Collaborating on homework is encouraged, but you must write your own solutions in your own words and list your collaborators. Copying someone else's solution will be considered plagiarism and may result in failing the whole course.

Please answer clearly and concisely. Explain your answers. Unexplained answers will get lower scores or even no credits.

(1) (8 points) For each of the following problems, say whether it is decidable or not. Justify your answer by describing an appropriate Turing machine, or by reducing from $\mathrm{ALL}_{\mathrm{CFG}}$ which was shown undecidable in class. Assume that the alphabet of CFG G contains the symbol $\mathtt{a}$.

(a) $L_1 = \{\langle G\rangle \mid \text{CFG } G \text{ generates at least one string that starts with } \mathtt{a}\}$.

(b) $L_2 = \{\langle G\rangle \mid \text{CFG } G \text{ generates all strings that start with } \mathtt{a}\}$.

(2) (8 points) For each of the following problems, show that it is NP-complete (namely, (1) it is in NP and (2) some NP-complete language reduces to it.) When showing NP-completeness, you can start from any language that was shown NP-complete in class or tutorial.

(a) $L_1 = \{\langle G, k\rangle \mid G \text{ is a graph that contains (at least) two cliques, each of size } k\}$

(b) $L_2 = \{\langle \varphi\rangle \mid \varphi \text{ is a satisfiable } \leqslant\text{3CNF formula where}$
each literal appears in at most 3 places$\}$.
(In a $\leqslant$3CNF formula, each clause contains *at most* 3 literals.)

Hint: Reduce from 3SAT. Make several copies of each variable in the 3SAT formula, and write clauses that require all copies of the same variable to be equal.

(3) (12 points) Throughout the semester, we looked at various models of computation and we came up with the following "hierarchy" of classes of languages:

$$\text{regular} \subseteq \text{context-free} \subseteq \text{P} \subseteq \text{NP} \qquad \text{decidable} \subseteq \text{recognizable}$$

We also gave examples showing that the containments are strict (e.g., a context-free language that is not regular), except for the containment $\text{P} \subseteq \text{NP}$, which is not known to be strict.

There is one gap in this picture between NP languages and decidable languages. In this problem you will fill this gap.

(a) Show that 3SAT is decidable, and the decider has running time $2^{O(n)}$. (Unlike a *verifier* for 3SAT which is given a 3CNF formula φ together with a potential satisfying assignment for φ, a *decider* for 3SAT is only given a 3CNF formula but not an assignment for it.)

(b) Argue that for every NP-language L there is a constant c such that L is decidable in time $2^{O(n^c)}$. (Use the Cook–Levin Theorem.)

(c) Let L' be the following language:

$$L' = \{\langle M, w\rangle \mid \text{Turing machine } M \text{ does not accept input } \langle M, w\rangle \text{ within } 2^{2^{|w|}} \text{ steps}\}.$$

It is not hard to see that L' can be decided in time $O(2^{2^n})$.
Show that L' cannot be decided in time $2^{O(n^c)}$ for any constant c, and therefore it is not in NP.
Hint: Assume that L' can be decided by a Turing machine D in time $2^{O(n^c)}$. What happens when D is given input $\langle D, w\rangle$, where w is a sufficiently long string?

(4) (12 points) A *heuristic* is an algorithm that often works well in practice, but it may not always produce the correct answer. In this problem, we will consider a heuristic for 3SAT.

Let φ be a CNF formula and x a literal in φ. Suppose we set x to TRUE. The *reduced form* of φ is the formula obtained by discarding all clauses of φ that contain x and removing $\overline{x}$ from all the other clauses. For example, if $\varphi = (x_1 \vee x_2) \wedge (\overline{x}_1 \vee x_3) \wedge (x_2 \vee x_3)$, then setting $x_1 = \text{TRUE}$ gives the reduced form $x_3 \wedge (x_2 \vee x_3)$, while setting $\overline{x}_1 = \text{TRUE}$ gives the reduced form $x_2 \wedge (x_2 \vee x_3)$. Consider the following heuristic H for 3SAT:

On input $\langle\varphi\rangle$, where φ is a 3CNF formula with n variables:
  For $i := 1$ to n, repeat the following:
    If x_i appears in φ more often than $\overline{x_i}$, set $x_i = true$.
    Otherwise, set $\overline{x}_i = true$.
    Replace φ with its reduced form. If φ contains an empty clause, **reject**.
  accept.

(a) Show that H runs in polynomial time.

(b) Show that if H accepts φ, then $\varphi \in 3SAT$.

(c) Show that it is possible that H rejects φ, even though $\varphi \in 3SAT$.