→ Propositional Logic

- propos° : statement T V F
- predicate : . propos° whose truth depends on vars.
  . func° {vars} → {T, F}

→ Satisfiability

P = set of yes/no pb that can be <u>solved</u> in polynomial time (poly time : depends on input length)

NP = set of yes/no pb that can be <u>verified</u> in polynomial time

theorem: $P = NP \iff SAT \in P$

→ Predicate logic formulas

- Occurrence of a var is quantified if it is in a subformula quantified $(Qx \in D, E(x))$. Otherwise it is <u>free</u> (unquantif.)
- <u>Interpreta°</u> of a formula :
  → <u>non-empty</u> sets for each domain
  → element of relevant domain for each cst symbol
  → func° for each cst symbol of a func° from relevant domain to {T, F} for each predicate symbol.
  → if it has free vars they have to be mapped to an element in the relevant domain (called evaluation)
- A PLF is valid / tautology if:
  → true for any interpreta°
  → unsatisfiable if false under all interpreta°
  → satisfiable if true under some interpreta°

→ Proofs

- See 165
- Note : by cases : $. P(x) \Rightarrow P_1(x) \vee P_2(x) \ldots$
  $. P_1(x) \Rightarrow Q(x) \wedge P_2(x) \Rightarrow Q(x) \ldots$
  $. (P_1(x) \vee P_2(x) \ldots) \Rightarrow Q(x)$ # modus ponens

→ Complete set of connectives

e.g. $\{\vee, \wedge, \neg\}$ is complete # because every formula can be rewritten in CNF/DNF.

$\{\wedge, \neg\}$ (or $\{\vee, \neg\}$) is complete # De Morgan.

$\{NAND\}$ is complete # $P \wedge Q \Leftrightarrow (P \text{ NAND } Q) \text{ NAND } (P \text{ NAND } Q)$

$\neg P \Leftrightarrow (P \text{ NAND } P)$

$\{$if-then-else$\}$ # $P \wedge Q \Leftrightarrow$ i-t-e (P, Q, F)

$P \vee Q \Leftrightarrow$ i-t-e (P, T, Q)

$\neg P \Leftrightarrow$ i-t-e (P, F, T)

▶ a set of connectives is complete iff any prop formula can be rewritten using its elements.

→ Substitution

Theorem Let R be a formula, let S be a subformula of R, let S' be a formula logically $\Leftrightarrow$ to S, let R' be the formula that results by replacing S by S' in R. Then R' is logically $\Leftrightarrow$ to R.

e.g. $R = (\neg A \wedge \neg B) \text{ XOR } (B \Rightarrow (\neg A \wedge \neg B))$ | $S = \neg A \wedge \neg B$, $S' = \neg(A \vee B)$

$R' = \neg(A \vee B) \text{ XOR } (B \Rightarrow \neg(A \vee B))$

→ $R \Leftrightarrow R'$

Theorem Suppose P is a prop variable and R is a prop tautology that contains some occurrences of P. Let R' be the formula obtained from R by replacing every occurrence of P by Q (where Q is a formula). Then R' is a tautology.

e.g. $R = (A \vee P) \Leftrightarrow (P \vee A)$ is a tautology

Let $Q = (C \vee D)$, then $R' = (A \vee C \vee D) \Leftrightarrow (C \vee D \vee A)$ is a taut.

$0.3999\ldots = 0.4000\ldots$

Proof
$0.3999\ldots = x$
$3.999\ldots = 10x$
substract: $3.6 = 9x$
$x = 0.4$

Reduce

Th there is no prog A that takes a prog P and determines that P is syntactically correct and halts on all inputs.

Proof $H'(P, x) = \{F$, if P is not syn corr / P...

→ H' solves halting pb.