DISCRETE MATHEMATICS

MATH 240 01

Author
Paul Beggs

2nd Semester, 2024

Contents

1 Logical Thinking **4**
1.1 Formal Logic . . . 4
1.1.1 Inquiry Problems . . . 4
1.2 The Integers . . . 5

2 Relational Thinking **12**
2.1 Graphs . . . 12
2.1.1 Definitions . . . 13
2.1.2 Directed Graph (Diagram) . . . 14
2.2 Sets . . . 15
2.3 Functions . . . 19
2.4 Relations . . . 22
2.4.1 Modular Arithmetic . . . 23
2.6 Graph Theory . . . 24
2.6.1 Graphs . . . 24
2.6.2 Circuits . . . 25
2.6.3 Bipartite . . . 26
2.6.4 Trees . . . 26
2.6.5 Euler Circuit . . . 28
2.6.6 Isomorphism . . . 28
2.6.7 Vertex Colorings . . . 29
2.6.8 Hamilton Graphs . . . 30

3 Recursive Thinking **31**
3.1 Recurrence Relations . . . 31
3.1.1 Fibonacci . . . 32
3.2 Mathematical Induction . . . 33
3.3 Recursive Definitions . . . 34
3.4 Induction Examples - Cont. . . . 36
3.5 Strong Induction . . . 38
3.6 Structural Induction . . . 40
3.7 Data Structures . . . 41
3.7.1 Lists – An Iterative Data structure . . . 41
3.7.2 Tree Traversal Algorithms . . . 42

4 Quantitative Thinking ('Counting') **43**
4.1 Basic Counting Techniques . . . 43
4.3 Selections and Arrangements . . . 47
4.3.1 Permutations . . . 47
4.5 Counting In Algorithms . . . 49
4.5.1 Big-O Classes (From handout p.3) . . . 49

5 Algorithms **50**
5.1 Greedy Algorithms 52
5.2 Traveling Salesperson Problem 52
5.3 Algorithms review 52

1 Logical Thinking

Definition. A statement (also known as a *proposition*) is either true or false, but not both.

1.1 Formal Logic

Inquiry Problems

if p is the statement "you are wearing shoes" and q is the statement "you can't cut your toenails," then,

$$p \Rightarrow q$$

Terminology 1. $p \Rightarrow q$ (i.e., if p, then, q). This is the *Statement* of a Theorem.

"If rain, then clouds."

Terminology 2. $\neg q \Rightarrow \neg p$ (i.e., if $\neg q$, then $\neg p$). This is the *Contra-positive* of the statement.

"If no clouds, then no rain"

Note: *statement* $\iff$ *contra-positive.*

Terminology 3. $q \Rightarrow p$ (i.e, if p then, q). This is the *Converse* of the statement.

"If clouds, then rain."

Terminology 4. $\neg p \Rightarrow \neg q$ (i.e., if not p, then not q). This is the *Inverse* of the statement.

"If no rain, then no clouds."

Note: *converse* $\iff$ *inverse.*

1.2 The Integers

The *integers*, denoted $\mathbb{Z}$, is the set $\{\dots,-2,-1,0,1,2,\dots\}$ with two binary operations, called *addition* and *multiplication*, denoted by $+$ and $\cdot$, respectively. We have the following axioms for the integers: /*9

Axiom 1. (*Closure*) If $a, b \in \mathbb{Z}$, then each of $a+b \in \mathbb{Z}$ and $a \cdot b \in \mathbb{Z}$.

Axiom 2. (*Commutativity*) If $a, b \in \mathbb{Z}$, then each of $a+b \in \mathbb{Z}$.

Axiom 3. (*Associativity*) If $a, b, c \in \mathbb{Z}$, then $(a+b)+c = a+(b+c)$ and $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.

Axiom 4. (*Distibutivity of Multiplication over Addition*) If $a, b, c \in \mathbb{Z}$, then $a \cdot (b+c) = a \cdot b + a \cdot c$.

Axiom 5. (*Identity Elements*) For each $a \in \mathbb{Z}, a+0 = a$ and $a \cdot 1 = a$. We call 0 and 1 the *additive* and *multiplicative identities*, respectively.

Axiom 6. (*Additive Inverse*) For each $a \in \mathbb{Z}, (\exists b) b \in \mathbb{Z}\colon a+b=0$. We will denote the additive inverse of a by $-a$.

Axiom 7. (*Cancellation*) Suppose that $a, b, c \in \mathbb{Z}$. Then:

- $a+b = b+c \iff a = b$.
- when $c \neq 0, a \cdot c = b \cdot c, \iff a = b$.

Axiom 8. (*Order*) There is a well-defined order relation $<$ on $\mathbb{Z}$ so that:

- $a \in \mathbb{Z} \Rightarrow a \not< a$,
- $a, b \in \mathbb{Z} \Rightarrow [(a<b) \vee (a=b) \vee (b<a)]$.
- $a, b, c \in \mathbb{Z}\colon (a<b) \wedge (b<c) \Rightarrow a<c$.

Axiom 9. (*Arithmetic Order*) Suppose that $a, b \in \mathbb{Z}\colon a<b$. Then,

- $c \in \mathbb{Z} \Rightarrow (a+c < b+c)$
- $(p \in \mathbb{Z}) \wedge (p>0) \Rightarrow (a \cdot p < b \cdot p)$.

Definition. We say that the integer $a \in \mathbb{Z}$ is *even* if there exists an integer k so that $a = 2 \cdot k + 1$.

Definition. We say that the integer $a \in \mathbb{Z}$ is *odd* if there exists an integer k so that $a = 2 \cdot k + 1$

Axiom 10. The particular integer, $0 \in \mathbb{Z}$ is not odd.

Theorem 1. Let $a \in \mathbb{Z}$.

$$\text{Then, } a \cdot 0 = 0$$

Proof.
Since $a, 0 \in \mathbb{Z}$, by Axiom 1, $a \cdot 0$ is also an integer, say b. We also note that $0 = 0 + 0$, by Axiom 5. Then,

$$\begin{aligned} b &= a \cdot 0 \\ &= a \cdot (0 + 0) \\ &= a \cdot 0 + a \cdot 0 && \text{by Axiom 4} \\ &= b + b \end{aligned}$$

Thus, we have shown that $b = b + b$. Now, since $b \in \mathbb{Z}$, there is an integer $-b$: $b + (-b) = 0$. Then,

$$\begin{aligned} b &= b + b \\ b + (-b) &= b + b + (-b) && \text{by Axiom 7} \\ 0 &= b + (b + (b + (-b)) && \text{by Axiom 3} \\ 0 &= b + 0 \\ 0 &= b \end{aligned}$$

Therefore, $((\forall 0)(\forall a)\ 0, a \in \mathbb{Z})\ a \cdot 0 = 0$ $\square$

Lemma. Suppose $a \in \mathbb{Z} \wedge b, c \in \mathbb{Z}\colon a + c = 0$.

Then, $b = c$.

Proof.

$$\begin{aligned} a + b &= 0 + c & \\ b &= c & \text{by Axiom 7} \end{aligned}$$

□

This Lemma shows that additive inverses are unique.

Theorem 2. Let $a \in \mathbb{Z}$.

Then, $-a = (-1) \cdot a$.

Abstract: Show $a + (-1) \cdot a = 0$.

Proof.
We wish to show that $a + (-1) \cdot a = 0$. This would prove that $(-1) \cdot a$ is an additive inverse of a. By Lemma 1, since inverses are unique, $-a = (-1) \cdot a$.

$$\begin{aligned} 0 &= 0 \cdot a & &\text{by Axiom 2 and Theorem 1} \\ &= (1 + (-1)) \cdot a & &\text{by Axiom 6, 1 has an additive inverse} \\ &= 1 \cdot a + (-1) \cdot a & &\text{by Axiom 4} \\ &= a + (-1) \cdot a & &\text{by Axiom 5} \end{aligned}$$

We have established that $-a + (-1) \cdot a = 0$; therefore, $-a = (-1) \cdot a$. □

Theorem 3. $0 < 1$.

