

6. Discrete Models (One Variable)

In this chapter we will investigate models that are derived from assumptions, as opposed to using a least squares fit. In particular, we will look at models of one variable, where the independent variable (usually time) is discrete. Models that are discrete in nature and consider quantities that change over time are called *Discrete Dynamical Systems (DDS)*.

In Sections 6.1 and 6.2 we will derive models for the drug content in the blood and for managing credit, both examples of Discrete Dynamical Systems. Section 6.3 introduces classifications for such models, in particular the distinction between linear and nonlinear DDS. An explicit solution for linear first-order Discrete Dynamical Systems will be derived in Section 6.4. This explicit solution will be used to determine the long-term model behavior and to make model predictions. Sections 6.5 and 6.6 introduce equilibrium values and ways to determine their stability for linear and nonlinear first-order DDS, respectively. In the last two sections of this chapter, Sections 6.7 and 6.8, we will look at exponential and logistic models, in the context of population growth and the spread of a disease.

6.1 A Simple Model for the Amount of a Drug Remaining in the Blood

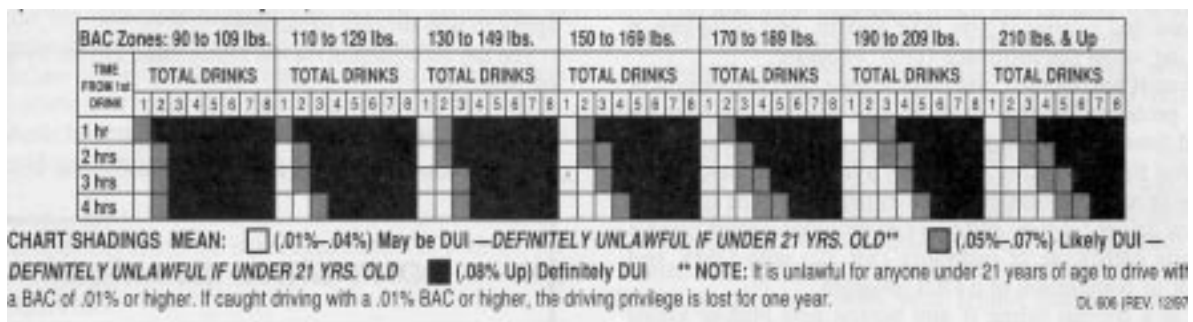
We will start with a simple model to determine the amount of a drug that remains in the body's system at a given time. To derive such a model, we need to understand the mechanism behind the way a drug breaks down and is removed from the blood by the kidneys. This waste removal function of the kidney is vital - people whose kidneys function only partially or not at all have to have a transplant. While waiting for a transplant, they need to undergo dialysis numerous times a week for several hours. In hemodialysis, the patient's blood is pumped through a machine that acts as an artificial kidney. Impurities and wastes are removed from the blood which is then returned to the patient's body. Being on hemodialysis is like having a healthy, functioning liver for about 9 hours a week. An alternative method is peritoneal dialysis, where the patient feeds a solution into his or her abdomen. This solution provides the body with chemicals the liver doesn't produce and removes wastes when the solution drains out of the body. This procedure takes about 45 minutes each time and needs to be done several times a day. It is more flexible than hemodialysis, but carries the risk of inflammation.

How does the kidney cleanse the blood? As the blood flows through the kidneys, the blood pressure forces some of the plasma to filter out of the bloodstream into part of the kidney. Blood

cells and large molecules are retained in the blood, whereas minerals, digested food, and waste products are taken out. Water and other substances required by the body such as glucose, chloride, and sodium are returned to the blood. The remaining material is transported out of the kidney in the form of urine. About 20 % of the blood plasma that enters the kidney is filtered and about 99% of the fluid portion of the filtrate is returned to the body. One cycle of moving all the body's blood once through the kidney takes about 45 minutes.

Why would we be interested in a model on the amount of a drug remaining in the blood? Beneficial drugs is one of the main applications of this type of model. For example, every time you see an advertisement that compares how much longer painkiller A is effective compared to painkiller B, information about the drug level is involved: If the amount of pain medication falls below a certain minimum level, the drug will no longer be effective. On the other hand, pain medication warns of the severe effects of an overdose: "In case of accidental overdose, contact a doctor or a poison control center immediately." Likewise, a patient who has to take medication to keep the blood from clotting needs to have a minimal amount of that medication in his or her system. Too much of the blood thinning medication is dangerous; in case of an injury, the patient may bleed to death. Generally, for most drugs, there is a range between an upper and a lower level for which the drug is both safe and effective. In order to administer the medication in such a way that it stays within these limits, we need to have a model that tells us how much of a drug remains in the blood. Based on such a model, a schedule for delivery of the medication can be devised.

