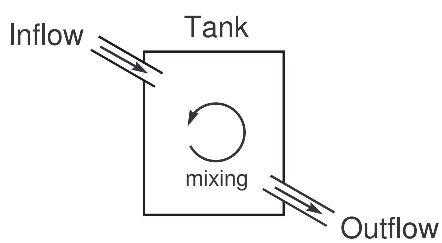
Math 133A: Week 5
Applications and Numerical Methods

Section 1: Inflow / Outflow Models

One application of first-order linear differential equations differential equations is modeling the amount (or concentration) of a substance in a well-stirred tank/vessel subject to constant in-flow and out-flow. Common simple applications are:

(1) an industrial mixing tank with an entry pipe and an exit pipe;

(2) a lake with a inflowing river feeding a pollutant in from upstream and an outflowing river removing pollutantdownstream;



In all cases, we are interested in the modeling the amount of substance in the mixing vessel over time.

We want to translate this description into mathematics. The fundamental equation we will use will be

$$[\text{rate of change}] = [\text{rate in}] - [\text{rate out}].$$

That is to say, at each instance in time, we believe that the rate of change of the overall amount of the quantity of interest to equal the amount that is flowing in minus the amount that is flowing out.

To characterize the inflow rate, we need to know the overall **flow rate** and **concentration** of the substance of interest in inflowing mixture. This could be the amount of pollutant in an inflowing stream, or the amount of chemical diluted in an inflowing pipe. We have

$$[\text{rate in}] = [\text{concentration in}] \cdot [\text{volume rate in}]$$

since

$$[\text{concentration in}] \cdot [\text{volume rate in}] = \left[\frac{\text{amount}}{\text{volume}}\right] \cdot \left[\frac{\text{volume}}{\text{time}}\right] = \left[\frac{\text{amount}}{\text{time}}\right] \tag{1}$$

The outflow is slightly different. We again need to know the **flow rate** and **concentration** of substance in the outflowing mixture. We do not, however, know the amount or concentration of the substance in the system—that is what we are trying to find! We will have to make some assumptions. The most basic will be to assume that the vessel is well-mixed so that that the concentration of the substance is uniform throughout the mixing vessel. That is, we do not allow pockets of high concentration of pollutants or pockets of low concentration.

Give this, by (1) we have

$$\begin{aligned}[\text{rate out}] &= [\text{concentration out}] \cdot [\text{volume rate out}] \\ &= \frac{[\text{current amount}]}{[\text{current volume}]} \cdot [\text{volume rate out}].\end{aligned}$$

We let $A(t)$ denote the amount of substance in the tank at time t and $V(t)$ denote the corresponding volume of the talk. We then have the combined model

$$\frac{dA}{dt} = [\text{concentration in}] \cdot [\text{volume rate in}] - \frac{A(t)}{V(t)} \cdot [\text{volume rate out}].$$

Note: The key difference between the inflow and outflow rates is that the *amount* and *volume* in the outflow rate depend upon the *current* amount and volume in the mixing vessel. In the inflow tank, these quantities are either controlled (for mixing tanks) or known (for rivers and streams). Notice also that if the volume of the in-flow and the volume of the out-flow do not balance, the volume of the tank may be a dynamic function of time (imagine filling or emptying a mixing tank).

Example 1

Suppose that there is a factory built upstream of a lake with a volume of 0.5 km^3. The factory introduces a new pollutant to a stream which pumps 1 km^3 of water into the lake every year. Suppose that the net outflow from the lake is also 1 km^3 per year and that the concentration of the pollutant in the inflow stream is 200 kg/km^3.

(a) Set up an initial value problem for the amount of pollutant in the lake and solve it.

(b) Assuming there is initially no pollutant in the lake, how much pollutant is there are one month?

(c) What is the limiting pollutant level?

