The 3-CNF satisfiability problem (3-CNF-SAT)

問題: 給定一 conjunction normal form (CNF) ϕ，其中每个 clause 皆具有 3个 literal

是否存在一組 truth assignment $T \to \phi$ 為 true?

e.g. $\phi = \underbrace{(x_1 \vee \overline{x_2} \vee \overline{x_3})}_{\text{clause}} \wedge (\overline{x_1} \vee x_3 \vee x_4)$

literal

每个 clause 中的所有 literal 以做 OR 運算

每个 conjunction normal form 中所有 clause 以做 AND 運算

Theorem: 3-SAT 為 NP complete

① 3-SAT $\in$ NP

給定一 $\langle x_1, x_2, \ldots, x_n \rangle$ 為 certificate

將 certificate 中 assignment 之值代入看是否為1，即可驗證是否為 satisfiable

此 verfication algorithm 為 polynomial-time sovable

故 3-SAT $\in$ NP

② SAT $\le_p$ 3-SAT

給定一 ϕ 為 SAT 之 problem instance

欲建構 ϕ''' 為 3-SAT 之 problem instance，建構 ϕ'' 之過程分為三步

step 1. 利用 ϕ 建構一 parse BT

其中 literal 為 leaves 而 connectives 為 internal node

並 assign 给每个 internal node 之 output 一个变數 y_i

可將 ϕ 転成 root variable AND 其它每个 node 之 operation 对應之

conjunction of clause 的型式 ϕ'

可得各个 ϕ' 中 conjunction of clause ϕ_i

且 $\forall i$，ϕ_i' 中最多只有3个 literal

step 2. 把 ϕ_i' 転变成 conjuctive normal form

利用各个 ϕ_i' 建構一 truth table，包含該 clause 中所有 literal 可能的 assignment

利用 truth table 可得 disjunctive normal form (DNF)，等價於 $\neg\phi_i'$

再用 negation 和 DeMorgan's Law 得 CNF formula ϕ_i''

Example:

$\phi = ((x_1 \to x_2) \vee \neg((\neg x_1 \leftrightarrow x_3) \vee x_4)) \wedge \neg x_2$.

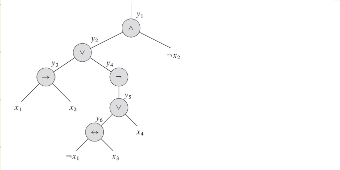

Figure 34.11 The tree corresponding to the formula $\phi = ((x_1 \to x_2) \vee \neg((\neg x_1 \leftrightarrow x_3) \vee x_4)) \wedge \neg x_2$.

$$
\begin{aligned}
\phi' = y_1 &\wedge (y_1 \leftrightarrow (y_2 \wedge \neg x_2)) \\
&\wedge (y_2 \leftrightarrow (y_3 \vee y_4)) \\
&\wedge (y_3 \leftrightarrow (x_1 \to x_2)) \\
&\wedge (y_4 \leftrightarrow \neg y_5) \\
&\wedge (y_5 \leftrightarrow (y_6 \vee x_4)) \\
&\wedge (y_6 \leftrightarrow (\neg x_1 \leftrightarrow x_3)) .
\end{aligned}
$$

y_1	y_2	x_2	$(y_1 \leftrightarrow (y_2 \wedge \neg x_2))$
1	1	1	0
1	1	0	1
1	0	1	0
1	0	0	0
0	1	1	1
0	1	0	0
0	0	1	1
0	0	0	1

Figure 34.12 The truth table for the clause $(y_1 \leftrightarrow (y_2 \wedge \neg x_2))$.

as follows. The truth table for ϕ_1' appears in Figure 34.12. The DNF formula equivalent to $\neg\phi_1'$ is

$$(y_1 \wedge y_2 \wedge x_2) \vee (y_1 \wedge \neg y_2 \wedge x_2) \vee (y_1 \wedge \neg y_2 \wedge \neg x_2) \vee (\neg y_1 \wedge y_2 \wedge \neg x_2) .$$

Negating and applying DeMorgan's laws, we get the CNF formula

$$
\begin{aligned}
\phi_1'' = &(\neg y_1 \vee \neg y_2 \vee \neg x_2) \wedge (\neg y_1 \vee y_2 \vee \neg x_2) \\
&\wedge (\neg y_1 \vee y_2 \vee x_2) \wedge (y_1 \vee \neg y_2 \vee x_2) ,
\end{aligned}
$$

故 $\forall i, \phi_i'$ 轉成了 CNF ϕ_i''，因此 ϕ' 轉成了 ϕ''

且 ϕ'' 中每个 clause 至多有 3个 literal

step 3. 將 ϕ'' 轉成各个 clause 恰含 3个 distinct literal 之 ϕ'''

对於 $C_i \in \phi''$：

(1). 若 C_i 有 3个不同之 literal，則一樣為 ϕ''' 中之 C_i

(2). 若 C_i 有 2个不同之 literal，即：$C_i = (l_1 \vee l_2)$

則 ϕ''' 中之 C_i 為 $(l_1 \vee l_2 \vee p) \wedge (l_1 \vee l_2 \vee \neg p)$

$\because$ 無論 p 為 0 或 1，一定有其一 clause 等價 $(l_1 \vee l_2)$

而另一 clause 為 1 \#.

(3). 若 C_i 有 1个不同之 literal，即：$C_i = l$

則 ϕ''' 中之 C_i 為 $(l \vee p \vee q) \wedge (l \vee p \vee \neg q) \wedge (l \vee \neg p \vee q) \wedge (l \vee \neg p \vee \neg q)$

$\because$ 無論 p, q 為何，一定有其一 clause 等價 l

而其它 clause 為 1 \#.

而 ϕ''' 為 ϕ 对應之 3-CNF-SAT 之 instance

該転換為 polynomial-time

又 ϕ is satisfiable $\Leftrightarrow$ ϕ''' is satisfiable

故 3-SAT $\in$ NPC \#.