③ multiple instances of a connective
band @ left first

e.g. $p \Rightarrow q \Rightarrow r$ means $(p \Rightarrow q) \Rightarrow r$

Semantics of PL

- having formalized how to write fmlas (syntax), want to define truth assignments and truth values of fmlas (semantics)

Def a truth assignment (or valuation) is a map

$$v : \{p_1, p_2, \dots\} \to \{0, 1\}$$

- think: 0 means false, 1 means true.
- so: v gives every var. p_i a truth value.

- we can extend a truth assignment v to an assignment v^* defined on all wffs:

Def. Sps v is a truth assignment. We define a truth assignment v^* on the class of wffs recursively:

(i) $v^*(p) = v(p)$

(ii) $v^*(\neg A) = 1 - v^*(A)$

(iii) Rules for binary connectives:

$$v^*((A \wedge B)) = 1 \quad \text{iff } v^*(A) = v^*(B) = 1$$
$$= 0 \quad \text{o/w}$$
$$v^*((A \vee B)) = 1 \quad \text{iff } v^*(A) = 1 \text{ or } v^*(B) = 1$$
$$= 0 \quad \text{if } v^*(A) = v^*(B) = 0$$
$$v^*((A \Rightarrow B)) = 0 \quad \text{iff } v^*(A) = 1 \text{ and } v^*(B) = 0$$
$$= 1 \quad \text{o/w}$$
$$v^*((A \Leftrightarrow B)) = 1 \quad \text{iff } v^*(A) = v^*(B)$$
$$= 0 \quad \text{o/w}$$

- can collect the rules for connectives in a <u>truth</u> <u>table</u>

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \Rightarrow B$	$A \Leftrightarrow B$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

- on one hand, can view these merely as formal rules that recursively define $v^*(A)$ for a wff A
- on the other, the def'n corresponds to our intuition for the words "and" "or" "implies" "iff"

... well, except maybe "implies"

- $(A \Rightarrow B)$ is F (i.e. $v^*((A \Rightarrow B)) = 0$) iff A is T and B is F
- in particular if A is F then $(A \Rightarrow B)$ is automatically (say: "vacuously") true
- def'n of $v^*((A \Rightarrow B))$ becomes more intuitive once we allow variables and quantifiers in our wffs (first-order logic)

  ∴ for now, we'll accept it and see that it has intuitive consequences.

<u>Notation</u>: though formally v^* extends v, we often identify them and write $v(A)$ for $v^*(A)$.

<u>Def</u>. The <u>support</u> of a wff A (write: supp(A)) is the set of variables appearing in A

- e.g. if $A = ((p \Rightarrow q) \wedge (q \Leftrightarrow \neg r))$ then $\mathrm{supp}(A) = \{p, q, r\}$

<u>Observe</u>: $v(A)$ depends only on $v(p)$ for $p \in \mathrm{supp}(A)$

<u>Also observe</u>: $v(\neg A)$ depends only on $v(A)$

and $v((A * B))$ depends only on $v(A)$ and $v(B)$.

- For a given A, we can evaluate $v(A)$ recursively using its parse tree

<u>Ex</u> Sps $A = ((p \Rightarrow q) \wedge (q \Leftrightarrow \neg r))$ and v is an assignment s.t. $v(p) = 1$, $v(q) = 0$, $v(r) = 1$

To compute $v(A)$, first find parse tree:

$((p \Rightarrow q) \wedge (q \Leftrightarrow \neg r))$

- $(p \Rightarrow q)$
  - p
  - q
- $(q \Leftrightarrow \neg r)$
  - q
  - $\neg r$
    - r

- now: $v(p) = v(r) = 1$, $v(q) = 0$

so: $v((p \Rightarrow q)) = 0$, $v(\neg r) = 0$

$v(q \Leftrightarrow \neg r) = 1$

$v((p \Rightarrow q) \wedge (q \Leftrightarrow \neg r)) = 0$

i.e. $v(A) = 0$ ✓

Models of A

Def'n Sps τ is a truth assign't and A is a wff.

We say τ satisfies or models A (or τ is a model of A) if $\tau(A)=1$.

In this case we write $\tau \models A$.

- if $\tau(A) = 0$, write $\tau \not\models A$.