By contrast, there are drugs that you would like to fall (or keep) below a certain level, such as your blood alcohol level after a party or a visit to a bar. Every time you renew your car registration, the DMV sends a flyer with a chart, like the one below, indicating the blood alcohol level depending on the person's body weight, the number of drinks consumed, and the time since the first drink. The latter influences how much of the alcohol has been removed; thus, the chart is based on a model indicating the remaining alcohol (drug) level in the blood over time.



In order to develop a model for the total amount of a drug in the blood, we need to make some assumptions. From the previous description of the kidney mechanism we can infer that a certain percentage of the drug is removed from the blood that enters the kidney. This will be our first assumption. The second assumption we will make is that the drug is uniformly distributed within the blood, and that a fixed percentage breaks down in a given time period. This assumption is not entirely obvious, but it simplifies the model. These two assumptions together ensure that in a given time segment, the **same** percentage of the total drug amount is removed. Let's summarize:

- A1:** A certain percentage of the drug amount entering the kidney is removed.
A2: The drug is uniformly distributed in the blood, and a given percentage of it breaks down in a fixed time period (such as one hour).

Now we need to translate these assumptions into a mathematical model. Before we can proceed, we need to identify the variables for our problem (determining the amount of drug left in the blood at a given time). The input (independent) variable should be time, and the output (dependent) variable should be the amount of drug remaining in the blood. Let's assume that 10% of the drug breaks down and is removed every two hours (the percentage and time period will depend on the specific drug). Thus, we know that after two hours, 90% of the original amount is left, and we can specify the drug amount after a two-hour period. This suggests that time should be measured in two-hour segments, i.e., we select a discrete model. Let's define the variables:

$$D(n) = \text{drug amount (in ml) after } n \text{ two-hour segments}$$
$$D(0) = \text{initial amount of the drug (in ml)}$$

It is very important to carefully state the meaning of the variables, as well as the units. First of all, function values do not make sense without units. (Are we talking about milliliters or liters? That makes a huge difference!) Furthermore, they often help in checking whether the model equations make sense by having the same units on both sides of the equation. If this is not the case, then the model equation is definitely incorrect.

Let's see what happens to the amount of drug in the body at different time intervals. After two hours (= 1 two-hour period $\Rightarrow n = 1$) we get

$$\text{amount of drug} = \text{previous amount} - 10\% \text{ of previous amount}$$

Translating this verbal description into mathematical terms leads to

$$D(1) = D(0) - .10 D(0)$$

$$D(1) = .90 D(0)$$

After 4 hours (= 2 two-hour periods $\Rightarrow n = 2$)

$$D(2) = D(1) - .10 D(1)$$

$$D(2) = .90 D(1)$$

After 6 hours (= 3 two-hour periods $\Rightarrow n = 3$)

$$D(3) = D(2) - .10 D(2)$$

$$D(3) = .90 D(2)$$

Can you detect a general pattern? If we replace the specific values 1, 2, and 3 by the (input) variable n , then we can write the pattern as the following *recursive* or *iterative model equation*:

$$D(n) = .90 D(n-1)$$

We call this equation iterative or recursive because the new output value is computed from the previous output value, as opposed to being a function of the input variable alone. This is a common feature of models that are derived from the paradigm

$$\mathbf{\text{new} = \text{old} + \text{change}}$$

In our case, the change in drug level stems from the breaking down of 10% of the current drug amount in the blood, leading to

$$\underbrace{D(n)}_{\text{new}} = \underbrace{D(n-1)}_{\text{old}} + \underbrace{(-.10 \cdot D(n-1))}_{\text{change}} = .90 \cdot D(n-1)$$

We will use this recursive model equation to answer the following question:

If 20 ml of the drug was injected into a patient's blood at 8 AM, how much of the drug is left in the blood at 8 PM?

When approaching a word problem like this, we first need to extract the information that is given. Secondly, we need to figure out what the question is asking. Instead of trying to immediately express both of these items as mathematical expressions, first state them in words.

Given: Drug amount 20 ml at 8 AM.

Question: Drug amount at 8 PM.

