Propositional Logic

"A formal system"

→ ? why we don't use it

→ ? some informality

Recall: Propositional logic is the logic of connectives, the most common of which are:
- "$\neg$" = not   - "$\rightarrow$" = if... then...
- "$\wedge$" = and   - "$\leftrightarrow$" = if and only if
- "$\vee$" = or

A	$\neg$A
T	F
F	T

"not A"

A	B	A$\wedge$B
T	T	T
T	F	F
F	T	F
F	F	F

"A & B"

A	B	A$\vee$B
T	T	T
T	F	T
F	T	T
F	F	F

"A or B"

A	B	A$\rightarrow$B
T	T	T
T	F	F
F	T	T
F	F	T

"if A then B"

A	B	A$\leftrightarrow$B
T	T	T
T	F	F
F	T	F
F	F	T

"A if and only if B"

Sidenote
- Book called "The Logic Book" by Merrie Bergman et al.
  → clean version of natural deductive logic
  → lots of rules for making deductions

Logic Systems:
A formal system to be stripped down to be almost as simple as possible:
1. what do we need?
  - atomic formulas (can't be broken down using connectives)
    $A_0, A_1, A_2, ..., A_k$
    → informal shortcut: A, B, C, ...
  - connectives:
    $\neg, \rightarrow$
    → $\wedge, \vee, \leftrightarrow$ are abbreviations for formulas using $\neg$ & $\rightarrow$
    → ex: we can simulate "$\vee$" using $\neg$ & $\rightarrow$

A	B	A$\vee$B	($\neg$A)$\rightarrow$($\neg$B)	($\neg$A)$\rightarrow$B
T	T	T	T	T
T	F	T	T	T
F	T	T	F	T
F	F	F	T	F

not quite there

  - grouping symbols (how to identify where things begin & end)
    "(,)"

2. How do we put these together to make "formulas"?
  a) every atomic formula, A_k, is a formula
  b) if α is a formula, then ($\neg\alpha$) is a formula
  c) if α & β are formulas, then ($\alpha\rightarrow\beta$) is a formula
  d) only finite sequences of symbols of the language in finitely many steps using rules 1 to 3 are formulas
  → official formula examples:

  A_{13} $((\neg A_{41})\rightarrow(\neg A_{42}))$ $(A_1\rightarrow(A_2\rightarrow(A_3\rightarrow(\neg A_4))))$

  → unofficially:
    A→B→C   gets grouped to the right to mean (A→(B→C))
    $\neg$A→B $\neg$ gets grouped first with as little as possible ($\neg$A)→B

A	B	((¬A) → (¬B)) → (((¬A) → B) → A)
T	T	F T F T F T T T
T	F	F T T T F T F T
F	T	T F F T T T T F
F	F	T T T T T F F T