# 2.3 Functions

## Section Summary

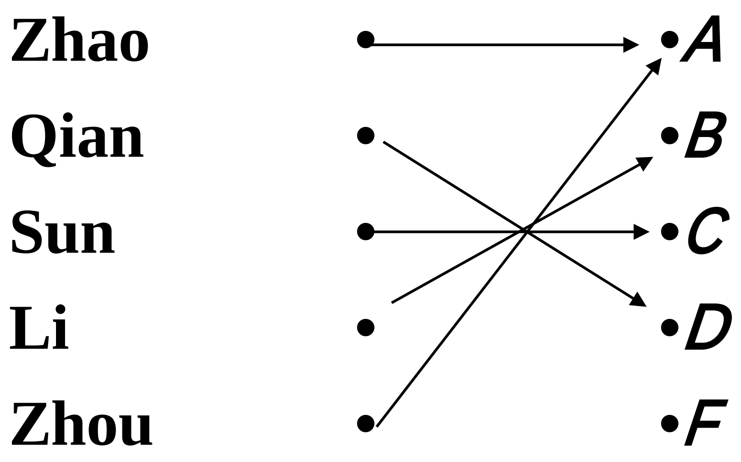
- Definition of a Function.
  - Domain, Codomain
  - Image, Preimage
- Injection, Surjection, Bijection
- Inverse Function
- Function Composition
- Graphing Functions
- Floor, Ceiling

# Functions

Definition: Let $A$ and $B$ be nonempty sets. A *function* f from $A$ to $B$, denoted $f: A \to B$ is an assignment of each element of $A$ to exactly one element of $B$.

We write $f(a) = b$ if $b$ is the unique element of $B$ assigned by the function $f$ to the element $a$ of $A$.

Functions are sometimes called *mappings* or *transformations*.



# Functions

- A function f: $A \rightarrow B$ can also be defined as a subset of $A \times B$ (a relation). This subset is restricted to be a relation where no two elements of the relation have the same first element.
- Specifically, a function f from $A$ to $B$ contains one, and only one ordered pair $(a, b)$ for every element $a \in A$.

$$\forall x[x \in A \rightarrow \exists y[y \in B \wedge (x, y) \in f]]$$

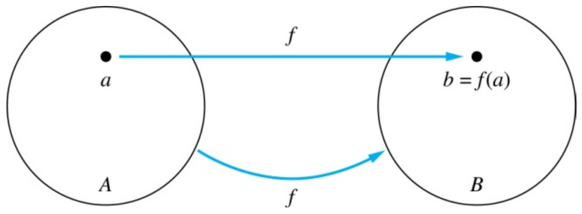
and

$$\forall x, y_1, y_2[[(x, y_1) \in f \wedge (x, y_2) \in f] \rightarrow y_1 = y_2]$$

# Functions

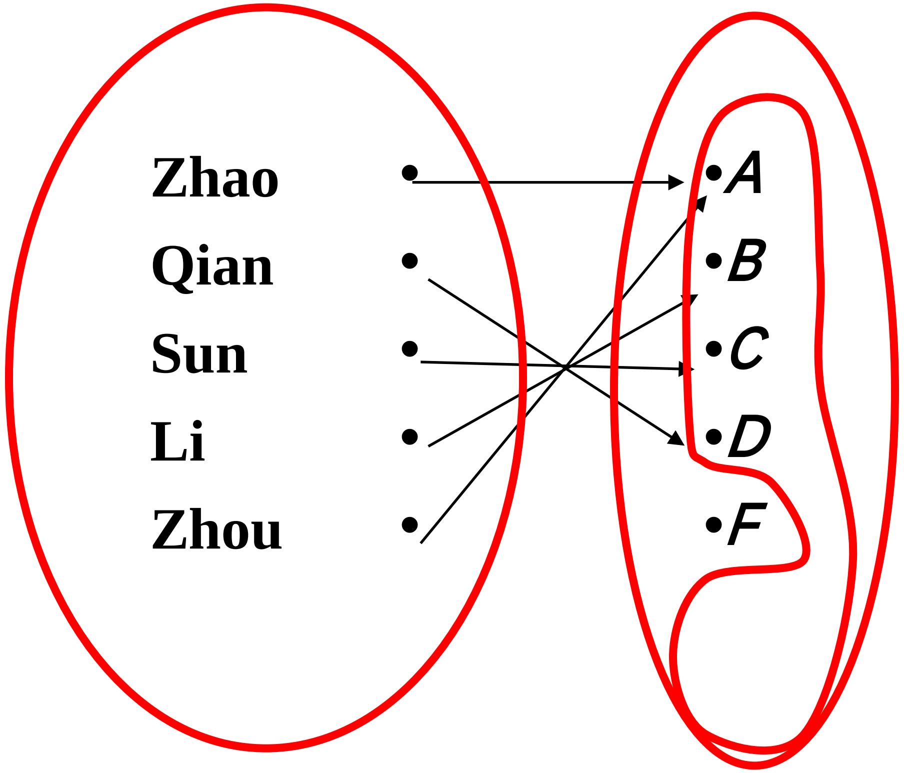
Given a function $f$: $A \rightarrow B$:

