Lecture Notes for CSCI 341: Theory of Computation
Set 2: Math Review

Edward Talmage

August 30, 2024

We need some vocabulary to be able to talk about computers and the models we use to represent them, so we will spend some time reviewing mathematical topics.

1 Sets

- If we have a collection of items, we call it a *set*. Like a list, we can write a set directly: $\{2, 3, 5, 7\}$.
- Sets are unordered and do not allow repetition. $\{2, 3, 5, 7\} = \{2, 3, 3, 3, 3, 3, 3, 5, 7, 7\} = \{3, 2, 5, 7\}$.
- We say that an item is an *element* of a set if it is in the set: $3 \in \{2, 3, 5, 6\}, 6 \notin \{2, 3, 5, 7\}$
- Sets may be empty, finite, or infinite, depending on how many elements they contain.
- We can relate sets A, B is several ways:
  - A is a *subset* of B: $A \subseteq B$: Every element in A is also in B.
  - A is a *proper subset* of B: $A \subsetneq B$: Every element in A is in B, and there is an element in B which is not in A. The empty set $\{\}$ or $\emptyset$ is a subset of every set (proper subset of all but itself).
  - The *union* of A and B, $A \cup B$, is the set of all elements that are in A or in B.
  - The *intersection* of A and B, $A \cap B$, is the set of all elements that are in A and in B.
  - $A - B$ is the set of elements in A, but not in B.
  - If $B \subseteq A$, the complement of B in A is $A - B$. In general, the complement of A, $\bar{A}$, is all (possible) elements not in A.
- We can also specify sets by rules, e.g. $\{x \mid x/2 \in \mathbb{Z}\}$.
  - In general, $\{f(x) \mid P(x)\}$ where f is a function and P is a predicate (truth statement). Read as: "For every x where $P(x)$ is true, $f(x)$ is in the set.".
  - Example: $\{x \mid x \text{ is a car}\}$

Exercise: What is the set $\{2x \mid x/2 \in \mathbb{Z}\}$?
What is the set $\{\{a, b\} \mid a \in \{1, 2, 3\} \text{ and } b \in \{x, y, z\}\}$?

Sets probably seem fairly intuitive. Why are we making a big deal of reviewing them (likely for the umpteenth time)?

- They are one of the foundational concepts in mathematics.
  - Algebra and topology study certain kinds of sets.
  - Analysis looks at functions, which we will soon see require sets.
- We need them to define functions, which is what computers compute.

2 Relations and Functions

Definition 1. The *Cartesian Product* of two sets A and B is the set $A \times B = \{(a, b) \mid (a \in A) \wedge (b \in B)\}$.

- (a, b) is a *tuple*, or *finite sequence*. It is like a set, but ordered.

> **Exercise:** Is $(a, b) = (a, a, b)$? Why or why not?

- Order matters means repetition matters.
- $(a, b) \neq (b, a) \neq (a, a, b)$

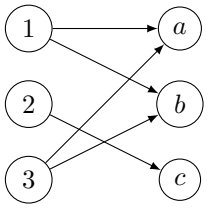
> **Exercise:** What is $\{dog, cat\} \times \{arf, meow\}$? What about $\{dog, cat\} \times \{arg, meow\} \times \{snoopy, garfield\}$?

- We cheat and don't nest tuples when doing multiple products. Thus, Cartesian Products are associative.

Definition 2. A *relation* is a subset of a Cartesian Product.

Example:

- Let $A = \{1, 2, 3\}$ and $B = \{a, b, c\}$.
- We can list a relation: $\{(1, a), (1, b), (2, c), (3, b), (3, a)\}$
- We can also draw it:



Definition 3. A *function* is a relation in which each element of the domain (first set) appears exactly once.

- Given an element of the domain, a function gives you exactly one element of the *codomain*.
- We write a function from A to B as $f : A \to B$.

