CS 70 Su23: Lecture 1

Propositional logic

Berkeley
UNIVERSITY OF CALIFORNIA

Propositions

- A proposition is a statement that is either **true** or **false**
- Examples of propositions
  - 2 + 2 = 4
  - 2 + 2 = 5
  - Victor took BART to get to class
- Examples that are **not** propositions
  - 2 + 2
  - x + 2 = 5
  - This statement is false
  - Victor is the best thing that has ever happened to CS 70

P, Q, R, and friends

- We bind propositions to variables to make them easier to work with
- Some common operations on propositions:
    - **Conjunction:** $P \wedge Q$
        - "P and Q"
        - True only when both P and Q are true
    - **Disjunction:** $P \vee Q$
        - "P or Q"
        - True when at least one of P and Q is true
    - **Negation:** $\neg P$
        - "not P"
        - True when P is false
    - These are called *propositional forms* (the result is itself a proposition)
- If this sounds similar to booleans, that's no coincidence
    - They're not exactly the same, but we won't go into details in this class

Truth tables

Enumeration of all possible values for propositions

P	Q	P ∧ Q	P ∨ Q	¬P	¬P ∧ ¬Q	¬P ∨ Q
T	T	T	T	F	F	T
T	F	F	T	F	F	F
F	T	F	T	T	F	T
F	F	F	F	T	T	T

A quick detour

Consider the following code snippet:

```python
def some_func(x):
    if x == 5:
        return 3
    ...
```

Take note of the ellipses! More can happen afterwards

Now consider the following input/output:

```
>>> some_func(5)
3

>>> some_func(5)
12

>>> some_func(1)
3

>>> some_func(1)
14
```

Which of these would convince us there was an error executing this code?

A quick detour

Let's write this test up:

```
def test(x):
    """returns True if we pass, False if not"""
    if x == 5 and some_func(x) != 3:
        return False
    return True
```

Simplifying:

```
def test(x):
    return not (x == 5 and some_func(x) != 3)
```

A quick detour

Now let's draw out the truth table for this:

- Let **P** = x == 5
- **Q** = *some_func(x)* == 3
- **R** = test(x)

P	Q	R	¬P ⋁ Q
T	T	T	T
T	F	F	F
F	T	T	T
F	F	T	T

Implication

This logical construction is so useful that it has its own name (and notation):

- **Implication: $P \Rightarrow Q$** (read as "P implies Q" or "if P, then Q")
  - $P \Rightarrow Q$ itself is either **true** or **false**, meaning that **it is also a proposition!**

Some terminology:

- P is the *antecedent* (or hypothesis/premise)
- Q is the *consequent* (or conclusion/outcome)
- Two propositional forms are **logically equivalent** if their truth tables are identical (for example, $P \Rightarrow Q \equiv \neg P \lor Q$)

Implication

Some example implications:

- If you stand in the rain, you will get wet
- If your total score is around 50%, you will receive some flavor of B
- If pigs can fly, cats can talk
    - implications for which P is always false are called “vacuously true”
- If you ask Victor if he likes boba, he will say yes

Implication is the basis for much of mathematics

Implication

We can define some other terms from an implication:

- The **converse** of $P \Rightarrow Q$ is $\mathbf{Q \Rightarrow P}$
- The **contrapositive** of $P \Rightarrow Q$ is $\mathbf{\neg Q \Rightarrow \neg P}$
  - sanity check: implication and contrapositive are logically equivalent

If an implication and its converse are both true, we write this as $\mathbf{P \Leftrightarrow Q}$

- read as **"P if and only if Q"** (iff)

Question: $(P \Rightarrow Q) \Rightarrow (Q \Rightarrow P)$? This is true iff $P \Leftrightarrow Q$ (does your head hurt yet?)

Quantifiers

Let's revisit our "not a proposition" from earlier: **x + 2 = 5**

- How do we turn this into a proposition?
  - Let P(x) = "x + 2 = 5" (this is still not a proposition)
  - We need to **quantify** it over some "universe"
- Some examples:
  - P(3): For x = 3, x + 2 = 5
  - P(1): For x = 1, x + 2 = 5
  - P(1) $\lor$ P(3): For x = 1 or x = 3, x + 2 = 5
  - For all integers x, x + 2 = 5 (universal quantifier)
  - There exists an integer x such that x + 2 = 5 (existential quantifier)

Quantifiers

Translating from english to math

- For every integer x, x + 2 = 5
  - $\forall x \in \mathbb{Z}, x + 2 = 5$
- There exists some integer x such that x + 2 = 5
  - $\exists x \in \mathbb{Z}, x + 2 = 5$
- For any integer x, there is some larger integer y
  - $\forall x \in \mathbb{Z}, \exists y \in \mathbb{Z}$ st $y > x$
  - $(\forall x \in \mathbb{Z})(\exists y \in \mathbb{Z})(y > x)$
- There is a largest integer
  - $\exists y \in \mathbb{Z}$ st $\forall x \in \mathbb{Z}, y > x$
  - $(\exists y \in \mathbb{Z})(\forall x \in \mathbb{Z})(y > x)$

DeMorgan's Laws

Naming something after yourself is the old way of commenting "first!"

- $\neg(P \land Q) \equiv \neg P \lor \neg Q$
- $\neg(P \lor Q) \equiv \neg P \land \neg Q$

A similar property holds for quantifiers:

- $\neg(\forall x P(x)) \equiv \exists x \neg P(x)$
- $\neg(\exists x P(x)) \equiv \forall x \neg P(x)$

Example from earlier: *not (x == 5 and some_func(x) != 3)*

- ¬(x == 5 ∧ some_func(x) != 3) ≡ x != 5 ∨ some_func(x) == 3
- Or, with P and Q, $\neg(P \land \neg Q) \equiv \neg P \lor Q$

Berkeley
UNIVERSITY OF CALIFORNIA

Next class: proofs

Let's put all of this together so we can start **proving** things are true or false!