# *Section* 3.5 – Applications of Recurrence Relations

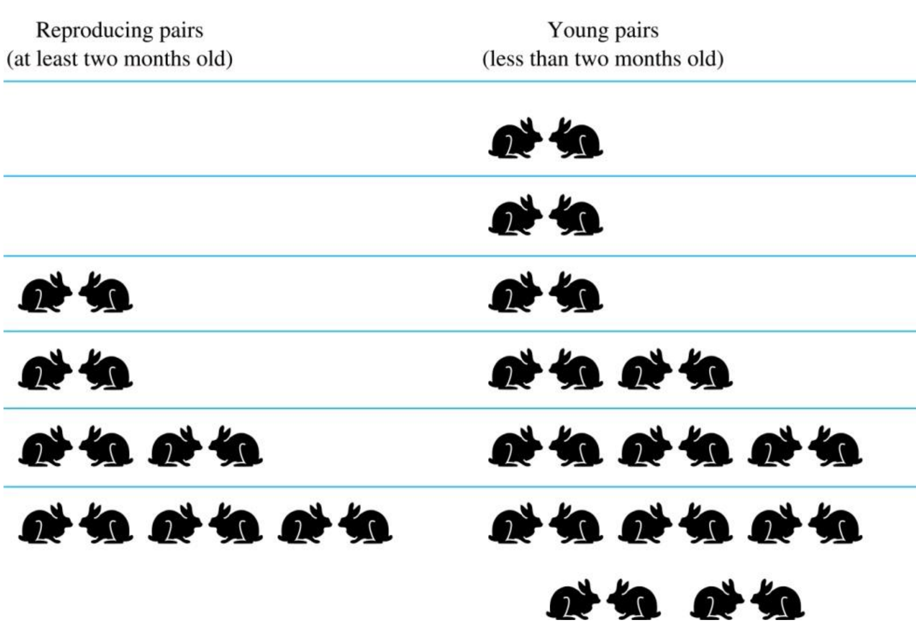
## Modeling with Recurrence Relations

### *Definition*

A *recurrence relation* for the sequence $\{a_n\}$ is an equation that expresses $a_n$ in terms of one or more of the previous terms of the sequence, namely, $a_0, a_1, \ldots, a_{n-1}$, for all integers $n$ with $n \geq n_0$, where $n_0$ is a nonnegative integer

### *Example*

A young pair of rabbits (one of each gender) is placed on an island. A pair of rabbits does not breed until they are 2 months old. After they are 2 months old, each pair of rabbits produces another pair each month. Find a recurrence relation for the number of pairs of rabbits on the island after *n* months, assuming that rabbits never die.




| Month | Reproducing pairs | Young pairs | Total pairs |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 2 | 0 | 1 | 1 |
| 3 | 1 | 1 | 2 |
| 4 | 1 | 2 | 3 |
| 5 | 2 | 3 | 5 |
| 6 | 3 | 5 | 8 |

***Modeling the Population Growth of Rabbits on an Island***

***Solution***

Let $f_n$ be the number of pairs of rabbits after *n* months.

- There are is $f_1 = 1$ pairs of rabbits on the island at the end of the first month.
- We also have $f_2 = 1$ because the pair does not breed during the first month.
- To find the number of pairs on the island after *n* months, add the number on the island after the previous month, $f_{n-1}$, and the number of newborn pairs, which equals $f_{n-2}$, because each newborn pair comes from a pair at least two months old.

Consequently the sequence $\{f_n\}$ satisfies the recurrence relation $f_n = f_{n-1} + f_{n-2}$ for $n \geq 3$ with the initial conditions $f_1 = 1$ and $f_2 = 1$.