> **Exercise:**
>
> - Draw the "divides" relation from $\{2, 3, 5, 7\}$ to $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Is this a function?
> - Reduce the above relation to an interesting function.
> - How many subsets does a set have? Why?
> - Let $A = \{1, 2, 3\}$ and $B = \{a, b, c, d\}$.
>   - How many Cartesian Products of A and B exist?
>   - How large is $A \times B$?
>   - How many relations from A to B are there?
>   - How many functions from A to B are there?

2.1 Properties of Functions and Relations

You are probably familiar with the basic properties a function might have:

- A function $f : A \to B$ is *onto* or *surjective* if $\forall b \in B, \exists a \in A s.t. f(a) = b$.

  Exercise: Define two sets. Write an onto function and function that is not onto between the sets. Be sure that your functions are, in fact, functions.

- A function $f : A \to B$ is *one-to-one* or *injective* if $\forall x \in f(A), \exists! a \in A s.t. x = f(a)$.

  Exercise: Using your two sets from the previous exercise, give an injection and a function which is not one-to-one.

- A function $f : B \to A$ is *bijective* if it is injective and surjective. Bijections are interesting becaus they are invertible: If you reverse the mapping for each element, you get a function from $B \to A$.

Relations also sometimes have nice properties.

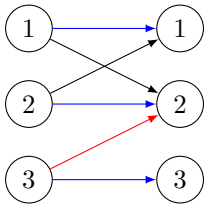
Aside: A *property* is a function from a set S to the set $\{True, False\}$. In this case, we are mapping from the set of all relations. If a relation has a property, it will map to True, and vice versa.

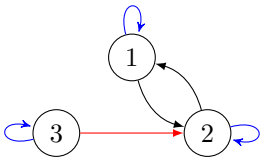
- If a property p is a mapping from the Cartesian Product of a set S with itself, then we call the function a *binary relation*.
  - Wait, what?!?
  - Consider an example: $g : \mathbb{Z} \times \mathbb{Z} \to \{T, F\}$ is a binary relation–a function from the product of a set with itself to True/False.
    $g(3, 5) = False, g(5, 3) = True, g(100, 100) = False, g(7, -1000) = True, \ldots$
    What function is this?
  - We often write binary relations differently than other functions. If R is a binary relation, then we often write aRb if and only if $R(a, b) = True$. Picture $a > b$ for the "greater than" binary relation. The idea here is that a binary relation indicates whether two elements of a set satisfy a certain property or have a certain relation.
- A binary relation $\sim$ is *reflexive* if $a \sim a$ for every $a \in S$. That is, $\sim (a, a) = True$ for every choice of a.
- A binary relation $\sim$ is *symmetric* if $a \sim b \Leftrightarrow b \sim a$ for every $a, b \in S$.
- A binary relation $\sim$ is *transitive* if $(a \sim b) \wedge (b \sim c) \Rightarrow (a \sim c)$ for every $a, b, c \in S$.
- A binary relation satisfying all three of these properties is an *equivalence relation*.

Exercise: Determine whether each of the following relations is reflexive, symmetric, and/or transitive.

- "in the same class year". What universe makes sense?
- "similar", among the set of triangles
- aRb, for integers a and b, if $a - b$ is even.
- $\geq$
- $>$
- > 4

- We can also draw binary relations, which can help visualize them:





Exercise: Draw a set with five elements, and draw two relations, one equivalence and one not. What properties of the picture are required for an equivalence relation?

- Equivalence relations *partition* a set into subsets of elements which are all related ("equivalent")to each other. These are known as *equivalence classes*.

Exercise: Determine the equivalence classes of each of the relations from the previous exercise which was an equivalence relation.

3 Graphs

Definition 4. An *undirected graph* G is a set of *vertices* V and a set of *edges* $E \subseteq V \times V$, such that E is a symmetric relation.

  - Symmetric means that $(i, j) \in E$ iff $(j, i) \in E$.
  - No need to draw double-headed arrows, just plain lines.
  - A *simple graph* is one with no loops or self-edges.

- The *degree* of a vertex $u \in V$ is the number of edges touching it: $|\{v \in V \mid (u, v) \in E\}|$.

Exercise: If E is a function, not just a relation, what can we say about degrees? What would the graph look like?