- We say $f$ *maps* $A$ to $B$ or
  $f$ is a *mapping* from $A$ to $B$.
- $A$ is called the domain of $f$.
- $B$ is called the codomain of $f$.
- If $f(a) = b$,
  - then $b$ is called the *image* of $a$ under $f$.
  - $a$ is called the *preimage* of $b$.
- The range of $f$ is the set of all images of points in **A** under $f$. We denote it by $f(A)$.
- Two functions are *equal* when they have the same domain, the same codomain and map each element of the domain to the same element of the codomain.



**For example,**

Suppose that each student in a class is assigned a letter grade from the set $\{A,B,C,D,F\}$. Let $G$ be the function that assigns a grade to a student.




Domain

Codomain

Range

# Representing Functions

**Functions may be specified in different ways:**

- **An explicit statement of the assignment**

  Students and grades example.

- **A formula**

  $f(x) = x + 1$

- **A computer program**

  A Java program that when given an integer $n$, produces the $n$th Fibonacci Number (covered in the next section and also in Chapter 5).

【**Definition**】 Let $f_1$ and $f_2$ be functions from A to **R**. Then $f_1 + f_2$ and $f_1 f_2$ are also functions from A to **R** defined by

$$(f_1 + f_2)(x) = f_1(x) + f_2(x)$$
$$(f_1 f_2)(x) = f_1(x)\, f_2(x)$$

【**Definition**】 Let $f$ be a function from $A$ to $B$ and let $S$ be a subset of $A$. The ***image*** of $S$ is the subset of $B$ that consists of the images of the elements of $S$. We denote the image of $S$ by $f(S)$, so that

$$f(S) = \{ f(s) \mid s \in S \}$$

**Remark:**

$f(S)$ denotes a set, and not the value of the function $f$ for the set $S$.

# One-to-one Functions

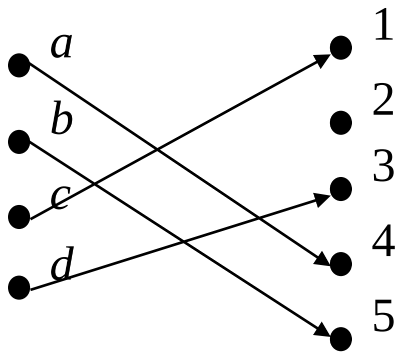
**A function f is one-to-one (denoted 1-1), or injective**

$$\forall a \forall b (a \in A \wedge b \in A \wedge (f(a) = f(b) \rightarrow a = b))$$

$$\forall a \forall b (a \in A \wedge b \in A \wedge (a \neq b \rightarrow f(a) \neq f(b)))$$

**Note: Preimages are unique.**

**Example ,**



# Onto Functions

**A function f from A to B is called onto, or surjective**

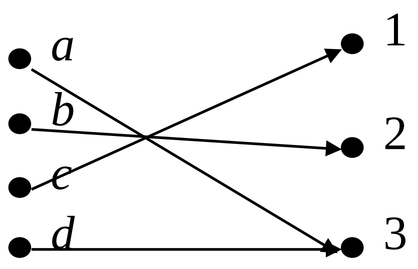
$$\forall b(b \in B \rightarrow \exists a\,(a \in A \wedge f(a) = b))$$

**Note:**

**This means that for every *b* in *B* there must be an *a* in *A* such that *f*(*a*) = *b*.**

**Every *b* in *B* has a pre-image.**

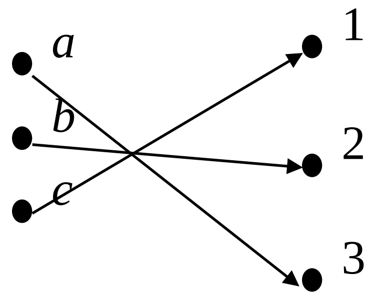
**Example ,**



# One-to-one Correspondence Functions

**The function f is a one-to-one correspondence, or a bijection, if it is both one-to-one and onto.**

**For example,**




**Note:**

**Whenever there is a bijection from *A* to *B*, the two sets must have the same number of elements or the same *cardinality*.**

**That will become our *definition*, especially for infinite sets.**

# Showing that f is one-to-one or onto

Suppose that $f : A \to B$.