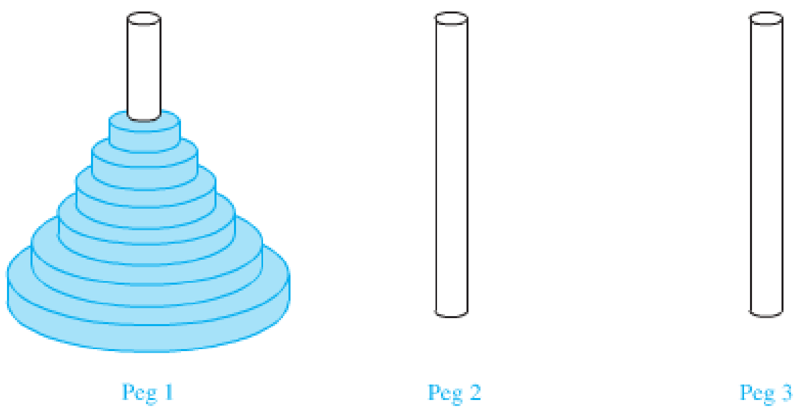
The number of pairs of rabbits on the island after $n$ months is given by the $n$th Fibonacci number.

## The Tower of Hanoi

In the late nineteenth century, the French mathematician Édouard Lucas invented a puzzle, called the Tower of Hanoi, consisting of three pegs on a board with disks of different sizes. Initially all of the disks are on the first peg in order of size, with the largest on the bottom

***Rules***: You are allowed to move the disks one at a time from one peg to another as long as a larger disk is never placed on a smaller.

***Goal***: Using allowable moves, end up with all the disks on the second peg in order of size with largest on the bottom.



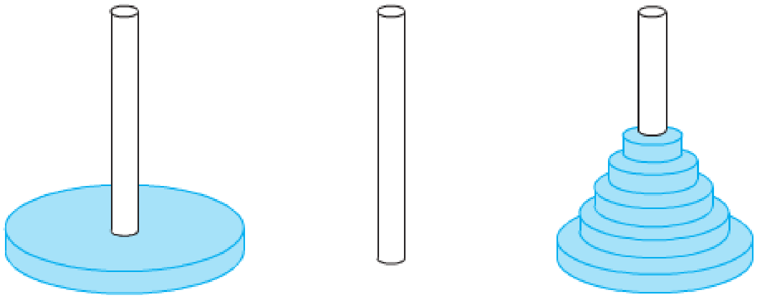

***Solution***

Let $\{H_n\}$ denote the number of moves needed to solve the Tower of Hanoi Puzzle with $n$ disks. Set up a recurrence relation for the sequence $\{H_n\}$. Begin with $n$ disks on peg 1. We can transfer the top $n-1$ disks, following the rules of the puzzle, to peg 3 using $H_{n-1}$ moves.

First, we use 1 move to transfer the largest disk to the second peg. Then we transfer the $n-1$ disks from peg 3 to peg 2 using $H_{n-1}$ additional moves. This cannot be done in fewer steps. Hence,

$$H_n = 2H_{n-1} + 1$$

The initial condition is $H_1 = 1$ since a single disk can be transferred from peg 1 to peg 2 in one move.




We can use an iterative approach to solve this recurrence relation by repeatedly expressing $H_n$ in terms of the previous terms of the sequence.

$$
\begin{aligned}
H_n &= 2H_{n-1} + 1 \\
&= 2\left(2H_{n-2} + 1\right) + 1 \\
&= 2^2 H_{n-2} + 2 + 1 \\
&= 2^2\left(2H_{n-3} + 1\right) + 2 + 1 \\
&= 2^3 H_{n-3} + 2^2 + 2 + 1 \\
&\vdots \quad \vdots \\
&= 2^{n-1}H_1 + 2^{n-2} + 2^{n-3} + \cdots + 2 + 1 && \textit{since } H_1 = 1 \\
&= 2^{n-1} + 2^{n-2} + 2^{n-3} + \cdots + 2 + 1 \\
& && \textit{Using the formula for the sum of the terms of geometric series} \\
&= 2^n - 1
\end{aligned}
$$

There was a myth created with the puzzle. Monks in a tower in Hanoi are transferring 64 gold disks from one peg to another following the rules of the puzzle. They move one disk each day. When the puzzle is finished, the world will end.
Using this formula for the 64 gold disks of the myth,

$2^{64} - 1$ = 18,446, 744,073, 709,551,615 days are needed to solve the puzzle, which is more than 500 billion years.