Now rephrase the given information and the question in terms of the input and output variables. Time is measured in two-hour segments counting from 8 AM. This makes 8 AM the starting time ($n = 0$) with an initial amount of 20 ml. Furthermore, we need to translate 8 PM into two-hour segments after 8 AM. Since the difference between 8 PM and 8 AM is twelve hours (=6 two-hour segments), $n = 6$ and we get

Given: $D(0) = 20$ (ml)

Question: $D(6) = ?$

So let's compute the value of $D(6)$. Using the recursive model equation, we get

$$D(6) = (.9)D(5)$$

However, we don't know the value of $D(5)$. Thus, we also have to apply the recursive model equation for $D(5)$:

$$D(5) = (.9)D(4)$$

This is no help either, as we do not know the value of $D(4)$. Therefore, we apply the recursive formula again (and again):

$$D(4) = (.9)D(3)$$

$$D(3) = (.9)D(2)$$

$$D(2) = (.9)D(1)$$

$$D(1) = (.9)D(0)$$

Finally, we are in business, as we know the value of $D(0)$. We get

$$D(1) = (.9)D(0) = (.9)20 = 18.$$

Working our way backwards, we can compute the values of $D(2)$, $D(3)$, $D(4)$, $D(5)$, and finally $D(6)$:

$$D(2) = (.9)D(1) = (.9)18 = 16.2$$

$$D(3) = (.9)D(2) = (.9)16.2 = 14.58$$

$$D(4) = (.9)D(3) = (.9)14.58 = 13.122 \approx 13.12$$

$$D(5) = (.9)D(4) = (.9)13.122 = 11.8098 \approx 11.81$$

$$D(6) = (.9)D(5) = (.9)11.8098 = 10.62882 \approx 10.63$$

Answer: The amount of the drug remaining in the blood at 8 PM is 10.63 ml.

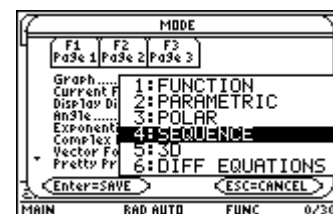
You may have predicted that the amount at 8 PM is 8 ml, assuming that in each two-hour period an amount of 2 ml (10% of 20 ml) was removed. However, after the first two hours, the total has decreased to 18 ml, and then 10% of 18 ml, 1.8 ml, was removed. Thus, in every consecutive two-hour period, a smaller amount is removed. Let's compare the effects of a fixed amount versus a fixed percentage of the drug being removed every two hours. The amounts in the first row give the initial amounts and all results are rounded to two decimal places.

# of two-hour segments	amount removed (fixed amount)	amount remaining	amount removed (10 %)	amount remaining
0		20		20
1	2	18	2	18
2	2	16	1.8	16.2
3	2	14	1.62	14.58
4	2	12	1.46	13.12
5	2	10	1.31	11.81
6	2	8	1.18	10.63

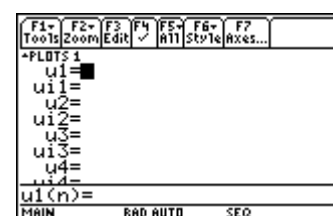
The two models show quite different behavior. In the case where a fixed amount is removed, the drug level would reach 0 in another 8 hours (4 two-hour segments). However, it is not clear when or even if that will happen in the case where 10% is removed. Thus, it is very important to distinguish between a fixed **amount** and a fixed **percentage**. In the latter case, when the total amount of drug decreases, so does the amount that is removed!

We will now use the sequence tools of the TI-89 to compute the sequence of output values (= drug levels) using the recursive model equation $D(n) = .90 \cdot D(n-1)$.

1. We first need to put the calculator into sequence mode. Press **MODE** and access the menu for Graph by using the **⏵** key. Move down to **4:Sequence**.

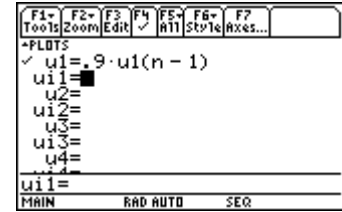


2. Press **ENTER** twice. To define the sequence, press **⏵[Y=]**. This will open up a screen in which you can define sequences, just like we defined y-functions before. The screen looks slightly different because recursive sequences require a starting or initial value. Sequences are denoted as u_1, u_2, u_3 , etc., and their respective initial values are denoted as u_{i1}, u_{i2}, u_{i3} , etc.

