Class starts after this song

_Noisestorm - Crab Rave (2018) requested by Alex Chao (TA-of-CM1)_

This is Leo (unfortunately not mine). Occasionally, I think about replacing my roommate with a cat, but unfortunately cats can't pay rent. I enjoy cooking food for others and climbing rocks.
Featured: Violet Pang

- I like surfing and skiing. In my leisure time at Durham, I enjoy going to gyms, playing switch games, and cooking. I have a ragdoll cat named Taro.
Logistic Bulletin Board

• Recitation section 08D class change:
  • was Gross 105
  • now moved to LSRC A247

Today’s class will also end a bit early
Come to the front after class if (1) you’re on the waitlist INCLUSIVE-or (2) you’ve emailed us about recitation sections
Before we start

• You don’t necessarily need to complete ALL the questions in the textbook before attempting the Canvas quiz (or at all)
  • Read the concepts
  • Use Canvas quiz to test your understanding of concepts
  • Depending on what you find yourself struggle on, read again or attempt the textbook questions

• It is okay to be a bit confused between prepare and class
Before we start

- Most of you got to *about* 80% in three tries or lower
  - Including $\frac{7}{9} = 77.78\%$ for this first quiz

- Is the 1-hour cooldown period too long? (Changed to 15 mins)
CS230 Spring 2024
Module 01: Logic
Why logic?

• You may have completed the required reading and the Canvas quiz, and likely already know the basics about logic

• But one thing the textbook does NOT say (explicitly) is **WHY should we learn logic?**
Rationale 1

• Logic is the foundation of all mathematical proofs, so if you want to learn proofs, you should learn logic (first)
• You want to learn proofs in CS230

• Therefore, you should learn logic
Rationale 2

• English is hard, because it is often imprecise

• *Learning logic helps us become vigilant in everyday talking and reasoning*
Peer Discussion!

• Turn to your neighbor
• Introduce yourself:
  • Name, year, major, etc.
  • What is another interesting course you’re taking?
• Then discuss the question in the QR code
  • Remember everyone should have an opinion
Natural language is often imprecise

- Waiter could’ve said “you can have cake or ice cream, *but not both*”
- Waiter could’ve also said “you can have cake *exclusive-or* ice cream”…
- What about the traffic sign?
Propositions/predicates

• Propositions have a truth value *per se*

• Strictly speaking, predicates *don’t* have a truth value … until we assign value(s) to the variable(s)
  • *But it’s often vague whether we’ve assigned value(s)*
  • *As a result, even the textbook sometimes calls predicates propositions*

• Fully quantified predicates like $\exists x \ [\text{Logic is the x-th module in CS230}]$ become *propositions* because they have truth values
Food for thought

• Is *CS230 is enjoyable* a proposition?

• Is *The Duke Blue Devils will win the national NCAA basketball championship in 2024* a proposition?

• Is $x^2 \geq 0$ a tautology?

---

Lesson: always declare the domain of variables (unless it’s REALLY REALLY clear from context)

Example 2.44. Assume that $x$ is a real number.
PI: Implication

- You all did okay on this question
- Let’s do a trickier one in peer instruction
What does $p \rightarrow q$ actually mean?

- It really means $\neg p \lor q$ and nothing else
- Even if it evaluates to True, it says *nothing* about whether there is any cause-and-effect relation between $p$ and $q$
- $F \rightarrow F$ and $F \rightarrow T$ are both True (in fact $F \rightarrow q$ is always True!)
- Similarly $(x \lor y) \rightarrow (x \land y)$ can be True if $(x \lor y)$ evaluates to False
Chains-of-equivalences

Example 2.52. Prove the idempotent law $p \lor p = p$ by using the other equivalences.

Proof: It is easier to prove backwards ($p = p \lor p$). We have

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$p$</td>
<td>$p \lor F$</td>
<td>(by identity)</td>
</tr>
<tr>
<td>$=$</td>
<td>$p \lor (p \land \neg p)$</td>
<td>(by negation)</td>
</tr>
<tr>
<td>$=$</td>
<td>$(p \lor p) \land (p \lor \neg p)$</td>
<td>(by distribution)</td>
</tr>
<tr>
<td>$=$</td>
<td>$(p \lor p) \land T$</td>
<td>(by negation)</td>
</tr>
<tr>
<td>$=$</td>
<td>$p \lor p$</td>
<td>(by identity)</td>
</tr>
</tbody>
</table>

Thus, $p \lor p = p$.

It's a Not Yet without the equal signs. If these are all omitted, it’s not a proof, just many logical statements (that happen to be equivalent) written together.

