# Discrete Structures

**Advanced Counting Techniques**

**Text book:** Kenneth H. Rosen, Discrete Mathematics and Its Applications

Section: 8.1

# Applications of Recurrence Relations

Section 8.1

# Section Summary

- Applications of Recurrence Relations
  - Fibonacci Numbers
  - The Tower of Hanoi
  - Counting Problems

# Recurrence Relations
## (recalling definitions from Chapter 2)

**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.

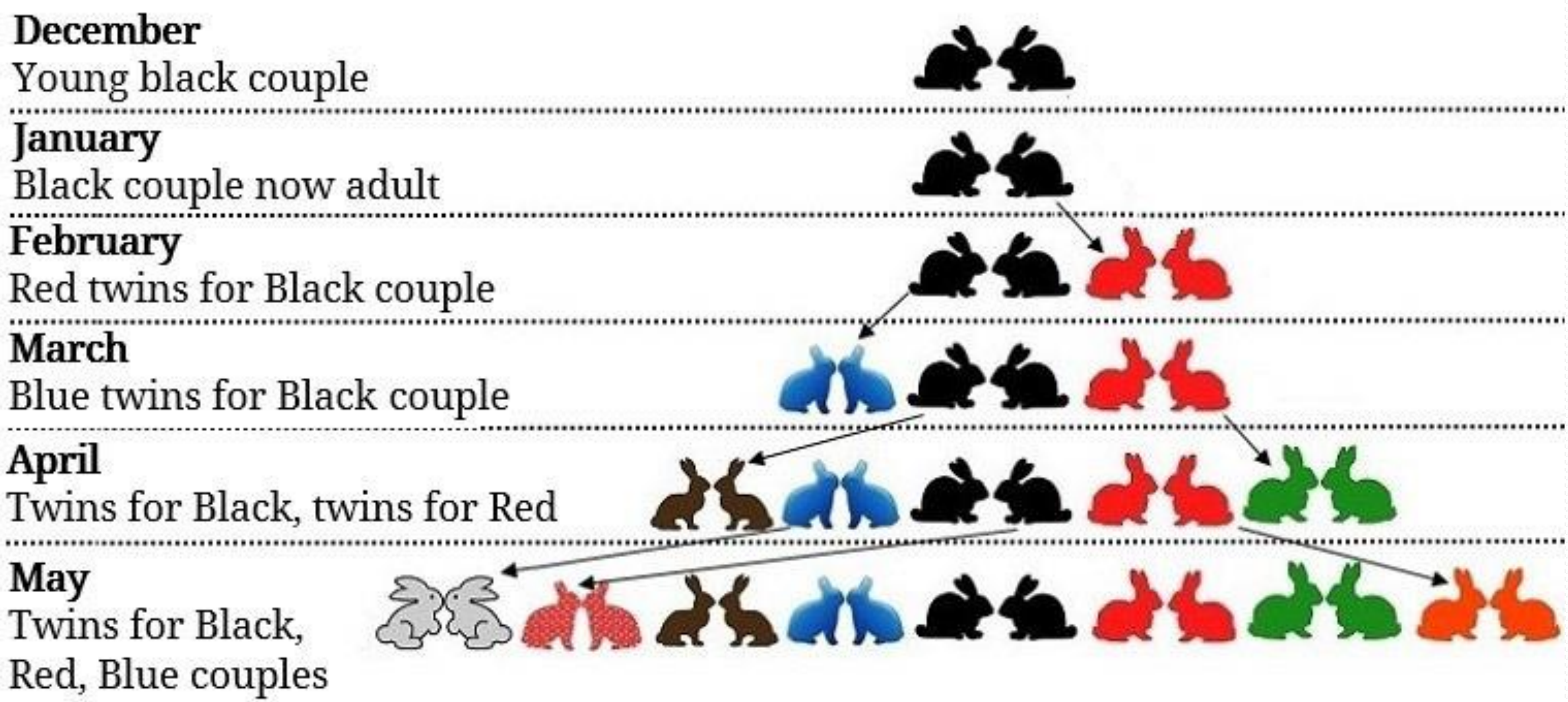
- A sequence is called a *solution* of a recurrence relation if its terms satisfy the recurrence relation.
- The *initial conditions* for a sequence specify the terms that precede the first term where the recurrence relation takes effect.