*To show that $f$ is injective* Show that if $f(x) = f(y)$ for arbitrary $x, y \in A$ with $x \neq y$, then $x = y$.

*To show that $f$ is not injective* Find particular elements $x, y \in A$ such that $x \neq y$ and $f(x) = f(y)$.

*To show that $f$ is surjective* Consider an arbitrary element $y \in B$ and find an element $x \in A$ such that $f(x) = y$.

*To show that $f$ is not surjective* Find a particular $y \in B$ such that $f(x) \neq y$ for all $x \in A$.

# Showing that f is one-to-one or onto

**Example** : Let $f$ be the function from $\{a,b,c,d\}$ to $\{1,2,3\}$ defined by $f(a) = 3$, $f(b) = 2$, $f(c) = 1$, and $f(d) = 3$. Is $f$ an onto function?

**Solution**: Yes, $f$ is onto since all three elements of the codomain are images of elements in the domain. If the codomain were changed to $\{1,2,3,4\}$, $f$ would not be onto.

**Example** : Is the function $f(x) = x^2$ from the set of integers onto?

**Solution**: No, $f$ is not onto because there is no integer $x$ with $x^2 = -1$, for example.

**A function is**

| | |
|---|---|
| ***increasing*** | $\forall x \forall y (x < y \rightarrow f(x) \leq f(y))$ |
| ***strictly increasing*** | $\forall x \forall y (x < y \rightarrow f(x) < f(y))$ |
| ***decreasing*** | $\forall x \forall y (x < y \rightarrow f(x) \geq f(y))$ |
| ***strictly decreasing*** | $\forall x \forall y (x < y \rightarrow f(x) > f(y))$ |

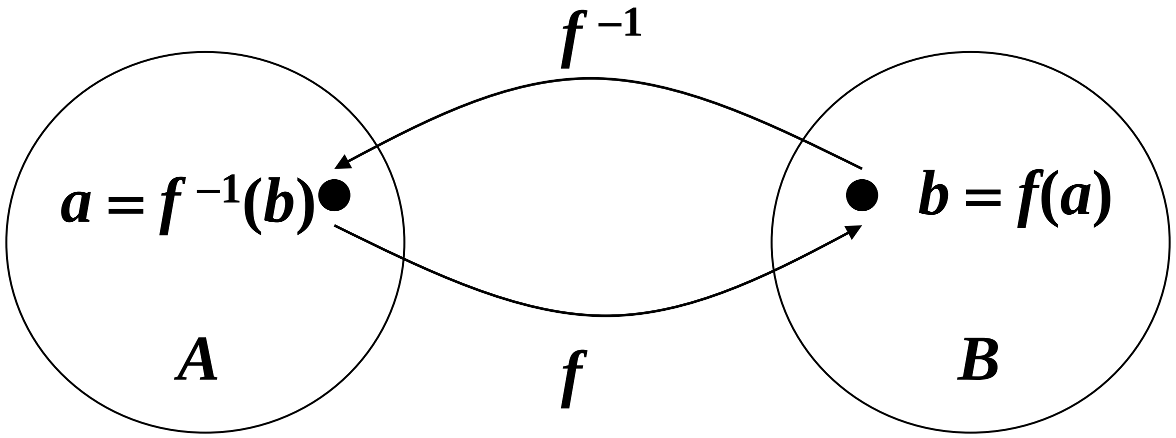
Questions:

1. A function that is either strictly increasing or strictly decreasing must be one to one?
2. The number of one to one functions form a set S to a set T?

# Inverse Functions

Let f be a bijection from A to B. Then the **inverse function** of f, denoted $f^{-1}$, is the function from B to A defined as

$$f^{-1}(b) = a \text{ iff } f(a) = b$$



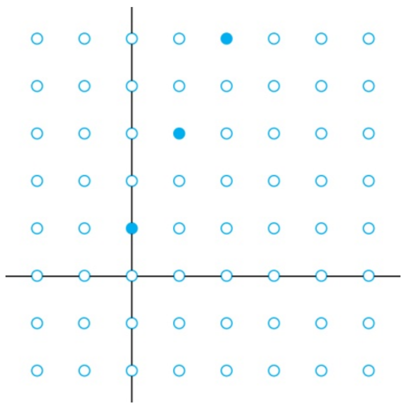

**Note:**

No inverse function exists unless $f$ is a bijection.

