Recap from last class:

For problems A & B,

$A \to B$ or $A \leq B$

means we can reduce A to B:

- given a poly-time algorithm for B we can construct a poly-time algorithm for A.

SAT:

input: Boolean formula f in CNF with n variables & m clauses $x_1, \ldots, x_n$

output: assignment to the variables that satisfies f, if one exists & NO otherwise

What's CNF?

CNF = conjunctive normal form

Variables $x_1, \ldots, x_n$ take values TRUE or FALSE

literals $x_1, \overline{x_1}, x_2, \overline{x_2}, \ldots, x_n, \overline{x_n}$

$\vee$ = OR, $\wedge$ = AND

Clauses are OR of several literals: example $(\overline{x_2} \vee x_3 \vee \overline{x_5})$

f is AND of clauses:

example: $(\overline{x_3}) \wedge (\overline{x_2} \vee x_3 \vee \overline{x_5}) \wedge (x_1 \vee x_5)$

Theorem: SAT is NP-complete

This means that:

a) SAT $\in$ NP

b) for all $A \in$ NP, $A \to$ SAT

So unlikely to be a poly-time alg. for SAT as that would imply a poly-time alg. for every problem in NP.

3SAT:

input: Boolean formula f in CNF with n variables & m clauses where each clause has ≤ 3 literals

output: satisfying assignment if one exists & NO otherwise.

We'll show 3SAT is NP-complete.

Need to show:

a) 3SAT $\in$ NP

b) SAT $\to$ 3SAT.

For (a): 3SAT $\in$ NP:

Given an assignment in $O(1)$ time per clause we can check that at least 1 liteal is satisfied in every clause. Thus $O(m)$ total time to check that f is satisfied.

For (b): SAT $\rightarrow$ 3SAT.

Take input f for SAT.
We need to create input f' for 3SAT.
Then given a satisfying assignment for f'
we need to define a satisfying assignment for f.
We also need to show that:

f is satisfiable $\Longleftrightarrow$ f' is satisfiable.

Example $f = \underbrace{(x_3)}_{C_1} \wedge \underbrace{(\overline{x_2} \vee x_3 \vee \overline{x_1} \vee \overline{x_4})}_{C_2} \wedge \underbrace{(x_2 \vee x_1)}_{C_3}$

Clauses C_1 & C_3 can stay the same but C_2 is too big for 3SAT.

Create a new variable y.

Look at:

$$C_2' = (\overline{x_2} \vee x_3 \vee y) \wedge (\overline{y} \vee \overline{x_1} \vee \overline{x_4})$$

Claim: C_2 is satisfiable $\Longleftrightarrow$ C_2' is satisfiable.

Proof:

($\Rightarrow$): Take a satisfying assignment to x_2, x_3, x_1, x_4 for C_2. This satisfies at least one of the clauses in C_2' & use y to satisfy the other.

($\Leftarrow$): Take sat. assig. to C_2'.

if $y = T$ then $x_1 = F$ or $x_4 = F$

if $y = F$ then $x_2 = F$ or $x_3 = T$

in either case, $\overline{x_2}$ or x_3 or $\overline{x_1}$ or $\overline{x_4}$ is sat., so C_2 is satisfied. $\blacksquare$

Suppose $C = (\overline{x}_2 \vee x_3 \vee \overline{x}_1 \vee \overline{x}_4 \vee x_5)$

then create 2 new variables y_1 & y_2

Let $C' = (\overline{x}_2 \vee x_3 \vee y_1) \wedge (\overline{y}_1 \vee \overline{x}_1 \vee y_2) \wedge (\overline{y}_2 \vee x_4 \vee x_5)$

<u>Claim:</u> fix an assignment to $x_1, \ldots, x_5$

then

C is satisfied $\iff$ there is an assignment to y_1, y_2 ~~that satisfies~~ so that C' is satisfied.

In general, for

$$C = (a_1 \vee a_2 \vee \cdots \vee a_k)$$

for literals $a_1, \ldots, a_k$

add $k-3$ new variables $y_1, \ldots, y_{k-3}$

and replace C by $k-2$ clauses:

$$C' = (a_1 \vee a_2 \vee y_1) \wedge (\overline{y}_1 \vee a_3 \vee y_2) \wedge (\overline{y}_2 \vee a_4 \vee y_3)$$
$$\wedge \cdots \wedge (\overline{y}_{k-4} \vee a_{k-2} \vee y_{k-3}) \wedge (\overline{y}_{k-3} \vee a_{k-1} \vee a_k)$$

Claim: for any assignment to $a_1, \dots, a_k$

C is satisfied $\iff$ there is an assignment to $y_1, \dots, y_{k-2}$ so that C' is satisfied

Proof:

$(\Rightarrow)$ Take assignment to $a_1, \dots, a_i$ satisfying C.

Let a_i be min i where a_i is satisfied.

So $a_i = T \Rightarrow$ clause $i-1$ in C' is satisfied

Set $y_1 = y_2 = \dots = y_{i-2} = T$

$\Rightarrow$ 1st $i-2$ clauses in C' are satisfied.

Set $y_{i-1} = \dots = y_{k-2} = F$

$\Rightarrow$ clauses $i, \dots, k-2$ in C' are satisfied.

$(\Leftarrow)$ Take assignment to $a_1, \dots, a_k, y_1, \dots, y_{k-2}$ sat. C'.

$\underbrace{\text{At least one of } a_i = T}_{\text{Why?}}$, thus C is satisfied.

Suppose $a_1 = a_2 = \cdots = a_k = F$

Then since C' is satisfied

for clause 1 we must have $y_1 = T$

for clause 2 " $y_2 = T$

$\vdots$

for clause $k-3$ we must have $y_{k-3} = T$

and then clause $(\overline{y_{k-3}} \vee a_{k-1} \vee a_k) = F$

So we have a contradiction.

SAT $\rightarrow$ 3SAT:

Given f for SAT

Create a new formula f' by the following procedure:

For each clause C in f,

if C has ≤ 3 literals keep it the same

if C has > 3 literals then add $k-3$ new variables & replace C by C' as described before.

Use f' as input for 3SAT.

f is satisfiable $\Leftrightarrow$ f' is satisfiable.

($\Rightarrow$) Given assignment to $x_1, \ldots, x_n$ satisfying f, for each C there is an assignment to the $k-3$ new variables so that C' is satisfied.

($\Leftarrow$) Given a satisfying assignment to f', for each C' in f' at least one of the literals in C is satisfied.

Given a satisfying assignment for f', ignore the new variables, and the ~~new~~ assignment for $x_1, \ldots, x_n$ is also a satisfying assignment for f.