# Rabbits and the Fibonacci Numbers

**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.

*This is the original problem considered by Leonardo Pisano (Fibonacci) in the thirteenth century*.

# Rabbits and the Fibonacci Numbers (*cont.*)



# Rabbits and the Fibonacci Numbers (*cont.*)

| Reproducing pairs (at least two months old) | Young pairs (less than two months old) | 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**

# Rabbits and the Fibonacci Numbers (*cont.*)

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

- There are $f_1 = 1$ 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.

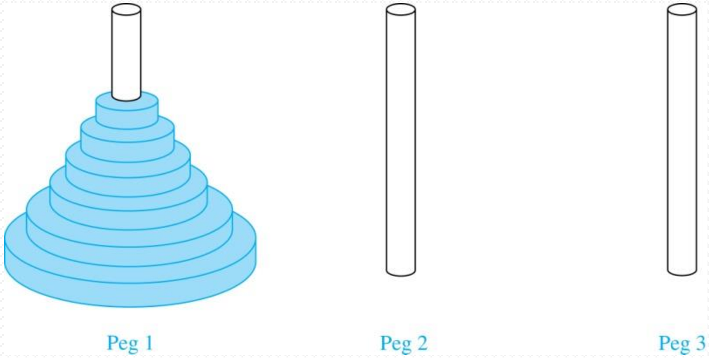
Consequently the sequence $\{f_n\}$ satisfies the recurrence relation $f_n = f_{n-1} + f_{n-2}$ for $n \geq 3$ with the initial conditions $f_1 = 1$ and $f_2 = 1$. The number of pairs of rabbits on the island after $n$ months is given by the $n$th Fibonacci number.

# The Tower of Hanoi

In the late nineteenth century, the French mathematician Édouard Lucas invented a puzzle 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.

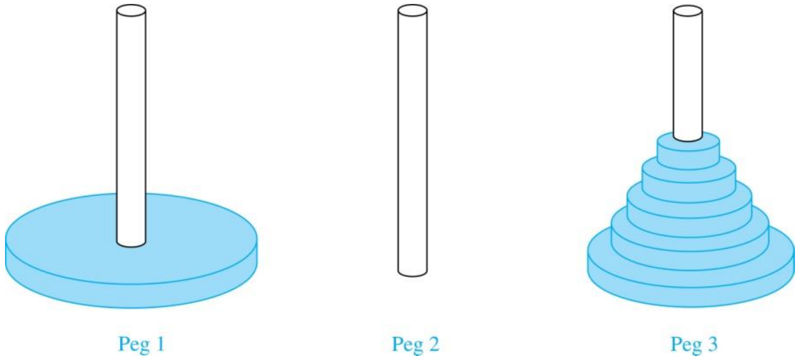
# The Tower of Hanoi (*continued*)




**The Initial Position in the Tower of Hanoi Puzzle**

# The Tower of Hanoi (*continued*)

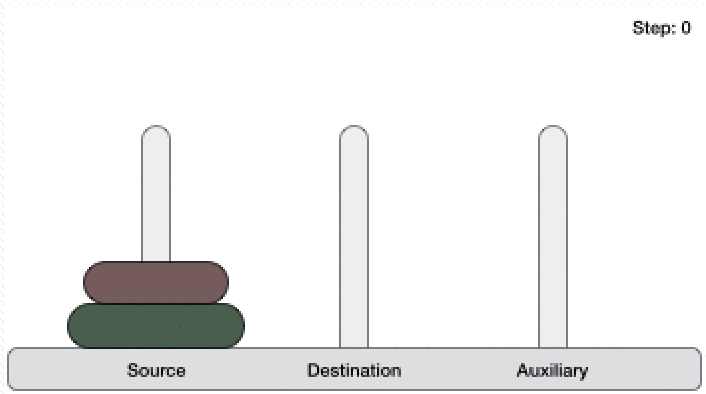
**Solution**: Let $\{H_n\}$ denote the number of moves needed to solve the Tower of Hanoi Puzzle with $n$ disks. Set up a recurrence relation for the sequence $\{H_n\}$. Begin with $n$ disks on peg 1. We can transfer the top $n-1$ disks, following the rules of the puzzle, to peg 3 using $H_{n-1}$ moves.