Proof.
Thus, we will use Axiom 8 to prove property a, and disprove b, and c. i.e., either $(0 < 1) \vee (0 = 1) \vee (1 < 0)$. If $0 = 1$, then,

$$\begin{aligned} -1 \cdot 0 &= (-1) \cdot 1 \\ 0 &= (-1) \end{aligned}$$

Clearly, there is more than a single integer, so this must be false.
If $1 < 0$, then,

$$\begin{aligned} -1 + (-1) &< 0 + (-1) && \text{by Axiom 9} \\ 0 &< (-1), && \text{by Axiom 5} \\ 0 \cdot (-1) &< (-1) \cdot (-1) && \text{by Axiom 9} \\ 0 &< 1, && \text{by Theorem 1 and 2} \end{aligned}$$

By Axiom 8, $1 < 0$ and $0 < 1$ implies $1 < 1$. But $1 \not< 1$, by Axiom 8. This is a contradiction. □

Theorem 4. Suppose a is odd.

Then, a^2 is odd.

Proof.
Since a is odd, there exists an integer k: $a = 2k + 1$. Then,

$$\begin{aligned} a &= 2k + 1 \\ a^2 &= (2k + 1)^2 && \text{multiply both sides by } a \\ &= 4k^2 + 4k + 1 \\ &= (4k^2 + 4k) + 1 && \text{by Axiom 1} \\ &= 2(2k^2 + 2k) + 1 \end{aligned}$$

Since $2k^2 + 2k$ is an integer (Axiom 1), $a^2 = 2(2k^2 + 2k) + 1$ means a^2 is also odd. □

Theorem 5. Suppose a^2 is odd.

Then, a is odd.

Proof.
By contra-position. Suppose that a is not odd (even). Then a^2 is not odd (even). If a is even, there exists an integer k such that,

$$a = 2k$$
$$a^2 = 4k^2$$
$$a^2 = 2(2k^2)$$

Since $2(2k^2)$ is an integer (Axiom 1), $a^2 = 2(2k^2)$ means a^2 is not odd. Since a^2 is even, it is not odd. Thus, if a is *not* odd, then a^2 is not odd. This is the contra-position of the statement that if a^2 *is* odd, then a is odd. □

So, this proved that since a is not odd, a^2 is not odd, which translated to a^2 is odd, so a is odd.

Definition. A number is rational iff it is the ratio of two integers, $\frac{a}{b}, b \neq 0$.

Fact: Each fraction has a reduced form when a and b lack common factors.

Theorem 6. $\sqrt{2}$ is not rational.

Proof.
By contradiction, suppose $\sqrt{2}$ is rational. Then, these are integers with no common factors for which $\sqrt{2} = \frac{a}{b}$

$$\sqrt{2}b = a$$
$$2b^2 = a^2$$

Thus, a^2 is even. Then, since a is even, $a = 2k$, for some integer k.

$$2b^2 = (2k)^2$$
$$2b^2 = 4k^2$$
$$b^2 = 2k^2$$

Thus, b^2 is even, so b must be even. But there is the contradiction: $\sqrt{2}$ was assumed to be rational. □

Definition. Let a, b be integers with $a < b$. WE say a and b are **consecutive** if $b = a + 1$.

Theorem 7. Suppose a, b are consecutive integers. Then, $b^2 = a^2 = a + b$.

Proof.
Suppose a, b are consecutive. Then, $b = a + 1$, such that:

$$\begin{aligned} b^2 - a^2 &= (a+1)^2 - a^2 && \text{since } b = a + 1 \\ &= a^2 + 2a + a - a^2 \\ &= 2a + 1 \\ &= a + a + 1 \\ &= a + (a+1) \\ &= a + b \end{aligned}$$

Thus, $b^2 - a^2 = a + b$. □

Definition. Suppose a, b are integers. We say that a divides b, denoted by $a|b$, if there exists an integer k such that $b = a \cdot k$.

Theorem 8. Suppose a is an integer. Then $1|a$.

Proof.
Let a be an integer. Then. we need to find integer k such that $a = 1 \cdot k$. Let $k = a$. Then, $a = 1 \cdot a$ so $1|a$. □

Theorem 9. Let a be an integer. Then, $a|a$.

Proof.
Since $a = 1 \cdot a$, then, $a|a$. □

Definition. an integer is *prime* if its only divisors are 1 and itself.

Theorem 10. (Euclid) There are infinitely many primes.

Proof.
Suppose instead that there are only finitely many, say n of them. We list the primes as $p_1, p_2, \ldots p_n$.

Let $m = p_1, p_2, \ldots p_n + 1$. Then, $p_1 \not| \ m$. For all i, $p_i \not| \ m$.. But then, m has no prime divisors. Thus, m is prime. But $m > p_i$ for all i, and so not in our list.

Contradiction! We assumed there was finitely many. $\square$

2 Relational Thinking

2.1 Graphs

Definition. *Vertices* are points defined on a graph.
Edges are lines or curves that connect points on a graph.

Definition. The *degree* of a vertex is the number of edges that include the vertex.

This implies, that we cannot design a tour (for the conceptual puzzle Königsberg bridge) if we have 3 or more vertices of odd degree.

Thus, if all degrees are even, you can start anywhere, and you will end there.

If you have *exactly* 2 point of odd degree, you can do it by starting at one end, and ending at the other.

For the missing case of vertices (*exactly* 1 point of odd degree), there cannot be a single vertex of odd degree.

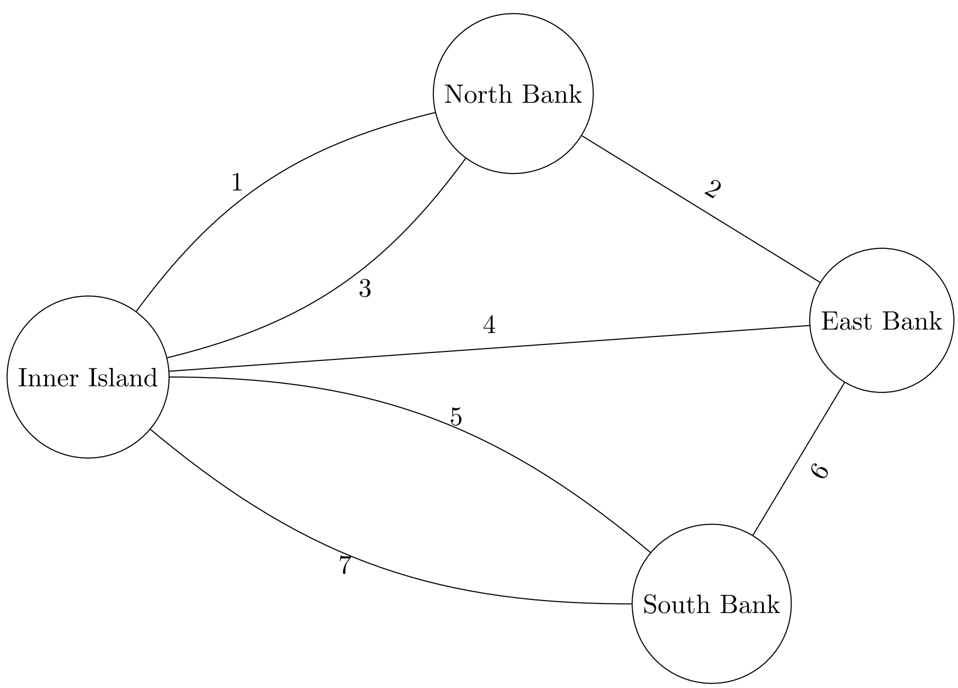

Figure 1: Illustration of the Königsberg Bridge Puzzle

Theorem 1. (Handshake Theorem) The sum of the degrees is always twice the number of the edges.

Corollary 1. The sum of degrees is always even.

Corollary 2. No graph has a single vertex of odd degree.

Definitions

Definition. A *loop* is an edge which connects a vertex to itself.

Definition. Two vertices are *adjacent* if there is an edge between them.

Definition. A *path* is a sequence of adjacent vertices.*

* Typically, edges are not used twice.

Definition. A *circuit* is a path with the same initial starting and ending vertex.

Definition. A graph is *connected* if each vertex can be reached by path from each other.