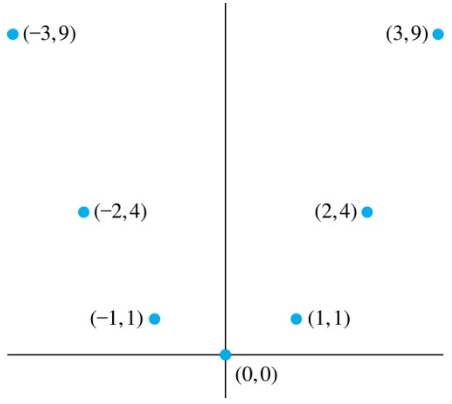
# The Graphs of Functions

Let $f$ be a function from the set $A$ to the set $B$. The **graph** of the function $f$ is the set of ordered pairs

$$\{(a, b) \mid a \in A \text{ and } f(a) = b\}.$$



Graph of $f(n) = 2n + 1$ from Z to Z




Graph of $f(x) = x^2$ from Z to Z

# Some Important Functions

The **floor function** f (x) is the largest integer less than or equal to x.

**Notation:** $\lfloor x \rfloor$

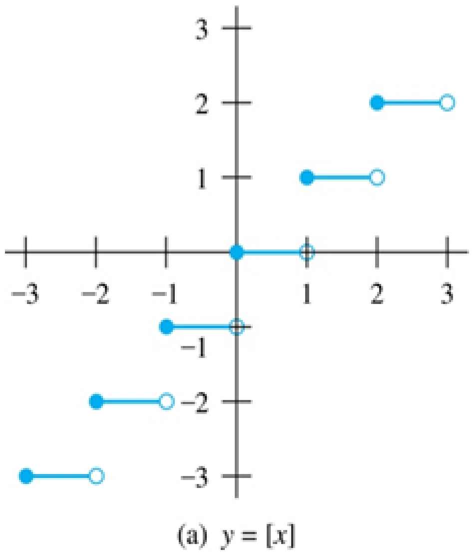
**For example:**

$\lfloor 0.5 \rfloor = 0$

$\lfloor 1.5 \rfloor = 1$

$\lfloor 2 \rfloor = 2$

$\lfloor -0.5 \rfloor = -1$




**Graph of Floor Functions**

**Remark:**

**The floor function is often also called the *greatest integer function*. It is often denoted by [*x*].**

# Some Important Functions

**The ceiling function f (x) is the smallest integer greater than or equal to x.**

**Notation:** $\lceil x \rceil$

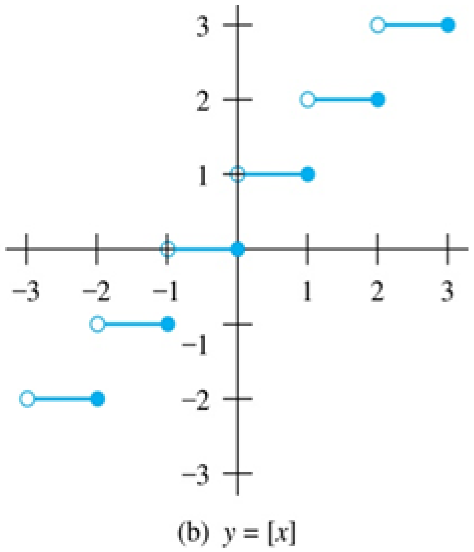
**For example:**

$\lceil 0.5 \rceil =$ 1

$\lceil 1.5 \rceil =$ 2

$\lceil 2 \rceil =$ 2

$\lceil -0.5 \rceil =$ 0




**Graph of Ceiling Functions**

# Some Important Functions

The floor and ceiling functions are useful in wide variety of application.