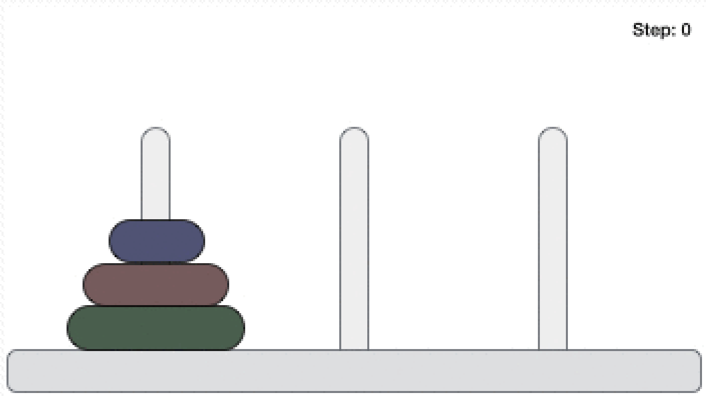


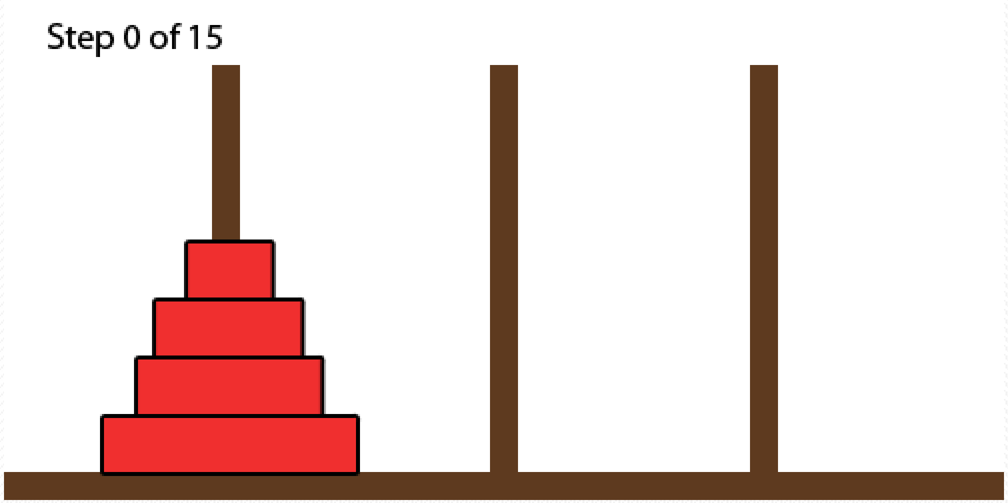

First, we use 1 move to transfer the largest disk to the second peg. Then we transfer the $n-1$ disks from peg 3 to peg 2 using $H_{n-1}$ additional moves. This can not be done in fewer steps. Hence,

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

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


Step: 0
Source
Destination
Auxiliary


Step: 0


Step 0 of 15

# The Tower of Hanoi (*continued*)

- 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.

$$
\begin{aligned}
H_n &= 2H_{n-1} + 1 \\
&= 2(2H_{n-2} + 1) + 1 = 2^2 H_{n-2} + 2 + 1 \\
&= 2^2(2H_{n-3} + 1) + 2 + 1 = 2^3 H_{n-3} + 2^2 + 2 + 1 \\
&\vdots \\
&= 2^{n-1}H_1 + 2^{n-2} + 2^{n-3} + \ldots + 2 + 1 \\
&= 2^{n-1} + 2^{n-2} + 2^{n-3} + \ldots + 2 + 1 \qquad \textit{because } H_1 = 1 \\
&= 2^n - 1 \qquad \textit{using the formula for the sum of the terms of a geometric series}
\end{aligned}
$$

- 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.