- A *path* in a graph is a sequence of vertices connected by edges.
  - A sequence of edges s.t. the first vertex of each edge after the first is the second vertex of the previous edge.
    * A *simple path* does not contain any node more than once.
    * A *cycle* is a path where the first and last vertices are the same.
- A graph is *connected* if for any $u, v \in V$, there exists a path from u to v, denoted $u \rightsquigarrow v$.
- A *subgraph* of a graph is a subset of the vertices and all edges which have both endpoints in that subset.
- A *directed graph (digraph)* is a graph in which the edge relation does not have to be symmetric.
  - Draw edges with arrows, since they have distinct start and end points.

Exercise: Represent the $\geq$ relation on $\{1, 2, 3, 4, 5\}$ as a digraph.

4 Strings and Languages

- An *alphabet* is a finite set, often denoted Σ, and a *language over alphabet* Σ is a set of *strings* (finite sequences) comprised of letters from Σ.
- String operations: Given two strings u, w
  - concatenation: uw

- reverse: u^R
- length: $|w|$
- u is a *substring* of w if u appears consecutively within w.
- lexicographic order: "dictionary order". Compare by position: If first elements differ, string with smaller first element is smaller. Else, compare second elements, order if different, move to third element if same, etc.
  - *shortlex* order is the same but the end of a string is ordered before any character.

5 Boolean Logic

- Let 1 represent True and 0 represent False. Define the following standard properties on the domain $\{0, 1\}$:

  Exercise: Why are these properties?

  - NOT ($\neg$): $\{0,1\} \to \{0,1\}, NOT(0) = 1, NOT(1) = 0$. Alternately, $NOT = \{(0,1),(1,0)\}$.
  - AND ($\wedge$): $\{0,1\} \times \{0,1\} \to \{0,1\}, AND = \{(0,0,0),(0,1,0),(1,0,0),(1,1,1)\}$

    Exercise: What other type of thing is AND? Is it an equivalence relation?

    Exercise: Define the following similarly:
    - OR ($\vee$)
    - XOR ($\oplus$)
    - IF ($IF-THEN$) ($\to$)
    - $ONLYIF$ ($\leftarrow$)
    - IFF ($\Leftrightarrow$)

- We can use these to combine statements with truth values (propositions).
- Specifically, we can create identities, statements which are always true. For example:
  - $(P \to Q) \Leftrightarrow (\neg P \vee Q)$
  - $P \wedge (Q \vee R) = (P \wedge Q) \vee (Q \wedge R)$
  - $P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$
  - $NOT(P \wedge Q) = (\neg P) \vee (\neg Q)$
  - $NOT(P \vee Q) = (\neg P) \wedge (\neg Q)$
  - $P = Q$ is the same as $P \Leftrightarrow Q$

  Exercise: Show that each of the above identities is always true.

6 Proof Techniques

While proving claims is an art form, more than a science, as each particular claim will have its own special features which one must grasp and manipulate to prove the desired fact, there are some common patterns that we use frequently enough to be worth reviewing.

- Direct Proof: Start with known facts, combine and convert to other facts until we reach the desired fact (conclusion).
  - Prove that $\overline{A \cup B} = \overline{A} \cap \overline{B}$

    Proof. If $x \in \overline{A \cup B}$, then $x \notin A \cup B$, so $x \notin A$ and $x \notin B$. Thus, $x \in \overline{A}$ and $x \in \overline{B}$, so $x \in \overline{A} \cap \overline{B}$.

    Exercise: Prove the converse.

□

- **Exercise:** Prove that in a graph, the sum of the degrees of all vertices is twice the number of edges.
- **Exercise:** Prove the corollary, that the sum of degrees is an even number.

Aside: A corollary is a subsidiary claim that follows directly from a previous one.

- Proof by Construction: To prove that something exists, build an example. All that is left is to prove that the example is what you claim.

Exercise: A *k-regular graph* is one in which all nodes have degree k.
- Prove that there is a 3-regular graph with 6 nodes.
- Prove that there is a 3-regular graph with any even number of nodes n, where $n > 2$.

