

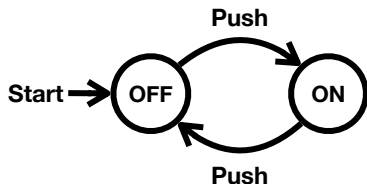
Lecture 1 – Mathematical Preliminaries

COSE215: Theory of Computation

Jihyeok Park



2023 Spring



Theorem

The current state is OFF if and only if the button is pushed even times.

- Is it possible to prove it?

Let's learn **mathematical background and notation**.

1. Mathematical Notations

- Notations in Logics

- Notations in Set Theory

2. Inductive Proofs

- Inductions on Integers

- Structural Inductions

- Mutual Inductions

3. Notations in Languages

- Symbols

- Words

- Languages

1. Mathematical Notations

Notations in Logics

Notations in Set Theory

2. Inductive Proofs

Inductions on Integers

Structural Inductions

Mutual Inductions

3. Notations in Languages

Symbols

Words

Languages

Notation	Description
A, B	arbitrary statements .
$P(x)$	a predicate that involves a variable x .
$A \wedge B$	the conjunction of A and B .
$A \vee B$	the disjunction of A and B .
$\neg A$	the negation of A .
$A \Rightarrow B$	the implication of A and B (i.e., A implies B / if A then B).
$A \Leftrightarrow B$	A if and only if (iff) B (i.e., $A \Rightarrow B \wedge B \Rightarrow A$).
$\forall x \in X. P(x)$	the universal quantifier (i.e., for all x in X , $P(x)$ holds).
$\exists x \in X. P(x)$	the existential quantifier (i.e., there exists x in X such that $P(x)$ holds).

- A **set** is a collection of elements, e.g.,
 - $\mathbb{N} = \{0, 1, 2, \dots\}$
 - $\{x \in \mathbb{N} \mid x \text{ is even}\} = \{0, 2, 4, 6, 8, 10, 12, \dots\}$
 - $\{x \in \mathbb{N} \mid x^2\} = \{0, 1, 4, 9, 16, 25, 36, \dots\}$
- The **empty set** is denoted by \emptyset .
- The **cardinality** of a set X is denoted by $|X|$.
- A **subset** X of a set Y is denoted by $X \subseteq Y$.

$$X \subseteq Y \iff \forall x \in X. x \in Y$$

- A **proper subset** X of a set Y is denoted by $X \subset Y$.

$$X \subset Y \iff X \subseteq Y \wedge X \neq Y$$

- The **union** of sets

$$\begin{aligned} X \cup Y &= \{x \mid x \in X \vee x \in Y\} \\ \bigcup \mathcal{C} &= X_1 \cup X_1 \cup \cdots \cup X_n = \{x \mid \exists X \in \mathcal{C}. x \in X\} \end{aligned}$$

where $\mathcal{C} = \{X_1, X_2, \dots, X_n\}$.

- The **intersection** of sets

$$\begin{aligned} X \cap Y &= \{x \mid x \in X \wedge x \in Y\} \\ \bigcap \mathcal{C} &= X_1 \cap X_1 \cap \cdots \cap X_n = \{x \mid \forall X \in \mathcal{C}. x \in X\} \end{aligned}$$

where $\mathcal{C} = \{X_1, X_2, \dots, X_n\}$.

- The **difference** of sets

$$X \setminus Y = \{x \mid x \in X \wedge x \notin Y\}$$

- The **complement** of a set X is denoted by \overline{X} .

$$\overline{X} = \{x \mid x \in U \wedge x \notin X\}$$

where U is the **universal set**.

- The **power set** of a set X is denoted by 2^X or $\mathcal{P}(X)$.

$$2^X = \mathcal{P}(X) = \{Y \mid Y \subseteq X\}$$

- The **Cartesian product** of sets X and Y is denoted by $X \times Y$.

$$X \times Y = \{(x, y) \mid x \in X \wedge y \in Y\}$$

1. Mathematical Notations

Notations in Logics

Notations in Set Theory

2. Inductive Proofs

Inductions on Integers

Structural Inductions

Mutual Inductions

3. Notations in Languages

Symbols

Words

Languages

Definition (Inductions on Integers)

Let $P(n)$ be a predicate on integers, and if

- **(Basis Case)** $P(k)$ is hold where k is an integer, and
- **(Induction Case)** for all $n \geq k$, $P(n) \Rightarrow P(n+1)$,

then $P(i)$ is hold for all $i \geq k$.

$P(n)$ is called **induction hypothesis**.

Example

Prove that $\forall n \geq 0. \sum_{i=0}^n i = \frac{n(n+1)}{2}$.

Proof)

Example (Exercise)

Prove that $\forall n \geq 0. \sum_{i=0}^n i^2 = \frac{n(n+1)(2n+1)}{6}$.

Proof)

In CS, we often define somethings as **inductively-defined** sets.

Example (Inductive Definition of Trees)

A **tree** is defined as follows:

- **(Basis Case)** A single **node** N is a tree.
- **(Induction Case)** If T_1, \dots, T_n are trees, then a graph defined with a new node N and edges from N to T_1, \dots, T_n is a tree as well.

Example (Inductive Definition of Arithmetic Expressions)

An **arithmetic expression** is defined as follows:

- **(Basis Case)** A **number** or a **variable** is an arithmetic expression.
- **(Induction Case)** If E and F are arithmetic expressions, then so are $E + F$, $E \times F$, and (E) .

