Introduction to Logic

Alice Gao

Lecture 1

Based on work by many people with special thanks to Collin Roberts, Jonathan Buss, Lila Kari and Anna Lubiw.

Outline

Learning goals

By the end of the lecture, you should be able to
(Introduction to Logic)

- Give a one-sentence high-level definition of logic.
- Give examples of applications of logic in computer science.

(Propositions)

- Define a proposition.
- Define an atomic proposition and a compound proposition.

Learning goals

By the end of the lecture, you should be able to
(Translations)

- Determine if an English sentence is a proposition.
- Determine if an English sentence is an atomic proposition.
- For an English sentence with no logical ambiguity, translate the sentence into a propositional formula.
- For an English sentence with logical ambiguity, translate the sentence into multiple propositional formulas and show that the propositional formulas are not logically equivalent using a truth table.

What is logic?

What comes to your mind when you hear the word “LOGIC”?

What is logic?

Logic is the science of reasoning, inference, and deduction.

The word “logic” comes from the Greek word *Logykos*, which means “pertaining to reasoning.”

Why should you study logic?

- Logic is fun!
- Logic improves one's ability to think analytically and to communicate precisely.
- Logic has many applications in Computer Science.

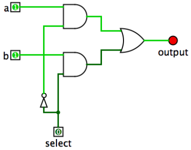
Logic and Computer Science

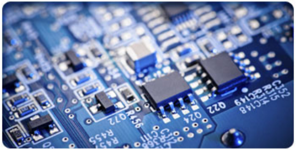
Name an application of logic in Computer Science.

Logic and computer science

Circuit Design

- Digital circuits are the basic building blocks of an electronic computer.





- CS 251: Computer Organization and Design
  CS 350: Operating Systems

Logic and computer science

Databases

- Structural Query Language (SQL) $\approx$ first-order logic
- Efficient query evaluation based on relational algebra
- Scale to large databases with parallel processors
- CS 348: Introduction to Database Management
  CS 448: Database Systems Implementation

Logic and computer science

Type Theory in Programming Language

- Propositions in logic $\leftrightarrow$ types in a programming language
- Proofs of a proposition $\leftrightarrow$ programs with the type
- Simplifications of proofs $\leftrightarrow$ evaluations of the programs

- CS 241: The compiler course
  CS 442: Principles of Programming Languages
  CS 444: Compiler Construction

Logic and computer science

Artificial Intelligence

- 19 billion FCC spectrum auction: Buy airwaves from television broadcasters and sell them to mobile phone carriers.
- IBM Watson won the Jeopardy Man vs. Machine Challenge
- CS 486: Artificial Intelligence
  CS 485: Machine Learning

Logic and computer science

Formal verification

- Prove that a program is bug free. Bugs can be costly and dangerous in real life.
- Intel's Pentium FDIV bug (1994) cost them half a billion dollars.
- Cancer patients died due to severe overdoses of radiation.
- CS 360: Theory of Computing (Finite Automata)

Logic and computer science

Algorithms and Theory of Computing

- How much time and memory space do we need to solve a problem?
- Are there problems that cannot be solved by algorithms?
- CS 341: Algorithm Design and Analysis
  CS 360: Introduction to the Theory of Computing

An example of logical deduction

Let's look at two clips of the TV series Sherlock.

An example of logical deduction

Let's look at two clips of the TV series Sherlock.

Argument 1:

- Watson's phone is expensive.
- Watson is looking for a person to share a flat with.
- Therefore, Watson's phone is a gift from someone else.

An example of logical deduction

Let's look at two clips of the TV series Sherlock.

Argument 1:

- Watson's phone is expensive.
- Watson is looking for a person to share a flat with.
- Therefore, Watson's phone is a gift from someone else.

Argument 2:

- Watson's phone is from a person named Harry Watson.
- The phone is expensive and a young person's gadget.
- Therefore, Watson's phone is a gift from his brother.

Propositions

A proposition is a declarative sentence that is either true or false.

CQ on Proposition

Examples of propositions

- The sum of 3 and 5 is 8.
- The sum of 3 and 5 is 35.
- Goldbach's conjecture: Every even number greater than 2 is the sum of two prime numbers.

Examples of non-propositions

- Question: Where shall we go to eat?
- Command: Please pass the salt.
- Sentence fragment: The dogs in the park
- Non-sensical: Green ideas sleep furiously.
- Paradox: This sentence is false.

Atomic and compound propositions

- An atomic proposition cannot be broken down into smaller propositions.
- A compound proposition is not atomic.

Propositional logic symbols

Three types of symbols in propositional logic:

- Propositional variables: p, q, r, p_1, etc.
- Connectives: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$.
- Punctuation: (and).

An atomic proposition = a propositional variable
A compound proposition = a formula with at least one connective and one set of brackets.

The meanings of the connectives

p	$(\neg p)$
T	F
F	T

p	q	$(p \wedge q)$	$(p \vee q)$	$(p \to q)$	$(p \leftrightarrow q)$
T	T	T	T	T	T
T	F	F	T	F	F
F	T	F	T	T	F
F	F	F	F	T	T

CQ on Atomic proposition

Well-formed propositional formulas

Let $\mathcal{P}$ be a set of propositional variables. We define the set of well-formed formulas over $\mathcal{P}$ inductively as follows.

1. A propositional variable in $\mathcal{P}$ is well-formed.
2. If α is well-formed, then $(\neg\alpha)$ is well-formed.
3. If α and β are well-formed, then each of $(\alpha \wedge \beta), (\alpha \vee \beta), (\alpha \to \beta), (\alpha \leftrightarrow \beta)$ is well-formed.

CQ on First symbol in a well-formed formula

English sentences with no logical ambiguity

Translate the following sentences to propositional logic formulas. If you came up with multiple translations, prove that they are logically equivalent using a truth table.

1. If I ace CS 245 then I can get a job at Google; otherwise I will apply for the Geek Squad.
2. Nadhi eats a fruit only if the fruit is an apple.
3. Soo-Jin will eat an apple or an orange but not both.
4. If it is sunny tomorrow, then I will play golf, provided that I am relaxed.

English sentences with logical ambiguity

Give multiple translations of the following sentences into propositional logic. Prove that the translations are not logically equivalent using a truth table.

1. Sidney will carry an umbrella unless it is sunny.
2. Pigs can fly and the grass is red or the sky is blue.

Translations: A reference page

- $\neg p$: p does not hold; p is false; it is not the case that p
- $p \wedge q$: p but q; not only p but q; p while q; p despite q; p yet q; p although q
- $p \vee q$: p or q or both; p and/or q;
- $p \rightarrow q$: p implies q; q if p; p only if q; q when p; p is sufficient for q; q is necessary for p
- $p \leftrightarrow q$: p is equivalent to q; p exactly if q; p is necessary and sufficient for q

Revisiting the learning goals

By the end of the lecture, you should be able to
(Introduction to Logic)

- Give a one-sentence high-level definition of logic.
- Give examples of applications of logic in computer science.

(Propositions)

- Define a proposition.
- Define an atomic proposition and a compound proposition.

Revisiting the learning goals

By the end of the lecture, you should be able to
(Translations)

- Determine if an English sentence is a proposition.
- Determine if an English sentence is an atomic proposition.
- For an English sentence with no logical ambiguity, translate the sentence into a propositional formula.
- For an English sentence with logical ambiguity, translate the sentence into multiple propositional formulas and show that the propositional formulas are not logically equivalent using a truth table.