Solution: We need to set up the model in the form [rate of change]=[rate in]−[rate out]. If we let A denote the amount of the pollutant (in kg), we have

$$[\text{rate of change}] = \frac{dA}{dt}.$$

In order to determine the rate in, we notice that the amount (in kg) coming from the inflow can be given by

$$\begin{aligned}[\text{rate in}] &= [\text{concentration in}] \cdot [\text{volume rate in}] \\ &= (200\ \text{kg/km}^3) \cdot (1\ \text{km}^3/\text{year}) = 200\ \text{kg/year}.\end{aligned}$$

The rate out is given by

$$\begin{aligned}[\text{rate out}] &= [\text{concentration out}] \cdot [\text{volume rate out}] \\ &= \left(\frac{A}{0.5}\ \text{kg/km}^3\right)(1\ \text{km}^3/\text{year}) = 2A\ \text{kg/year}.\end{aligned}$$

We can see the units have worked as desired. We can drop them and just focus on the initial value problem

$$\frac{dA}{dt} = 200 - 2A, \quad A(0) = A_0.$$

This is a first-order linear differential equation which in standard form is given by

$$\frac{dA}{dt} + 2A = 200.$$

We can see that we have $p(x) = 2$ and $q(x) = 200$. The necessary integration factor is

$$\rho(t) = e^{\int 2\;dt} = e^{2t}$$

so that we have

$$\begin{aligned} & e^{2t}\frac{dA}{dt} + 2e^{2t}A = 200e^{2t} \\ \implies \quad & \frac{d}{dt}\left[e^{2t}A\right] = 200e^{2t} \\ \implies \quad & e^{2t}A = \int 200e^{2t}\;dt = 100e^{2t} + C \\ \implies \quad & A(t) = 100 + Ce^{-2t}. \end{aligned}$$

In order to solve for C, we use $A(0) = A_0$ to get

$$A(0) = A_0 = 100 + C \quad \implies \quad C = A_0 - 100.$$

This gives the solution

$$A(t) = 100 + (A_0 - 100)e^{-2t}.$$

For this form, we can easily answer part (b). Given an initial pollutant level of zero (i.e. $A_0 = 0$), we have

$$A(t) = 100 - 100e^{-2t}.$$

After one month has passed, we have $t = 1/12$ so that the amount of pollutant is given by

$$A(1/12) = 100 - 100e^{-2(1/12)} \approx 15.3528 \text{ kg}.$$

We can also easily determine the limiting pollutant level by evaluating

$$\lim_{t\to\infty} A(t) = \lim_{t\to\infty}\left[100 + (A_0 - 100)e^{-2t}\right] = 100.$$

In other words, no matter what the initial amount is in the lake, we will always converge toward 100 kg of pollutant distributed throughout the lake. This makes sense. The limiting level is going to be when the rate in and the rate out are balanced. That occurs for this model when $200 = 2A$ which implies $A = 100$.

Example 2

Consider a 50 gallon tank which is initial filled with 20 gallons of brine (salt/water mixture) with a concentration of 1/4 lbs/gallon of salt. Suppose that there is an inflow tube which infuses 3 gallons of brine into the tank per minute with a concentration of 1 lbs/gallon. Suppose that there is an outflow tube which flows at a rate of 2 gallons per minute.

(a) Set up and solve a differential equation for the amount of salt in the tank.

(b) How much salt is in the tank when the tank is full?

Solution: This is slightly different than the previous example because the volume of mixture in the tank *changes* because the inflow and outflow volume rates are different. There is more mixture flowing into the tank than flowing out. Nevertheless, we can incorporate this into our model by noting that the volume of the tank at time t can be given by

$$V(t) = 20 + (3-2)t = 20 + t.$$

We can now complete the model as before. We have

$$\frac{dA}{dt} = (3)(1) - (2)\frac{A}{20+t} = 3 - \frac{2A}{20+t}, \quad A(0) = 20(1/4) = 5.$$

Again, this is a first-order linear differential equation. We can solve it by rewriting

$$\frac{dA}{dt} + \left(\frac{2}{20+t}\right) A = 3$$

and determining the integrating factor

$$\rho(t) = e^{\int 2/(20+t)\, dt} = e^{2\ln(20+t)} = (20+t)^2.$$

This gives

$$
\begin{aligned}
& (20+t)^2\frac{dA}{dt} + 2(20+t)A = 3(20+t)^2 \\
\implies \quad & \frac{d}{dt}\left[(20+t)^2 A\right] = 3(20+t)^2 \\
\implies \quad & (20+t)^2 A = (20+t)^3 + C \\
\implies \quad & A(t) = (20+t) + \frac{C}{(20+t)^2}.
\end{aligned}
$$

Using the initial condition $A(0) = 5$, we have

$$A(0) = 5 = 20 + \frac{C}{400} \quad \implies \quad C = -6000$$

so that the particular solution is

$$A(t) = (20+t) - \frac{6000}{(20+t)^2}.$$

To answer the question of how much salt will be in the tank when the tank is full, we notice that the tank will be full when $V(t) = 20+t = 50$, which implies $t = 30$ (i.e. it will take thirty minutes). This gives

$$A(30) = (20+30) - \frac{6000}{(20+30)^2} = 50 - \frac{6000}{2500} = 47.6.$$

It follows that there will be 47.6 lbs of salt in the tank when it is full.

