y} = t + 2\sqrt{3} - 2\,arctanh\,\sqrt{3}\,.$$

10. See Prob. 4 .

(a). For $h = 0.1$:

	$n = 5$	$n = 10$	$n = 15$	$n = 20$
t_n	0.5	1.0	1.5	2.0
y_n	1.61262	2.48091	3.74548	5.49587

(b). For $h = 0.05$:

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	1.61262	2.48091	3.74548	5.49587

12. See Prob. 5 . for the *exact* solution.

(a). For $h = 0.1$:

	$n = 5$	$n = 10$	$n = 15$	$n = 20$
t_n	0.5	1.0	1.5	2.0
y_n	0.590909	0.800000	1.166667	1.75000

(b). For $h = 0.05$:

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	0.590909	0.800000	1.166667	1.75000

13. The ODE is linear, with $f(t, y) = 1 - t + 4y$. The Runge-Kutta algorithm requires

the evaluations

$$k_{n1} = f(t_n, y_n)$$
$$k_{n2} = f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n1}\right)$$
$$k_{n3} = f\left(t_n + \frac{1}{2}h\,, y_n + \frac{1}{2}hk_{n2}\right)$$
$$k_{n4} = f(t_n + h\,, y_n + hk_{n3}).$$

The next estimate is given as the weighted average

$$y_{n+1} = y_n + \frac{h}{6}(k_{n1} + 2\,k_{n2} + 2\,k_{n3} + k_{n4}).$$

The exact solution of the IVP is $y(t) = \frac{19}{16}e^{4t} + \frac{1}{4}t - \frac{3}{16}$.

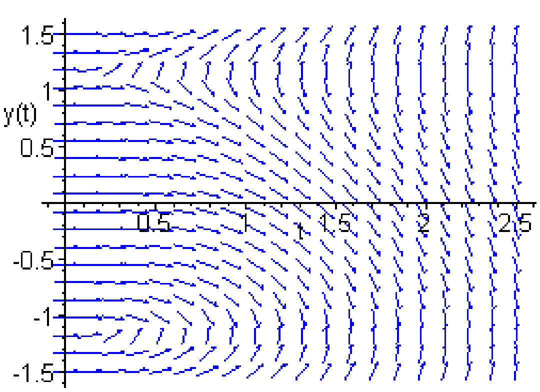
(a). For $h = 0.1$:

	$n = 5$	$n = 10$	$n = 15$	$n = 20$
t_n	0.5	1.0	1.5	2.0
y_n	8.7093175	64.858107	478.81928	3535.8667

(b). For $h = 0.05$:

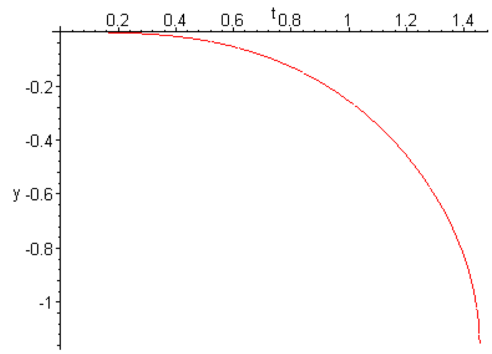
	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	8.7118060	64.894875	479.22674	3539.8804

$15(a)$.



(b). For the integral curve starting at $(0,0)$, the slope becomes *infinite* near $t_M \approx 1.5$. Note that the exact solution of the IVP is defined implicitly as

$$y^3 - 4y = t^3.$$



Using the classic Runge-Kutta algorithm, with $h = 0.01$, we obtain the values

	$n=70$	$n=80$	$n=90$	$n=95$
t_n	0.7	0.8	0.9	0.95
y_n	-0.08591	-0.12853	-0.18380	-0.21689

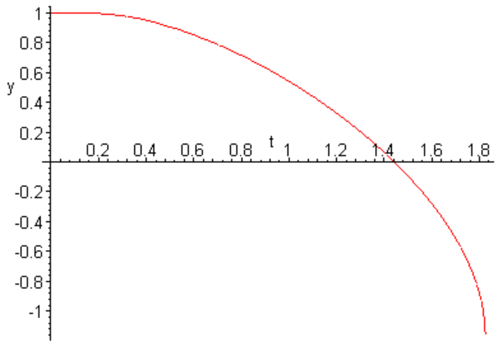
(c). Based on the direction field, the solution should *decrease* monotonically to the limiting value $y = -2/\sqrt{3}$. In the following table, the value of t_M corresponds to the approximate time in the iteration process that the calculated values begin to *increase*.

h	t_M
0.1	1.9
0.05	1.65
0.025	1.55
0.01	1.455

(d). Numerical values will continue to be generated, although they will *not* be associated with the integral curve starting at $(0,0)$. These values are approximations to nearby integral curves.

(e). We consider the solution associated with the initial condition $y(0) = 1$. The exact solution is given by

$$y^3 - 4y = t^3 - 3.$$



For the integral curve starting at $(0\,,1)$, the slope becomes *infinite* near $t_M \approx 2.0$. In the following table, the values of t_M corresponds to the approximate time in the iteration process that the calculated values begin to *increase*.

h	t_M
0.1	1.85
0.05	1.85
0.025	1.86
0.01	1.835

Section 8.4

$1(a)$. Using the notation $f_n = f(t_n\,, y_n)$, the *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\ .$$

We use the starting values generated by the Runge-Kutta method :

	$n=0$	$n=1$	$n=2$	$n=3$
t_n	0.0	0.1	0.2	0.3
y_n	1.0	1.19516	1.38127	1.55918

	$n=4(pre)$	$n=4(cor)$	$n=5(pre)$	$n=5(cor)$
t_n	0.4	0.4	0.5	0.5
y_n	1.72967690	1.72986801	1.89346436	1.89346973

