CE 474 - LOGIC OF COMPUTER SCIENCE

Lecture 1 Notes

Instructor: Dr. Sylvester Akpah
Email: sakpah@umat.edu.gh, Mobile: +233 (0)20-754-1684 / (0)24-422-0361
Demonstrator: Emmanuel Kwesi Tandoh
Email: pg-ektandoh7023@st.umat.edu.gh, Mobile: +233 (0)557511017

1. Learning Objectives

By the end of this lecture, students should be able to:

- Recall basic set operations
- Reinforce proof techniques on set identities

2. Key Definitions

2.1 Set

A **set** is a collection of distinct elements. For example:

$$A = \{1, 2, 3\}$$

denotes a set containing the elements 1, 2, and 3.

2.2 Element

An **element** is an object that belongs to a set. We write:

$$x \in A \quad \text{(x is an element of A)} \qquad \text{and} \qquad x \notin A \quad \text{(x is not in A)}$$

2.3 Subset

For two sets A and B, $A \subseteq B$ means every element of A is also in B. If $A \subseteq B$ and $A \neq B$, then $A \subset B$ is called a **proper subset**.

2.4 Union

The **union** of sets A and B, written $A \cup B$, is defined as:

$$A \cup B = \{x \mid x \in A \text{ or } x \in B\}$$

2.5 Intersection

The **intersection** of sets A and B, written $A \cap B$, is defined as:

$$A \cap B = \{x \mid x \in A \text{ and } x \in B\}$$

2.6 Difference

The **difference** of sets A and B, written $A - B$, is defined as:

$$A - B = \{x \mid x \in A \text{ and } x \notin B\}$$

2.7 Complement

If U is the universal set, the **complement** of A, written A^c or $U - A$, is:

$$A^c = \{x \in U \mid x \notin A\}$$

3. Relations & Functions

3.1 Relation

A **relation** R on a set S is a subset of the Cartesian product $S \times S$, i.e.,

$$R \subseteq S \times S$$

If $(a, b) \in R$, we say "a is related to b".

3.2 Domain

The **domain** of a relation $R \subseteq A \times B$ is:

$$\text{domain}(R) = \{a \in A \mid \exists b \in B \text{ such that } (a, b) \in R\}$$

3.3 Codomain

The **codomain** is the set B from which second elements of ordered pairs in $R \subseteq A \times B$ are drawn.

3.4 Function

A **function** $f : A \to B$ is a relation such that for every $a \in A$, there exists a **unique** $b \in B$ with $(a, b) \in f$.

3.5 Injective (One-to-One)

A function $f : A \to B$ is **injective** if:

$$f(a_1) = f(a_2) \Rightarrow a_1 = a_2$$

3.6 Surjective (Onto)

A function is **surjective** if:

$$\forall b \in B,\ \exists a \in A \text{ such that } f(a) = b$$

3.7 Bijective

A function is **bijective** if it is both injective and surjective. A bijection has an inverse $f^{-1} : B \to A$.

4. Proof Techniques Refresher

4.1 Direct Proof

To prove an implication $P \to Q$, assume P is true and show through logical steps that Q must follow.

4.2 Proof by Contrapositive

To prove $P \to Q$, prove the logically equivalent contrapositive:

$$\neg Q \to \neg P$$

5. Example Proof

Theorem:

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

Proof:
(1) Show $A \cap (B \cup C) \subseteq (A \cap B) \cup (A \cap C)$
Let $x \in A \cap (B \cup C)$. Then:

- $x \in A$ and $x \in B \cup C$
- So either $x \in B$ or $x \in C$
  - If $x \in B$, then $x \in A \cap B$, so $x \in (A \cap B) \cup (A \cap C)$
  - If $x \in C$, then $x \in A \cap C$, so $x \in (A \cap B) \cup (A \cap C)$

(2) Show $(A \cap B) \cup (A \cap C) \subseteq A \cap (B \cup C)$
Let $x \in (A \cap B) \cup (A \cap C)$. Then:

- Either $x \in A \cap B$ or $x \in A \cap C$
  - If $x \in A \cap B$, then $x \in A$ and $x \in B$, so $x \in B \cup C$, hence $x \in A \cap (B \cup C)$
  - If $x \in A \cap C$, then $x \in A$ and $x \in C$, so $x \in B \cup C$, hence $x \in A \cap (B \cup C)$

Hence both directions are proven, and the sets are equal:

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

6. In-Class Activity

Group Activity: Prove one of the following set identities. Each group will present and receive feedback.

1. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
2. $A - (B \cup C) = (A - B) \cap (A - C)$
3. $(A \cap B)^c = A^c \cup B^c$ (De Morgan's Law)

7. Summary & Takeaways

- Set theory is foundational for logic and programming semantics.
- Precise definitions and notation enable clear reasoning.
- Proof techniques add rigor and are essential for building formal logic systems.
- This lecture sets the stage for propositional and predicate logic.

References

- Huth, M., & Ryan, M. (2004). *Logic in Computer Science: Modelling and Reasoning about Systems*. Cambridge University Press.
- Gries, D., & Schneider, F. B. (1993). *A Logical Approach to Discrete Math*. Springer-Verlag.
- Rosen, K. H. (2011). *Discrete Mathematics and Its Applications* (7th ed.). McGraw-Hill.
- Halmos, P. (1960). *Naive Set Theory*. Van Nostrand.