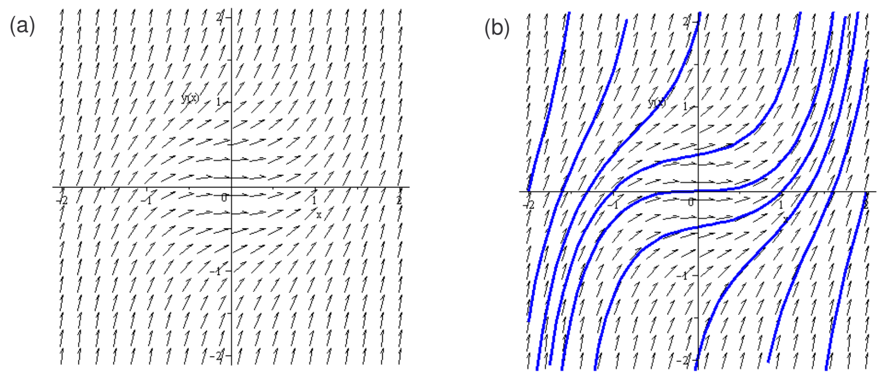
Section 2: Numerical Methods

We have seen examples of first-order differential equations which can, by one method of another, be solved to yield a solution $y(x)$. In real world applications, however, the equations encountered are more complicated and it is often unreasonable to be expected to solve differential equations directly.

Now consider the following example:

$$y' = x^2 + y^2. \tag{2}$$

The slopes are constant along circles, i.e. $x^2 + y^2 = C^2$. As the radius C grows, the slopes became steeper. This gives the following slope field, where we have overlain candidate solutions on the right:



We can clearly see what the qualitative behavior of solutions is, but as we go through the solution tools we have accumulated so far to explicitly find the solution $y(x)$, we find ourselves frustrated. This differential equation is not directly integrable, separable, or first-order linear. It is also not power homogeneous or Bernoulli, or any of the other classifications which can be solved by more sophisticated first-order methods.

In fact, this is an example of a differential equation for which there is **no solution which can be represented in terms of elementary functions** (e.g. x^n, $\sin(x)$, $\cos(x)$, e^x, $\ln(x)$, etc.). The best we can do with this example is represent the solution in terms of a particular series of (potentially non-integer) powers of x known as **Bessel functions**.

Note: There is a difference between saying there is no solution representable by elementary functions and there is no solution at all. The existence of a solution through every point (x, y) is guaranteed for this equation by basic existence theorems since $f(x, y) = x^2 + y^2$ is continuous. We did not cover this theorem in class, but the point is that we know there is a solution; we just lack a convenient way to describe what it is.

Our interest in differential equations does not stop when a closed-form solution eludes our grasp. Most differential equations which model real-world phenomena (e.g. predicting the weather, modeling the motion of a space shuttle) do not permit such solutions, but we cannot simply give up because the task is messier than we would like. Weather still changes, space shuttles still move in orbit, and the solution to our differential equation is still out there waiting to be described. Our technique for characterizing the solution will be to build an approximate solution called a **numerical solution**. We will discuss two such methods: the **forward Euler Method** and **fourth-order Runge-Kutta method**.

Section 3: Euler's Method

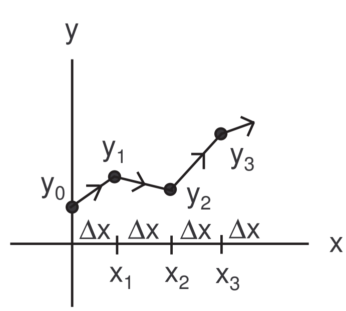
In the example (2), consider attempting to obtain an estimate for $y(1)$ for the solution through $y(0) = 0$. We could certainly obtain a ballpark estimate of the number from the slope field (e.g. it is larger than 0 and less than 0.5), but if we were modeling a shuttle re-entering the Earth's atmosphere, we would want to do better. In fact, lives would depend upon it. We now make this process of numerically tracking solutions more rigorous.

