Propositional Logic

Definitional CNF
The **definitional CNF** of a formula is obtained in 2 steps:

1. Repeatedly replace a subformula $G$ of the form $\neg A$, $A \land B$ or $A \lor B$ by a new atom $A$ and conjoin $A \leftrightarrow G$. This replacement is not applied to the “definitions” $A \leftrightarrow G$ but only to the (remains of the) original formula.

2. Translate all the subformulas $A \leftrightarrow G$ into CNF.

**Example**

$\neg(A_1 \lor A_2) \land A_3$

$\Rightarrow$

$\neg A_4 \land A_3 \land (A_4 \leftrightarrow (A_1 \lor A_2))$

$\Rightarrow$

$A_5 \land A_3 \land (A_4 \leftrightarrow (A_1 \lor A_2)) \land (A_5 \leftrightarrow \neg A_4)$

$\Rightarrow$

$A_5 \land A_3 \land \text{CNF}(A_4 \leftrightarrow (A_1 \lor A_2)) \land \text{CNF}(A_5 \leftrightarrow \neg A_4)$
Definitional CNF: Complexity

Let the initial formula have size $n$.

1. Each replacement step increases the size of the formula by a constant.
   There are at most as many replacement steps as subformulas, linearly many.

2. The conversion of each $A \leftrightarrow G$ into CNF increases the size by a constant.
   There are only linearly many such subformulas.

Thus the definitional CNF has size $O(n)$. 
Definition
The notation $F[G/A]$ denotes the result of replacing all occurrences of the atom $A$ in $F$ by $G$. We pronounce it as “$F$ with $G$ for $A$”.

Example
$(A \land B)[(A \rightarrow B)/B] = (A \land (A \rightarrow B))$

Definition
The notation $\mathcal{A}[\nu/A]$ denotes a modified version of $\mathcal{A}$ that maps $A$ to $\nu$ and behaves like $\mathcal{A}$ otherwise:

$$(\mathcal{A}[\nu/A])(A_i) = \begin{cases} 
\nu & \text{if } A_i = A \\
\mathcal{A}(A_i) & \text{otherwise}
\end{cases}$$
Substitution Lemma

Lemma
\[ \mathcal{A}(F[G/A]) = (\mathcal{A}[\mathcal{A}(G)/A])(F) \]

Example
\[ \mathcal{A}((A_1 \land A_2)[G/A_2]) = (\mathcal{A}[\mathcal{A}(G)/A_2])(A_1 \land A_2) \]

Proof by structural induction on \( F \).
Definitional CNF: Correctness

Each replacement step produces an equisatisfiable formula:

**Lemma**

*Let* \( A \) *be an atom that does not occur in* \( G \).

*Then* \( F[G/A] \) *is equisatisfiable with* \( F \land (A \leftrightarrow G) \).

**Proof** If \( F[G/A] \) is satisfiable by some assignment \( \mathcal{A} \), then by the Substitution Lemma, \( \mathcal{A}' = \mathcal{A}[\mathcal{A}(G)/A] \) is a model of \( F \). Moreover \( \mathcal{A}' \models (A \leftrightarrow G) \) because \( \mathcal{A}'(A) = \mathcal{A}(G) \) and \( \mathcal{A}(G) = \mathcal{A}'(G) \) by the Coincidence Lemma (Exercise 1.2).

Thus \( F \land (A \leftrightarrow G) \) is satisfiable (by \( \mathcal{A}' \)).

Conversely we actually have \( F \land (A \leftrightarrow G) \models F[G/A] \).

Suppose \( \mathcal{A} \models F \land (A \leftrightarrow G) \). This implies \( \mathcal{A}(A) = \mathcal{A}(G) \).

Therefore \( \mathcal{A}[\mathcal{A}(G)/A] = \mathcal{A} \).

Thus \( \mathcal{A}(F[G/A]) = (\mathcal{A}[\mathcal{A}(G)/A])(F) = \mathcal{A}(F) = 1 \) by the Substitution Lemma.

Does \( F[G/A] \models F \land (A \leftrightarrow G) \) hold?
Summary

Theorem
For every formula $F$ of size $n$
there is an equisatisfiable CNF formula $G$ of size $O(n)$.

Similarly it can be shown:

Theorem
For every formula $F$ of size $n$
there is an equivalid DNF formula $G$ of size $O(n)$. 
Validity of CNF

Validity of formulas in CNF can be checked in linear time.

A formula in CNF is valid iff all its disjunctions are valid.

A disjunction is valid iff it contains both an atomic $A$ and $\neg A$ as literals.

Example

Valid: $(A \lor \neg A \lor B) \land (C \lor \neg C)$

Not valid: $(A \lor \neg A) \land (\neg A \lor C)$
Satisfiability of formulas in DNF can be checked in linear time.

A formula in DNF is satisfiable iff at least one of its conjunctions is satisfiable. A conjunction is satisfiable iff it does not contain both an atomic $A$ and $\neg A$ as literals.

Example

Satisfiable: $\neg B \land A \land B \lor \neg A \land C$

Unsatisfiable: $A \land \neg A \land B \lor C \land \neg C$
Satisfiability/validity of DNF and CNF

Theorem
Satisfiability of formulas in **CNF** is NP-complete.

Theorem
Validity of formulas in **DNF** is coNP-complete.