Tutorial 01

University of Victoria

CSC 320 - Spring 2023

Foundations of Computer Science

Teaching Team

Learning Outcomes:

- Remember essential mathematical concepts.
- Become familiar with countability.
- Become familiar with set theory.
- Become familiar with languages.

Interesting Article:

"Undergraduates' Example Use in Proof Construction: Purposes and Effectiveness" [1]

January 17th, 2023

Outline

A review on essential mathematical concepts and an overview on languages.

1. **Countability**: A brief overview.
2. **Set Theory**: A review of set theory and describing sets.
3. **Proof Types**: A review on proof by contradiction, contrapositive, induction, and construction.
4. **Languages**: A review on alphabets, symbols, strings, etc..

Countability

We know that there are three types. Finite, Countably Infinite, and Uncountably Infinite.

Finite means that you can count the elements up to some number n.

Countably Infinite maps to N, the set of natural numbers.

Uncountably Infinite means there exists no way of counting that maps to N, the set of natural numbers.

subsection 8.4 - Countable Sets and Sequences in Dr. Gary's Notes [2] is a good resource if you need further examples.

Set Theory

A Few Basic Definitions

- set: $S = \{a, b, c, d\}$
- membership: $a \in S$, $f \notin S$
- empty set: $\emptyset$
- singleton set: set with exactly 1 element
- unordered pair set: set with 2 elements

subsection 0.2 - Mathematical Notions And Terminology in Introduction to the Theory of Computation is a good resource if you need further examples.

A Few Basic Definitions

- union: $A \cup B$
- intersection $A \cap B$
- complement: $\bar{A}$
- set difference: $A - B$
- cartesian / cross product: $A \times B$

subsection 6.1 - Cartesian Products in Dr. Gary's Notes [2] is a good resource if you need further examples.

Describing Sets

$$\{x \mid x = 2m \text{ for each m in } \mathbb{N},\ m > 5\}$$

Writing Sets

- The set of all integers greater than 5.
- The set of all strings 0...01...1 where all 0's come before 1's and there are twice as many 0's as 1's.
- The set of all odd number ≥ 1.

Proof Types

I highly recommend reading subsection 0.4 - Types of Proof in Introduction to the Theory of Computation. It goes over contradiction, induction, and construction.

Contradiction

Prove that $\sqrt{2}$ is an irrational number by contradiction.

Remember: You'll want to state your goal (i.e., prove that $\sqrt{2}$ is an irrational number). You'll then want to assume the opposite (i.e., assume that $\sqrt{2}$ is a rational number). And finally, you'll want to draw a conclusion that results in a contradiction (i.e., a statement that contradicts $\sqrt{2}$ is a rational number). Lastly, you'll need to write a formal conclusion.

Contrapositive

Given $p \longrightarrow q$, prove $\neg q \longrightarrow \neg p$.

Prove that for any integer n, if n^2 is even then n is even.

1. Suppose n is odd.
2. Prove that n^2 is odd.

subsection 1.7 - Converse and Contrapositive of an Implication in Dr. Gary's Notes is a good resource if you need further examples.

Induction

Prove that $1 + 2 + ... + n = n(n + 1)/2$.

1. Prove base case(s).
2. State inductive hypothesis.
3. Perform inductive step.

4. Write conclusion.

Construction

Theorems often state that an object exists. Proof by Construction is a way to prove the theorem by constructing the object (i.e., proof by look here's one).

Example: Prove that there is a program that can be used to calculate $A + B$. Solution: Write a program that can calculate $A + B$.

Languages

Very Informally

- alphabets and languages will be key components of this course.
- alphabets are a set of symbols, just as you know from daily life.
  - for example, the alphabet for English is a, b, c, ..., z
  - while the alphabet for German is a, b, c, ..., β, z, ü, ö, ä
- similarly, language are a set of words (strings) made up from a given alphabet.
  - for example, the English language contains words like "hello", "goodbye"
  - the French language contains words like "bonjour", "chat"

Note: Even when given the exact same alphabet, you can create different languages.

subsection 0.2 - Mathematical Notions And Terminology in Introduction to the Theory of Computation is a good resource if you need further examples.

Formally

- an alphabet is a finite set of symbols denoted Σ
- a string is any combination of symbols in Σ
- empty string $= \epsilon$
- set of all strings of an alphabet is Σ^*
- length of strings, $|\epsilon| = 0$, if $w = ab$, then $|w| = 2$
- position, $w = aba$, then $w_2 = b$, and $w_1 = w_3 = a$
- concatenation, $x = hi$ and $y = bye$, then $xy = hibye$

Languages

- a language is a set of strings
- Kleene star, L^*
  - concatenate all substrings of L with L (infinite)

Example $L_1 = \{00, 11\}$

$$L_1^* = \{\epsilon, 00, 11, 0011, 1100, ...\}$$

where $*$ means 0 or more occurrences.

$$L_1^+ = \{00, 11, 0011, 1100, ...\}$$

where $+$ means at least 1 occurrence.

Bibliography

[1] L. Alcock and K. Weber, "Undergraduates' Example Use in Proof Construction: Purposes and Effectiveness," English, *Investigations in mathematics learning*, vol. 3, no. 1, pp. 1–22, 2010, ISSN: 1947-7503.

[2] G. MacGillivray, *Notes for Math 122 Logic and Foundations*, English. [Online]. Available: `https://www.math.uvic.ca/faculty/gmacgill/LFNotes/M122Notes.pdf`.