Consider the following intuition:

1. The value of $f(x, y)$ at every point (x, y) agrees with the solution through the point.

2. In a small neighborhood of (x, y), the value of $f(x, y)$ does not change significantly (so long as $f(x, y)$ is continuous).

Consider starting at some initial condition (x_0, y_0). The above intuition means that we may approximate the solution *locally* with a straight line solution with slope $f(x_0, y_0)$. We know the straight-line solution and the real solution will diverge at some point, but we can adjust for this by simply changing the value of $f(x, y)$ in the straight-line solution we are following, as in the following picture:



If we take sufficiently small increments in x for each time we change (say $0 < \Delta x \ll 1$) we imagine each step forward in the state space is not far away from the analytic solution corresponding to the same initial condition.

We formally introduce the following.

Definition 1

Consider a first-order initial value problem $y' = f(x, y)$, $y(x_0) = y_0$, where $f(x, y)$ is continuous. The **forward Euler method** for generating a **numerical solution** is given by

$$y_{n+1} = y_n + f(x_n, y_n)\Delta x. \tag{3}$$

where $\Delta x > 0$ is the step-size.

This formula corresponds exactly to the intuition was offered above. At a point (x_n, y_x), we compute the next state (x_{n+1}, y_{n+1}) by updating the current point by the slope of the vector field at that point $(f(x_n, y_n))$ over a small increment (Δx). We then repeat the process to extend the solution. We may then either graph the result or use the piecewise linear function obtained to approximate specific values of the system.

Example 3

Use the step-size $\Delta x = 0.1$ to estimate the value of $y(1)$ for the initial value problem (2) with $y(0) = 0$.

Solution: We have the update scheme

$$y_{n+1} = y_n + f(x_n, y_n)\Delta x = y_n + (x_n^2 + y_n^2)\Delta x$$

and $x_{n+1} = x_n + \Delta x$. For $\Delta x = 0.1$ and $(x_0, y_0) = (0, 0)$, we have

$$y_1 = y_0 + (x_0^2 + y_0^2)\Delta x = (0) + ((0)^2 + (0)^2)(0.1) = 0.$$

We also have that $x_1 = x_0 + \Delta x = (0) + (0.1) = 0.1$ so that $(x_1, y_1) = (0.1, 0)$. Applying the procedure again, we have

$$y_2 = y_1 + (x_1^2 + y_1^2)\Delta x = (0) + ((0.1)^2 + (0)^2)(0.1) = 0.001.$$

It follows that $(x_2, y_2) = (0.2, 0.001)$. Continuing this procedure, we arrive at the following table of values given below:

n	x_n	y_n
0	0	0
1	0.1	0
2	0.2	0.001
3	0.3	0.0050001
4	0.4	0.0140026
5	0.5	0.030022207
6	0.6	0.05511234
7	0.7	0.091416077
8	0.8	0.141251767
9	0.9	0.207246973
10	1.0	0.292542104

These values represent a **numerical solution**. They are the analogue of plugging specific x values into our solution form $y(x)$. We have the estimate $y(1) \approx 0.292542104$. Of course, for this example, we do not have an explicit solution $y(x)$ with which to compare the accuracy of this estimate, but it can be determined that the "true" value up to 10 significant digits of accuracy is 0.3502318443 for a relative error of 0.0576897403. We should agree that, if our lives depend upon the NASA technicians correctly calibrating our shuttle for re-entry, this error is probably too much to stomach.

Note on numerical solutions:

(1) Beyond a few iterations, numerically simulating solutions is impossible to do by hand—**computers are a necessity**. As computational power has increased in the past fifty years, the emphasis in applied mathematics has shifted significantly toward numerical and computational methods.

(2) Numerical simulation requires specified **initial conditions** and **parameter values**. A numerical solution can suggest whether a model permits certain behavior (e.g. growth/decay/stability, oscillations, etc.) but it can only do so for *one* solution at a time.