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\ .$$

In this problem, $f_{n+1} = 3 + t_{n+1} - y_{n+1}$. Since the ODE is *linear*, we can solve for

$$y_{n+1} = \frac{1}{24 + 9\,h}[24\,y_n + 27\,h + 9h\,t_{n+1} + h(19\,f_n - 5\,f_{n-1} + f_{n-2})]\ .$$

	$n=4$	$n=5$
t_n	0.4	0.5
y_n	1.7296800	1.8934695

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\,f_{n+1}].$$

In this problem, $f_{n+1} = 3 + t_{n+1} - y_{n+1}$. Since the ODE is *linear*, we can solve for

$$y_{n+1} = \frac{1}{25 + 12\,h}[36\,h + 12h\,t_{n+1} + 48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3}]\ .$$

	$n=4$	$n=5$
t_n	0.4	0.5
y_n	1.7296805	1.8934711

The exact solution of the IVP is $y(t) = 2 + t - e^{-t}$.

$2(a)$. Using the notation $f_n = f(t_n\,, y_n)$, the *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\;.$$

We use the starting values generated by the Runge-Kutta method :

	$n=0$	$n=1$	$n=2$	$n=3$
t_n	0.0	0.1	0.2	0.3
y_n	2.0	1.62231	1.33362	1.12686

	$n=4(pre)$	$n=4(cor)$	$n=5(pre)$	$n=5(cor)$
t_n	0.4	0.4	0.5	0.5
y_n	0.993751	0.993852	0.925469	0.925764

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\;.$$

In this problem, $f_{n+1} = 5t_{n+1} - 3\sqrt{y_{n+1}}$. Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\left[\,45t_{n+1} - 27\sqrt{y_{n+1}} + 19\,f_n - 5\,f_{n-1} + f_{n-2}\right]$$

at each time step. We obtain the approximate values:

	$n=4$	$n=5$
t_n	0.4	0.5
y_n	0.993847	0.925746

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h f_{n+1}].$$

Since the ODE is *nonlinear*, an equation solver is used to approximate the solution of

$$y_{n+1} = \frac{1}{25}\left[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\left(5t_{n+1} - 3\sqrt{y_{n+1}}\,\right)\right]$$

at each time step.

	$n = 4$	$n = 5$
t_n	0.4	0.5
y_n	0.993869	0.925837

The exact solution of the IVP is given *implicitly* by

$$\frac{1}{\left(2\sqrt{y} + 5t\right)^5\left(t - \sqrt{y}\right)^2} = \frac{\sqrt{2}}{512}\,.$$

$3(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\,.$$

Using the starting values generated by the Runge-Kutta method :

	$n = 0$	$n = 1$	$n = 2$	$n = 3$
t_n	0.0	0.1	0.2	0.3
y_n	1.0	1.205350	1.422954	1.655527

	$n = 4(pre)$	$n = 4(cor)$	$n = 5(pre)$	$n = 5(cor)$
t_n	0.4	0.4	0.5	0.5
y_n	1.906340	1.906382	2.179455	2.179567

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\,.$$

In this problem, $f_{n+1} = 2\,y_{n+1} - 3\,t_{n+1}$. Since the ODE is *linear*, we can solve for

$$y_{n+1} = \frac{1}{24 - 18\,h}[24\,y_n - 27h\,t_{n+1} + h(19\,f_n - 5\,f_{n-1} + f_{n-2})]\ .$$

	$n = 4$	$n = 5$
t_n	0.4	0.5
y_n	1.906385	2.179576

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h f_{n+1}].$$

In this problem, $f_{n+1} = 2\,y_{n+1} - 3\,t_{n+1}$. Since the ODE is *linear*, we can solve for

$$y_{n+1} = \frac{1}{25 - 24\,h}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} - 36h\,t_{n+1}]\ .$$

	$n = 4$	$n = 5$
t_n	0.4	0.5
y_n	1.906395	2.179611

The exact solution of the IVP is $y(t) = e^{2t}/4 + 3t/2 + 3/4$.

$5(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\ .$$

Using the starting values generated by the Runge-Kutta method :

	$n=0$	$n=1$	$n=2$	$n=3$
t_n	0.0	0.1	0.2	0.3
y_n	0.5	0.51016950	0.52413795	0.54210529

	$n=4(pre)$	$n=4(cor)$	$n=5(pre)$	$n=5(cor)$
t_n	0.4	0.4	0.5	0.5
y_n	0.56428532	0.56428577	0.59090816	0.59090918

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\;.$$

In this problem,

$$f_{n+1} = \frac{y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}}{3 + t_{n+1}^2}\;.$$

Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\left[9\,\frac{y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}}{3 + t_{n+1}^2} + 19\,f_n - 5\,f_{n-1} + f_{n-2}\right]$$

at each time step.

	$n=4$	$n=5$
t_n	0.4	0.5
y_n	0.56428578	0.59090920

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h f_{n+1}].$$

Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = \frac{1}{25}\left[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\frac{y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}}{3 + t_{n+1}^2}\right]$$

at each time step. We obtain the approximate values:

	$n=4$	$n=5$
t_n	0.4	0.5
y_n	0.56428588	0.59090952

The exact solution of the IVP is $y(t) = (3 + t^2)/(6 - t)$.