Directed Graph (Diagram)

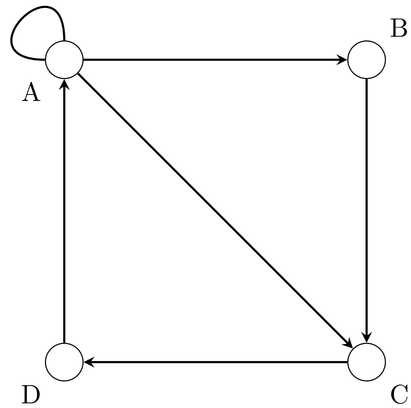

Figure 2: Directed Graph pt. 1

in-degree:
A - 1, B - 1, C - 2, D - 1
out-degree:
A - 2, B - 1, C - 1, D - 1

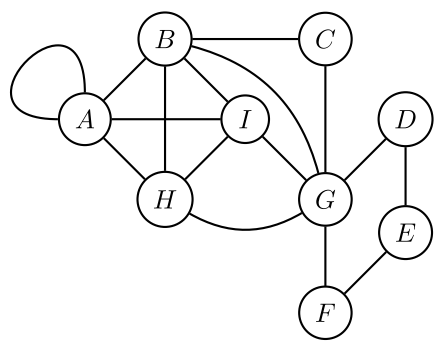

Figure 3: Social Network (Undirected Diagram)

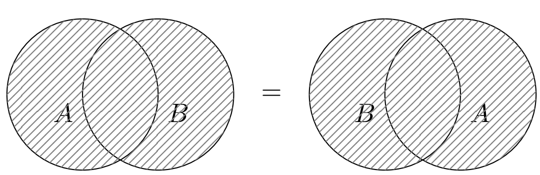

Figure 4: Demonstrating the commutative property of union, $A \cup B = B \cup A$, with sets A and B and their positions switched.

2.2 Sets

Definition. A *set* is a collection of objects, called the *elements*, or *members*.

We can list the elements like so:
$\{1, 2, 3\}, \{5, 10, 21, 23\}, \{1, 3, \ldots, 5\}, \{1, 3, 5, \ldots, 11\}, \{1, 3, 5, 7, \ldots\}$

Terminology 5. Typically, sets are upper-case Latin: $A, B, X, Y, \ldots$. Elements are lower-case Latin: $a, b, x, y, \ldots$.
And, if a is an element of X, $a \in X$. Likewise, if b is not in X, $b \notin X$.

Definition. There exists a set whose elements are exactly the elements of A for which $s(x)$ is true. $\{x \in A \colon P(x)\}$ where P is any predicate with A as its domain.

Example 2. $A = \{1, 3, 5, 7, 9\}$, then $\{x \in A \colon x^2 < 10\} = \{1, 3\}$. This is called set builder notation.

Definition. The statement that set A is *equal* to set B means $x \in A \iff x \in B$.

Theorem 2. Suppose A, B, C are sets with $A = B$ and $B = C$. Then,

$$A = C$$

Proof. Let $x \in A$. Since $A = B, x \in B$.
Since $B = C$, then $x \in C$. Now, let $y \in C$. Since $B = C, y \in B$. Since $A = B$ then $y \in A$.
Therefore, $A = C$ $\square$

Definition. Suppose A and B are sets. The statement that A is a subset of B, denoted by $A \subseteq B$, means if $x \in A$, then $x \in B$

Theorem 3. Let A be a set and P a predicate. $\{x \in A \colon P(x)\} \subseteq A$.

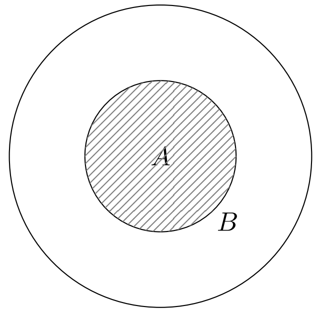

Figure 5: Set A as a subset of set B.

Proof. Let $x \in \{x \in A \colon P(x)\}$.

By definition, $x \in A$.

Therefore, $x \in A$. $\square$

Definition. The *empty set*, (i.e., $\emptyset$, $\{\}$), the set with no elements: if x exists, $x \notin \emptyset$.

Definition. Union and Intersection are defined as:

- Union: $A \cup B\colon x \in A \cup B$ if $x \in A$ or $x \in B$
- Intersection: $A \cap B\colon x \in A \cap B$ if $x \in A$ and $x \in B$.

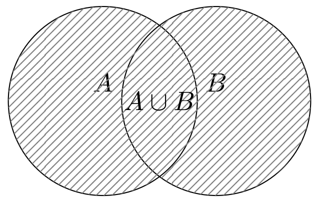

Figure 6: Venn diagram demonstrating the union of sets A and B.

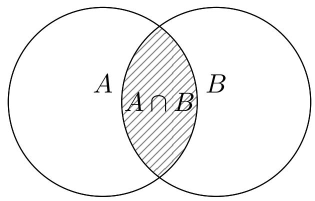

Figure 7: Venn diagram demonstrating the intersection of sets A and B.

Definition. Product: $A \times B$ is the set of all order pairs, (a, b), where $a \in A, b \in B$. $A = \{1, 2\}, B = \{2, 4, 6\}$ $A \times B = \{(1, 2), (1, 4), (1, 6), (2, 2), (2, 4), (2, 6)\}$

Definition. Suppose $A \subseteq U$. The *complement* of A, relative to U is $\{x \in U\colon x \notin A\} = A'$ (preferably, $\overline{A}$ or, A^c).

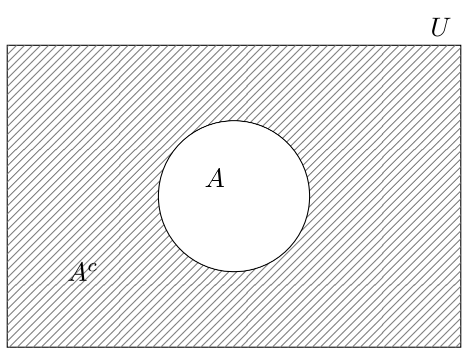

Figure 8: Set diagram with U, set A, and its complement, A^c (or $\overline{A}$), within U.

Theorem 4. Let A and B be sets. Then,

$$A \subseteq B \iff A \cap B = A$$

Proof. ($\Rightarrow$)
Show that if $A \subseteq A \cap B = A$

Let $x \in A \cap B$.
Then, $x \in A$ and $x \in B$. Therefore, $x \in A$.

Let $y \in A$.
Since, $A \subseteq B$, $y \in B$. So, $y \in A$ and $y \in B$.
Therefore, $y \in A \cap B$.

Proof. ($\Leftarrow$)
Show that $A \cap B = A$ then, $A \subseteq B$.

Let $x \in A$.
Since $A = A \cap B$, $x \in A \cap B$. So, $x \in A$ and $x \in B$.
Therefore, $x \in B$. ∎

2.3 Functions

Definition. Let X and Y be sets. A *function* f is a well-defined rule that associates each element $x \in X$ with a unique $y \in Y$. We write this as $f(x) = y$.

X Domain:
$g\colon \mathbb{R} \to \mathbb{R}$ by $g(x) = \frac{1}{x-2}$.

Y Co-Domain:
We require that f is defined for each $x \in X$ (i.e., the domain). However, it is okay if not for each $y \in Y$ is matched.

Terminology 6. $f\colon X \to Y$ is a subset of $X \times Y$ so that each $x \in X$ belongs to exactly one pair.

If $f\colon X \to X$ is a function, then this generates a directed graph:
$X = \{0, 1, 2, 3\}$. so, $f(0) = 1, f(1) = 3, f(2) = 1, f(3) = 3$

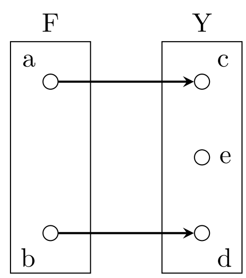

Figure 9: Directed Graph of a Function

Definition. Let $f\colon X \to Y$ be a function. we say f is a *one-to-one* or *injective* if for each $x_1, x_2 \in X$ if $f(x_1) = f(x_2)$, and $x_1 = x_2$.