# Counting Bit Strings

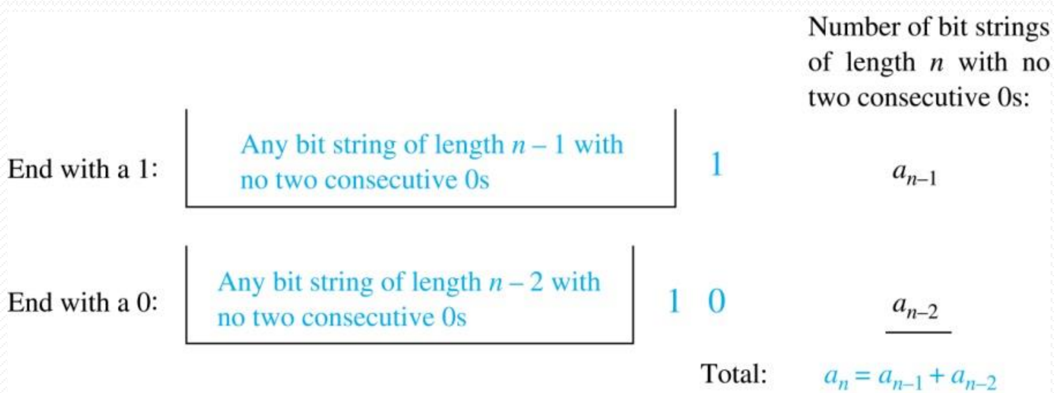
**Example 3**: Find a recurrence relation and give initial conditions for the number of bit strings of length $n$ without two consecutive 0s. How many such bit strings are there of length five?

**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 $n \geq 3$.



# Bit Strings (*continued*)

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:

- $a_3 = a_2 + a_1 = 3 + 2 = 5$
- $a_4 = a_3 + a_2 = 5 + 3 = 8$
- $a_5 = a_4 + a_3 = 8 + 5 = 13$

Note that $\{a_n\}$ satisfies the same recurrence relation as the Fibonacci sequence. Since $a_1 = f_3$ and $a_2 = f_4$, we conclude that $a_n = f_{n+2}$.

# Example 4

**Codeword Enumeration** 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 120987045608 is not valid. Let $a_n$ 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.

First, a valid string of $n$ 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 nine ways. Hence, a valid string with $n$ digits can be formed in this manner in $9a_{n-1}$ ways.

Second, a valid string of $n$ digits can be obtained by appending a 0 to a string of length $n - 1$ that is not valid. (This produces a string with an even number of 0 digits because the invalid string of length $n - 1$ has an odd number of 0 digits.) 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

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

valid strings of length $n$. ◀

# Exercise

3. 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 matters.
   b) What are the initial conditions?
   c) How many ways are there to deposit \$10 for a book of stamps?

- Solution

(a) Let $a_n$ be the number of ways to deposit $n$ dollars. Consider depositing a one-dollar coin, then we have $n-1$ dollars left to deposit. This will give us $a_{n-1}$ ways to deposit $n$ dollars. Depositing \$1 bills will also be the same as depositing one-dollar coins, so this will also give us $a_{n-1}$ ways to deposit $n$ dollars. Lastly, if we deposit a \$5 bill, then we have $n-5$ dollars left to deposit, so we have $a_{n-5}$ ways to deposit $n$ dollars. This would mean that our recurrence relation is $a_n = a_{n-1} + a_{n-1} + a_{n-5}$. Simplify the equation and we get $a_n = 2a_{n-1} + a_{n-5}$. Note that this is only valid when $n \geq 5$.

(b) The initial conditions are the different ways to deposit $n$ dollars up to $n = 4$. So, $a_0 = 1$ because there is only one way to deposit 0 dollars (do nothing). To find $a_1$, we know that there are two ways to deposit 1 dollar, by using the dollar coin or the \$1 bill, so $a_1 = 2$. To find $a_2$, we know there are $2^2$ ways to deposit 2 dollars, so $a_2 = 4$. Following the same line of thinking, we know $a_3 = 8$ and $a_4 = 16$.

(c) We need to find $a_{10}$, so we will have to work our way up from $a_5$.

$a_5 = 2a_4 + a_0 = 2(16) + 1 = 33$
$a_6 = 2a_5 + a_1 = 2(33) + 2 = 68$
$a_7 = 2a_6 + a_2 = 2(68) + 4 = 140$
$a_8 = 2a_7 + a_3 = 2(140) + 8 = 288$
$a_9 = 2a_8 + a_4 = 2(288) + 16 = 592$
$a_{10} = 2a_9 + a_5 = 2(592) + 33 = 1217$

There are 1217 ways to deposit \$10.

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

**b)** What are the initial conditions?

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

# Exercise

- <u>Solution</u>

(a) Let $a_n$ be the number of bit strings of length $n$ that contains a pair of consecutive 0s. Consider that the first bit of the bit string is 1, then the remaining $n-1$ bits of the string would contain the pair of consecutive 0s. Let us also consider if the start of the bit string was 01, then the remaining $n-2$ bits of the string would contain the pair of consecutive 0s. Lastly, if our bit string started 00, then it follows that any string of $n-2$ would satisfy the condition. This would mean that our recurrence relation is $a_n = a_{n-1} + a_{n-2} + 2^{n-2}$, valid for $n \geq 2$. Remember that there are $2^n$ bit strings of length $n$.