$6(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\, f_n - 59\, f_{n-1} + 37\, f_{n-2} - 9\, f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\, f_{n+1} + 19\, f_n - 5\, f_{n-1} + f_{n-2})\;.$$

We use the starting values generated by the Runge-Kutta method :

	$n = 0$	$n = 1$	$n = 2$	$n = 3$
t_n	0.0	0.1	0.2	0.3
y_n	− 1.0	− 0.924517	− 0.864125	− 0.816377

	$n = 4(pre)$	$n = 4(cor)$	$n = 5(pre)$	$n = 5(cor)$
t_n	0.4	0.4	0.5	0.5
y_n	− 0.779832	− 0.779693	− 0.753311	− 0.753135

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\, f_{n+1} + 19\, f_n - 5\, f_{n-1} + f_{n-2})\;.$$

In this problem, $f_{n+1} = (t_{n+1}^2 - y_{n+1}^2) sin\, y_{n+1}$. Since the ODE is *nonlinear*, we obtain the *implicit* equation

$$y_{n+1} = y_n + \frac{h}{24}\left[\,9\left(t_{n+1}^2 - y_{n+1}^2\right) sin\, y_{n+1} \;+ 19\, f_n - 5\, f_{n-1} + f_{n-2}\right].$$

	$n = 4$	$n = 5$
t_n	0.4	0.5
y_n	− 0.779700	− 0.753144

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\,f_{n+1}].$$

Since the ODE is *nonlinear*, we obtain the *implicit* equation

$$y_{n+1} = \frac{1}{25}\left[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\left(t_{n+1}^2 - y_{n+1}^2\right) sin\, y_{n+1}\right].$$

	$n = 4$	$n = 5$
t_n	0.4	0.5
y_n	$-\,0.779680$	$-\,0.753089$

$8(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\ .$$

We use the starting values generated by the Runge-Kutta method :

	$n = 0$	$n = 1$	$n = 2$	$n = 3$
t_n	0.0	0.05	0.1	0.15
y_n	2.0	1.7996296	1.6223042	1.4672503

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	0.9257133	1.285148	2.408595	4.103495

(b). Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\left[\,45t_{n+1} - 27\sqrt{y_{n+1}}\ + 19\,f_n - 5\,f_{n-1} + f_{n-2}\right]$$

at each time step. We obtain the approximate values:

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	0.9257125	1.285148	2.408595	4.103495

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\, y_n - 36\, y_{n-1} + 16\, y_{n-2} - 3\, y_{n-3} + 12h f_{n+1}].$$

Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = \frac{1}{25}\left[48\, y_n - 36\, y_{n-1} + 16\, y_{n-2} - 3\, y_{n-3} + 12h\left(5t_{n+1} - 3\sqrt{y_{n+1}}\,\right)\right]$$

at each time step.

	$n=10$	$n=20$	$n=30$	$n=40$
t_n	0.5	1.0	1.5	2.0
y_n	0.9257248	1.285158	2.408594	4.103493

The exact solution of the IVP is given *implicitly* by

$$\frac{1}{\left(2\sqrt{y} + 5t\right)^5\left(t - \sqrt{y}\right)^2} = \frac{\sqrt{2}}{512}\,.$$

$9(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\, f_n - 59\, f_{n-1} + 37\, f_{n-2} - 9\, f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\, f_{n+1} + 19\, f_n - 5\, f_{n-1} + f_{n-2})\,.$$

Using the starting values generated by the Runge-Kutta method :

	$n=0$	$n=1$	$n=2$	$n=3$
t_n	0.0	0.05	0.1	0.15
y_n	3.0	3.087586	3.177127	3.268609

	$n=10$	$n=20$	$n=30$	$n=40$
t_n	0.5	1.0	1.5	2.0
y_n	3.962186	5.108903	6.431390	7.923385

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\, f_{n+1} + 19\, f_n - 5\, f_{n-1} + f_{n-2})\,.$$

In this problem, $f_{n+1} = \sqrt{t_{n+1} + y_{n+1}}$. Since the ODE is *nonlinear*, an equation solver must be implemented in order to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\left[9\sqrt{t_{n+1} + y_{n+1}} + 19\, f_n - 5\, f_{n-1} + f_{n-2}\right]$$

at each time step.

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	3.962186	5.108903	6.431390	7.923385

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\, y_n - 36\, y_{n-1} + 16\, y_{n-2} - 3\, y_{n-3} + 12h f_{n+1}].$$

Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = \frac{1}{25}\left[48\, y_n - 36\, y_{n-1} + 16\, y_{n-2} - 3\, y_{n-3} + 12h\sqrt{t_{n+1} + y_{n+1}}\right]$$

at each time step.

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	3.962186	5.108903	6.431390	7.923385

The exact solution is given *implicitly* by

$$ln\left[\frac{2}{y + t - 1}\right] + 2\sqrt{t + y} - 2\, arctanh\, \sqrt{t + y} = t + 2\sqrt{3} - 2\, arctanh\, \sqrt{3}\ .$$

$10(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\, f_n - 59\, f_{n-1} + 37\, f_{n-2} - 9\, f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\, f_{n+1} + 19\, f_n - 5\, f_{n-1} + f_{n-2})\,.$$

We use the starting values generated by the Runge-Kutta method :

	$n = 0$	$n = 1$	$n = 2$	$n = 3$
t_n	0.0	0.05	0.1	0.15
y_n	1.0	1.051230	1.104843	1.160740

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	1.612622	2.480909	3.7451479	5.495872

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\, f_{n+1} + 19\, f_n - 5\, f_{n-1} + f_{n-2})\,.$$

In this problem, $f_{n+1} = 2\, t_{n+1} + exp(-t_{n+1}\, y_{n+1})$. Since the ODE is $nonlinear$, an equation solver must be implemented in order to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\{9[2\, t_{n+1} + exp(-t_{n+1}\, y_{n+1})] + 19\, f_n - 5\, f_{n-1} + f_{n-2}\}$$

at each time step.