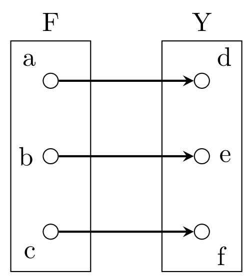

Figure 10: Directed Graph of an One-To-One Function

Example 1. Let $f\colon \mathbb{N} \to \mathbb{N}$ by $f(n) = 3n + 1$. Then, f is one-to-one.

Proof. Let $x_1, x_2 \in \mathbb{N}$ so that $f(x_1) = f(x_2)$. We hope to show $x_1 = x_2$.

$f(x_1) = 3x_1 + 1$ and $f(x_2) = 3x_2 + 1$. Then,

$$\begin{aligned} 3x_1 + 1 &= 3x_2 + 1 \\ 3x_1 &= 3x_2 \\ x_1 &= x_2 \end{aligned}$$

Therefore, f is one-to-one. $\square$

Example 2. Let $f\colon \mathbb{Z} \to \mathbb{Z}$ by $f(z) = z^2 - 3$. f is not one-to-one. Find $x_1 \neq x_2$ so that $f(x_1) = f(x_2)$.

Solution. consider $x_1 = 2, x_2 = -2$. Since $x_1 \neq x_2$, but $f(x_1) = 1 = f(x_2)$, f is not one-to-one.

Example 3. Let $f\colon \mathbb{N} \to \mathbb{N}$ by $f(x) = \frac{n}{n} = 1$.

Solution. Consider $n_1 = 2, n_2 = 3. f(n_1) = 1 = f(n_2)$.

Definition. $f: X \to Y$ is *onto* or *surjective*. If for-each $y \in Y$. there exists at least one $x \in X$ so that $f(x) = y$.

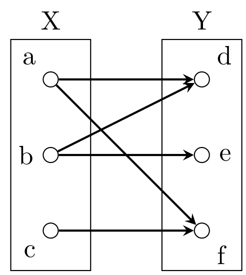

Figure 11: Directed Graph of an Onto Function

Example 4. $f: \mathbb{N} \to \mathbb{N}$ by $f(n) = n^2$ is not onto.

Solution. Consider $2 \in \mathbb{N}$ (the co-domain). there is no $n \in \mathbb{N}$ for which $n^2 = 2$.

Example 5. $g: \mathbb{N} \to \mathbb{N}$ by $g(n) = |n-1| + 1$ is onto.

Solution. Show g is onto: Let $y \in \mathbb{N}$. Goal: Find n so that $g(n) = y$. Consider $y \in \mathbb{N}$. Then, $g(y) = |y-1| + 1$. Since $y \geq 1, |y-1| = y-1$, so $g(y) = y - 1 + 1 = y$.

Example 6. Show h is onto: $h(n) = |n-7| + 1$

Solution. Let $y \in \mathbb{N}$. Consider $y + 6 \in \mathbb{N}$. Then, $(y+6) \in \mathbb{N}$. Then, $(y+6) \geq 0$, so $|(y+6) - 7| = (y+6) - 7$. Then, $h(y) = |y + 6 - 7| + 1 = y + 6 - 7 + 1 = y$.

Example 7. Suppose $f: X \to Y$ is one-to-one. Then we can define f^{-1} by $f^{-1}(y) = x \iff f(x) = y$

Solution. $f: \mathbb{R}$ by $f(x) = x^2$ does *not* have an inverse. $g: [0, \infty) \to [0, \infty)$ by $g(x) = x^2$ *does* have an inverse: $g^{-1}(x) = \sqrt{x}$.

Example 8. Suppose $f: X \to Y$ and $g: Y \to Z$ are functions that we define as $h = g \circ f$ by $h(x) = g(f(x))$ so $h: X \to Z$.

2.4 Relations

Definition. A *relation* is a generalization of the idea of a function. Thus,

$$R\colon X \to Y \text{ is a relation if } R \subseteq X \times Y.$$

Note: All Functions are Relations, but not all Relations are Functions.
If an x value can be mapped to more than one y value, the statement is not a function.

Example 9. For the given relation, is it a function? Consider the relation $R\colon \mathbb{R} \to \mathbb{R}$, such that $R := \{\{2,3\},\{4,3\},\{3,4\},\{2,8\}\}$.

This is not a function because an x-value, 2, is mapped to multiple y-values.

Terminology 7.

$$R = \{(\ , \ ), (\ , \ ), (\ , \ )\}$$

Assume both p and $\neg p$ are true and let q be a statement.

$$\begin{aligned} < \ &: \ \mathbb{N} \to \mathbb{N} \\ < \ &= \ \{(1,2),(1,3),(1,4),\dots, \\ &\qquad (2,3),(2,4),(2,5),\dots, \\ &\qquad (3,4),(3,5),(3,6),\dots, \\ &\qquad \dots\} \end{aligned}$$

Example 10. $X, Y =$ set of Hendrix Students: $a\ R\ b$ if a has taken a class with b.

Definition. If $R\colon X \to X$ is a relation, we say that R is:

- *Reflexive* if for each $x \in X$, $x\ R\ x$.
- *Symmetric* if when $a\ R\ b$, then $a\ R\ b$.
- *Transitive* if when $a\ R\ b$, and $b\ R\ c$, then $a\ R\ c$.

Note: There exists relations that can be none, one, two, or all of the previous traits.

Definition. If it a relation is satisfies all three traits, then it is called an *equivalence relation.*

Note: This allows us to *partition* a domain.

Definition. Each partition is called an *equivalence class.* Denoted as $[a]$.

Example 11. Let $a, b \in \mathbb{N}$ and say $a \; R \; b$. Such that if a divides into b.

- $3 \; R \; 12$
- $7 \; R \; 49$
- $5 \not R \; 12$

R is reflexive. Let $a \in \mathbb{N}$. a is divisible by a, so $a \; R \; a$.
R is not symmetric. Consider $7 \; R \; 49$, but $49 \not R \; 4$.
R is transitive. Consider $a \; R \; b$, and $b \; R \; c$.
Then, $a \cdot k$, and $c = b \cdot l$ for some k, l.
Then, $c = b \cdot l$
$c = (a \cdot k) \cdot l$
$c = a \cdot (k \cdot l)$

Example 12. If [a] is a set (in fact, a subset of X), then $a \in [a]$ if:

- $b \in [a]$, then $a \in [b]$. $a \; R \; b \Rightarrow [a] = [b]$.
- $a \in [b], b \in [c], a \in [c]$.

Modular Arithmetic

Definition. Let $n \in \mathbb{N}$. Let $a, b \in \mathbb{Z}$.
We say a is *equivalent to* b, modulo (%), $a \equiv b \mod n$, if $(a - b)$ is divisible by n.

- Is $a \equiv a \mod n$ reflexive? Yes.
  $a - a = 0$ which is certainly divisible by n.
- Is $a \equiv a \mod n$ symmetric? Yes.
  If $a \equiv b \mod n$ is $b \equiv a \mod n$?
  $(a - b)$ is divisible by n. Well, $b - a$ is also divisible by n.
- Is $a \equiv a \mod n$ transitive? Yes.
  $a \equiv b \mod n$, and $b \equiv c \mod n$, is $a \equiv c \mod n$?

2.6 Graph Theory

Graphs

Definition. A *graph* $G = (V, E)$, is a pair of sets V, the *vertex set*, and E, the *edgeset*, so that each element of E has the form $\{v_i, v_j\}, v_i, v_j \in V$.

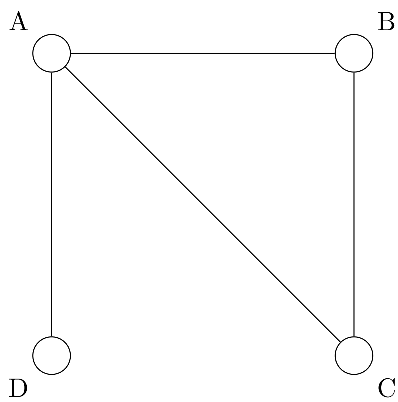

Figure 12: Graph with vertices and edges.

G:
$V = \{a, b, c, d\}$,
$E = \{\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}\}$

Definition. Given $v \in V$, the *degree* of v (denoted as $\deg(v)$) is the number of edges which include v.