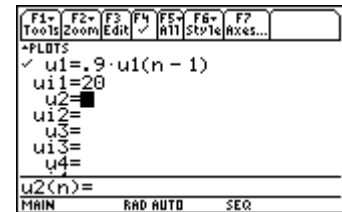


3. We now define the sequence. Remember that the name of the sequence is u_1 ; thus, we replace every D in the recursive model equation ($D(n) = .90 \cdot D(n-1)$) by u_1 . Move the cursor to $u_1 =$ and type

\square 90 \times \square alpha \square [U] 1 \square (\square alpha \square [N] \square - 1 \square) \square [ENTER].



4. To get the sequence computation started, we need the initial value. In our case, the initial value was given by 20. The cursor is already at the correct position. Type 20, followed by \square [ENTER].



5. Now that the sequence is defined, we can look at a table of values. First, we need to set up the table parameters. Press \blacklozenge [TblSet]. Two parameters need to be defined: **tblStart** gives the value for n at which the table display begins, and Δ **tbl** is the difference between two consecutive n -values in the table. As we want to see all the values from the initial time, set **tblStart** to **0** and Δ **tbl** to **1**. **Graph** \leftrightarrow **Table** should always be set to **OFF** \rightarrow and **Independent** should always be **AUTO** \rightarrow .



6. Press \square [ENTER]. This will return you to the previous screen. Press \blacklozenge [TABLE] to see the table of values.

n	u_1		
0.	20.		
1.	18.		
2.	16.2		
3.	14.58		
4.	13.122		

Calculator screen showing a table of values for n from 0 to 4. The status bar at the bottom shows 'n=0.', 'MAIN', 'RAD AUTO', and 'SEQ'.

7. To see additional values, in particular the one for $n = 6$ (which corresponds to 8 PM), use the \blacktriangledown key or the \square 2nd \blacktriangledown keys (which move down one screen at a time). Note that there is no end to this table--the further you move down, the more values get computed.

n	u_1		
5.	11.81		
6.	10.629		
7.	9.5659		
8.	8.6093		
9.	7.7484		

Calculator screen showing a table of values for n from 5 to 9. The status bar at the bottom shows 'n=6.', 'MAIN', 'RAD AUTO', and 'SEQ'.

Sometimes you may be interested only in specific values of n and may not want to have to move through the whole list of values (especially since you have to wait until all the intermediate values are displayed on the screen before you can move to the next screen). Or, you may want to see only every tenth or hundredth value. In both cases, you can modify the table display by changing the value of Δ tbl.

8. To just see the drug level at time $n = 6$, press \blacklozenge [TblSet]. Note that tblStart shows, by default, the value of n that is currently displayed at the top of the table and may have changed from the value of 0 you entered earlier. Set **tblStart** to **0** and **Δ tbl** to **6**.
9. Press \square [ENTER] twice. You now see only every sixth value displayed in the table; the first two values of n give the initial value and the value of interest for $n = 6$.
10. To see every tenth value, press \blacklozenge [TblSet] and set **tblStart** to **0** and **Δ tbl** to **10**. Press \square [ENTER] twice. Now the table only displays values for values of n that are multiples of 10.



n	u1		
0.	20.		
6.	10.629		
12.	5.6486		
18.	3.0019		
24.	1.5953		

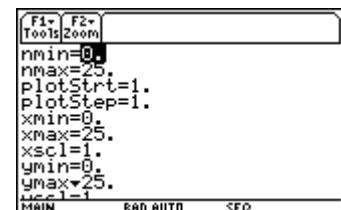
n=6.
MAIN RAD AUTO SEQ

n	u1		
0.	20.		
10.	6.9736		
20.	2.4315		
30.	.84782		
40.	.29562		

n=0.
MAIN RAD AUTO SEQ

Having created a table of values, we may also want to see a graph of the drug level remaining in the blood. Again, we need to be in sequence mode, just as before for creation of the table.

1. Press \square [MODE] and verify that the Graph mode is set to SEQUENCE \rightarrow .
2. Press \square [ENTER], then \blacklozenge [Y=] to return to the screen where you defined the recursive sequence. Press \square [2nd][F7]. Use the \blacktriangleright key to open the **Axes** menu and select **1:TIME**. This will ensure that the horizontal axis displays time ($= n$).
3. Press \square [ENTER] twice. Press \blacklozenge [WINDOW] to set the appropriate window parameters. Since we are graphing a sequence, there are additional parameters that need to be defined. The first four parameters, **nmin**, **nmax**, **plotStrt** and **plotStep**, are new, all others have the same meaning as before.