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	1.612622	2.480909	3.7451479	5.495872

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\, y_n - 36\, y_{n-1} + 16\, y_{n-2} - 3\, y_{n-3} + 12h f_{n+1}].$$

Since the ODE is $nonlinear$, we obtain the *implicit* equation

$$y_{n+1} = \frac{1}{25}\{48\, y_n - 36\, y_{n-1} + 16\, y_{n-2} - 3\, y_{n-3} + 12h[2\, t_{n+1} + exp(-t_{n+1}\, y_{n+1})]\}.$$

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	1.612623	2.480905	3.7451473	5.495869

$11(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\,.$$

Using the starting values generated by the Runge-Kutta method :

	$n = 0$	$n = 1$	$n = 2$	$n = 3$
t_n	0.0	0.05	0.1	0.15
y_n	− 2.0	− 1.958833	− 1.915221	− 1.868975

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	− 1.447639	− 0.1436281	1.060946	1.410122

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\,.$$

In this problem,

$$f_{n+1} = \frac{4 - t_{n+1}\,y_{n+1}}{1 + y_{n+1}^2}\,.$$

Since the differential equation is *nonlinear*, an equation solver is used to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\left[9\,\frac{4 - t_{n+1}\,y_{n+1}}{1 + y_{n+1}^2} + 19\,f_n - 5\,f_{n-1} + f_{n-2}\right]$$

at each time step.

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	− 1.447638	− 0.1436767	1.060913	1.410103

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12hf_{n+1}].$$

Since the ODE is *nonlinear*, an equation solver must be implemented in order to approximate the solution of

$$y_{n+1} = \frac{1}{25}\left[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\frac{4 - t_{n+1}\,y_{n+1}}{1 + y_{n+1}^2}\right]$$

at each time step.

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	-1.447621	-0.1447619	1.060717	1.410027

$12(a)$. The *predictor* formula is

$$y_{n+1} = y_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3}).$$

With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the *corrector* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\,.$$

We use the starting values generated by the Runge-Kutta method :

	$n = 0$	$n = 1$	$n = 2$	$n = 3$
t_n	0.0	0.05	0.1	0.15
y_n	0.5	0.5046218	0.5101695	0.5166666

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	0.5909091	0.8000000	1.166667	1.750000

(b). With $f_{n+1} = f(t_{n+1}\,, y_{n+1})$, the fourth order *Adams-Moulton* formula is

$$y_{n+1} = y_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})\,.$$

In this problem,

$$f_{n+1} = \frac{y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}}{3 + t_{n+1}^2}\ .$$

Since the ODE is *nonlinear*, an equation solver is needed to approximate the solution of

$$y_{n+1} = y_n + \frac{h}{24}\left[9\,\frac{y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}}{3 + t_{n+1}^2} + 19\,f_n - 5\,f_{n-1} + f_{n-2}\right]$$

at each time step.

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	0.5909091	0.8000000	1.166667	1.750000

(c). The fourth order *backward differentiation* formula is

$$y_{n+1} = \frac{1}{25}[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h f_{n+1}].$$

Since the ODE is *nonlinear*, we obtain the *implicit* equation

$$y_{n+1} = \frac{1}{25}\left[48\,y_n - 36\,y_{n-1} + 16\,y_{n-2} - 3\,y_{n-3} + 12h\,\frac{y_{n+1}^2 + 2\,t_{n+1}\,y_{n+1}}{3 + t_{n+1}^2}\right].$$

	$n = 10$	$n = 20$	$n = 30$	$n = 40$
t_n	0.5	1.0	1.5	2.0
y_n	0.5909092	0.8000002	1.166667	1.750001

The exact solution of the IVP is $y(t) = (3 + t^2)/(6 - t)$.

13. Both *Adams* methods entail the approximation of $f(t\,,y)$, on the interval $[t_n\,, t_{n+1}]$, by a polynomial. Approximating $\phi'(t) = P_1(t) \equiv A$, which is a *constant* polynomial, we have

$$\begin{aligned}\phi(t_{n+1}) - \phi(t_n) &= \int_{t_n}^{t_{n+1}} A\,dt \\ &= A(t_{n+1} - t_n) = Ah\,.\end{aligned}$$

Setting $A = \lambda\,f_n + (1-\lambda)f_{n-1}$, where $0 \le \lambda \le 1$, we obtain the approximation

$$y_{n+1} = y_n + h[\lambda\,f_n + (1-\lambda)f_{n-1}].$$

An appropriate choice of λ yields the familiar Euler formula. Similarly, setting

$$A = \lambda\,f_n + (1-\lambda)f_{n+1}\,,$$

where $0 \le \lambda \le 1$, we obtain the approximation

$$y_{n+1} = y_n + h[\lambda f_n + (1-\lambda) f_{n+1}].$$

14. For a *third order* Adams-Bashforth formula, we approximate $f(t, y)$, on the interval $[t_n, t_{n+1}]$, by a *quadratic* polynomial using the points (t_{n-2}, y_{n-2}), (t_{n-1}, y_{n-1}) and (t_n, y_n). Let $P_3(t) = At^2 + Bt + C$. We obtain the system of equations

$$\begin{aligned} At_{n-2}^2 + Bt_{n-2} + C &= f_{n-2} \\ At_{n-1}^2 + Bt_{n-1} + C &= f_{n-1} \\ At_n^2 + Bt_n + C &= f_n\,. \end{aligned}$$