- Data storage and data transmission
- The pigeonhole principle

**Example, Date stored on a computer disk or transmitted over a data network are usually represented as a string of bytes. Each byte is made up of 8 bits. How many bytes are required to encode 100 bits of data?**

$$\lceil 100/8 \rceil = \lceil 12.5 \rceil = 13$$

# Some Important Functions

## ◆Useful Properties of the Floor and Ceiling Functions

(1a) $\lfloor x \rfloor = n$ if and only if $n \le x < n + 1$ where $n$ is an integer

(1b) $\lceil x \rceil = n$ if and only if $n - 1 < x \le n$ where $n$ is an integer

(1c) $\lfloor x \rfloor = n$ if and only if $x - 1 < n \le x$ where $n$ is an integer

(1d) $\lceil x \rceil = n$ if and only if $x \le n < x + 1$ where $n$ is an integer

(2) $x - 1 < \lfloor x \rfloor \le x \le \lceil x \rceil < x + 1$

(3a) $\lfloor -x \rfloor = -\lceil x \rceil$

(3b) $\lceil -x \rceil = -\lfloor x \rfloor$

(4a) $\lfloor x + m \rfloor = \lfloor x \rfloor + m$ where $m$ is an integer

(4b) $\lceil x + m \rceil = \lceil x \rceil + m$ where $m$ is an integer

## ◆ Prove property (4a)

**(4a) $\lfloor x + m \rfloor = \lfloor x \rfloor + m$ where $m$ is an integer**

***Proof:***

**Suppose that $\lfloor x \rfloor = n$, where n is a positive integer.**

**By property (1*a*), it follows that if $n \le x < n + 1$.**

**Then $n + m \le x + m < n + m + 1$.**

**Using property (1*a*) again, we see that $\lfloor x + m \rfloor = n + m = \lfloor x \rfloor + m$.**

**Example,** **Prove that if *x* is a real number, then**

$$\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + 1/2 \rfloor.$$

***Hint:***

**The floor function:**

**Let $x=n+\varepsilon$, where $n = \lfloor x \rfloor$ is an integer, and $\varepsilon$, the fractional part of $x$, $0 \le \varepsilon < 1$.**

**The ceiling function:**

**Let $x=n-\varepsilon$, where $n = \lceil x \rceil$ is an integer, and $0 \le \varepsilon < 1$.**

**Example, Prove that if $x$ is a real number, then**

$$\lfloor 2x \rfloor = \lfloor x \rfloor + \lfloor x + 1/2 \rfloor.$$

***Solution*: Let $x = n + \varepsilon$, where $n$ is an integer and $0 \leq \varepsilon < 1$.**

***Case 1:* $\varepsilon < ½$**

**$2x = 2n + 2\varepsilon$ and $\lfloor 2x \rfloor = 2n$, since $0 \leq 2\varepsilon < 1$.**

**$\lfloor x + 1/2 \rfloor = n$, since $x + ½ = n + (1/2 + \varepsilon)$ and $0 \leq ½ + \varepsilon < 1$.**

**Hence, $\lfloor 2x \rfloor = 2n$ and $\lfloor x \rfloor + \lfloor x + 1/2 \rfloor = n + n = 2n$.**

***Case 2:* $\varepsilon \geq ½$**

**$2x = 2n + 2\varepsilon = (2n + 1) + (2\varepsilon - 1)$ and $\lfloor 2x \rfloor = 2n + 1$, since $0 \leq 2\varepsilon - 1 < 1$.**

**$\lfloor x + 1/2 \rfloor = \lfloor n + (1/2 + \varepsilon) \rfloor = \lfloor n + 1 + (\varepsilon - 1/2) \rfloor = n + 1$ since $0 \leq \varepsilon - 1/2 < 1$.**

**Hence, $\lfloor 2x \rfloor = 2n + 1$ and $\lfloor x \rfloor + \lfloor x + 1/2 \rfloor = n + (n + 1) = 2n + 1$.**

# 2.4
# Sequences and Summations

# Introduction

- Sequences are ordered lists of elements.
  - 1, 2, 3, 5, 8
  - 1, 3, 9, 27, 81, …….
- Sequences arise throughout mathematics, computer science, and in many other disciplines
  - Sequences can be used to represent solutions to certain counting problems.
  - Sequence is also an important data structure in computer science

# Sequences

**Definition**: A ***sequence*** is a function from a subset of the integers (usually either the set {0, 1, 2, 3, 4, …..} or {1, 2, 3, 4, ….} ) to a set *S*.

The notation $a_n$ is used to denote the image of the integer *n*. We can think of $a_n$ as the equivalent of *f(n)* where *f* is a function from {0,1,2,…..} to *S*.

We call $a_n$ a ***term*** of the sequence.

Notation for sequence: $\{\boldsymbol{a_n}\}$

# Sequences

**Example1**: Consider the sequence $\{a_n\}$ where $a_n = \dfrac{1}{n}$ .

The sequence can be described by listing the terms of the sequence in order of increasing subscripts.