Reve's puzzle (proposed in 1907 by Henry Dudeney) is similar but has 4 pegs. There is a well-known unsettled conjecture for the minimum number of moves needed to solve this puzzle.

## *Example*

Find a recurrence relation and give initial conditions for the number of bit strings of length $n$ without two consecutive 0s. How many such bit strings are there of length five?

***Solution***

Let $a_n$ denote the number of bit strings of length $n$ without two consecutive 0s. To obtain a recurrence relation for $\{a_n\}$ note that the number of bit strings of length $n$ that do not have two consecutive 0s is the number of bit strings ending with a 0 plus the number of such bit strings ending with a 1.

Now assume that $n \geq 3$.

The bit strings of length $n$ ending with 1 without two consecutive 0s are the bit strings of length $n-1$ with no two consecutive 0s with a 1 at the end. Hence, there are $a_{n-1}$ such bit strings.

The bit strings of length $n$ ending with 0 without two consecutive 0s are the bit strings of length $n-2$ with no two consecutive 0s with 10 at the end. Hence, there are $a_{n-2}$ such bit strings.

We conclude that $a_n = a_{n-1} + a_{n-2}$ for $n \geq 3$.

The initial conditions are:

$a_1 = 2$, since both the bit strings 0 and 1 do not have consecutive 0s.

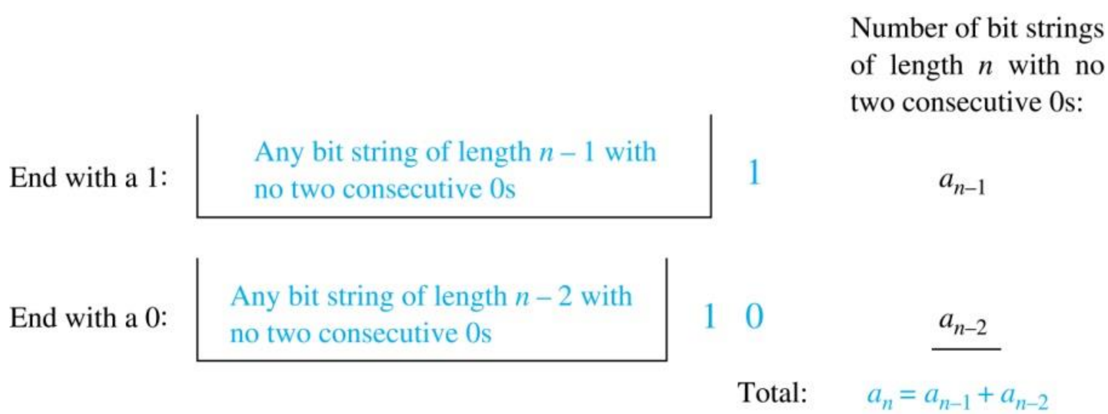
$a_2 = 3$, since the bit strings 01, 10, and 11 do not have consecutive 0s, while 00 does.

To obtain $a_5$, we use the recurrence relation three times to find that:

$$a_3 = a_2 + a_1 = 3 + 2 = 5$$

$$a_4 = a_3 + a_2 = 5 + 3 = 8$$

$$a_5 = a_4 + a_3 = 8 + 5 = 13$$



### *Example*

A computer system considers a string of decimal digits a valid codeword if it contains an even number of 0 digits. For instance, 1230407869 is valid, whereas 1230407869 is not valid. Let $a_0$ be the number of valid *n*-digit codewords. Find a recurrence relation for $a_n$.

***Solution***

Note that $a_1 = 9$ because there are 10 one-digit strings, and only one, namely the string 0, is not valid. A recurrence relation can be derived for this sequence by considering how a valid *n*-digit string can be obtained from strings of *n* – 1 digits. There are two ways to form a valid string with *n* digits from a string with one fewer digit.