(3) Each step in Euler's method has a error associated to it, so we might wonder about how these errors accumulate. Is the overall error still guaranteed to be small? Omitting details, we will make the following notes about ways to increase accuracy:

   (a) Choose a smaller time step Δx.

   (b) Choose a better numerical scheme. (*Note:* We have already seen that the forward Euler method is *terrible* for bounding the accumulation of errors!)

Example 4

With the help of a computer, use the Forward Euler method with step sizes $\Delta x = 0.5, 0.1, 0.01$, and 0.001 to estimate the value of $y(1.5)$ for the differential equation

$$y' = x^2 + y^2, \qquad y(0) = 0. \tag{4}$$

Comment on how these results compare to the "true" value of $y(1.5) = 1.517447537$.

Solution: To re-iterate, the formula for the Forward Euler method is

$$\textbf{Forward Euler:} \qquad y_{n+1} = y_n + f(x_n, y_n)\Delta x. \tag{5}$$

Notice that, when we specify a particular point in the future, we can determine the number of steps required to get there. For this problem, we need $(x_{final} - x_{initial})/\Delta x = 1.5/0.001 = 1500$ computations to produce the estimate for $y(1.5)$ using $\Delta x = 0.001$. We have our work cut out for us! To do this by hand is infeasible; fortunately, computers can implement such recursive algorithms as Euler's method (and other numerical schemes) very, very quickly.

We can carry out the procedure outlined in the lecture notes by hand to get the first few estimates with our calculators, but for small step-sizes we will definitely have to use a computer. This gives the following table:

Δx	$y(1.5)$	error	steps
0.5	0.6328125	0.884635	3
0.1	1.213352104	0.3040954	15
0.01	1.479113716	0.0383338	150
0.001	1.513502037	0.0039455	1500

We can see that we have obtained a marked improvement by reducing the step size. This makes sense—we have less chance of floating far away from the true trajectory if we take smaller steps before correcting ourselves. We should, however, be disappointed by the tremendous number of steps associated with the refined estimate. Fifteen-hundred steps represents a significant computational expenditure for a few decimal places of accuracy, and it would be reasonable at this point to wonder if there are alternative schemes by which to construct numerical solutions.

Section 4: Runge-Kutta Method

The fourth-order Runge-Kutta method is a very popular numerical simulation algorithm as it is known to produce less error per step than the forward Euler method. The trade-off is that each step of the algorithm is significantly more computationally intensive.

Formally, we introduce the following.

Definition 2

Consider a first-order initial value problem $y' = f(x, y),\ y(x_0) = y_0$, where $f(x, y)$ is continuous. The **fourth-order Runge-Kutta method** for generating a numerical solution is given by

$$y_{n+1} = y_n + \frac{\Delta x}{6} (k_1 + 2k_2 + 2k_3 + k_4) \tag{6}$$

where

$$\text{Runge-Kutta:} \quad \left\{ \begin{array}{ll} k_1 & = f(x_n, y_n) \\ k_2 & = f(x_n + \frac{1}{2}\Delta x, y_n + \frac{1}{2}k_1\Delta x) \\ k_3 & = f(x_n + \frac{1}{2}\Delta x, y_n + \frac{1}{2}k_2\Delta x) \\ k_4 & = f(x_n + \Delta x, y_n + k_3\Delta x). \end{array} \right. \tag{7}$$

These equations look crazy at first glance, but before we throw our hands up in defeat, let's compare how it performs for the previous example compared to the forward Euler method.

Example 5

With the help of a computer, use the fourth-order Runge-Kutta method with step sizes $\Delta x = 0.5, 0.1$, and 0.01 to estimate the value of $y(1.5)$ for the differential equation

$$y' = x^2 + y^2, \qquad y(0) = 0. \tag{8}$$

Comment on how these results compare to the "true" value of $y(1.5) = 1.517447537$ and also the values obtained by the forward Euler method.

