Some definitions:

- A *Boolean variable* x can take values 0,1.
- A *Boolean formula* is an expression constructed from Boolean variables and connectives, negation ($\neg$ or $\overline{\phi}$), disjunction ($\vee$) and conjunction ($\wedge$).
- A Boolean formula ϕ is *satisfiable* if there is a *truth assignment* $T : X \rightarrow \{0,1\}$ to the variables in ϕ such that $T(\phi) = 1$.

  For example, for $X = \{x_1, x_2, x_3\}$,

  $$\phi = (x_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (x_2 \vee x_3)$$

  is satisfiable, take $T(x_1) = T(x_2) = 0, T(x_3) = 1$ then $T(\phi) = 1$.
- A *literal* is a Boolean variable x or a negation of a Boolean variable$\bar{x}$.
- A *clause* is a disjunction (conjunction) of literals.
- A Boolean formula ϕ in *Conjunctive Normal Form (CNF)* is a conjunction of (disjunction) clauses, $\phi = \bigwedge_{i=1}^{m}(C_i)$, where each clause $C_i = \bigvee_{j=1}^{k_i}\{l_j\}$.

  For example, for $X = \{x_1, x_2, x_3\}$, a CNF formula is

  $$\phi = (x_1 \vee \bar{x}_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (x_2 \vee x_3)$$
- A Boolean formula ϕ in *Disjunctive Normal Form (DNF)* is expressed as a disjunction of (conjunction) clauses, $\phi = \bigvee_{i=1}^{m}(C_i)$, where each clause $C_i = \bigwedge_{j=1}^{k_i}\{l_j\}$.

Some computational problems:

- SAT: Given a boolean formula ϕ in CNF, is ϕ satisfiable?
- k-SAT: Given a boolean formula in CNF where each clause has exactly k literals, is ϕ satisfiable?
- DNF-SAT: Given a boolean formula ϕ in DNF, is ϕ satisfiable?
- k-DNF-SAT: Given a boolean formula in DNF where each clause has exactly k literals, is ϕ satisfiable?

8. (DNF vs CNF)

   (a) Show that DNF-SAT and CNF-SAT belong to NP, and that DNF-SAT has a polynomial time algorithm.

   (b) Show that a CNF formula can be converted into an equivalent DNF formula on the same variables. How much time takes this computation?

9. Show that k-SAT belongs to NP, for each $k \in \mathbb{N}$.

10. Recall that the 3-SAT problem is a restricted version of SAT where each clause has exactly 3 literal. Let $\phi = \{C_i\}_{i=1}^{m}$ be a CNF formula on a set X of variables. let z_i be the literal x_i or $\bar{x}_i$. We construct a formula $\phi' = f(\phi)$ on a set X' of variables ($X \subseteq X'$) having all clauses with 3 literals.

    For each clause in ϕ, f determines a set of clauses to be included in ϕ' replacing the clause in ϕ. We add variables when needed. The replacements depend on the size k of clause C_j.

    - If $k = 1$, $C_j = z$, we add variables $\{y_{j1}, y_{j2}\}$ and clauses
    $$C'_j = \{(z \vee y_{j1} \vee y_{j2}), (z \vee \bar{y}_{j1} \vee y_{j2}), (z \vee y_{j1} \vee \bar{y}_{j2}), (z \vee \bar{y}_{j1} \vee \bar{y}_{j2})\}.$$
    - If $k = 2$, $C_j = z_1 \vee z_2$, we add variable y_j and clauses
    $$C'_j = \{(z_1 \vee z_2 \vee y), (z_1 \vee z_2 \vee \bar{y})\}.$$
    - If $k = 3$, we add $C_j = (z_1 \vee z_2 \vee z_3)$ to ϕ'.
    - If $k > 3$, $C_j = (z_1 \vee z_2 \vee \cdots \vee z_k)$, add variables $\{y_{j1}, y_{j2}, \ldots, y_{jk-3}\}$ and the clauses
    $$C'_j = \{(z_1 \vee z_2 \vee y_{j1}), (\bar{y}_{j1} \vee z_3 \vee y_{j2}), \ldots, (\bar{y}_{jk-3} \vee z_{k-1} \vee z_k)\}$$

    Does this construction provide a reduction from SAT to 3-SAT? Can f be computed in polynomial time?

11. Consider a 2-SAT instance ϕ. Define an associated directed graph G_ϕ having one vertex for each literal appearing in ϕ. For each clause $(\ell_1 \vee \ell_2)$ in ϕ, add the edges $(\overline{\ell}_1, \ell_2)$ and $(\overline{\ell}_2, \ell_1)$ to G_ϕ. Show that ϕ is satisfiable iff there is no strongly connectyed component containing both x and $\overline{x}$ in G_ϕ.

    Can you use this result to show that 2-SAT belongs to P?