It's a Satisfactory without these elaborations/steps. Omitting these doesn’t affect technical correctness, but it makes the proof harder to understand.
Modus Ponens

- Logic is the foundation of all mathematical proofs, so if you want to learn proofs, you should learn logic (first)
- You want to learn proofs in CS230

\[ p \rightarrow q \]

\[ p \]

\[ \therefore q \]

- Therefore, you should learn logic
Modus Ponens

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$T$</td>
<td>$F$</td>
<td>$F$</td>
</tr>
<tr>
<td>$F$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
</tbody>
</table>

$p \rightarrow q$

$p$

$\therefore q$
Modus Ponens

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$T$</td>
<td>$F$</td>
<td>$F$</td>
</tr>
<tr>
<td>$F$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
</tbody>
</table>

$p \rightarrow q$  

$\therefore q$
**Modus Tollens**

$p \rightarrow q$  
$\neg q$  
$\therefore \neg p$

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
<th>$\neg q$</th>
<th>$\neg p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>$T$</td>
<td>$T$</td>
<td>$F$</td>
<td>$F$</td>
</tr>
<tr>
<td>$T$</td>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
<td>$F$</td>
</tr>
<tr>
<td>$F$</td>
<td>$T$</td>
<td>$T$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
<tr>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
</tbody>
</table>
Modus Tollens

• If the sky is blue, then it is raining
• It is not raining
• Therefore, the sky is not blue

\[ p \rightarrow q \]
\[ \neg q \]
\[ \therefore \neg p \]
“Scientific method”?

• If my hypothesis is true, I will see this result.
• I run an experiment and get this result.
• Therefore, my hypothesis is true.

\[ p \rightarrow q \]

\[
\begin{array}{c}
q \\
\hline
\vdots \ p
\end{array}
\]
"Scientific method"?

<table>
<thead>
<tr>
<th>$p$</th>
<th>$q$</th>
<th>$p \rightarrow q$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$T$</td>
<td>$F$</td>
<td>$F$</td>
</tr>
<tr>
<td>$F$</td>
<td>$T$</td>
<td>$T$</td>
</tr>
<tr>
<td>$F$</td>
<td>$F$</td>
<td>$T$</td>
</tr>
</tbody>
</table>

$p \rightarrow q$

$q$

$\therefore p$
Late Policies

• If:
  • we don’t want to measure your time management skills and
  • deadlines for Gradescope assignments are for logistic reasons,

• Then:
  • no one should be punished if they don’t incur any logistic cost/inconvenience to the staff
Class starts after this song

I am a senior studying computer science with a concentration in AI/ML with minors in math and gender, sexuality, and feminist studies. I am a member of the rowing team at Duke. In my free time, I like to bake and crochet.

*Bon Jovi - Livin’ on a Prayer (1986)*
requested by Paige Knudsen (TA-of-CM1, Recitation 04D)
1. Beginner-friendly!
   Training Programs in ML and Software Engineering

2. Opportunities for all interests!
   Client Projects, Startup Fellowships, Internal Products
Duke Applied Machine Learning
Learn more at our Info Session!

http://tinyurl.com/damlsp24info

Join our mailing list here

Use the URL above
Boolean algebra/logic = Propositional logic?

- If you have taken hardware/systems courses, you may have already seen *Boolean algebra/logic*
- You may even feel Boolean algebra/logic is awfully similar to propositional logic
PI: DNF/CNF

• DNF: **disjunctions (or's)** of conjunctive clauses (and’s)
  - \( q \land \neg r \)
  - \( (d \land n \land f) \lor (d \land n \land \neg f) \lor (\neg d \land n \land f) \lor (\neg d \land \neg n \land f) \)

• CNF: **conjunctions (and’s)** of disjunctive clauses (or’s)
  - \( q \land \neg r \)
  - \( (c \lor n \lor f) \land (c \lor n \lor \neg f) \land (\neg c \lor n \lor f) \land (\neg c \lor \neg n \lor f) \) Which one is satisfiable?
Applications

• DNF: \textbf{disjunctions (or’s) of} conjunctive clauses (and’s)
  - \( q \land \neg r \)
  - \( (d \land n \land f) \lor (d \land n \land \neg f) \lor (\neg d \land n \land f) \lor (\neg d \land \neg n \land f) \)

• A lot easier to check for \textit{satisfiability} (just check every clause)
• Similar to the idea of \textit{proof-by-cases}
Applications

• CNF: **conjunctions (and’s)** of disjunctive clauses (or’s)
  - \( q \land \neg r \)
  - \((c \lor n \lor f) \lor (c \lor n \land \neg f) \lor (\neg c \lor n \lor f) \lor (\neg c \land \neg n \lor f)\)

• The problem of whether a CNF* is **satisfiable**
  is one of the 21 fundamental problems in theory-of-computing
  - *CNF with exactly 3 literals per clause
Every student has a dean (true)

There’s a dean who is every student’s dean (false)

Takeaway: Order of quantifiers matters. A LOT.
Recap: common sets of numbers

• \( \mathbb{Z} \): the set of all integers
  • \( \mathbb{Z}^+ \): the set of all positive integers
  • \( \mathbb{Z}^- \): the set of all negative integers

• \( \mathbb{R} \): the set of all real numbers (\( \mathbb{R}^+ \) and \( \mathbb{R}^- \) follow similarly)

• \( \mathbb{N} \): the set of natural numbers
  • \( 0 \notin \mathbb{N} \) or \( 0 \in \mathbb{N} \)? No consensus

• \( \mathbb{Q} \): the set of rational numbers
  • Rational numbers can be represented in the form of \( \frac{a}{b} \), where \( a, b \in \mathbb{Z}, b > 0 \)
Recap: division