 **TIP:**

nmin and **nmax** give the bounds on the range of n -values that are calculated for the graph. (Only calculated values can be displayed.) **plotStrt** and **plotStep** are the equivalents of **tblStart** and Δtbl (used for table creation), i.e., they indicate which term of the sequence of n -values is the first one to be displayed, and the distance between consecutive (displayed) n -values, respectively. Usually, **plotStrt** is set to **1** (to start plotting with the **first** n -value, namely $n = \text{nmin}$). Do not set **plotStrt** to 0, as this will result in an error message.

4. Set the window parameters to the following values:

$\text{nmin} = 0$ $\text{nmax} = 25$ $\text{plotStrt} = 1$ $\text{plotStep} = 1$

(this choice will display all 26 n -values)

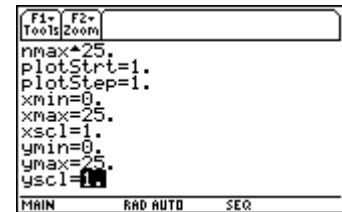
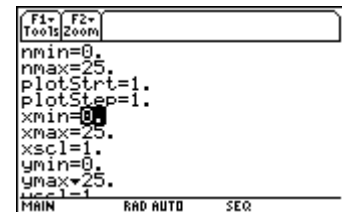
$\text{xmin} = 0$ $\text{xmax} = 25$ $\text{xscl} = 1$

(since the horizontal axis displays the n -values)

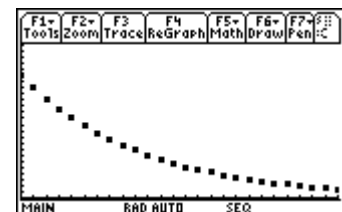
$\text{ymin} = 0$ $\text{ymax} = 25$ $\text{yscl} = 1$

(since drug levels decrease from the initial value of 22 and should not become negative).

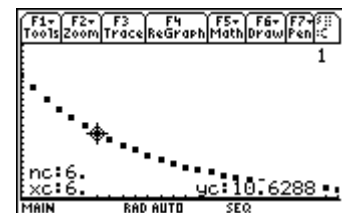
Press **[ENTER]** to confirm your choice of values.



5. Press **[GRAPH]** to see the graph of the drug levels over the time period 8 AM ($n = 0$) to 10 AM ($n = 25$) two days later.



6. We can now use the trace feature to explore the graph. Press **[F3]** to activate the trace. You will see three values displayed, namely **nc**, **xc**, and **yc**. In our case, **nc = xc**, since n -values are plotted on the horizontal (x -) axis. **yc = u1(nc)** gives the drug level at time **nc**. Recall that you can move the trace cursor along the graph by using the **[RIGHT]** and **[LEFT]** keys.



Now that you know how to use the sequence tools of the TI-89 to both create and graph a sequence of values from the recursive model equation, you can use them to answer the questions in the next activity.

Activity 6.1.1

Use the sequence tools of the TI-89 to answer the following questions for the example about the drug amount remaining in the blood:

- a) When has the amount of drug remaining in the patient's body been reduced to half of the original amount?
- b) At what time will the drug level be reduced to 1 ml?
- c) Repeat a) and b) if the initial drug amount is 30 ml instead of 20 ml. Compare your answers to those in a) and b). Summarize your findings.

You should have noticed that it takes the same amount of time for the drug to decrease to half the original amount, whether the initial drug amount was 20 ml or 30 ml. This is a special property resulting from the assumption of the fixed percentage change. We will return to it after we have derived the explicit solution for the model.

The *explicit model solution* is an expression for the output values in terms of the **input** values (explicit function). By contrast, the recursive or iterative model equation is expressed as a function of previous **output** values (implicit function). To understand the differences between the two types of functions, let's compare the recursive model equation with the explicit model solution. The table below indicates the advantages and disadvantages of each of these two functions.