For computational purposes, assume that $t_0 = 0$, and $t_n = nh$. It follows that

$$\begin{aligned} A &= \frac{f_n - 2f_{n-1} + f_{n-2}}{2\,h^2} \\ B &= \frac{(3-2n)f_n + (4n-4)f_{n-1} + (1-2n)f_{n-2}}{2\,h} \\ C &= \frac{n^2 - 3n + 2}{2} f_n + \left(2n - n^2\right) f_{n-1} + \frac{n^2 - n}{2} f_{n-2}\,. \end{aligned}$$

We then have

$$\begin{aligned} y_{n+1} - y_n &= \int_{t_n}^{t_{n+1}} \left[At^2 + Bt + C\right] dt \\ &= Ah^3\left(n^2 + n + \frac{1}{3}\right) + Bh^2\left(n + \frac{1}{2}\right) + Ch\,, \end{aligned}$$

which yields

$$y_{n+1} - y_n = \frac{h}{12}(23\,f_n - 16\,f_{n-1} + 5\,f_{n-2})\,.$$

15. For a *third order* Adams-Moulton formula, we approximate $f(t, y)$, on the interval $[t_n, t_{n+1}]$, by a *quadratic* polynomial using the points (t_{n-1}, y_{n-1}), (t_n, y_n) and (t_{n+1}, y_{n+1}). Let $P_3(t) = \alpha t^2 + \beta t + \gamma$. This time we obtain the system of algebraic equations

$$\begin{aligned} \alpha t_{n-1}^2 + \beta t_{n-1} + \gamma &= f_{n-1} \\ \alpha t_n^2 + \beta t_n + \gamma &= f_n \\ \alpha t_{n+1}^2 + \beta t_{n+1} + \gamma &= f_{n+1}\,. \end{aligned}$$

For computational purposes, again assume that $t_0 = 0$, and $t_n = nh$. It follows that

$$\alpha = \frac{f_{n-1} - 2f_n + f_{n+1}}{2\,h^2}$$
$$\beta = \frac{-\,(2n+1)f_{n-1} + 4nf_n + (1-2n)f_{n+1}}{2\,h}$$
$$\gamma = \frac{n^2+n}{2}f_{n-1} + \left(1-n^2\right)f_n + \frac{n^2-n}{2}f_{n+1}\,.$$

We then have

$$\begin{aligned} y_{n+1} - y_n &= \int_{t_n}^{t_{n+1}} \left[\alpha t^2 + \beta t + \gamma\right] dt \\ &= \alpha h^3\left(n^2 + n + \frac{1}{3}\right) + \beta h^2\left(n + \frac{1}{2}\right) + \gamma h\,, \end{aligned}$$

which results in

$$y_{n+1} - y_n = \frac{h}{12}(5f_{n+1} + 8f_n - f_{n-1})\,.$$

Section 8.5

$1(a)$. The *general* solution of the ODE is $y(t) = c\, e^t + 2 - t$. Imposing the initial condition, $y(0) = 2$, the solution of the IVP is $\phi_1(t) = 2 - t$.

(b). If instead, the initial condition $y(0) = 2.001$ is given, the solution of the IVP is $\phi_2(t) = 0.001\, e^t + 2 - t$. We then have $\phi_2(t) - \phi_1(t) = 0.001\, e^t$.

3. The solution of the initial value problem is $\phi(t) = e^{-100\,t} + t$.

(a, b). Based on the exact solution, the *local truncation error* for both of the Euler methods is

$$|e_{loc}| \le \frac{10^4}{2} e^{-100\,\bar{t}_n} h^2.$$

Hence $|e_n| \le 5000\, h^2$, for all $0 < \bar{t}_n < 1$. Furthermore, the local truncation error is *greatest* near $t = 0$. Therefore $|e_1| \le 5000\, h^2 < 0.0005$ for $h < 0.0003$. Now the truncation error accumulates at each time step. Therefore the *actual* time step should be much smaller than $h \approx 0.0003$. For example, with $h = 0.00025$, we obtain the data

	Euler	*B.Euler*	$\phi(t)$
$t = 0.05$	0.056323	0.057165	0.056738
$t = 0.1$	0.100040	0.100051	0.100045

Note that the total number of time steps needed to reach $t = 0.1$ is $N = 400$.

(c). Using the Runge-Kutta method, comparisons are made for several values of h :

$h = 0.1$:

	$\phi(t)$	y_n	$y_n - \phi(t_n)$
$t = 0.05$	0.056738	0.057416	0.000678
$t = 0.1$	0.100045	0.100055	0.000010

$h = 0.005$:

	$\phi(t)$	y_n	$y_n - \phi(t_n)$
$t = 0.05$	0.056738	0.056766	0.000027
$t = 0.1$	0.100045	0.100046	0.0000004

$6(a)$. Using the method of *undetermined coefficients*, it is easy to show that the general solution of the ODE is $y(t) = c\, e^{\lambda t} + t^2$. Imposing the initial condition, it follows that $c = 0$ and hence the solution of the IVP is $\phi(t) = t^2$.

(b). Using the Runge-Kutta method, with $h = 0.01$, numerical solutions are generated for various values of λ :

$\lambda = 1$:

	$\phi(t)$	y_n	$\lvert y_n - \phi(t_n)\rvert$
$t = 0.25$	0.0625	$0.0624999..$	2×10^{-11}
$t = 0.5$	0.25	0.25	0
$t = 0.75$	0.5625	0.5625	0
$t = 1.0$	1.0	1.0	0

$\lambda = 10$:

	$\phi(t)$	y_n	$\lvert y_n - \phi(t_n)\rvert$
$t = 0.25$	0.0625	$0.0624998..$	2.215×10^{-7}
$t = 0.5$	0.25	0.249997	2.920×10^{-6}
$t = 0.75$	0.5625	0.562464	3.579×10^{-5}
$t = 1.0$	1.0	0.999564	4.362×10^{-4}

