3SAT

A Boolean expression is in 3CNF if it is in conjunctive normal form and each clause contains at most 3 literals.

3SAT is defined as the language consisting of those expressions in 3CNF that are satisfiable.

3SAT is NP-complete, as there is a polynomial time reduction from CNF-SAT to 3SAT.

Composing Reductions

Polynomial time reductions are clearly closed under composition. So, if \( L_1 \leq_P L_2 \) and \( L_2 \leq_P L_3 \), then we also have \( L_1 \leq_P L_3 \).

Note, this is also true of \( \leq_L \), though less obvious.

If we show, for some problem \( A \) in NP that

\[ \text{SAT} \leq_P A \]

or

\[ \text{3SAT} \leq_P A \]

it follows that \( A \) is also NP-complete.

Independent Set

Given a graph \( G = (V, E) \), a subset \( X \subseteq V \) of the vertices is said to be an independent set, if there are no edges \((u, v)\) for \( u, v \in X \).

The natural algorithmic problem is, given a graph, find the largest independent set.

To turn this optimisation problem into a decision problem, we define IND as:

\[ \text{IND} \text{ is the set of pairs } (G, K), \text{ where } G \text{ is a graph, and } K \text{ is an integer, such that } G \text{ contains an independent set with } K \text{ or more vertices.} \]

IND is clearly in NP. We now show it is NP-complete.
Reduction

We can construct a reduction from 3SAT to IND.

A Boolean expression $\phi$ in 3CNF with $m$ clauses is mapped by the reduction to the pair $(G, m)$, where $G$ is the graph obtained from $\phi$ as follows:

$G$ contains $m$ triangles, one for each clause of $\phi$, with each node representing one of the literals in the clause. Additionally, there is an edge between two nodes in different triangles if they represent literals where one is the negation of the other.

Clique

Given a graph $G = (V, E)$, a subset $X \subseteq V$ of the vertices is called a clique, if for every $u, v \in X$, $(u, v)$ is an edge.

As with IND, we can define a decision problem version:

**CLIQUE** is defined as:

The set of pairs $(G, K)$, where $G$ is a graph, and $K$ is an integer, such that $G$ contains a clique with $K$ or more vertices.

Clique 2

**CLIQUE** is in NP by the algorithm which *guesses* a clique and then verifies it.

**CLIQUE** is NP-complete, since $\text{IND} \leq_p \text{CLIQUE}$

by the reduction that maps the pair $(G, K)$ to $(\bar{G}, K)$, where $\bar{G}$ is the complement graph of $G$. 

Example

$$(x_1 \lor x_2 \lor \neg x_3) \land (x_3 \lor \neg x_2 \lor \neg x_1)$$
**$k$-Colourability**

A graph $G = (V, E)$ is $k$-colourable, if there is a function

$$\chi : V \rightarrow \{1, \ldots, k\}$$

such that, for each $u, v \in V$, if $(u, v) \in E$,

$$\chi(u) \neq \chi(v)$$

This gives rise to a decision problem for each $k$.

For all $k > 2$, $k$-colourability is $\text{NP}$-complete.

**3-Colourability**

3-Colourability is in $\text{NP}$, as we can guess a colouring and verify it.

To show $\text{NP}$-completeness, we can construct a reduction from 3SAT to 3-Colourability.

For each variable $x$, have two vertices $x, \bar{x}$ which are connected in a triangle with the vertex $a$ (common to all variables).

In addition, for each clause containing the literals $l_1, l_2$ and $l_3$ we have a gadget.

With a further edge from $a$ to $b$. 

**Gadget**