The list of the terms of this sequence, beginning with $a_1$, namely,

$$a_1, a_2, a_3, a_4, \ldots$$

starts with

$$1, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots$$

# Some Familiar Sequences

**Definition**: A *geometric progression* is a sequence of the form:

$$a, ar, ar^2, \ldots, ar^n, \ldots$$

where the *initial term a* and the *common ratio r* are real numbers.

**Definition**: A *arithmetic progression* is a sequence of the form:

$$a, a+d, a+2d, \ldots, a+nd, \ldots$$

where the *initial term a* and the *common difference d* are real numbers.

# Strings

**Definition**: A *string* is a finite sequence of characters from a finite set (an alphabet).

Note:

- This string is also denoted by $a_1\, a_2 \ldots a_n$
- Sequences of characters or bits are important in computer science.
- The ***length*** of a string is the number of terms in this string.
  - The string *abcde* has *length* 5.
- The ***empty string*** is represented by $\lambda$.

# 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.

- 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.

# Questions about Recurrence Relations

**Example** 2: Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} + 3$ for $n = 1,2,3,4,\ldots.$ and suppose that $a_0 = 2$. What are $a_1$, $a_2$ and $a_3$?

[Here $a_0 = 2$ is the initial condition.]

**Solution**: We see from the recurrence relation that

$a_1 = a_0 + 3 = 2 + 3 = 5$

$a_2 = 5 + 3 = 8$

$a_3 = 8 + 3 = 11$

# Questions about Recurrence Relations

**Example** 3: Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} - a_{n-2}$ for $n = 2,3,4,\ldots.$ and suppose that $a_0 = 3$ and $a_1 = 5$. What are $a_2$ and $a_3$?

[Here the initial conditions are $a_0 = 3$ and $a_1 = 5$. ]

**Solution**: We see from the recurrence relation that

$$a_2 = a_1 - a_0 = 5 - 3 = 2$$

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

# Fibonacci Sequence

**Definition**: Define the *Fibonacci sequence*, $f_0, f_1, f_2, \ldots$, by:

- Initial Conditions: $f_0 = 0$, $f_1 = 1$
- Recurrence Relation: $f_n = f_{n-1} + f_{n-2}$

Example: Find $f_2, f_3, f_4, f_5$ and $f_6$.

Answer:

$$f_2 = f_1 + f_0 = 1 + 0 = 1,$$
$$f_3 = f_2 + f_1 = 1 + 1 = 2,$$
$$f_4 = f_3 + f_2 = 2 + 1 = 3,$$
$$f_5 = f_4 + f_3 = 3 + 2 = 5,$$
$$f_6 = f_5 + f_4 = 5 + 3 = 8.$$

# Solving Recurrence Relations

- Finding a formula for the $n$th term of the sequence generated by a recurrence relation is called *solving the recurrence relation*.
- Such a formula is called a *closed formula*.
- Various methods for solving recurrence relations will be covered in Chapter 8 where recurrence relations will be studied in greater depth.
- Here we illustrate by example the method of iteration in which we need to guess the formula. The guess can be proved correct by the method of induction (Chapter 5).

# Iterative Solution Example

**Method 1**: Working upward, forward substitution

Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} + 3$ for $n = 2,3,4,\ldots.$ and suppose that $a_1 = 2$.

$$a_2 = 2 + 3$$
$$a_3 = (2 + 3) + 3 = 2 + 3 \cdot 2$$
$$a_4 = (2 + 2 \cdot 3) + 3 = 2 + 3 \cdot 3$$
$$\vdots$$
$$a_n = a_{n-1} + 3 = (2 + 3 \cdot (n - 2)) + 3 = 2 + 3(n - 1)$$

# Iterative Solution Example

**Method 2**: Working downward, backward substitution

Let $\{a_n\}$ be a sequence that satisfies the recurrence relation $a_n = a_{n-1} + 3$ for $n =$ 2,3,4,…. and suppose that $a_1 = 2$.

$$
\begin{aligned}
a_n &= a_{n-1} + 3 \\
&= (a_{n-2} + 3) + 3 = a_{n-2} + 3 \cdot 2 \\
&= (a_{n-3} + 3) + 3 \cdot 2 = a_{n-3} + 3 \cdot 3 \\
&\;\;\vdots \\
&= a_2 + 3(n-2) = (a_1 + 3) + 3(n-2) = 2 + 3(n-1)
\end{aligned}
$$

**Homework:**

SE: P.153 12,40, 56, 72, 76

P.168 15

EE: P.153 12,42, 58, 74, 78

P.178 15