Iterative model equation	Explicit model solution
Output value expressed as a function of earlier output value(s)	Output value expressed as a function of the input value
Good for model setup, especially when using the paradigm $\text{new} = \text{old} + \text{change}$	Bad for model setup; usually no direct relation between output value and input value known (otherwise we would have the model already)
Bad for computations (tedious by hand, lengthy with calculator)	Good for computations and to determine the long-term behavior of the model

How do we derive this explicit solution? We start out just as we did when we wanted to compute the value of $D(6)$, except we will not substitute the intermediate values. Any pattern that results is a candidate for the explicit solution. To verify that the particular pattern is actually the explicit solution, we substitute the pattern into the iterative model equation.

n	$D(n)$	$D(n)$ simplified
0	$D(0)$	$D(0)$
1	$D(1) = (.9)D(0)$	$D(1) = (.9)D(0)$
2	$D(2) = (.9)D(1) = (.9)[(.9)D(0)] = (.9)^2 D(0)$	$D(2) = (.9)^2 D(0)$
3	$D(3) = (.9)D(2) = (.9)[(.9)^2 D(0)] = (.9)^3 D(0)$	$D(3) = (.9)^3 D(0)$
4	$D(4) = (.9)D(3) = (.9)[(.9)^3 D(0)] = (.9)^4 D(0)$	$D(4) = (.9)^4 D(0)$
5	$D(5) = (.9)D(4) = (.9)[(.9)^4 D(0)] = (.9)^5 D(0)$	$D(5) = (.9)^5 D(0)$
6	$D(6) = (.9)D(5) = (.9)[(.9)^5 D(0)] = (.9)^6 D(0)$	$D(6) = (.9)^6 D(0)$

The pattern that seems to emerge is

$$D(n) = (.9)^n D(0)$$

This also works for $n = 0$ (as $(.9)^0 = 1$) and $n = 1$ (as $(.9)^1 = .9$). Thus, we can propose the following explicit solution:

$$D(n) = (.9)^n D(0)$$

Notice that now the value of $D(n)$ only depends on n and the value of $D(0)$. Since the latter is known, this pattern satisfies the requirement for an explicit solution. To verify that this is indeed the explicit solution, we substitute the expression for $D(n-1)$ into the iterative model equation and check whether we get the correct pattern for time n :

$$\begin{aligned}
 D(n) &= (0.9)D(n-1) && \text{recursive model equation} \\
 &= (0.9)[(0.9)^{n-1}D(0)] && \text{substitute pattern for } D(n-1) \\
 &= (0.9)^n D(0) && \text{simplify} \\
 D(n) &= (0.9)^n D(0) && \text{correct pattern for time } n
 \end{aligned}$$

This shows that the pattern holds true, and therefore, we have found the explicit solution. We can actually generalize our result by replacing the specific value of 0.9 by a constant, since nothing in our derivation was based on this specific value.

Theorem 1 (Explicit Solution)

A model with iterative model equation of the form

$$x(n) = a \cdot x(n-1), \quad a \neq 0$$

has explicit solution

$$x(n) = a^n x(0).$$

We thus have succeeded in finding an explicit expression for $x(n)$ that depends only on the constant a , the initial value $x(0)$ and the value of the independent variable n .

Let's see the difference between finding an answer using the iterative model equation and the explicit solution. Using the iterative equation repeatedly, we found that $D(6) \approx 10.6$ (rounded to one decimal place). If we use the explicit solution, we simply apply Theorem 1. Our iterative model equation was given by

$$D(n) = (.9)D(n-1).$$

Thus, we can read off that $a = 0.9$. We also know that $D(0) = 20$. Theorem 1 now tells us that we can compute $D(n)$ as

$$D(n) = (.9)^n \cdot 20$$

Therefore,

$$D(6) = (.9)^6 \cdot 20 = (.53144) \cdot 20 = \mathbf{10.6288}.$$

Note how much easier it is to use the explicit solution rather than the iterative model equation, especially if we have to do the computations by hand. In addition, there are questions that cannot be answered by only using the iterative model equation; for example, "What is the long-term behavior of the model?" or "When will the drug level decrease to half the original amount?"

To answer the last question in general, we set

$$\begin{aligned} \underbrace{x(n)}_{\text{amount at time } n} &= \underbrace{\frac{1}{2} x(0)}_{\text{half of original amount}} \\ a^n x(0) &= \frac{1}{2} x(0) && \text{substitute explicit solution for } x(n) \\ a^n &= \frac{1}{2} && \text{divide by } x(0) \neq 0 \end{aligned}$$

Using analytical methods for solving exponential equations, we get

$$n = \frac{\ln(\frac{1}{2})}{\ln(a)}$$

where \ln refers to the natural logarithm. (On the TI-89, this function can be accessed by pressing $\boxed{2\text{nd}}\boxed{[\text{LN}]}$.) This specific time, at which exactly half of the original drug level remains, is called *half-life* and is often denoted as $n_{1/2}$ or $t_{1/2}$ (in a continuous model).