$\lambda = 20$:

	$\phi(t)$	y_n	$\lvert y_n - \phi(t_n)\rvert$
$t = 0.25$	0.0625	$0.062889..$	1.10×10^{-5}
$t = 0.5$	0.25	0.248342	1.658×10^{-3}
$t = 0.75$	0.5625	0.316458	0.246042
$t = 1.0$	1.0	-35.5139	36.5139

$\lambda = 50$:

	$\phi(t)$	y_n	$\lvert y_n - \phi(t_n)\rvert$
$t = 0.25$	0.0625	$-0.044803..$	0.107303
$t = 0.5$	0.25	-28669.55	28669.804
$t = 0.75$	0.5625	-7.66014×10^{9}	7.66014×10^{9}
$t = 1.0$	1.0	-2.04668×10^{15}	2.04668×10^{15}

The following table shows the calculated value, y_1 , at the *first* time step :

$\phi(t)$	$y_1(\lambda = 1)$	$y_1(\lambda = 10)$	$y_1(\lambda = 20)$	$y_1(\lambda = 50)$
10^{-4}	9.99999×10^{-5}	9.99979×10^{-5}	9.99833×10^{-5}	9.97396×10^{-5}

(c). Referring back to the *exact* solution given in Part(a), if a *nonzero* initial condition, say $y(0) = \varepsilon$, is specified, the solution of the IVP becomes

$$\phi_\varepsilon(t) = \varepsilon\, e^{\lambda t} + t^2.$$

We then have $\lvert\phi(t) - \phi_\varepsilon(t)\rvert = \lvert\varepsilon\rvert\, e^{\lambda t}$. It is evident that for any $t > 0$,

$$\lim_{\lambda \to \infty} |\phi(t) - \phi_\varepsilon(t)| = \infty\,.$$

This implies that virtually any error introduced early in the calculations will be magnified as $\lambda \to \infty$. The initial value problem is inherently *unstable*.

Section 8.6

1. In vector notation, the initial value problem can be written as

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x+y+t \\ 4x-2y \end{pmatrix}, \ \mathbf{x}(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

(a). The Euler formula is

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \end{pmatrix} + h\begin{pmatrix} x_n+y_n+t_n \\ 4x_n-2y_n \end{pmatrix}.$$

That is,

$$x_{n+1} = x_n + h(x_n + y_n + t_n)$$
$$y_{n+1} = y_n + h(4x_n - 2y_n).$$

With $h = 0.1$, we obtain the values

	$n=2$	$n=4$	$n=6$	$n=8$	$n=10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.26	1.7714	2.58991	3.82374	5.64246
y_n	0.76	1.4824	2.3703	3.60413	5.38885

(b). The Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = (x_n + y_n + t_n\,, 4x_n - 2y_n)^T$$
$$\mathbf{k}_{n2} = \left[x_n + \frac{h}{2}k^1_{n1} + y_n + \frac{h}{2}k^2_{n1} + t_n + \frac{h}{2}\,, 4\left(x_n + \frac{h}{2}k^1_{n1}\right) - 2\left(y_n + \frac{h}{2}k^2_{n1}\right)\right]^T$$
$$\mathbf{k}_{n3} = \left[x_n + \frac{h}{2}k^1_{n2} + y_n + \frac{h}{2}k^2_{n2} + t_n + \frac{h}{2}\,, 4\left(x_n + \frac{h}{2}k^1_{n2}\right) - 2\left(y_n + \frac{h}{2}k^2_{n2}\right)\right]^T$$
$$\mathbf{k}_{n4} = [x_n + hk^1_{n3} + y_n + hk^2_{n3} + t_n + h\,, 4(x_n + hk^1_{n3}) - 2(y_n + hk^2_{n3})]^T.$$

With $h = 0.2$, we obtain the values:

	$n=1$	$n=2$	$n=3$	$n=4$	$n=5$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.32493	1.93679	2.93414	4.48318	6.84236
y_n	0.758933	1.57919	2.66099	4.22639	6.56452

(c). With $h = 0.1$, we obtain

	$n=2$	$n=4$	$n=6$	$n=8$	$n=10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.32489	1.9369	2.93459	4.48422	6.8444
y_n	0.759516	1.57999	2.66201	4.22784	6.56684

The exact solution of the IVP is

$$x(t) = e^{2t} + \frac{2}{9}e^{-3t} - \frac{1}{3}t - \frac{2}{9}$$
$$y(t) = e^{2t} - \frac{8}{9}e^{-3t} - \frac{2}{3}t - \frac{1}{9}\,.$$

$3(a)$. The Euler formula is

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \end{pmatrix} + h\begin{pmatrix} -\,t_n\,x_n - y_n - 1 \\ x_n \end{pmatrix}.$$

That is,

$$x_{n+1} = x_n + h(-\,t_n\,x_n - y_n - 1)$$
$$y_{n+1} = y_n + h(x_n).$$

With $h = 0.1$, we obtain the values

	$n=2$	$n=4$	$n=6$	$n=8$	$n=10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	0.582	0.117969	-0.336912	-0.730007	-1.02134
y_n	1.18	1.27344	1.27382	1.18572	1.02371

(b). The Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = (-\,t_n\,x_n - y_n - 1, x_n)^T$$
$$\mathbf{k}_{n2} = \left[-\left(t_n + \frac{h}{2}\right)\left(x_n + \frac{h}{2}k^1_{n1}\right) - \left(y_n + \frac{h}{2}k^2_{n1}\right) - 1\,, x_n + \frac{h}{2}k^1_{n1}\right]^T$$
$$\mathbf{k}_{n3} = \left[-\left(t_n + \frac{h}{2}\right)\left(x_n + \frac{h}{2}k^1_{n2}\right) - \left(y_n + \frac{h}{2}k^2_{n2}\right) - 1\,, x_n + \frac{h}{2}k^1_{n2}\right]^T$$
$$\mathbf{k}_{n4} = [-\,(t_n + h)(x_n + hk^1_{n3}) - (y_n + hk^2_{n3}) - 1\,, x_n + hk^1_{n3}]^T.$$