Observe: for all wffs A, B we have:

$\tau \models \neg A$ iff $\tau \not\models A$

$\tau \models (A \wedge B)$ iff $\tau \models A$ and $\tau \models B$

$\tau \models (A \vee B)$ iff $\tau \models A$ or $\tau \models B$

$\tau \models (A \Rightarrow B)$ iff either $\tau \not\models A$ or $\tau \models B$

$\tau \models (A \oplus B)$ iff either ($\tau \models A$ and $\tau \not\models B$) or ($\tau \not\models A$ and $\tau \not\models B$

- indeed, we could have defined the relation $\models$ using the above rules, starting from $\tau \models p$ iff $\tau(p)=1$

Def'n a wff A is a tautology (or, logically valid) if $\tau \models A$ for every assignment τ

- so: a tautology A is true regardless of truth values of vars in A.

<u>Ex</u> (i) for any A:
$(A \Leftrightarrow \neg\neg A)$
is a tautology
↳ clear... but how to justify? Truth Table!
↳ have a column for every subfmla

A	$\neg A$	$\neg\neg A$	$(A \Leftrightarrow \neg\neg A)$
0	1	0	1
1	0	1	1

possibilities for $\nu(A)$

always 1
⇒ tautology

(ii) for any A, B
$(\neg(A \wedge B) \Leftrightarrow (\neg A \vee \neg B))$ is a tautology

A	B	$(A \wedge B)$	$\neg(A \wedge B)$	$\neg A$	$\neg B$	$(\neg A \vee \neg B)$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

- one of "De Morgan's laws" .. $(\neg(A \wedge B) \Leftrightarrow (\neg A \vee \neg B))$

$(\neg(A \wedge B) \Leftrightarrow (\neg A \vee \neg B))$
1
1
1
1

- likewise
$(\neg(A \vee B) \Leftrightarrow (\neg A \wedge \neg B))$ ✓
is a tautology

"assoc-iative laws" (iii) $((A\wedge B)\wedge C) \Leftrightarrow (A\wedge(B\wedge C))$
$((A\vee B)\vee C) \Leftrightarrow (A\vee(B\vee C))$ are tautologies
(you try!)

~~[illegible]~~
- these justify dropping parentheses to abbreviate long strings of $\wedge$'s or $\vee$'s

"distrib-utive laws" (iv) $(A\wedge(B\vee C)) \Leftrightarrow ((A\wedge B)\vee(A\wedge C))$
$(A\vee(B\wedge C)) \Leftrightarrow ((A\vee B)\wedge(A\vee C))$
are tautologies (you try!)

(v.) $(p\Rightarrow q)\Rightarrow(q\Rightarrow p)$ is not a tautology: if $v(p)=1$, $v(q)=0$
then $v((p\Rightarrow q)\Rightarrow(q\Rightarrow p))=0$
$\neq 1$

(vi.) $(p\Rightarrow q) \Leftrightarrow (\neg p\vee q)$
is a tautology (you try).

Def'n - a wff A is satisfiable if there is some assignment v s.t. $v\models A$
- o/w A is unsatisfiable
(or contradictory)

- e.g. $A=(p\wedge q)$ is satisfiable, since if $v(p)=v(q)=1$ then $v(A)=1$

- but $A = (p \wedge \neg p)$ is unsatisfiable

Observe: A is unsatisfiable iff $\neg A$ is a tautology.

- we can extend the notion of satisfaction from one fmla to sets of fmlas

Def'n Sps S is a set of wffs and v is a truth assignment. We say v satisfies or models S if $v \models A$ for every $A \in S$.

In this case we write $v \models S$.

Def'n S is satisfiable if there is at least one v s.t. $v \models S$.

Ex (i) $S = \{(p_1 \vee p_2), (\neg p_2 \vee \neg p_3), (p_3 \vee p_4), (\neg p_4 \vee \neg p_5), \ldots\}$ is satisfiable, namely by

$$v(p_i) = \begin{cases} 1 & \text{if } i \text{ odd} \\ 0 & \text{if } i \text{ even} \end{cases}$$

Do any other assignments satisfy S?

(ii) $S = \{p \Rightarrow q, q \Rightarrow r, r \Rightarrow p, \neg p, q\}$
is not satisfiable (why?)
but $T = \{p \Rightarrow q, q \Rightarrow r, r \Rightarrow p, \neg p, \neg q\}$
is satisfiable (why?)