Definition (Structural Inductions)

Let $P(x)$ be a predicate on a **inductively-defined set** X , and if

- **(Basis Case)** $P(x)$ is hold for all base cases x , and
- **(Induction Case)** for all $x \in X$,

$$P(x_1) \wedge \cdots \wedge P(x_n) \Rightarrow P(x)$$

where x_1, \cdots, x_n are the **sub-structures** of x .

then $P(x)$ is hold for all $x \in X$.

$P(x_1), \cdots, P(x_n)$ are called **induction hypotheses**.

Example

Prove that for all tree T , the number of nodes in T is equal to the number of edges in T plus one.

Proof)

Example (Exercise)

Prove that for all arithmetic expression E , the number of left parentheses in E is equal to the number of right parentheses in E .

Proof)

Definition (Mutual Inductions)

Let $P(x)$ and $Q(x)$ are predicates on a **inductively-defined** set X , and if

- **(Basis Case)** $P(x)$ and $Q(x)$ are hold for all base cases x , and
- **(Induction Case)** for all $x \in X$,

$$P(x_1) \wedge \cdots \wedge P(x_n) \wedge Q(x_1) \wedge \cdots \wedge Q(x_n) \Rightarrow P(x) \wedge Q(x)$$

where x_1, \cdots, x_n are the **sub-structures** of x .

then $P(x)$ and $Q(x)$ are hold for all $x \in X$.

$P(x_1), \cdots, P(x_n)$ and $Q(x_1), \cdots, Q(x_n)$ are called **induction hypotheses**.

Theorem

The current state is OFF if and only if the button is pushed even times.

Proof)

1. Mathematical Notations

Notations in Logics

Notations in Set Theory

2. Inductive Proofs

Inductions on Integers

Structural Inductions

Mutual Inductions

3. Notations in Languages

Symbols

Words

Languages

We first define a finite and non-empty set of **symbols**.

- $\Sigma = \{0, 1\}$ – binary symbols.
- $\Sigma = \{a, b, \dots, z\}$ – lowercase letters.
- $\Sigma = \{a \mid a \text{ is an ASCII character}\}$ – ASCII characters.

A **word** $w \in \Sigma^*$ is a sequence of symbols:

- $\Sigma = \{0, 1\} - \epsilon, 0, 1, 00, 01, 10010, \dots$
- $\Sigma = \{a, b, \dots, z\} - \epsilon, a, b, abc, \text{hello}, \text{cs}, \text{students}, \dots$

Notations:

Notation	Description
ϵ	the empty word .
$w_1 w_2$	the concatenation of w_1 and w_2 . (w_1 is a prefix of $w_1 w_2$ and w_2 is a suffix of $w_1 w_2$)
w^R	the reverse of w .
$ w $	the length of w .
Σ^k	the set of all words of length k .
Σ^*	the set of all words (the Kleene star). (i.e., $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \dots = \bigcup_{k=0} \Sigma^k$)
Σ^+	the set of all words except ϵ (the Kleene plus). (i.e., $\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \dots = \bigcup_{k=1} \Sigma^k$)

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.
- $L = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.
- $L = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$ – all binary words.

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.
- $L = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$ – all binary words.
- $L = \{0^n 1^n \mid n \geq 0\}$

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.
- $L = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$ – all binary words.
- $L = \{0^n 1^n \mid n \geq 0\}$ – equal number of consecutive zeros and ones.

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.
- $L = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$ – all binary words.
- $L = \{0^n 1^n \mid n \geq 0\}$ – equal number of consecutive zeros and ones.
- $L = \{10, 11, 101, 111, 1011, \dots\}$

A **language** $L \subseteq \Sigma^*$ is a set of words. When $\Sigma = \{0, 1\}$, we can define the following languages:

- $L = \{\epsilon, 0, 1\}$ – the empty word, zero, and one.
- $L = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$ – all binary words.
- $L = \{0^n 1^n \mid n \geq 0\}$ – equal number of consecutive zeros and ones.
- $L = \{10, 11, 101, 111, 1011, \dots\}$ – prime numbers in a binary format.

- The **union**, **intersection**, and **difference** of languages:

$$L_1 \cup L_2 \quad L_1 \cap L_2 \quad L_1 \setminus L_2$$

- The **reverse** of a language:

$$L^R = \{w^R \mid w \in L\}$$

- The **complement** of a language:

$$\overline{L} = \Sigma^* \setminus L$$

- The **concatenation** of languages:

$$L_1 L_2 = \{w_1 w_2 \mid w_1 \in L_1 \wedge w_2 \in L_2\}$$

- The **power** of a language:

$$\begin{aligned} L^0 &= \{\epsilon\} \\ L^n &= L^{n-1}L \quad (n \geq 1) \end{aligned}$$

- The **Kleene star** of a language:

$$L^* = L^0 \cup L^1 \cup L^2 \cup \dots = \bigcup_{n \geq 0} L^n$$

- The **Kleene plus** of a language:

$$L^+ = L^1 \cup L^2 \cup L^3 \cup \dots = \bigcup_{n \geq 1} L^n$$

1. Mathematical Notations

Notations in Logics

Notations in Set Theory

2. Inductive Proofs

Inductions on Integers

Structural Inductions

Mutual Inductions

3. Notations in Languages

Symbols

Words

Languages

- Basic Introduction of Scala

Jihyeok Park

jihyeok_park@korea.ac.kr

<https://plrg.korea.ac.kr>