Outline

- Syllabus & Introductions
- What is logic?
- Logic propositions
Syllabus

https://student.cs.uwaterloo.ca/~cs245/spring2024/
Who am I?

My name is Alena Gusakov, I grew up in Delaware (USA) and moved here for grad school.

Research: Mathematics formalization, theorem provers, matroid theory, algebraic graph theory

My education history:

▶ Masters, Combinatorics & Optimization, University of Waterloo

▶ Undergraduate, Mathematics & Computer Science, University of Delaware

Hobbies: Salsa dancing, figure skating, Dungeons & Dragons
Logic Puzzle

You have 2 minutes to work with the people around you and try to solve this logic puzzle:

At the Pet Show recently I noticed that all except two of the entries were cats, all except two were dogs, and all except two were fish.

How many of each animal were at the Pet Show?
What is logic?

What comes to 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.”
Different kinds of logic

- **Propositional logic**
  - e.g. \((p \lor q) \rightarrow (\neg r)\)

- **First order Logic**
  - e.g. \(\forall n \in \mathbb{N}, \text{ } n \text{ is even or odd}\)

- **Higher order Logic**
  - quantifiers over sets, e.g.
    - “every finite set of natural numbers has a largest element”

- **Constructive Logic**
  - Law of Excluded Middle (LEM)

- **Modal Logic**
  - temporally, epistemically

- and many more...
Why study logic?

- Humans are inherently kind of bad at logic.
- Logic improves one’s ability to think analytically and communicate precisely.
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
Artificial Intelligence

- AI is insanely widespread: search engines, content recommendation systems, ChatGPT and image generators, automation in various industries, etc

“A lot of cutting edge AI has filtered into general applications, often without being called AI because once something becomes useful enough and common enough it’s not labeled AI anymore.”

- CS 486: Artificial Intelligence
- CS 485: Machine Learning
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
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: Compilers
- CS 442: Principles of Programming Languages
- CS 444: Compiler Construction
- CS 448: Database Systems Implementation
A proposition is a declarative sentence that is either true or false (but not both).
Examples & Non-examples
Examples & Non-examples

Examples

▶ Serik is cute
▶ The sum of 3 and 5 is 8

Non-examples

▶ Question: What time is it?
▶ Command: Please pass the salt.
▶ Sentence fragment: The dogs in the park
▶ Non-sensical: Green ideas sleep furiously.
▶ Paradox: This sentence is false.
Atomic & Compound Propositions

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

We denote the formal language of propositional logic as $\mathcal{L}^p$. There are three types of symbols in propositional logic:

- **Proposition symbols**: $p, q, p_1$, etc.
- **Connectives**: $\neg, \land, \lor, \rightarrow, \leftrightarrow$.
- **Punctuation**: ( and ).
The meanings of the connectives

<table>
<thead>
<tr>
<th>$p$</th>
<th>$\neg p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$(p \land q)$</th>
<th>$(p \lor q)$</th>
<th>$(p \rightarrow q)$</th>
<th>$(p \leftrightarrow q)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>