(iii) in general if $S = \{A_1, \ldots, A_n\}$ is a finite set of fmlas, then $\upsilon \models S$ iff $\upsilon \models A_1 \wedge A_2 \wedge \ldots \wedge A_n$

Logical implication and equivalence

Def'n Sps S is a set of wffs and A is a wff. We say S logically implies A (and write $S \models A$) if every υ satisfying S also satisfies A

Notes (i) overloading "$\models$" notation here: $S \models A$ means "if $\upsilon \models S$ then $\upsilon \models A$"

(ii) if $S = \emptyset$ write $\models A$ instead of $\emptyset \models A$ (this just means A is a tautology — why?)

(iii) if $S = \{A\}$ sometimes write $A \models B$ instead of $\{A\} \models B$.

Ex: (i) $\{B \vee C, \neg C\} \models B$
why: if $v(B \vee C) = 1$ and $v(\neg C) = 1$ must have $v(B) = 1$

(ii) $\{p \vee q, p\} \not\models q$
why: could have $v(p) = 1$ $v(q) = 0$
then $v \models \{p \vee q, q\}$ but $v \not\models q$

"modus ponens" (iii) $\{A, A \Rightarrow B\} \models B$
why: if $v(A) = v(A \Rightarrow B) = 1$ must have $v(B) = 1$

(iv) $\{p_1 \Rightarrow p_2, p_2 \Rightarrow p_3, p_3 \Rightarrow p_4, \ldots\} \models p_1 \Rightarrow p_n$
for any n – you try.

(iv) (Deduction rule) For any S, A, B
$S \models (A \Rightarrow B)$ iff $S \cup \{A\} \models B$

Pf: (forward direction) Sps $S \models (A \Rightarrow B)$ and fix v s.t. $v \models S \cup \{A\}$. WTS $v \models B$
- then $v \models S$ and $v(A) = 1$
- since $S \models (A \Rightarrow B)$, $v(A \Rightarrow B) = 1$
- since $v(A) = 1$, must have $v(B) = 1$
- since v was arbitrary, $S \cup \{A\} \models B$ ✓

(backward) Sps $S \cup \{A\} \models B$ and $v \models S$
- either $v(A) = 0$, in which case $v(A \Rightarrow B) = 1$ vacuously

or $v(A) = 1$. But then $v \models S \cup \{A\}$
– hence $v(B) = 1$ and so $v(A \Rightarrow B) = 1$
in this case as well.
$\Rightarrow S \models A \Rightarrow B$ ✓

(vii) $S \models A$ iff $S \cup \{\neg A\}$ is unsatisfiable
pf: you try.

Def'n Two wffs A, B are logically equivalent (write: $A \equiv B$) if
$A \models B$ and $B \models A$.

– equivalently: $A \equiv B$ iff $(A \Leftrightarrow B)$
is a tautology (why?)

Observe: $\equiv$ is an equivalence relation on wffs, i.e. for all wffs A, B, C

$A \equiv A$

$A \equiv B$ implies $B \equiv A$

$A \equiv B$ and $B \equiv C$ implies $A \equiv C$

– think of $A \equiv B$ as saying A, B
same semantically
– can still be very different as formulas!
(i.e. syntactically)
– e.g. $A \equiv \neg\neg\neg\neg\neg\neg A$

Ex: can rewrite our tautologies above as logical equivalences

(i) $A \equiv \neg\neg A$

(ii) (De Morgan's laws)

$\neg(A \wedge B) \equiv \neg A \vee \neg B$

$\neg(A \vee B) \equiv \neg A \wedge \neg B$

(iii) (Distributive laws)

$A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$

$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$

- other useful equivalences:

(iv) $(A \Rightarrow B) \equiv (\neg A \vee B)$

(v) $(A \Leftrightarrow B) \equiv (A \wedge B) \vee (\neg A \wedge \neg B)$

(vi) $(A \wedge B) \Rightarrow C \equiv (A \Rightarrow C) \vee (B \Rightarrow C)$

Pf: you try

↳ "Appendix A" in Files tab on Canvas contains list of useful tautologies/ equivalences

- Some other facts you can use:

Fact 1: if A is a wff containing the variables $p_1, \dots, p_n$ and $B_1, \dots, B_n$ are wffs then: if A is a tautology so is $A[p_1/B_1, \dots, p_n/B_n]$

(i.e. fmla obtained by replacing each instance of p_i in A w/ B_i)