Wrapping Up Loose Ends of Time Complexity Module

---

1. Exam 1 grades, qs 1d full marks to all, qs 2 reduction gets full marks if introduction of new literals is mentioned.

2. Exactly-1 3SAT solution using 4 dummy literals

   3SAT → Exactly-1 3SAT

   given φ , for each clause (x ∨ y ∨ z), create 4 literals only for this clause, a,b,c,d, and replace the original clause with:

   (~x ∨ a ∨ b) ∧ (b ∨ y ∨ c) ∧ (c ∨ d ∨ ~z)

R(~x, a, b)	R(b, y, c)	R(c, d, ~z)
(1, a, b)	(b, 0, c)	(c, d, 1)
(1, a, b)	(b, 0, c)	(c, d, 0)
(1, a, b)	(b, 1, c)	(c, d, 1)
(1, a, b)	(b, 1, c)	(c, d, 0)
(0, a, b)	(b, 0, c)	(c, d, 1)
(0, a, b)	(b, 0, c)	(c, d, 0)
(0, a, b)	(b, 1, c)	(c, d, 1)
(0, a, b)	(b, 1, c)	(c, d, 0)

3. Why *Each co-NP-complete problem is the complement of an NP-complete problem*, e.g. why TAUT $\leq_p$ 3UNSAT but TAUT $\leq_p^?$ 3SAT

Proof: Consider L ∈ coNP-Complete, i.e. L is in coNP and any language in coNP, suppose language A, reduces to L. Now consider L-complement ∈ NP (by definition), then A-complement can be reduced to L-complement by the same reduction function as was used for A ≤ L.

explanation:

∀x x ∈ A iff f(x) ∈ L

also means

∀x x ∉ A iff f(x) ∉ L

∀x x ∈ A-complement iff f(x) ∈ L-complement

But what about A-complement? Does it not reduce to L as well?

$A \leq_p L$

$a \in A \rightarrow f(a)$, s.t. Machine_L outputs 1 iff $a \in A$

$a' \in$ A-complement $\rightarrow f(a')$, s.t. Machine_L 0 iff $a' \in$ A-complement

TAUT ≤ 3UNSAT: On input φ: output ~φ

reduction function: negation

TAUT-complement ≤ 3UNSAT

TAUT-complement ≤ 3SAT (the same reduction function will work as for TAUT ≤ 3UNSAT)

4. If EXP ≠ NEXP then P ≠ NP (*padding technique*)

   To prove that if P = NP then EXP = NEXP

1. Consider $L \in \mathrm{NTIME}(2^{n^{\wedge}c})$ <2 to the power n to the power c> that can be decided on NTM, M.
2. Now consider the language $L_{pad} = \{ \langle x, 1^{|x|^{\wedge}c} \rangle \mid x \in L \}$
3. Since $L \in$ NEXP, $L_{pad} \in$ NP, because padding the input changes the relation between input length and computation time.
4. Since P = NP, $L_{pad} \in$ P.
5. But then $L \in$ EXP.
6. Therefore, EXP = NEXP

5. equalities between complexity classes “scale up”