Solutions

2.1

- B (the target): if A is hard, B must be at least as hard ✓.

2.2

- The problem is in NP (polynomial verification); every NP problem reduces to it ✓.

2.3

- $(\bar{G}, n-k) = (\bar{G}, 1)$: complement edges A–D, B–D, C–D; cover {D} ✓.

3.1

- Translator: $(G, k) \mapsto (\bar{G}, n-k)$ – build the complement in $O(V^2)$ ✓.
- Forward: a k-clique's nodes are mutually non-adjacent in $\bar{G}$, so every $\bar{G}$-edge has an endpoint outside it – the other $n-k$ nodes form a cover ✓.
- Backward: a cover S of size $n-k$ leaves $n-|S| = k$ nodes touching no $\bar{G}$-edge – an independent set in $\bar{G}$, i.e. a clique in G ✓.
- Run: $\bar{G}$'s edges are exactly A–D, B–D, C–D; {D} covers them all at size $1 = n-k$ ✓.
- Implication: a polynomial vertex-cover solver would solve CLIQUE, hence SAT, hence every NP problem – P = NP ✓.

3.2

- They proved WARS $\leq_p$ SAT: a SAT solver solves WARS instances – it says nothing about WARS being hard ✓.
- Instead: reduce a known NP-complete problem (SAT, 3-SAT, CLIQUE, …) TO WARS with a polynomial translator ✓.
- NP-completeness needs: membership in NP (a verifiable certificate in polynomial time) plus NP-hardness by such a reduction ✓.
- Skipping membership: WARS might be harder than NP (e.g. outside NP) – then calling it NP-complete is simply unproven ✓.

exercises.lol exercises.com.cn