(b) Since there are no bit strings of length 0 and 1 that contains two consecutive 0s, we know $a_0 = 0$ and $a_1 = 0$.

(c) Compute $a_7$ starting from $a_2$.

$a_2 = a_1 + a_0 + 2^0 = 0 + 0 + 1 = 1$

$a_3 = a_2 + a_1 + 2^1 = 1 + 0 + 2 = 3$

$a_4 = a_3 + a_2 + 2^2 = 3 + 1 + 4 = 8$

$a_5 = a_4 + a_3 + 2^3 = 8 + 3 + 8 = 19$

$a_6 = a_5 + a_4 + 2^4 = 19 + 8 + 16 = 43$

$a_7 = a_6 + a_5 + 2^5 = 43 + 19 + 32 = 94$

There are 94 bit strings of length seven that contains a pair of consecutive 0s.

11. **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 ways can this person climb a flight of eight stairs?

# Exercise

- solution

**a)** Let $a_n$ be the number of ways to climb $n$ stairs. In order to climb $n$ stairs, a person must either start with a step of one stair and then climb $n-1$ stairs (and this can be done in $a_{n-1}$ ways) or else start with a step of two stairs and then climb $n-2$ stairs (and this can be done in $a_{n-2}$ ways). From this analysis we can immediately write down the recurrence relation, valid for all $n \geq 2$: $a_n = a_{n-1} + a_{n-2}$.

**b)** The initial conditions are $a_0 = 1$ and $a_1 = 1$, since there is one way to climb no stairs (do nothing) and clearly only one way to climb one stair. Note that the recurrence relation is the same as that for the Fibonacci sequence, and the initial conditions are that $a_0 = f_1$ and $a_1 = f_2$, so it must be that $a_n = f_{n+1}$ for all $n$.

**c)** Each term in our sequence $\{a_n\}$ is the sum of the previous two terms, so the sequence begins $a_0 = 1$, $a_1 = 1$, $a_2 = 2$, $a_3 = 3$, $a_4 = 5$, $a_5 = 8$, $a_6 = 13$, $a_7 = 21$, $a_8 = 34$. Thus a person can climb a flight of 8 stairs in 34 ways under the restrictions in this problem.

# Exercise

**19.** Messages are transmitted over a communications channel using two signals. The transmittal of one signal requires 1 microsecond, and the transmittal of the other signal requires 2 microseconds.

**a)** Find a recurrence relation for the number of different messages consisting of sequences of these two signals, where each signal in the message is immediately followed by the next signal, that can be sent in $n$ microseconds.

**b)** What are the initial conditions?

**c)** How many different messages can be sent in 10 microseconds using these two signals?

- Solution

(a) Let $a_n$ be the number of different messages that can be sent in $n$ microseconds. To send a message, we can first start off with a signal that is 1 microsecond and then send the remaining $n-1$ microseconds. We can also start off with a signal that is 2 microseconds and then send the remaining $n-2$ microseconds. Then it follows that our recurrence relation is $a_n = a_{n-1} + a_{n-2}$, for $n \geq 2$.

(b) There is only 1 way to send a 0 microsecond message, and that is to send nothing. So, $a_0 = 1$. To send 1 microsecond, it is obvious that there is only one way to accomplish this, so $a_1 = 1$.

(c) Compute $a_{10}$.

$a_2 = a_1 + a_0 = 1 + 1 = 2$

$a_3 = a_2 + a_1 = 2 + 1 = 3$

$a_4 = a_3 + a_2 = 3 + 2 = 5$

$a_5 = a_4 + a_3 = 5 + 3 = 8$

$a_6 = a_5 + a_4 = 8 + 5 = 13$

$a_7 = a_6 + a_5 = 13 + 8 = 21$

$a_8 = a_7 + a_6 = 21 + 13 = 34$

$a_9 = a_8 + a_7 = 34 + 21 = 55$

$a_{10} = a_9 + a_8 = 55 + 34 = 89$

There are 89 messages that are 10 microseconds long.