- In the following, $a, b, c \in \mathbb{Z}$
- $a \mid b$ (a divides b): there exists $k \in \mathbb{Z}$ such that $ak = b$
  - Also said as $b$ is divisible by $a$, $b$ is a multiple of $a$, $a$ is a factor of $b$
  - \( \forall x \in \mathbb{Z} [1 \mid x] \)
- $a \nmid b$ : there exists no such $k \in \mathbb{Z}$ (a does not divide b)
  - \( \forall x \in \mathbb{Z}^+ [0 \nmid x] \)
- $c$ is a common divisor of $a, b$ if $c \mid a$ and $c \mid b$
- $c$ is the greatest common divisor (GCD) of $a, b$ if there exists no other common divisor of $a, b$ larger than $c$
Recap: primes

• $p \in \mathbb{Z}^+$ is a prime number (or $p$ is prime) if $p$ is divisible by exactly two natural numbers, 1 and $p$

• $p \in \mathbb{Z}^+$ is a composite number (or $p$ is composite) if $p$ is divisible by more than two natural numbers, i.e., there exists $1 < q < p$ such that $q \mid p$

• $p, q$ are coprime if the only natural number that divides both is 1

• 1 is neither a prime number nor a composite number
Recap: prime factorization

• Every positive integer greater than 1 can be written as multiplications of (not necessarily distinct) prime numbers.
  • \(16 = 2 \times 2 \times 2 \times 2\)
  • \(525 = 3 \times 5 \times 5 \times 7 = 5 \times 7 \times 3 \times 5\)

• (Uniqueness) There is only one such way to do this for every positive integer such that the prime numbers are *sorted*. 
Let the domain of $x$ be integers. Let $P(x)$ represent $x \leq 0$, i.e., $x$ is negative; let $Q(x)$ represent $x^2 \leq 0$, i.e., $x^2$ is negative. Accordingly, $Q(x)$ is always false regardless of $x$, while $P(x)$ is a contingency (so $\forall x P(x)$ is false).

This is false because $P(x) \rightarrow Q(x)$ is false when $x$ is negative. We may represent $P(x) \rightarrow Q(x)$ by $R(x)$; then $R(x)$ is false when $x$ is negative, so $\forall x R(x)$ is false.

This is true simply because $\forall x P(x)$ is false. Anything is implied by a false proposition (think back to the “vacuous promise” argument. It doesn’t even matter whether $\forall x Q(x)$ is true or false (it is false).
Higher order logic systems

• Propositional logic (0-order logic) is simple yet very restrictive
• Predicate logic (1-order logic) is richer and still simple enough, so it is a good “playground” for us
• There are still many things that cannot be represented by predicate logic statements, e.g.,
  “every non-empty set of positive integers contains a smallest element”
Memorization?

- This table will be part of our “official reference sheet” for exam 1

- Memorizing things in this course almost never helps in the long term

<table>
<thead>
<tr>
<th>Name</th>
<th>Equivalence</th>
</tr>
</thead>
<tbody>
<tr>
<td>commutativity</td>
<td>$p \lor q = q \lor p$</td>
</tr>
<tr>
<td></td>
<td>$p \land q = q \land p$</td>
</tr>
<tr>
<td>associativity</td>
<td>$p \lor (q \lor r) = (p \lor q) \lor r$</td>
</tr>
<tr>
<td></td>
<td>$p \land (q \land r) = (p \land q) \land r$</td>
</tr>
<tr>
<td>distributive</td>
<td>$p \land (q \lor r) = (p \land q) \lor (p \land r)$</td>
</tr>
<tr>
<td></td>
<td>$p \lor (q \land r) = (p \lor q) \land (p \lor r)$</td>
</tr>
<tr>
<td>identity</td>
<td>$p \lor \neg p = T$</td>
</tr>
<tr>
<td></td>
<td>$p \land \neg p = F$</td>
</tr>
<tr>
<td>negation</td>
<td>$p \lor \neg p = T$</td>
</tr>
<tr>
<td></td>
<td>$p \land \neg p = F$</td>
</tr>
<tr>
<td>domination</td>
<td>$p \lor T = T$</td>
</tr>
<tr>
<td></td>
<td>$p \land F = F$</td>
</tr>
<tr>
<td>idempotent</td>
<td>$p \lor p = p$</td>
</tr>
<tr>
<td></td>
<td>$p \land p = p$</td>
</tr>
<tr>
<td>double negation</td>
<td>$\neg(\neg p) = p$</td>
</tr>
<tr>
<td>DeMorgan's</td>
<td>$\neg(p \lor q) = \neg p \land \neg q$</td>
</tr>
<tr>
<td></td>
<td>$\neg(p \land q) = \neg p \lor \neg q$</td>
</tr>
<tr>
<td>absorption</td>
<td>$p \lor (p \land q) = p$</td>
</tr>
<tr>
<td></td>
<td>$p \land (p \lor q) = p$</td>
</tr>
</tbody>
</table>

Table 2.3: Common Logical Equivalences