Definition. We say vertex u is *adjacent* to vertex v if, and only if, $\{u, v\} \in E$.

Note: For us, no vertex is adjacent to itself because we are ignoring graphs that have loops.

Definition. A *path* from vertex v_0 to vertex v_n is a sequence $v_0, v_1, v_2, \ldots, v_n$, where each $v_i \in V$ and $\{v_i, v_{i+1}\} \in E$.

Definition. A path is *simple* if no edge occurs twice.

Definition. Two vertices are *connected* if there is a path (edge) from one to the other. In other words, a graph is connected if each pair of vertices are connected.

Circuits

Definition. A *circuit* is a path with the same starting and ending vertex.

Definition. The *complete graph* on n vertices, K_n, is the connected graph where each vertex is adjacent to each other.

Figure 13: k_1 (just a single vertex)

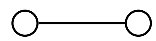

Figure 14: k_2 (two vertices and an edge)

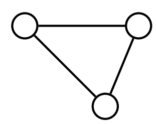

Figure 15: k_3 (three vertices and three edges)

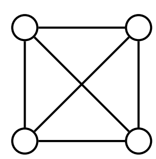

Figure 16: k_4 (four vertices and six edges)

Bipartite

Definition. A graph is *bipartite* if the vertex set $V = v_1 \cup v_2$, $v_1 \cap v_2 = \emptyset$, and no vertex in V_1 is adjacent to any other in V_1 and no vertex in V_2 is adjacent to any other in V_2.

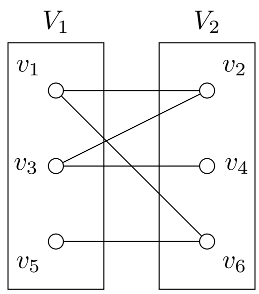

Figure 17: Bipartite Graph

Trees

Definition. A graph is *tree* if it is connected and has no circuit.

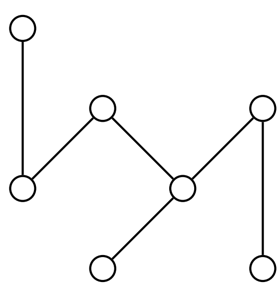

Figure 18: Tree

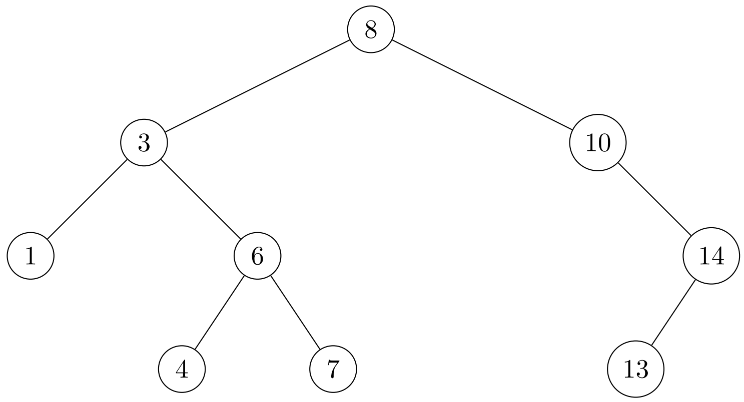

Figure 19: Binary Search Tree

Lemma. If G is a tree, G has at least one vertex of degree 1.

Proof. For the sake of contradiction, suppose each vertex has degree ≥ 2. Pick a vertex, v_0. Since, $\deg(v_0) \geq 2$, it is adjacent to some v_j. Because $\deg(v_1) \geq 2$, it has an edge distinct from $\{v_0, v_1\}$, follow it to v_2. Then, v_2 has edge distinct from $\{v_1, v_2\}$, follow it to $v-3$. If $v_3 = v_0(v_1, v_2)$ we have a circuit. v_3 has edge distinct from $\{v_2, v_3\}$. Go to v_4. Continue .... Either some v_j is visited again, or $v_0, v_1, v_2, \dots v_n$. Therefore, it must be a circuit.

Hence, G must have a vertex with degree of at least 1 such that $1 \leq$. □

Theorem 1. A tree with n vertices always has $n-1$ edges.

Proof. By the Lemma, there exists a vertex of degree 1. Remove it and its edge. We still have a tree. This new tree has vertex of degree 1. Remove it and its edge. Continue until you get to k_2, then k_1. We stop with 1 vertex, 0 edges, we have removed $n-1$ vertices. Each edge was removed.

Thus, we threw out *all* $n-1$ edges. □

Euler Circuit

Definition. A graph has an *Euler circuit* if there is a circuit which uses each edge only once. Similarly, an *Euler path* is a path which uses each edge – start and end vertices are distinct.

Notes: G has an Euler circuit if, and only if, it is connected and each vertex has an even degree. Intuitively, if each vertex has an even degree, then if you come into the vertex through the entrance (first edge), and you leave through the exit (second edge) you have used up both openings.

G has an Euler path if, and only if, it is connected and has exactly two vertices of odd degree.

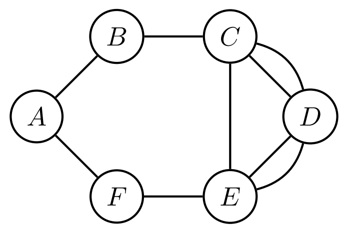

Figure 20: Euler Circuit

For this to be an Euler path, simply remove the line that connects C to E.

Isomorphism

Definition. Two graphs, $G = (V, E)$ and $H = (W, f)$ are *isomorphic* if $\exists f : V \to W$ which is one-to-one, onto, and $\{v_i, v_j\} \in E \iff \{f(v_i), f(v_j)\} \in F$.

Vertex Colorings

> **Definition.** Given a graph G, assign a color to each vertex so that two adjacent vertices have different colors.

If G contains a triangle (i.e., if it has a copy of k_3), we need at least 3. In G contains a copy of K_n, we need at least n.

If a graph has no overlapping paths, the graph requires no more than 4 colors.

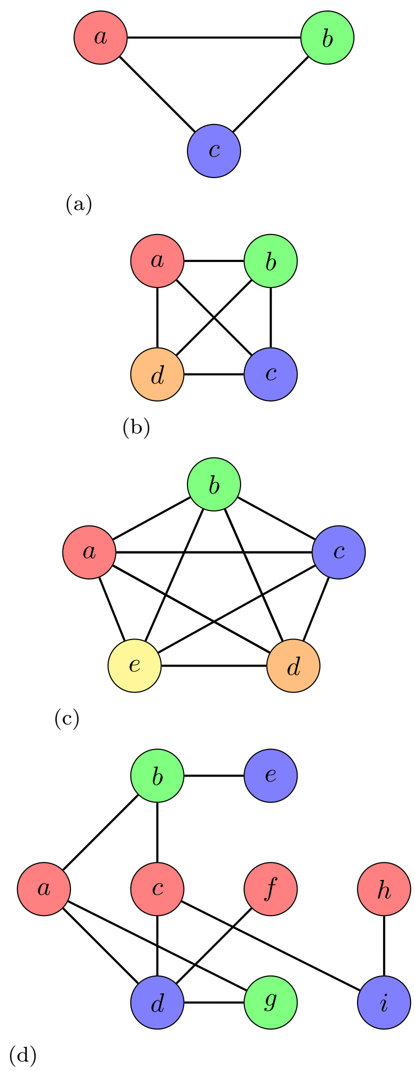

Figure 21: Vertex Colorings

Hamilton Graphs

Definition. A graph has a *Hamilton Circuit* if there is a circuit that uses each vertex once.

Notes:
This is different from Euler, as Euler uses edges. This is specifically for vertices.
If it has a vertex of degree 1, it cannot have a Hamilton circuit.

