**Algorithms**

NP - Completeness

**Hardness**

---

**Algorithm**

- **S**tupid!
- **F**ew steps in classes
  - 2CNF
  - \( n \) variables in \( 2^{2n} \) clauses
  - Try all possible assignments

---

\[(x \lor y) \land (\neg x \lor y) \land (\neg x \lor \neg y)\]

\(n\) variables in classes.

\(F\)

- \(x=0\) and \(y=1\)
- Satisfied

\((x \lor y) \land (\neg x \lor \neg y) \land (x \lor \neg y)\)

3CNF

---

**Graph**

- 2n vertices
- 2m edges

---

**DAG**

- 1 vertex
- 2 vertices

---

**Graph**

- 3 vertices
- 2 edges
2SAT Linear Time
3SAT 3CNF
\[(\sum_{x \in \{0,1\}} (x \lor y \lor z)) \land (x \lor y \lor z) \land \ldots \land (x \lor y \lor z)\]
TRY EVERYTHING

Brute force search
\[O(2^n) - \text{can we do better?}\]
P = NP? We don't know
\[O\left(2^{\frac{c}{n^2}}\right)\]

2SAT/3SAT
dimension \[2^n\] c is some constant
any lower-bounds

Seth
Strong exponential time hypothesis
3SAT (kSAT for k satisfiable) can not be solved in time faster than \[2^n\]

\[O(n^c) \downarrow 2^n \leq 2^n\]
c long \[cn \log\]

3SAT
Clique
Brute force solving
\[1, 1.7, 4.49 \leq \frac{k}{n}\]
\[O(k^2) \leq O(n^2)\]
**NP: Not Deterministic Polynomial**

All decision problems.

S.T.: A yes/true answer can be verified in polynomial time.

**Cook-Levin Theorem**

If 3SAT can be solved in polynomial time, then all the problems in NP can be solved in polynomial time!