Thm 7.32

3SAT $\leq_p$ CLIQUE

"if CLIQUE $\in$ P, then so is 3SAT"

3SAT = { ⟨φ⟩ | φ is a Boolean expression in 3CNF form, and φ has a satisfying assignment. }

CLIQUE = { ⟨G, K⟩ | G is an undirected graph that contains a clique of size K }

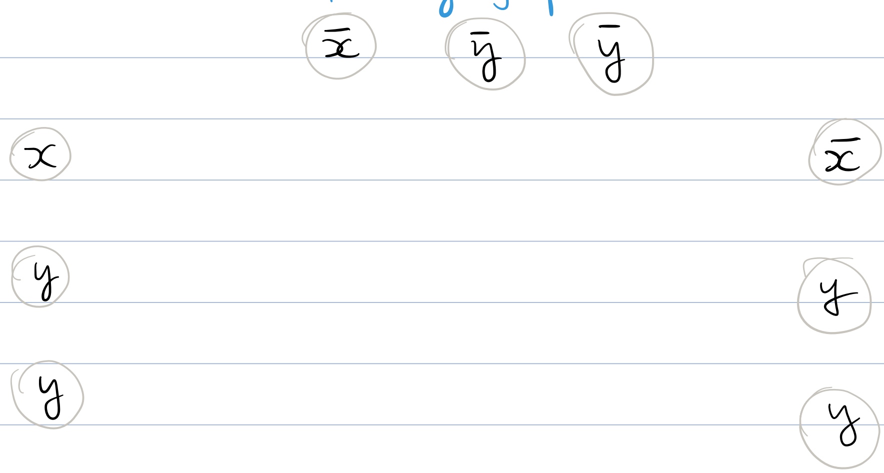
Proof: We polyreduce $\overset{X}{\text{3SAT}}$ to $\overset{Y}{\text{CLIQUE}}$.

We will show how it is done using an example, then give the general instructions.

Suppose our 3SAT input is

$$(x \vee x \vee y) \wedge (\bar{x} \vee \bar{y} \vee \bar{y}) \wedge (\bar{x} \vee y \vee y)$$

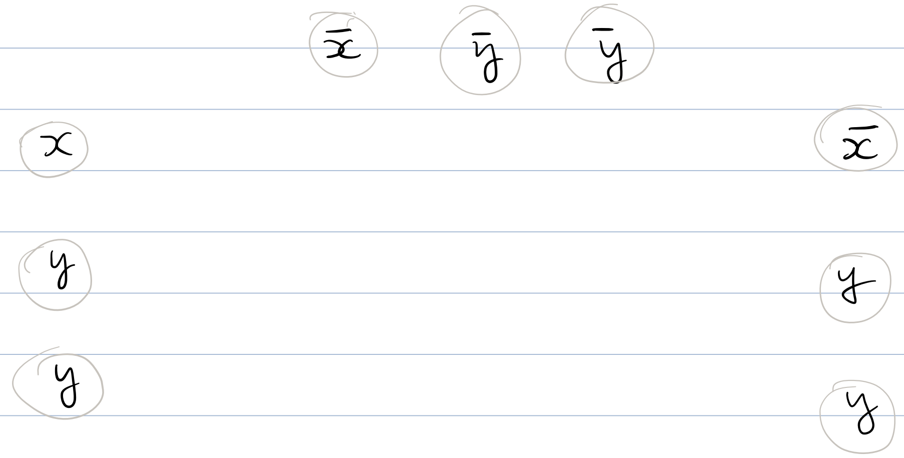
We construct the following graph:



add edges between any 2 literals in different clauses if the two do not contradict one another.

(i.e. could both be true simultaneously).

c clauses — what does it mean if $\exists$ a clique of size c?



X = "on input $\langle \phi \rangle$ where ϕ is a 3SAT Boolean expression,

1. Make a graph G where
   - for each clause, make 3 vertices, labelled with a literal from the clause } clause gadget
   - $\forall$ u, v where u and v are in different clause-gadgets, and u and v are not labelled with contradicting literals (x_1 and $\bar{x}_1$, for example) add the edge (u,v)
2. Let c be number of clauses in ϕ.
   Run Y on $\langle G, c \rangle$ // Y is CLIQUE-solver.
   If Y accepts, ACCEPT.
   If Y rejects, REJECT."

Since all steps EXCEPT perhaps the call to Y are polynomial, then if Y is polynomial, so is X.

Y correctly decides if ϕ is satisfiable because:

a clique of size c in G must

contain a vertex from each of the c
clause-gadgets (there being no edges within
a clause-gadget)
and each of those vertices is logically consistent
with all the others (being adjacent to all the others)
and so yields a satisfying assignment to the
variables.