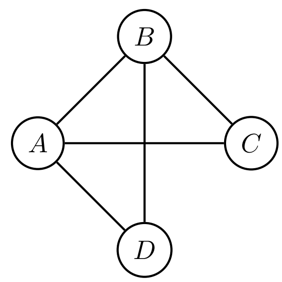

Figure 22: Hamilton Circuit

3 Recursive Thinking

3.1 Recurrence Relations

$$A(n) = \begin{cases} 1 & \text{if } n = 1 \\ A(n-1) + n & \text{if } n > 1 \end{cases}$$

Where $A(n) = 1$ is the base case, and $A(n) = A(n-1) + n)$ is the recursive case.
For example, consider $A(2)$

$$\begin{aligned} A(2) &= A(2-1) + 2 \\ &= A(1) + 2 \\ &= 1 + 2 \\ &= 3 \end{aligned}$$

n	$A(n)$
1	1
2	3
3	6
4	10
5	15
6	21
7	28

Table 1: $A(n)$ Bottom Up

$$\begin{aligned} A(7) &= A(6) + 7 \\ &= A(5) + 6 + 7 \\ &= A(4) + 5 + 6 + 7 \\ &\vdots \\ &= A(1) + 2 + 3 + 4 + 5 + 6 + 7 \end{aligned}$$

Table 2: $A(n)$ Top Down

What about $A(100)$? We would use $\frac{n(n+1)}{2}$. We can always do bottom-up or top-down, but it might be tedious. Can we find and prove closed form explicit expression?

$$S\colon \mathbb{N} \to \mathbb{N} \text{ by: } S(n) = \begin{cases} 1 & \text{if } n = 1 \\ A(n-1) + 2n - 1 & \text{if } n > 1 \end{cases}$$

This can be modelled like $S(n) = n^2$.

$$f\colon \mathbb{N} \to \mathbb{N} \text{ by: } f(n) = \begin{cases} 1 & \text{if } n = 0 \\ n \cdot f(n-1) & \text{if } n > 0 \end{cases}$$

This can be modelled like $f(n) = n!$.

Fibonacci

Start with a pair of rabbits in month 1. After 2 total months, any rabbit pair makes a new pair. This can be modelled with the piece-wise function below:

$$F\colon \mathbb{R} \to \mathbb{R} \text{ by: } F(n) = \begin{cases} 1 & \text{if } n = 1 \text{ or } n = 2 \\ F(n-1) + F(n-2) & \text{if } n > 2 \end{cases}$$

n	$F(n)$
1	1
2	1
3	2
4	3
5	5
6	8
7	13

Table 3: Fibonacci Bottom Up

$$\begin{aligned} F(7) &= F(6) + F(5) \\ &= F(5) = F(4) + F(4) + F(3) \\ &= A(4) + 5 + 6 + 7 \\ &= \vdots \\ &= F(1) + F(2) + 11 \\ &= 13 \end{aligned}$$

Table 4: Fibonacci Top Down

3.2 Mathematical Induction

Example 1. Suppose we want to prove some logical predicate $P(n)$ is true for all integers $n \geq 1$.

- Base Case: Show that $P(1)$ is true.

- Inductive Hypothesis: Suppose that for $n > 1$, $P(n-1)$ is true.

- Inductive Step: Use IH to show $P(n-1) \to P(n)$.

$$\begin{aligned}
&P(1) \\
&P(1) \to P(2) \\
&P(2) \\
&P(2) \to P(3) \\
&P(3) \\
&P(3) \to P(4)
\end{aligned}$$

Example 2. $S(n) \begin{cases} 1 & \text{if } n = 1 \\ s(n-1) + 2n - 1 & \text{if } n > 1 \end{cases}$

We are trying to show that $s(n) = n^2$.

- Base Case: $n = 1$
  $S(n) = S(1) = 1$, by definition, and is also equal to 1^2, which is equal to n^2.
- Inductive Hypothesis:
  Suppose for $n > 1$, $S(n-1) = (n-1)^2$.
- Inductive Step:

$$\begin{aligned}
S(n) &= S(n-1) + 2n - 1 \\
&= (n-1)^2 + 2n - 1 \\
&= n^2 - 2n + 1 + 2n - 1 \\
&= n^2
\end{aligned}$$

Use induction to prove $t(n) = 3^n$.

$$t(n) = \begin{cases} 1, & n = 0 \\ 3, & n = 1 \\ 2t(n-1) + 3t(n-2), & n > 1 \end{cases}$$

- Base Case (1): $n = 0 : t(n) = t(0) = 1 = 3^0 = 3^n$
- Base Case (2): $n = 1 : t(n) = t(1) = 3 = 3^1 = 3^n$
- Inductive Hypothesis: Suppose for $n > 1$, $t(n-1) = t^{n-1}$ and $t(n-2) = 3^{n-2}$
- Inductive Step:

$$\begin{aligned} t(n) &= 2t(n-1) + 3t(n-2) \\ &= 2 \times 3^{n-1} + 3 \times 3^{n-2} \\ &= 2 \times 3^{n-1} + 3^{n-1} \\ &= 3 \times 3^{n-1} \\ &= 3^n \end{aligned}$$

3.3 Recursive Definitions

In general, one or more base cases imply that some recurrence on the things themselves.

Example 3. Let $X \subseteq \mathbb{R}$ have the following definition.

Proof.

Base Case: $1 \in X$

Recursion: If $x \in X$, then $x + 2 \in X$.

X is all odd positive integers. □

Example 4. What if we want R to include all odd integers, not just the positive?

Proof.
B: $1 \in X$.
R1: if $x \in X$, $x + 2 \in X$.
R2: if $x \in X$, $x - 2 \in X$. □

Example 5. What if we want a set of all possible strings?

Proof. Suppose that we have a finite alphabet of symbols. $a_1, a_2, \ldots, a_3$. Then, let the set S be defined by:
B1: The empty string $\lambda \in S$
B2: $a_i \in S_i$ for all i.
R: If $x, y \in S$ the concatenation is $xy \in S$.
□

Example 6. What if we want a set of all palindromes?

Proof. Define $P \subseteq S$ by,
B1: $\lambda \in P$
B2: $a_i \in P$
R: if $x, y \in P$, $xyx \in P$.
Given an alphabet, $A = \{a, b, c, d\}$, what would the set of P look like? $P = \{\lambda, a, b, c, d, aba, bab, bcb, cbc, aaa, aa, bcbaa$
Think, recursive step changes from $xy \in P$ in example 5 to $xyx \in P$ in example 6. □

Example 7. Let T be the set of all tress and define $B \subseteq T$ by:

1. **B1**: The empty tree (i.e., no vertices) is in B.
2. **B2**: A simple vertex tree is also in B.
3. **R**: Suppose $T_1, T_2 \in B$ with roots r_1 and r_2, respectively.

3.4 Induction Examples - Cont.

Example 8. Prove that $n < 2^n$ for all integers $n \geq 1$.

Proof.

Base Case: $n = 1$
$n = 1 < 2 = 2^1 = 2^n$

Inductive Hypothesis:
Let $n > 1$ and suppose that $n - 1 < 2^{n-1}$.

Inductive Step:

$$
\begin{aligned}
n &= (n-1) + 1 \\
&< 2^{n-1} + 1 \\
&< 2^{n-1} + 2 \\
&\leq 2^{n-1} + 2^{n-1} \\
&= 2(2^{n-1}) \\
&= 2^n
\end{aligned}
$$

□

Example 9. Prove that $n^3 - n$ is divisible by 3 for all integers $n \geq 0$.

Proof.

Base Case:
For $n = 0$, $n^3 - n = 0^3 - 0 = 0 = 3 \times 0$, therefore divisible by 3.

Inductive Hypothesis:
Suppose that $n > 0$ and $(n-1)^3 - (n-1)$ is divisible by 3.

Inductive Step:
We know that since $(n-1)^3 - (n-1)$ is divisible by 3, then $(n-1)^3 - (n-1) = 3k$ for some integer k.

$$
\begin{aligned}
3k &= (n-1)^3 - (n-1) \\
&= n^3 - 3n^2 + 3n - 1 - n - 1 \\
&= n^3 - 3n^2 + 2n
\end{aligned}
$$

and then,

$$
\begin{aligned}
n^3 - n &= n^3 - n - 3n^2 + 3n^2 - 3n + 3n \\
&= (n^3 - 3n^2 + 2n) + 3n^2 - 3n \\
&= 3k + 3n^2 - 3n \\
&= 3(k + n^2 - n)
\end{aligned}
$$

Thus, $n^3 - n$ is divisible by 3. □

Example 10. Suppose A is a finite set with $n \geq 0$ elements. Then $\mathcal{P}(A)$ has 2^n elements.

