CSci 423/523 Finite Automata and Theory of Computation

1 Introduction

1.1 Three areas of Theory of Computation

Reading: Sipser 0.1
Theory of Computation is to study the fundamental capabilities and limitations of computers. It contains three areas.

- Automata theory: Models of computation. Seeking a precise but concise definition of a computer.
- Computability theory: What can and cannot a computer do? Computationally unsolvable versus computationally solvable problems. Determining whether a mathematical statement is true or false is unsolvable by any computer algorithms.
- Complexity theory: What can a computer do efficiently? Computationally hard versus computationally easy problems. For example, factorization and sorting. Cryptography needs hard problems such as factorization to ensure security.

1.2 Basic math

Reading: Sipser 0.2
Sets, sequences, functions, relations, graphs, and Boolean logic.
Here is a discussion on strings and languages, which are central concepts in automata theory.

- Alphabet Σ: A finite and nonempty set of symbols. For example, $\Sigma = \{0,1\}$ and $\Sigma = \{a,b,\ldots,z\}$.
- String (or word): A finite sequence of symbols chosen from some alphabet. The empty string ε is a string with zero occurrences of symbols. For string w, the length of the string, $|w|$, is the number of positions for symbols in the string. $|\varepsilon| = 0$. If $w_1 = a_1 \cdots a_n$ and $w_2 = b_1 \cdots b_m$, then the concatenation $w_1 w_2 = a_1 \cdots a_n b_1 \cdots b_m$. If $w = a_1 \cdots a_n$, then the reverse $w^R = a_n \cdots a_1$. A string x is a substring of w if x appears consecutively within w. The empty string ε is a substring of any string.
- Language: A language is a set of strings. Some special languages include $\{\varepsilon\}$ and $\emptyset$. Other examples of languages are $A = \{w | w \text{ has an equal number of 0s and 1s}\}$ anda $B = \{0^n 1^n | n \geq 1\}$. An alphabet Σ can be treated as a language.
- Regular operators: Let A and B be two languages.
  - Union: $A \cup B = \{x | x \in A \text{ or } x \in B\}$.
  - Concatenation: $AB = \{xy | x \in A \text{ and } y \in B\}$.
  - Star: $A^* = \{x_1 x_2 \cdots x_k | \text{ all } k \geq 0 \text{ and each } x_i \in A\}$. A^* is infinite unless $A = \emptyset$ or $A = \{\varepsilon\}$.
- Power of a language: Let A be a language.
  - $A^k = \{x_1 x_2 \cdots x_k | x_i \in A \text{ for } i = 1,2,\cdots,k\}$
  - $A^0 = \{\varepsilon\}$
  - $A^* = A^0 \cup A^1 \cup A^2 \cup \cdots$
  - $A^+ = A^1 \cup A^2 \cup \cdots$
  - $A^* = A^+ \cup \{\varepsilon\}$
- Why languages, not problems:
  - Decision problems: Given an input x, does x satisfy property P, or is $x \in \{y | y \text{ satisfies } P\}$?
  - Membership in a language: Given a language A and a string $w \in \Sigma^*$, is w a member of A, or is $w \in A$?

1.3 Forms of theorems and deductive proofs

Reading: Sipser 0.3
Here are a few common forms of theorems:

- If H (hypothesis) then C (conclusion). Also equivalently, H implies C, H only if C, C if H, whenever H holds, C follows, or $H \rightarrow C$.
- A if and only if B. Also equivalently, A iff B, A is equivalent to B, A exactly when B, $A \equiv B$, or $A \leftrightarrow B$.
- S (a statement without any hypothesis).
- For all x, $S(x)$ holds. Also equivalently, $\forall x S(x)$.
- There is x such that $S(x)$ holds. Also equivalently, $\exists x S(x)$.

A deductive proof consists of a sequence of statements $S_0, S_1, \ldots, S_k$, where S_k is the theorem to prove. Each step, from S_i to S_{i+1}, must follow, by some accepted logical principle, from either the given facts, or some of the previous statements in the proof , or a combination of these.

1.4 Proof techniques

Reading: Sipser 0.4
In addition to the most common direct proofs, there are some other proof techniques:

- By definition: Convert terms in the hypothesis to their definitions.
- By construction: To prove the existence of something, just construct one.
- By counterexample: Used to disprove something seemingly true but actually not true.

  Example: There is no integers a and b such that $a \bmod b = b \bmod a$.
- By contradiction: $H \rightarrow C$ is equivalent to $\neg C \rightarrow \neg H$ or $\neg C \wedge H \rightarrow \neg T$, where T is an axiom, a known truth, or a proven fact.

  Example: The number of primes is infinite.

  Example: $\sqrt{2}$ is irrational.
- By induction: Used to prove a statement $S(n)$ about an integer $n \geq c$. There are several forms of inductions:
  - Simple integer induction: $S(n)$ for $n \geq c$ iff $S(c) \wedge \forall k(S(k) \rightarrow S(k+1))$. (What are the basis step, inductive hypothesis, and inductive step?)

    Example: For $n \geq 1$, $\sum_{i=1}^{n} i^2 = \frac{1}{6}n(n+1)(2n+1)$.
  - General (strong) integer induction: $S(n)$ for $n \geq c$ iff $S(c) \wedge \forall k(S(c) \wedge S(c+1) \wedge \cdots \wedge S(k) \rightarrow S(k+1))$.

    Example: For any $n \geq 8$, n can be written as a sum of 3s and 5s.
  - Structural induction: To prove a property $S(X)$ of a recursively defined structure X, it is suffice to prove that $S(X)$ is true for the basis structures X and $S(Y_1) \wedge S(Y_2) \wedge \cdots \wedge S(Y_k) \rightarrow S(X)$, where X is constructed from $Y_1, Y_2, \ldots, Y_k$.

    Example: Every tree has one more node than it has edges.
  - Mutual induction: Used in the case when a single statement $S_1(n)$ can not be proved by induction but rather $S_1(n)$ together with some other statements $S_2(n), \ldots$ can be proved successfully by induction.

There is one important point about induction. Sometimes it is necessary to consider more than one basis cases. See the example for the general integer induction $(S(8), S(9), S(10))$.