Solution: The computations are easy enough to perform for $\Delta x = 0.5$ that we will do one step by hand. Thereafter, we will either rely on a computer—otherwise, it would take remainder of the semester to do this single problem. As always, we have $x_1 = x_0 + \Delta x = (0) + (0.5) = 0.5$. To compute y_1, we need k_1, k_2, k_3, and k_4. We have

$$k_1 = f(x_0, y_0) = x_0^2 + y_0^2 = (0)^2 + (0)^2 = 0$$

and

$$\begin{aligned}
k_2 &= f(x_0 + \frac{1}{2}\Delta x, y_0 + \frac{1}{2}k_1\Delta x) \\
&= (x_0 + \frac{1}{2}\Delta x)^2 + (y_0 + \frac{1}{2}k_1\Delta x)^2 \\
&= ((0) + \frac{1}{2}(0.5))^2 + ((0) + \frac{1}{2}(0)(0.5))^2 = 0.0625
\end{aligned}$$

and

$$\begin{aligned}
k_3 &= f(x_0 + \frac{1}{2}\Delta x, y_0 + \frac{1}{2}k_2\Delta x) \\
&= (x_0 + \frac{1}{2}\Delta x)^2 + (y_0 + \frac{1}{2}k_2\Delta x)^2 \\
&= ((0) + \frac{1}{2}(0.5))^2 + ((0) + \frac{1}{2}(0.0625)(0.5))^2 \\
&= 0.06274414
\end{aligned}$$

and

$$
\begin{aligned}
k_4 &= f(x_0 + \Delta x, y_0 + k_3 \Delta x) \\
&= (x_0 + \Delta x)^2 + (y_0 + k_3 \Delta x)^2 \\
&= ((0) + (0.5))^2 + ((0) + (0.06274414)(0.5))^2 \\
&= 0.250984206.
\end{aligned}
$$

That was a lot of work, and we haven't even computed the estimate y_1 yet! We finally have

$$
\begin{aligned}
y_1 &= y_0 + \frac{\Delta x}{6} (k_1 + 2k_2 + 2k_3 + k_4) \\
&= (0) + \frac{0.5}{6} ((0) + 2(0.0625) + 2(0.06274414) + (0.250984206)) \\
&= 0.041789373.
\end{aligned}
$$

At this point, we are probably about to throw our hands up and swear off the Runge-Kutta method once and for all. This was a pile of work just to do one time-step! Before we despair too much, however, we should recognize that all the work have done is easily programmed into a computer, and that is exactly what is done in application.

Letting my laptop do the rest of the work, in a fraction of a second we have the following estimates:

Δx	$y(1.5)$	error	steps
0.5	1.521061677	0.00361414	3
0.1	1.517473413	0.000025876	15
0.01	1.517447548	0.000000004	150
0.001	1.517447653	0.000000000	1500

The reason we have gone through all of this trouble—or rather, let our computers go through all this trouble—should now be clear. The Runge-Kutta method gives a *significantly* better estimate of the true value per step. No matter how ridiculous we find the amount of computation necessary in each step to be, we cannot escape the *overall* efficiency. We have obtained a better estimate of $y(1.5)$ in three steps of the Runge-Kutta method (error$= 0.00361414$) than we obtained in 1500 iterations of the forward Euler method (error$= 0.0039455$). It should come as no surprise, therefore, to learn that the forward Euler method—while illustrative and intuitive—is never, ever, *ever* use in practice. Even though each step is easy to compute, the overall burden of cumulative errors makes it tremendously inefficient.

You might wonder why we have we classify the Forward Euler Method as first order and the Runge-Kutta method as fourth order. The reason is highly suggested by the error tables we have already considered:

Forward Euler:

Δx	error	steps
0.1	$3.041 \cdot 10^{-1}$	15
0.01	$3.833 \cdot 10^{-2}$	150
0.001	$3.946 \cdot 10^{-3}$	1500

Runge-Kutta:

Δx	error	steps
0.1	$2.588 \cdot 10^{-5}$	15
0.01	$3.818 \cdot 10^{-9}$	150
0.001	$7.278 \cdot 10^{-13}$	1500

Without too much effort, we can see that: (a) for the Forward Euler method, for each decimal place we add to Δx we gain a decimal place of accuracy in the error; and (b) for the Runge Kutta method, for each decimal place we add to Δx we gain *four* decimal places of accuracy in the error.

Another way to state this is that, for the Forward Euler method, we have error $= O(\Delta x)$ while for the Runge-Kutta method, we have error $= O((\Delta x)^4)$ (with the "big-O" notation means that the corresponding quantity is "on the order of"). Note that this works for all decreases in Δx, not just those which are increments of powers of 10! Numerical results like this can be proved rigorously, but will not be studied in any more detail that this in this course. Further details can be found in **Math 143C**.