Proof.

Base Case:
For $n = 0$: If A has 0 elements then $A = \emptyset$ and $\mathcal{P}(A) = \{\emptyset\}$
The number of elements of $\mathcal{P}(A) = 1 = 2^0 = 2^n$

Inductive Hypothesis:
Suppose $n > 0$, and any set of size $n-1$ has a power set of size 2^{n-1}.

Inductive Step:
Let A be a set with n elements. Since $n > 0$, A has at least one element, let this element be a. Let $B = \{x \in A \mid x \neq a\}$, B has $n-1$ elements. $\mathcal{P}(B)$ has 2^{n-1} elements. Any subset of B is a subset of A. Any subset of A which contains a is a union of some $C \subseteq B$ and a. B has 2^{n-1} subsets. Thus, each subset of A is either:

- a subset of B (2^{n-1} of them)
- a subset of B union with a (2^{n-1} of them)

and A has $2^{n-1} + 2^{n-1} = 2^n$ subsets. □

3.5 Strong Induction

- Base Case(s): Let b be the smallest base case.
- Inductive Hypothesis: Suppose $n >$ largest base case so that for all $b \leq k < n$, the desired property holds.
- Show the inductive hypothesis holds for n.

Example 11. For Problem 4 on Practice Set V, prove $f(n) = 2^n$, for $n \geq 0$.

Proof.

Base Case:
Exactly the same as the practice set base case.

Inductive Hypothesis:
Suppose $n > 1$ and that for each k, $0 \leq k < n$, $F(k) = 2^k$.

Inductive Step:

$$\begin{aligned} F(n) &= F(n-1) + 2 \cdot F(n-2) \\ &= 2^{n-1} + 2 \cdot 2^{n-2} \\ &= 2^{n-1} + 2^{n-1} \\ &= 2^n \end{aligned}$$

$\square$

Example 12. Suppose you have an unlimited number of \$0.03 and \$0.08 stamps. What values can you make? In other words, prove that we can make all values that are at least \$0.14.

Proof.

Base Case:

$$\begin{aligned} n &= \$0.14 \colon \$0.03 + \$0.03 + \$0.08 \\ n &= \$0.15 \colon 5 \cdot \$0.03 \\ n &= \$0.16 \colon 2 \cdot \$0.08 \end{aligned}$$

Inductive Hypothesis:
Suppose $n > \$0.16$ and if k is $\$0.14 \leq k < n$, we can make k cents.

Inductive Step:
Our goal is to make n cents. Since $n > 16$, $n - 3 \geq 14$.
We can make $n - \$0.03$ cents by the inductive hypothesis.
Choose a correct number of \$0.03 and \$0.08 stamps to make $n - \$0.03$. Then, add one \$0.03 stamp. We have made n stamps. $\square$

3.6 Structural Induction

Intended for recursively defined sets. In other words:

- Show Base Case elements work.
- Suppose the "small" elements in R work, and show their combinations works.

Example 13. Recursively defined set: Let X be defined as:
B1: $2 \in X$
B2: $-2 \in X$
R: If $x, y \in X$, $x + y \in X$
Prove that if $z \in X$, z is even. AND Prove that if z is even, then $z \in X$. (Will require TWO proofs.)

Proof.

Base Cases:
For $z = 2$: Then, $z = 2$, which is even.
For $z = -2$: Then, $z = 2(-1)$, which is even.

Inductive Hypothesis:
Suppose $x, y \in X$ are both are even.

Inductive Step:
Show $x + y$ is even (i.e, $x = 2k$ and $y = 2l$ for $k, l \in \mathbb{Z}$). Then,

$$\begin{aligned} x + y &= 2k + 2l \\ &= 2(k + l) \end{aligned}$$

and so is also even. □

Proof.

Base Cases:
First, prove if $z \geq 0$. then, for $z = 2 \cdot k$ for $k \geq 0$. For $k = 0$, $z = 0 = 2 + (-2) \in x$

Inductive Hypothesis:
Suppose for $k > 0, 2(k - 1) \in X$.

Inductive Hypothesis:
$2k = 2(k - 1) + 2 \in X$, then by the inductive hypothesis, $\in X$, then by B1, so $2k \in X$. □

3.7 Data Structures

Lists – An Iterative Data structure

Definition. *Lists* are linearly ordered. E.g., $a \to b \to c$, or $a \leftarrow b \leftarrow c$

Terminology 8. We write these as `lst` $= [x_1, x_2, \ldots x_n]$ for math, and $[x_0, x_1, \ldots x_{n-1}]$. For an index in the list, you could write that as $[x]_i \to$ `lst`$[i]$

Example 14. Given a list whose length is guaranteed to be ≥ 1, write an algorithm that returns the maximum value of the list.

Recursive:

```python
# Define the function maximize
def maximize(lst):
    # Check if the list contains only one element
    if len(lst) == 1:
        return lst[0]
    else:
        # Call maximize()
        pass
```

Iterative:

```python
def find_maximum(lst):
    # Initialize the maximum with the first list element
    m = lst[0]
    # Iterate over each item in the list
    for item in lst:
        # If the current item is greater than the current
            maximum, update m
        if m < item:
             m = item
    # Return the maximum value found
    return m
```

Tree Traversal Algorithms

```python
class Node:
    def __init__(self, value):
        self.value = value
        self.left = None
        self.right = None

# Traversal:
def inorder_traversal(node):
    if node:
        inorder_traversal(node.left)
        print(node.value)
        inorder_traversal(node.right)

def preorder_traversal(node):
    if node:
        print(node.value)
        preorder_traversal(node.left)
        preorder_traversal(node.right)

def postorder_traversal(node):
    if node:
        postorder_traversal(node.left)
        postorder_traversal(node.right)
        print(node.value)
```

4 Quantitative Thinking ('Counting')

4.1 Basic Counting Techniques

Yap 1. Counting exists under the umbrella of combinatorics. Counting can be deceptively hard, it's easy to talk yourself into the wrong answer.

Definition. *Addition Rule*: If we have two sets of options, A and B, and $A \cap B = \emptyset$, the total number of options is $|A| + |B|$

But what if $A \cap B \neq \emptyset$?

$|A| + |B|$ counts everyone in A as well as everyone in B. We have counted anyone in both twice. Thus, we need a method that can count each

Definition. *Inclusion-Exclusion*: $|A \cup B| = |A| + |B| - |A \cap B|$. : $|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$

Yap 2. This definition is like counting majors in the class. We can count the number of math majors, then we can count the number of computer science majors. But, the total number of students isn't just those two numbers added together, because there are students who are both math and computer science majors, they must only be counted once.

Example 1. Suppose a survey asks about sports viewing habits. 28% of people say they watch **Basketball**; 29% of people say they watch **Baseball**; 19% of people say they watch **Soccer**.

What if:
14% $\sim$ Basketball and Baseball,
12% $\sim$ Basketball and Soccer,
10% $\sim$ Baseball and Soccer,
8% $\sim$ All 3?

Solution. What percentage of people watched at least one sport?

We get the solution from adding the survey numbers together (i.e., $28 + 29 + 19$); then, add $(-14 - 12 - 10)$ to the total to get $28 + 29 + 19 - 14 - 12 - 10$; but because we have under-counted, add in 8 to get the total percentage.

Yap 3. The name inclusion-exclusion for counting comes from counting everyone, then excluding the duplicates in order to achieve the accurate count.

Definition. *Multiplication*: Suppose you have sets of options, A and B. Select one from A and <u>one</u> from B. Hence, there are $|A| \times |B|$ options.

Example 2. Old style Arkansas License plate looked like 3-digits, with 3-letters. How many distinct license plates are available?

Solution. Well, there are 10 numbers that can be picked for each digit slot. Hence, we know that for the first 3 digits, it is simply 10^3. For the alphabet part, that is simply 26^3. Multiply these odds together, you get $= 17,576,000$.

Yap 4. Over 17 million numbers, Wow! that a lot of plates.

Example 3. How many binary strings exist for length of n? How many binary strings of length n has repeated 1s?

Solution. For the first question, simply 2^n. For the second question refer to Yap 5.