In our example, $a = 0.9$, therefore,

$$n_{1/2} = \frac{\ln(\frac{1}{2})}{\ln(0.9)} = \frac{-0.693147}{-0.105361} = 6.57881.$$

Recall that n is the number of two-hour periods; thus, the drug amount has decreased to half the original amount after $2 \cdot 6.57881 = 13.16$ hours. Note that the half-life depends only on the value of a , not on the initial amount $x(0)$. This is the reason why it takes the same amount of time, no matter what initial amount we start from.

Note that the explicit solution to the model $x(n) = a \cdot x(n-1)$ is an exponential function whenever $a > 0$:

$$x(n) = a^n \cdot x(0).$$

Thus, we always get an exponential model if we start with a **proportional** relationship in the iterative model equation. (Recall that x and y are *proportional* if there is a constant k such that $x = k \cdot y$.) Therefore, exponential functions are associated with proportional growth (if $a > 1$) or proportional decay (if $0 < a < 1$).

We will now look at the long-term behavior of models such as the drug removal problem. In this type of model, we have two parameters, namely a and $x(0)$. Parameters are constants, (as opposed to variables) and are assumed to be fixed for a given problem. However, they can change from problem to problem, and we are often interested in how they influence the long-term behavior of the model.

In the next activity, you will explore what happens to the model values in the long run for specific values of a and $x(0)$. From these examples, we will make predictions as to what happens in general.

Activity 6.1.2

For each of the eight possibilities of combining the two iterative model equations (a) and (b) with the four initial values (i-iv), compute a table of values and display their graph to determine what happens to the output-values in the long run. Start by defining the two sequences on the [Y=] screen, and change the initial values for the four different cases. Use \blacklozenge [TblSet] and \blacklozenge [TABLE] to create the table of values and \blacklozenge [WINDOW] and \blacklozenge [GRAPH] to display the corresponding graphs. For each combination write a short paragraph summarizing the long-term behavior.

a) $x(n) = 2 \cdot x(n-1)$

b) $x(n) = -\frac{3}{4} \cdot x(n-1)$

i) $x(0) = 5$

ii) $x(0) = 0.2$

iii) $x(0) = 0$

iv) $x(0) = -1$

In these examples, you should have seen quite different long-term behaviors for the two iterative model equations. Either the values for $x(n)$ became large without bound (positive or negative), tended toward 0, or were always 0 (in the special case where $x(0) = 0$). Sometimes, the values for $x(n)$ fluctuated between positive and negative values. Can we predict which type of behavior results just by looking at the formula for the explicit solution? This formula contains two factors, namely $x(0)$ and a^n . The first factor is not going to change over time; however, the second one depends on n and will change as n gets large.

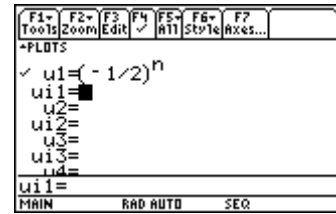
Activity 6.1.3

Complete the table given below. For each interval given in the left column, select a specific value for a from that interval (e.g. you may select $a = -\frac{1}{2}$ for the interval $-1 < a < 0$). Then use the TI-89 commands below for your specific value of a to define the sequence of values for a^n . Graph the sequence and determine what happens to the value of a^n as n gets larger.

Press **MODE** and select SEQUENCE→ as the Graph mode. Press **Y=** and define a new sequence $u1(n) = (-\frac{1}{2})^n$ by typing

$$\left(\left(- \right) 1 \div 2 \right) \wedge \alpha [N]$$

Note that you need not define an initial value as this is not a recursive sequence. Repeat with other values of a . Make sure to set appropriate window parameters before graphing.



	Behavior of a^n as a gets large
$a < -1$	a^n grows w/o bound and toggles between positive and negative values
$a = -1$	a^n toggles between -1 and 1
$-1 < a < 0$	
$a = 0$	$a^n = 0$ always
$0 < a < 1$	
$a = 1$	
$a > 1$	

Now that we have discovered the effect of a^n , we have to determine what happens when we multiply by $x(0)$. As $x(0)$ is a fixed number, the only effect it can have is to change the sign of the answer (if $x(0) < 0$) or to make the result 0 always (if $x(0) = 0$). If $x(0) > 0$, then the overall behavior does not change. These results are summarized in the table below. We now have completely determined the long-term behavior of the model. Thus, we can predict what will happen as soon as we know the value of a and $x(0)$.

