What's 3-SAT?

"AND" of "ORs"
$\wedge$ outside parens
$\vee$ inside parens

Input: A list of Boolean variables $x_1, \ldots, x_n$

An expression in Conjunctive Normal Form, where each clause has exactly 3 literals.

One of the subexpressions inside parens

Something like:

$$(z_i \vee z_j \vee z_k) \wedge (z_i \vee z_\ell \vee z_a) \wedge \cdots \wedge (z_a \vee z_b \vee z_c)$$

Where z is a "literal" a variable or the negation of a variable (x_i, $\neg x_j$, etc.).

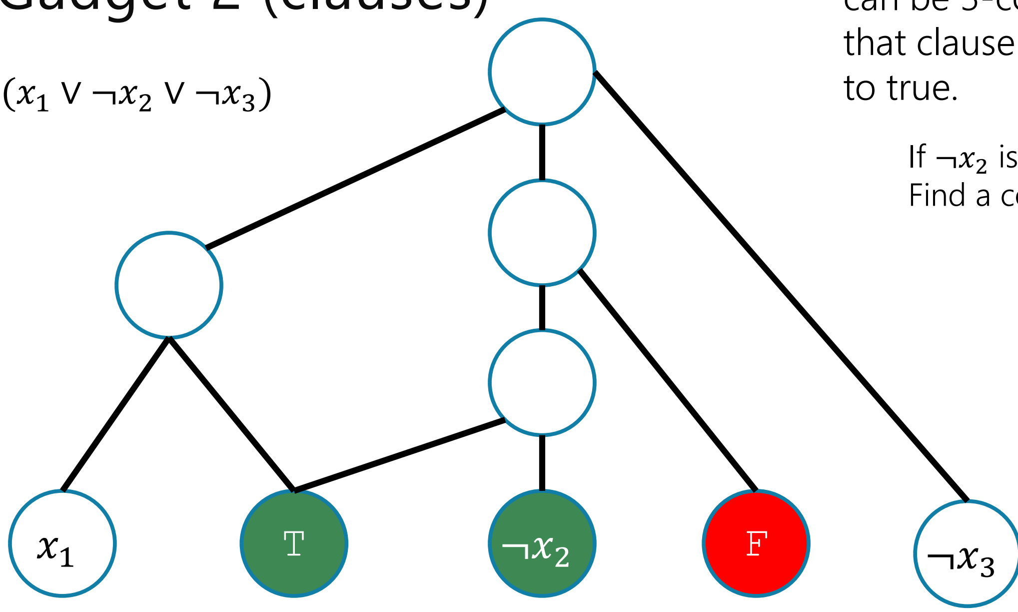
Output: true if there is a setting of the variables where the expression evaluates to true, false otherwise.

Why is it called 3-SAT? 3 because you have 3 literals per clause
SAT is short for "satisfiability" can you satisfy all of the constraints?

Gadget 2 (clauses)

This tricky little graph can be 3-colored iff that clause evaluates to true.

$(x_1 \vee \neg x_2 \vee \neg x_3)$



If $\neg x_2$ is true…
Find a coloring!

Putting it together

If the graph is 3-colorable, then there must be a satisfying assignment

"Size"

Normally all (reasonable) representations give you the same behavior.

But *occasionally* it really matters.

The most common time where it matters is when (potentially very large) integers are a part of the input.

Consider the following problem:

PRIMES (on input n, n represented in binary, return true if n is prime)

Your algorithm? Trial division (is it divisible by 2, 3, 4, 5,…)

What's the running time? Is it polynomial?