1. A valid string with *n* digits can be obtained by appending a valid string of *n* – 1 digits with a digit other than 0. This appending can be done in 9 ways. Hence, a valid string with n digits can be formed in this manner in $9a_{n-1}$ ways.
2. A valid string with *n* digits can be obtained by appending a 0 to a string of length *n* – 1 that is not valid. The number of ways that this can be done equals the number of invalid (*n* – 1)-digit strings. Because there are $10^{n-1}$ strings of length *n* – 1, and $a_{n-1}$ are valid, there are $10^{n-1} - a_{n-1}$ valid *n* –digit strings obtained by appending an invalid string of length *n* – 1 with a 0.

   Because all valid strings of length *n* are produced in one of these two ways, it follows that there are

$$\begin{aligned} a_n &= 9a_{n-1} + \left(10^{n-1} - a_{n-1}\right) \\ &= 8a_{n-1} + 10^{n-1} \end{aligned}$$

Valid strings of length *n*.

### *Example*

Find a recurrence relation for $C_n$, the number of ways to parenthesize the product of $n + 1$ numbers, $x_0 \cdot x_1 \cdot x_2 \cdots \cdot x_n$, to specify the order of multiplication. For example, $C_3 = 5$, since all the possible ways to parenthesize 4 numbers are

$$\left(\left(x_0 \cdot x_1\right) \cdot x_2\right) \cdot x_3 \qquad \left(x_0 \cdot \left(x_1 \cdot x_2\right)\right) \cdot x_3 \qquad \left(x_0 \cdot x_1\right) \cdot \left(x_2 \cdot x_3\right)$$

$$x_0 \cdot \left(\left(x_1 \cdot x_2\right) \cdot x_3\right) \qquad x_0 \cdot \left(x_1 \cdot \left(x_2 \cdot x_3\right)\right)$$

***Solution***

Note that however parentheses are inserted in $x_0 \cdot x_1 \cdot x_2 \cdots \cdot x_n$, one "·" operator remains outside all parentheses. This final operator appears between two of the $n + 1$ numbers, say $x_k$ and $x_{k+1}$. Since there are $C_k$ ways to insert parentheses in the product $x_0 \cdot x_1 \cdot x_2 \cdots \cdot x_k$ and $C_{n-k-1}$ ways to insert parentheses in the product $x_{k+1} \cdot x_{k+2} \cdots \cdot x_n$, we have

$$\begin{aligned} C_n &= C_0 C_{n-1} + C_1 C_{n-2} + \cdots + C_{n-2} C_1 + C_{n-1} C_0 \\ &= \sum_{k=0}^{n-1} C_k C_{n-k-1} \end{aligned}$$

The initial conditions are $C_0 = C_1 = 1$.

# *Exercises* *Section* 3.5 – Applications of Recurrence Relations

**1.** *a*) Find a recurrence relation for the number of permutation of a set with $n$ elements

*b*) Use the recurrence relation to find the number of permutations of a set with $n$ elements using iteration.

**2.** A vending machine dispensing books of stamps accepts only one-dollar coins, \$1 bills, and \$5 bills.

*a)* Find a recurrence relation for the number of ways to deposit n dollars in the vending machine, where the order in which the coins and bills are deposited matter.

*b)* What are the initial conditions?

*c)* How many ways are there to deposit \$10 for a book of stamps?

**3.** *a*) Find a recurrence relation for the number of bit strings of length $n$ that contain three consecutive 0s.

*b*) What are the initial conditions?

*c*) How many bit strings of length seven contain three consecutive 0s?

**4.** *a*) Find a recurrence relation for the number of bit strings of length $n$ that do not contain three consecutive 0s.

*b*) What are the initial conditions?

*c*) How many bit strings of length seven do not contain three consecutive 0s?

**5.** *a*) Find a recurrence relation for the number of ways to climb $n$ stairs if the person climbing the stairs can take one stair or two stairs at a time.

*b*) What are the initial conditions?

*c*) In how many can this person climb a flight of eight stairs