Yap 5. This is deceptively harder than just filling the slots, because each number effects the slot after it. You can't just immediately write an answer down. You need to make a decision tree, which makes it easier to find the pattern of possible combinations.

Definition. *Decision Tree*: For Example 3, we would construct something like below:

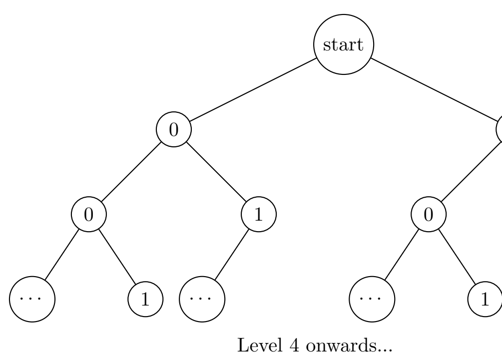

Figure 23: Binary Tree Illustrating Counting of Binary Strings with Repeated 1s

Yap 6. FIBONACCI WHY ARE YOU HERE?? HMMM?? Also, if you look at the sequence of possible lists of each length, take length 4, the answer would be 8. It's easy if you only checked strings of length 4 to say that there are half of the total possible strings, 2^4.

4.3 Selections and Arrangements

Permutations

Definition. We define a *permutation* as where we choose r people from n choices, where order matters!

$$nPr = P(n, r)$$

Example 4. In how many distinct ways can n people line up? Such that $n, n-1, n-2 \ldots, 1$ (n slots).

Solution. $n!$

Example 5. What if we want to have r people from n line up such that $(r \leq n)$. Such that $n, n-1, n-2, \ldots, n-r+1$ (r slots).

Solution. $\dfrac{n!}{(n-r)!}$

Definition. What if order does not matter?

$$\frac{nPr}{r!} = nCr = \frac{n!}{(n-r)!r!}$$

Example 6. We want to choose 5 people from 32 (32 being the amount of people in the class). How many possible combinations are there?

Solution. $\binom{32}{5} = 201,376.$

Example 7. An urn contains 12 numbered marbles, $1, 2 \ldots, 12$. 5 are selected. How many distinct outcomes are possible? Answer the following:

- With or without replacement

Solution. Consider the following chooses:

- With replacement, order matters. 12^5.
- With replacement, order does not matter.*

- Without replacement, order matters. ${}_{12}P_5 = \frac{12!}{7!}$
- Without replacement, order does not matter. $\binom{12}{5} = \dfrac{12!}{7! \cdot 5!}$

* We must use the Binomial Theorem:

$$(a+b)^n = \sum_{r=0}^{n} \binom{n}{r} a^r b^{n-r}$$

Consider the following solution for $(x+3)^5$ with the bimodal theorem:

$$\binom{5}{5} x^5 \cdot 3^0 + \binom{5}{4} x^4 \cdot 3 + \binom{5}{3} x^3 \cdot 3^2 + \binom{5}{2} x^2 \cdot 3^3 + \binom{5}{1} x^1 \cdot 3^4 + \binom{5}{0} x^0 \cdot 3^5$$

Note that $\binom{5}{4}$ and $\binom{5}{1}$ will always produce the same answer because $\binom{n}{n-r}$

4.5 Counting In Algorithms

```
x = 4
for i in [1,2,...,n]:
    x = x + 1

# If n = 3, what is the final value of x?

i = 1, x = 5 # There are three additions.
i = 2, x = 6
i = 3, x = 7

# In general, the line x = x + 1 runs n times total.

for i in [1,2,3,...,n]:
    c = c + i + 5 + b
    for j in [1,2,3,...n]:
        a = a + b + 1

# Note the outer for loop is n additions, and the inner
   loop is 2n additions.

for i in [1,2,3,...,n]:
    for j in [i, i + 1,..., n]:
        a = a + b + 1

# Total: 2 * (n(n+1)) / 2 = n ** 2 + n additions.
```

Big-O Classes (From handout p.3)

$n^3 + 3n^2 - 8n + 1 \in \Theta(n^3)$

Look at the highest power or piece that grows the fastest. **Important**: Constants do not matter.

$(n^3 + 4n)(\log_2(n) + 5) \in \Theta(n^3 \log_2(n))$

Multiply the largest growing terms.

5 Algorithms

Definition. Terms:
Pre condition: Initial input/state.
Post condition: Final state/output.

Yap 7. blah blah if you're familiar with algorithms its like a computer program, its what we really tell the computer what to do.

Definition. *Linear Search*: Given a list of integers and a target value, return True, if target is in the list, False if otherwise.

Say, for the list, lst, consider $[x_0, x_1, \ldots, x_{n-1}]$.
For the target value, t, we want to know two things:

1. Is t in the list?
2. Where is it?

Pseudo-code:

```
i = 0
x_n = t # Sentinal value
while t < > x_i: # "While t is not equal to x_i..."
    i = i + 1
if i = n:
    print ('Not␣there')
else:
    print('Found␣at␣i')

i = 0
x_n = t # Sentinal value
while t < > x_i: # "While t is not equal to x_i..."
if i = n:
    print ('Not␣there')
else:
    print('Found␣at␣i')
i = i + 1

# Now it checks index 0 ? before why would we have it as 0
   if we just skip to 1 anyway before our comparison
```

The maximum number of additions with this algorithm is n.
And the minimum number of additions is 0.

Yap 8. When reasoning with post-conditions, we can use pre-conditions that we know are true in order to simplify the algorithm.

Post-condition Claim	Pre-condition claim
$x_i = t$	$t \in list$

Table 5: something about conditions matching

Post-condition claim: $x_i = t$
Pre-condition claim: $t \in lst$

We want to prove this using proof by induction. (Note: The base case is always that the loop is not running.)

Base Case:
$t = x_0$.

Inductive Hypothesis:
Assume $x_i = t$ if $t \in lst$, assume x_i if t is later (?), and assume t appears only once in the list.

Inductive Step:
We need to prove that $x_{i+1} = t$ if t is at $i + 1$, and $x_{i+1} = t$, if t is not at $i + 1$.
Cases:

1. t is at $i + 1$: then x_i is not equal to t

Yap 9. Apparently coders don't need to prove their code works if they believe their code is sound in logic and would not be cost effective to prove.

1. What is the biggest value in a list?

```
big = x_0
for x_n in list
    if x_n > big
        big = x_n
print(big)
```

The maximum and minimum number of comparisons with this algorithm is $n - 1$.

Post-condition Claim	Pre-condition claim
$\forall x_i \in list, big > x_i$	$n \geq 1$

Table 6: something about conditions matching

We want to prove this by induction. (Note: The base case is always that the loop is not running.)

Base Case: $i = n \to n = 1$
big $= x_0$
$x_0 \geq x_0$ is true.

Inductive Hypothesis:
$i \leq n, \forall x_i \in$ list, big $\geq x_i$

Inductive Step:
We need to prove that $x_{i+1} = t$ if t is at $i + 1$, and $x_{i+1} = t$, if t is not at $i + 1$.
Cases:

1. t is at $i + 1$: then x_i is not equal to t

5.1 Greedy Algorithms

Definition. *Prim's Algorithm* finds a minimal spanning tree in an edge-weighted graph. Where a *Spanning Tree* is a tree that uses only the edges (and of course, the vertices) of the main graph.

To make a spanning tree: Make the original blank graph (just the vertices), throw edges in, and never make a circuit.
A *minimal* spanning tree is one that gives the total, smallest weight of the edges that are on the graph. That is, if $A--B$ is weight 5, and $A--C$ is weight 3, and $B--C$ is weight 1, how can you choose all edges that are of the smallest total weight?

5.2 Traveling Salesperson Problem

Given a connected weighted graph, find a minimal cost Hamilton circuit (use each vertex one time).

5.3 Algorithms review

Definition. Linear Search: iterate through list, one element at a time. $\theta(n)$

Definition. Binary Search: Given a sorted list, find a target value.

Unsorted list: $[7, 1, 8, 2, 4, 9]$, linear search, 6 iterations.
Sorted list: $[1, 2, 4, 7, 8, 9]$, binary search, 3 iterations or in general, $log_2 n$.