Algorithm for Clause Form:

(‘$A \to B$’ = rewrite A as B)

1. Convert wff to Prenex Normal Form:
   a) $(P \Leftrightarrow Q) \to ((P \Rightarrow Q) \wedge (Q \Rightarrow P))$
   b) $(P \Rightarrow Q) \to (\neg P \vee Q)$
   c) repeat: (i) $\neg\neg P \to P$
      (ii) $\neg(P \wedge Q) \to (\neg P \vee \neg Q)$
      (iii) $\neg(P \vee Q) \to (\neg P \wedge \neg Q)$
      (iv) $\neg\exists x[\alpha] \to \forall x[\neg\alpha]$
      (v) $\neg\forall x[\alpha] \to \exists x[\neg\alpha]$

      until: ‘$\neg$’ only applies to atomic wffs
   d) Rename vars s.t. vars bound by diff qfrs have unique names
   e) Move all qfrs left, w/o changing order

2. Convert PNF to Skolem NF:
   a) $\forall x_1 \ldots \forall x_n \exists y[\alpha(y)] \to \forall x_1 \ldots \forall x_n[\alpha(f(x_1, \ldots, x_n))]$

3. Convert SNF to Conjunctive NF:
   a) $\forall x[\alpha(x)] \to \alpha(x)$
   b) repeat: $(P \vee (Q \wedge R)) \to ((P \vee Q) \wedge (P \vee R))$

      until: wff is a conjn of disjns of literals

4. Convert CNF to Clause Form:
   a) $(P \vee Q) \to PQ$; each PQ is a clause
   b) $(P \wedge Q) \to \{P, Q\}$
   c) Rename vars s.t. each clause has diff vars