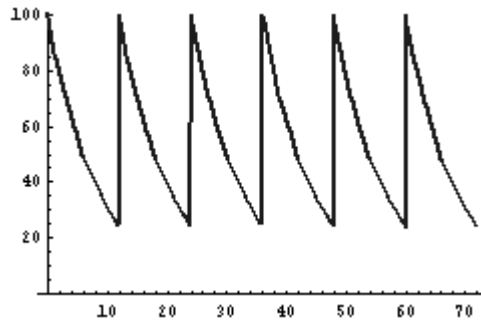
Parameter values	Behavior
$x(0) = 0$ (or $a = 0$)	$x(n) = 0$ always
$x(0) \neq 0$ and $a = 1$	$x(n) = x(0)$ always
$x(0) \neq 0$ and $a = -1$	$x(n)$ toggles between $x(0)$ and $-x(0)$
$x(0) \neq 0$ and $ a > 1$	$x(n)$ will increase or decrease without bound, toggling between positive and negative values if a is negative (e.g. third graph). <div style="text-align: center;"> </div>
$x(0) \neq 0$ and $0 < a < 1$	$x(n)$ will tend toward zero, toggling between positive and negative values if a is negative (e.g. second graph). <div style="text-align: center;"> </div>

Let's see how we can use this summary. Going back to the original example with iterative model equation

$$D(n) = (.9)D(n-1)$$

with $a = .9$ and $D(0) = 20$, we can read off the long-term behavior (our values fit the description in the last row: $D(0) \neq 0$ and $0 < |.9| < 1$). We see that the drug amount decreases to zero in the long run, and therefore, decreases eventually below any given value.

This is good news for party-goers, as they eventually will be able to drive without risking a DUI conviction. However, a patient who needs to have a minimal amount of a beneficial drug in his or her system needs to get a new dose before the drug amount in the blood falls below the minimal effective level. Thus, a physician has to determine how often the drug should be administered and in which amount. The resulting drug level follows a “see-saw” function as illustrated in the graph below.



In the next activity, you will develop such a drug administration schedule. To aide you with this task, you may use the fact that the time it takes to reduce the original amount of the drug, $x(0)$, to $k \cdot x(0)$ (where $0 < k < 1$) is given by $\ln(k)/\ln(a)$, where the function \ln represents the natural logarithm, which can be accessed via [2nd](#)[LN]. (This formula is similar to the half-life, where $k = \frac{1}{2}$). Keep in mind the following practical issues:

- 1) The time intervals should be relatively convenient (full or half hours, rather than 2.67 hours, for example) to make the nurse's job easier.
- 2) You should build in a safety margin, i.e., not let the drug amount decrease to the minimal level of 20 ml before administering the next dose (in case the nurse is late).

Activity 6.1.4

Assume the amount of a certain drug remaining in the blood (in ml) is described by the iterative model equation $D(n) = .89D(n-1)$, where $D(n)$ gives the drug amount n hours after the initial drug dose of 100 ml. Furthermore, the safe and effective levels for this drug are between 20 ml (minimum) and 105 ml (maximum). If a patient's drug level is to stay within the safe and effective zone and the number of times the drug is given is to be kept small, devise a good (see remarks above) drug administration schedule (i.e., indicate the drug amount per dose and the interval between drug doses).

References:

Brum, G.D., McKane, L. and G. Karp, *Biology Fundamentals*, John Wiley & Sons, 1995

WWW References for Kidney

http://www.med.umich.edu/trans/develop/twu/twu_content/kidney_failure.html

<http://www.niddk.nih.gov/health/kidney/kidney.htm>

<http://www.nkfg.org/kidndownt.htm>

http://www.imcpl.lib.in.us/nov_kidn.htm

<http://www.ultranet.com/~jkimball/BiologyPages/K/Kidney.html>

WWW References for Alcohol Charts

<http://www.dmv.ca.gov/pubs/hdbk/98dl600.pdf> (page 84)

<http://www.health.org/pubs/qdocs/alcohol/bac-chrt.htm>