  - $E = \{\{i, i+1\} \mid 0 \leq i \leq n-2\} \cup \{\{n-1, 0\}\} \cup \{\{i, i+n/2\} \mid 0 \leq i \leq n/2-1\}$
  - Nodes in a circle, connect to neighbors on each side and to the node directly opposite.

- Proof by Contradiction: Assume what you want is false, directly prove that that is impossible (by showing a contradiction). Thus, the desired claim must be true.

  - Claim: $\sqrt{2}$ is irrational.
    - Proof by Contradiction is great when there are too many (typically infinitely many) cases to check. Here, we could try squaring all rational numbers, but that would take a while.
  - *Proof.* Assume in contradiction that $\sqrt{2} = a/b$, where a and b are relatively prime integers. It follows that $2 = (a/b)^2 = a^2/b^2$. We can rearrange this to get $2b^2 = a^2$, which implies that a^2 is even. We then conclude that a is even, as the square of an odd number is odd (This is a miniature proof by contradiction!). We can then write $a = 2k$ for some integer k, and $a^2 = (2k)^2 = 4k^2$.

    Substituting this back into our previous equality, we now have $2b^2 = 4k^2$, or $b^2 = 2k^2$. Similar reasoning as before tells us that b is even, or $b = 2j$ for some integer j. But now a and b are both divisible by 2, which contradicts our assumption that they are relatively prime, and the claim holds. □

- Proof by Induction: Prove that property $P(n)$ is true for all integers $n \geq 1$. This is the most structured of our proof techniques, and is great because it lets us prove an infinite number of cases in a (small) finite number of steps.

  - Base Case: Prove, typically directly, that $P(1)$ is true.
  - Inductive Hypothesis: Assume that for some $k \geq 1$, $P(k)$ is true.
  - Inductive Step: Prove that $P(k+1)$ must then also be true.

    Really, showing that $P(k)$ implies $P(k+1)$, so we do not need to know in advance that $P(k)$ is true. The trick here is to massage $P(k+1)$ to a form that includes $P(k)$, then substitue from the Inductive Hypothesis.
  - Conclude that $P(n)$ is true for all $n \geq 1$.

Exercise: Prove inductively that $1 + 4 + 9 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6}$

  - BC: $1 = \frac{1*2*3}{6}$
  - IH: Assume $\sum_{i=1}^{k} i^2 = \frac{k(k+1)(2k+1)}{6}$.
  - IS: $\sum_{i=1}^{k+1} i^2 = \sum_{i=1}^{k} i^2 + (k+1)^2 = \frac{k(k+1)(2k+1)}{6} + (k^2 + 2k + 1)$. Working from the other direction, $(k+1)(k+2)(2k+3) = 2k^3 + 3k^2 + 4k^2 + 2k^2 + 6k + 3k + 4k + 6 = 2k^3 + 9k^2 + 13k + 6$. Comparing, we can determine that the two expressions are equal, proving the claim.

6.1 Inductive Definitions

You may notice, or recall from Algorithms, that induction uses the same fundamental idea as recursion, only in reverse. We can use a similar structure for defining objects, known as *inductive definitions*.

Definition 5. Let Σ be an alphabet. Define the language Σ^* as:

$$\Sigma^* := \begin{cases} \varepsilon \in \Sigma^* \\ ua \in \Sigma^* \quad \forall u \in \Sigma^*, a \in \Sigma \end{cases}$$

Exercise: Give an English description of Σ^*.

Exercise: Define Σ^+ (the language of all non-empty strings over Σ) inductively.

Exercise: Inductively define the set of all even-length strings over $\Sigma = \{0, 1\}$. Similarly for odd-length strings.

$$\begin{cases} \varepsilon \in L \\ u00, u01, u10, u11 \in L \quad \forall u \in L \end{cases}$$

Exercise: Inductively define the length of a string.

$$\begin{cases} |\varepsilon| = 0 \\ |u| = |w| + 1 \quad u = wa, a \in \Sigma \end{cases}$$