With $h = 0.2$, we obtain the values:

	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	0.568451	0.109776	− 0.32208	− 0.681296	− 0.937852
y_n	1.15775	1.22556	1.20347	1.10162	0.937852

(c). With $h = 0.1$, we obtain

	$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	0.56845	0.109773	− 0.322081	− 0.681291	− 0.937841
y_n	1.15775	1.22557	1.20347	1.10161	0.93784

$4(a)$. The Euler formula gives

$$x_{n+1} = x_n + h(x_n - y_n + x_n\, y_n)$$
$$y_{n+1} = y_n + h(3x_n - 2y_n - x_n\, y_n).$$

With $h = 0.1$, we obtain the values

	$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	− 0.198	− 0.378796	− 0.51932	− 0.594324	− 0.588278
y_n	0.618	0.28329	− 0.0321025	− 0.326801	− 0.57545

(b). Given

$$f(t,x,y) = x - y + x\,y$$
$$g(t,x,y) = 3x - 2y - x\,y\,,$$

the Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = [f(t_n, x_n, y_n), g(t_n, x_n, y_n)]^T$$
$$\mathbf{k}_{n2} = \left[f\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n1}, y_n + \frac{h}{2}k^2_{n1}\right), g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n1}, y_n + \frac{h}{2}k^2_{n1}\right)\right]^T$$
$$\mathbf{k}_{n3} = \left[f\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n2}, y_n + \frac{h}{2}k^2_{n2}\right), g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n2}, y_n + \frac{h}{2}k^2_{n2}\right)\right]^T$$
$$\mathbf{k}_{n4} = \left[f\left(t_n + h, x_n + hk^1_{n3}, y_n + hk^2_{n3}\right), g\left(t_n + h, x_n + hk^1_{n3}, y_n + hk^2_{n3}\right)\right]^T.$$

With $h = 0.2$, we obtain the values:

	$n = 1$	$n = 2$	$n = 3$	$n = 4$	$n = 5$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	-0.196904	-0.372643	-0.501302	-0.561270	-0.547053
y_n	0.630936	0.298888	-0.0111429	-0.288943	-0.508303

(c). With $h = 0.1$, we obtain

	$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	-0.196935	-0.372687	-0.501345	-0.561292	-0.547031
y_n	0.630939	0.298866	-0.0112184	-0.28907	-0.508427

$5(a)$. The Euler formula gives

$$x_{n+1} = x_n + h[x_n(1 - 0.5\,x_n - 0.5\,y_n)]$$
$$y_{n+1} = y_n + h[y_n(-0.25 + 0.5\,x_n)].$$

With $h = 0.1$, we obtain the values

	$n = 2$	$n = 4$	$n = 6$	$n = 8$	$n = 10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	2.96225	2.34119	1.90236	1.56602	1.29768
y_n	1.34538	1.67121	1.97158	2.23895	2.46732

(b). Given

$$f(t, x, y) = x(1 - 0.5\,x - 0.5\,y)$$
$$g(t, x, y) = y(-0.25 + 0.5\,x)\,,$$

the Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = [f(t_n, x_n, y_n), g(t_n, x_n, y_n)]^T$$
$$\mathbf{k}_{n2} = \left[f\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n1}, y_n + \frac{h}{2}k^2_{n1}\right), g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n1}, y_n + \frac{h}{2}k^2_{n1}\right)\right]^T$$
$$\mathbf{k}_{n3} = \left[f\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n2}, y_n + \frac{h}{2}k^2_{n2}\right), g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k^1_{n2}, y_n + \frac{h}{2}k^2_{n2}\right)\right]^T$$
$$\mathbf{k}_{n4} = \left[f\left(t_n + h, x_n + hk^1_{n3}, y_n + hk^2_{n3}\right), g\left(t_n + h, x_n + hk^1_{n3}, y_n + hk^2_{n3}\right)\right]^T.$$

With $h = 0.2$, we obtain the values:

	$n=1$	$n=2$	$n=3$	$n=4$	$n=5$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	3.06339	2.44497	1.9911	1.63818	1.3555
y_n	1.34858	1.68638	2.00036	2.27981	2.5175

(c). With $h = 0.1$, we obtain

	$n=2$	$n=4$	$n=6$	$n=8$	$n=10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	3.06314	2.44465	1.99075	1.63781	1.35514
y_n	1.34899	1.68699	2.00107	2.28057	2.51827

$6(a)$. The Euler formula gives

$$x_{n+1} = x_n + h[exp(-x_n + y_n) - cos\,x_n]$$
$$y_{n+1} = y_n + h[sin(x_n - 3\,y_n)].$$

With $h = 0.1$, we obtain the values

	$n=2$	$n=4$	$n=6$	$n=8$	$n=10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.42386	1.82234	2.21728	2.61118	2.9955
y_n	2.18957	2.36791	2.53329	2.68763	2.83354

