Overview (0.1)

What is *computability*?
- What are the fundamental capabilities and limitations of computers?
- Why are some problems harder than others?
  - Sorting is pretty easy...
  - ...but scheduling is very hard
- Why are some problems flat-out impossible?
  - The halting problem
  - Determining the truth or falsehood of a statement
- What are *automata*?
  - Why are they important?
  - More importantly, why are they *useful*?
Review of Mathematical Essentials

SECTION 0.2
Sets

Given elements $x$ and $y$, and sets $A$ and $B$:

**Containment**
- $x \in A$ - $A$ contains $x$.
- $x \notin A$ - $A$ doesn’t contain $x$.
- $A = \{x, y\}$ - $A$ contains only $x$ and $y$.
- $A = \{x \mid x \in \mathbb{N}, x > 50\}$ - $A$ contains the natural numbers higher than 50.

**Operators**
- $A \cup B$ – union
- $A \cap B$ – intersection
- $\overline{A}$ - complement

**Subsets**
- $A \subseteq B$ - $A$ is a subset of $B$.
  - $\forall x \in A, x \in B$
- $A \subset B$ - $A$ is a proper subset of $B$.
  - $\forall x \in A, x \in B$ and $A \neq B$.
- The power set of $A$ is the set of all subsets of $A$.

**Common sets**
- $\mathbb{Z}$ – the set of all integers
- $\mathbb{N}$ – the set of all natural numbers
- $\emptyset$ or $\phi$ - the empty set
Sequences and Functions

Sequences
- Like ordered sets
- Finite sequences are called $k$-tuples
- 2-tuples are also known as *ordered pairs*

Cartesian products of sets:
- $A \times B = \{(a, b) \mid a \in A \text{ and } b \in B\}$
- Can take it of any number of sets
- $A \times A = A^2, A \times A \times A = A^3$, etc.

Functions
- Map a *domain* onto a *range*
- $n$-ary functions take $n$ arguments
- $f: D \rightarrow R$
  - $abs: \mathbb{Z} \rightarrow \mathbb{Z}$
  - $add: \mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$

A function is...
- *One-to-one* (an *injection*) if it maps every element of the range from at most one element of the domain
- *Onto* (a *surjection*) if it maps every element of the range from at least one element of the domain
- A *bijection* if every element of the range is mapped by exactly one element of the domain
Relations

A *predicate* or *property* is a function with range \{TRUE, FALSE\}

A property with a domain of \(n\)-tuples \(A^n\) is an \(n\)-ary relation

Binary relations are common, and like binary functions, we use infix notations for them

Let \(R\) be a binary relation on \(A^2\). \(R\) is:

- *Reflexive* if \(\forall x \in a, x R x\)
- *Symmetric* if \(x R y \rightarrow y R x\)
- *Transitive* if \((x R y, y R z) \rightarrow x R z\)
- An *equivalence* relation if it is reflexive, symmetric and transitive
An undirected graph is a collection of nodes (or vertices) and edges that connect them:

- The degree of a node is the number of edges that connect to that node.
- Edges are unique – you can’t have two edges between the same pair of nodes.
- Nodes can have self-loops.
- Edges can also be labeled.
An undirected graph is a collection of nodes (or vertices) and edges that connect them

- The degree of a node is the number of edges that connect to that node
- Edges are unique – you can’t have two edges between the same pair of nodes
- Nodes can have self-loops
- Edges can also be labeled

A graph $G$ is a subgraph of graph $H$ if it has a subset of $H$’s nodes and all the related edges
Graphs: Paths

A *path* is a sequence of nodes connected by edges
- A *simple path* doesn’t repeat any nodes
- A graph is *connected* if every two nodes have a path
Graphs: Cycles

A path is a sequence of nodes connected by edges

- A simple path doesn’t repeat any nodes
- A graph is connected if every two nodes have a path
- A path is a cycle if it starts and ends on the same node
- A simple cycle contains at least three nodes and repeats only the first/last
Graphs: Trees

A *path* is a sequence of nodes connected by edges
- A *simple path* doesn’t repeat any nodes
- A graph is *connected* if every two nodes have a path
- A path is a *cycle* if it starts and ends on the same node
- A *simple cycle* contains at least three nodes and repeats only the first/last
- A graph is a *tree* if it is connected and has no simple cycles
A directed graph is a graph with arrows instead of lines.