Suggested Problems

1. Consider a mixing tank with a total volume of 20 gallons, initially filled with 10 gallons of pure water. Suppose there is an inflow pipe which pumps in a 0.5 lb/gallon brine (salt/water) mixture at a rate of 4 gallons per minute, and there is an outflow pipe which removes the mixture from the tank at a rate of 2 gallon per minute.

   (a) Use the given information to derive a differential equation which models the amount of salt in the tank.

   (b) Find the general solution of the differential equation derived in part (a).

   (c) How much salt is in the tank when it is full?

2. Consider a filled mixing tank with a volume of 20 gallons. Suppose there is an inflow pipe which pumps in a 0.5 lb/gallon brine (salt/water mixture) at a rate of 2 gallons per minute, and there is an outflow pipe which removes the mixture from the tank at a rate of 2 gallons per minute.

   (a) Use the given information to derive a differential equation which models the amount of salt in the tank.

   (b) Find the general solution of the differential equation derived in part (a).

   (c) Suppose there is initially no salt in the tank. How much salt is in the tank after ten minutes?

3. Suppose a factory is built upstream of a lake with a volume of 0.5 km^3. The factory introduces a pollutant into the upstream water system. Suppose the affected water system pumps 0.25 km^3 of water into the lake each year and the downstream water system removes water from the lake at the same rate. Suppose the concentration of pollutant in the feeding water system is 40 kg/km^3.

   (a) Set up a first-order differential equation which models the amount of pollutant (in kg) in the lake.

   (b) Suppose that there is initially no pollutant in the lake. How much pollutant is in the lake after (i) one month; (ii) seven months; (iii) five years? What is the limiting amount of pollutant in the lake?

(c) Suppose now that the inflow and outflow rates of the upstream and downstream water systems vary based on the seasons. Suppose this variance can be modeled by the form [volume rate in] = [volume rate out] $= 0.25 + 0.25\cos(2\pi t)$. Derive the corresponding first-order differential equation which models the amount of pollutant (in kg) in Lake Mendota.

(d) Suppose that there is initially no pollutant in the lake. Under the assumptions of part (c), determine how much pollutant is in the lake after (i) one month; (ii) seven months; (iii) five years. What is the limiting amount of pollutant in the lake? Is it the same as in part (b)? Does it converge to this value faster or slower than part (b)? Provide a brief explanation for the observed differences.

4. Use the forward Euler method with the given step sizes to approximate the given values for the given initial value problems:

(a) Estimate $y(5)$
$\Delta x = 5, 2.5, 1$
$$\begin{cases} y' = 0.5y(1-y) \\ y(0) = 0.1 \end{cases}$$

(b) Estimate $y(1)$
$\Delta x = 1, 0.5, 0.1$
$$\begin{cases} y' = \sin(x) + \cos(y) \\ y(0) = 0 \end{cases}$$

(c) Estimate $y(5)$
$\Delta x = 5, 2.5, 1$
$$\begin{cases} y' = 0.5y(1-y) \\ y(0) = -0.1 \end{cases}$$

(d) Estimate $y(2)$
$\Delta x = 2, 1, 0.5$
$$\begin{cases} y' = xy \\ y(0) = 1 \end{cases}$$

5. Use the fourth-order Runge Kutta method with the given step sizes to approximate the given values for the given initial value problems:

(a) Estimate $y(1)$
$\Delta x = 0.5$
$$\begin{cases} y' = y^2 - x \\ y(0) = 1 \end{cases}$$

(b) Estimate $y(2)$
$\Delta x = 1$
$$\begin{cases} y' = -\ln(xy) \\ y(0) = 0 \end{cases}$$

6. Consider the following initial value problem:

$$\begin{cases} y' = 2y(4-y) \\ y(0) = 2 \end{cases}$$

(a) Draw the slope field in the (x, y)-plane.

(b) Determine the particular solution to the IVP.

(c) Determine the values of the solution up to $x = 1$ using the step sizes $\Delta x = 1$, $\Delta x = 0.5$, $\Delta x = 0.25$, $\Delta x = 0.2$, and $\Delta x = 0.1$. What do you observe in each case? Does this match the qualitative behavior of the corresponding solution through $y(0) = 2$? Can you explain why this might occur?