(b). The Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = [f(t_n, x_n, y_n), g(t_n, x_n, y_n)]^T$$
$$\mathbf{k}_{n2} = \left[f\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_{n1}^1, y_n + \frac{h}{2}k_{n1}^2\right), g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_{n1}^1, y_n + \frac{h}{2}k_{n1}^2\right)\right]^T$$
$$\mathbf{k}_{n3} = \left[f\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_{n2}^1, y_n + \frac{h}{2}k_{n2}^2\right), g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_{n2}^1, y_n + \frac{h}{2}k_{n2}^2\right)\right]^T$$
$$\mathbf{k}_{n4} = \left[f\left(t_n + h, x_n + hk_{n3}^1, y_n + hk_{n3}^2\right), g\left(t_n + h, x_n + hk_{n3}^1, y_n + hk_{n3}^2\right)\right]^T.$$

With $h = 0.2$, we obtain the values:

	$n=1$	$n=2$	$n=3$	$n=4$	$n=5$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.41513	1.81208	2.20635	2.59826	2.97806
y_n	2.18699	2.36233	2.5258	2.6794	2.82487

(c). With $h = 0.1$, we obtain

	$n=2$	$n=4$	$n=6$	$n=8$	$n=10$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.41513	1.81209	2.20635	2.59826	2.97806
y_n	2.18699	2.36233	2.52581	2.67941	2.82488

7. The Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = [x_n - 4\,y_n,\ -x_n + y_n]^T$$
$$\mathbf{k}_{n2} = \left[x_n + \frac{h}{2}k^1_{n1} - 4\left(y_n + \frac{h}{2}k^2_{n1}\right),\ -\left(x_n + \frac{h}{2}k^1_{n1}\right) + y_n + \frac{h}{2}k^2_{n1}\right]^T$$
$$\mathbf{k}_{n3} = \left[x_n + \frac{h}{2}k^1_{n2} - 4\left(y_n + \frac{h}{2}k^2_{n2}\right),\ -\left(x_n + \frac{h}{2}k^1_{n2}\right) + y_n + \frac{h}{2}k^2_{n2}\right]^T$$
$$\mathbf{k}_{n4} = \left[x_n + hk^1_{n3} - 4\left(y_n + hk^2_{n3}\right),\ -\left(x_n + hk^1_{n3}\right) + y_n + hk^2_{n3}\right]^T.$$

Using $h = 0.04$, we obtain the following values:

	$n=5$	$n=10$	$n=15$	$n=20$	$n=25$
t_n	0.2	0.4	0.6	0.8	1.0
x_n	1.3204	1.9952	3.2992	5.7362	10.227
y_n	− 0.25085	− 0.66245	− 1.3752	− 2.6435	− 4.9294

The exact solution is given by

$$\phi(t) = \frac{e^{-t} + e^{3t}}{2}\,,\ \psi(t) = \frac{e^{-t} - e^{3t}}{4}\,,$$

and the associated tabulated values:

	$n=5$	$n=10$	$n=15$	$n=20$	$n=25$
t_n	0.2	0.4	0.6	0.8	1.0
$\phi(t_n)$	1.3204	1.9952	3.2992	5.7362	10.227
$\psi(t_n)$	− 0.25085	− 0.66245	− 1.3752	− 2.6435	− 4.9294

8. Let $y = x'$. The second order ODE can be transformed into the first order system

$$x' = y$$
$$y' = t - 3x - t^2 y\,,$$

with initial conditions $x(0) = 1\,,\ y(0) = 2\,.$ Given

$$f(t, x, y) = y$$
$$g(t, x, y) = t - 3x - t^2 y\,,$$

the Runge-Kutta method uses the following intermediate calculations:

$$\mathbf{k}_{n1} = \left[y_n, t_n - 3\,x_n - t_n^2\,y_n\right]^T$$
$$\mathbf{k}_{n2} = \left[y_n + \frac{h}{2}k_{n1}^2, g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_{n1}^1, y_n + \frac{h}{2}k_{n1}^2\right)\right]^T$$
$$\mathbf{k}_{n3} = \left[y_n + \frac{h}{2}k_{n2}^2, g\left(t_n + \frac{h}{2}, x_n + \frac{h}{2}k_{n2}^1, y_n + \frac{h}{2}k_{n2}^2\right)\right]^T$$
$$\mathbf{k}_{n4} = \left[y_n + hk_{n3}^2, g\left(t_n + h, x_n + hk_{n3}^1, y_n + hk_{n3}^2\right)\right]^T.$$

With $h = 0.1$, we obtain the following values:

	$n = 5$	$n = 10$
t_n	0.5	1.0
x_n	1.543	0.07075
y_n	1.14743	− 1.3885

9. The *predictor* formulas are

$$x_{n+1} = x_n + \frac{h}{24}(55\,f_n - 59\,f_{n-1} + 37\,f_{n-2} - 9\,f_{n-3})$$
$$y_{n+1} = y_n + \frac{h}{24}(55\,g_n - 59\,g_{n-1} + 37\,g_{n-2} - 9\,g_{n-3}).$$

With $f_{n+1} = x_{n+1} - 4\,y_{n+1}$ and $g_{n+1} = -\,x_{n+1} + y_{n+1}$, the *corrector* formulas are

$$x_{n+1} = x_n + \frac{h}{24}(9\,f_{n+1} + 19\,f_n - 5\,f_{n-1} + f_{n-2})$$
$$y_{n+1} = y_n + \frac{h}{24}(9\,g_{n+1} + 19\,g_n - 5\,g_{n-1} + g_{n-2}).$$

We use the starting values from the *exact solution* :

	$n=0$	$n=1$	$n=2$	$n=3$
t_n	0	0.1	0.2	0.3
x_n	1.0	1.12883	1.32042	1.60021
y_n	0.0	-0.11057	-0.250847	-0.429696

One time step using the *predictor-corrector* method results in the approximate values:

	$n=4(pre)$	$n=4(cor)$
t_n	0.4	0.4
x_n	1.99445	1.99521
y_n	-0.662064	-0.662442