- Edges between nodes $i$ and $j$ are ordered pairs $(i, j)$
- Directed paths are paths that follow the direction of the edges
A directed graph is a graph with arrows instead of lines

- Edges between nodes $i$ and $j$ are ordered pairs $(i, j)$
- Directed paths are paths that follow the direction of the edges
- A directed graph is strongly connected if every pair of nodes has a directed path
Directed Graphs and Binary Relations

Consider the relation “beats”
Proofs

SECTIONS 0.3-0.4
Proofs and Friends
All of these *should* be clear and concise; they *must* be precise

- **Definitions** describe the mathematical objects and ideas we want to work with
- **Statements** or assertions are things we say about mathematics; they can be true or false
- **Proofs** are unassailable logical demonstrations that statements are true
- **Theorems** are statements that have been proven true
- **Lemmas** are theorems that are only any good for proving other theorems
- **Corollaries** are follow-on theorems that are easy to prove once you prove their parent theorems
How To Prove Something

1. Understand the statement
2. Convince *you*rself of whether it is true or false
3. Work out its implications until you have a general sense of *why* it is true or false
   ◦ “Warm fuzzy feelings” don’t prove anything – but they can help you get *ready* to prove something
4. Break down any sub-cases you will need to prove
   ◦ After this you may need to cycle back to step 2
5. Get started
Formats of Proofs

- The book uses a highly narrative proof format
- There are several other valid ones
- Let’s look at two
Quasi-Narrative Format

Prove $A \cup B = \bar{A} \cap \bar{B}$

We can show this by showing $A \cup B \subseteq \bar{A} \cap \bar{B}$ and $\bar{A} \cap \bar{B} \subseteq A \cup B$.

Suppose $x \in A \cup B$.
Then by definition of complement, $x \notin A \cup B$.
Then by definition of union, $x \notin A$ and $x \notin B$.
Then by def. of complement, $x \in \bar{A}$ and $x \in \bar{B}$.
Then by definition of intersection, $x \in \bar{A} \cap \bar{B}$.
We have shown that if $x \in A \cup B$, $x \in \bar{A} \cap \bar{B}$.
Hence by definition of subset, $A \cup B \subseteq \bar{A} \cap \bar{B}$.

Now suppose $x \in \bar{A} \cap \bar{B}$.
Then by def. of intersection, $x \in \bar{A}$ and $x \in \bar{B}$.
Then by def. of complement, $x \notin A$ and $x \notin B$.
Then by definition of union, $x \notin A \cup B$.
Then by definition of complement, $x \in A \cup B$.
We have shown that if $x \in \bar{A} \cap \bar{B}$, $x \in A \cup B$.
Hence by definition of subset, $\bar{A} \cap \bar{B} \subseteq A \cup B$.

We have shown that $A \cup B \subseteq \bar{A} \cap \bar{B}$ and $\bar{A} \cap \bar{B} \subseteq A \cup B$.
Hence by set equality, $A \cup B = \bar{A} \cap \bar{B}$, QED.
Two-Column Format

Prove $\overline{A \cup B} = \overline{A} \cap \overline{B}$

1. Set equality

2. Let $x \in \overline{A \cup B}$
3. $\therefore x \notin A \cup B$ complement
4. $\therefore x \in \overline{A}$, $x \in \overline{B}$ union
5. $\therefore x \in \overline{A \cap B}$ intersection
6. $x \in \overline{A \cup B} \Rightarrow x \in \overline{A} \cap \overline{B}$ 2-5 subset
7. $\therefore \overline{A \cup B} \subseteq \overline{A} \cap \overline{B}$

8. Let $x \in \overline{A \cap B}$
9. $\therefore x \in \overline{A}$, $x \in \overline{B}$ intersection
10. $\therefore x \notin A$, $x \notin B$ complement
11. $\therefore x \notin A \cup B$ union
12. $\therefore x \in \overline{A \cup B}$ complement
13. $x \in \overline{A} \cap \overline{B} \Rightarrow x \in \overline{A \cup B}$ 9-13 subset
14. $\therefore \overline{A \cap B} \subseteq \overline{A \cup B}$

15. $\overline{A \cup B} \subseteq \overline{A} \cap \overline{B}$, $\overline{A \cap B} \subseteq \overline{A \cup B}$ 7, 14 set equality

16. $\therefore \overline{A \cup B} = \overline{A} \cap \overline{B}$
Types of Proofs

Direct Argument
- What we just did

Construction
- Prove something exists by showing how to make it

Contradiction
- Prove something is true by showing it can’t be false

Weak Induction
- Show that a statement is true for the case of 0
- Show that if it’s true for the case of $i$, it’s true for the case of $i + 1$

Strong Induction
- Show that it’s true for the case of 0
- Show that if it’s true for all of the cases $< i$, it’s true for the case of $i$
Next Time: Finite Automata