CSCI 3130: Formal Languages and Automata Theory — Homework 6

The Chinese University of Hong Kong, Fall 2011 — due Thursday 8 December

---

Each of the problems is worth 10 points. Write your name, student ID, and your TA's name on the solution sheet

Please write your solutions clearly and concisely. If you do not explain your answer you will be given no credit. You are encouraged to collaborate on the homework, but you must write your own solutions and list your collaborators on your solution sheet. Copying someone else's solution or solutions found in external references will be considered plagiarism and may result in failing the whole course.

Please turn in the solutions by 11.59pm on Thursday 8 December. The homework should be dropped into the box labeled CSCI 3130 on the 9th floor of SHB. Late homeworks will not be accepted.

Problem 1

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 ALL_{CFG} which was shown undecidable in class. Assume that the alphabet of CFG G contains the symbol a.

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

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

Problem 2

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) $L_1 = \{\langle \phi \rangle : \phi \text{ is a boolean formula that has at least two satisfying assignments}\}$.

(b) $L_2 = \{\langle G, k \rangle : G \text{ is a graph that contains a clique of size } k \textbf{ or } \text{an independent set of size } k\}$.

Problem 3

Throughout the semester, we looked at various models of computation and we came up with the following hierarchy:

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

We also gave examples showing that the containments are strict (e.g., a context-free language that is not regular), except for the containment P $\subseteq$ 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 ϕ together with a potential satisfying assignment for ϕ, a *decider* for 3SAT is only given a 3CNF but not an assignment for it.)

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

(c) Let D be the following Turing Machine:

> D: On input $\langle M, w\rangle$, where M is a Turing Machine,
> Run M on input $\langle M, w\rangle$ for at most $2^{2^{|w|}}$ steps.
> If M accepts $\langle M, w\rangle$ within this many steps, *reject*;
> Otherwise (if M rejects or hasn't halted), *accept*.

Show that the language decided by D cannot be decided in time $2^{O(n^c)}$ for any constant c, and therefore it is not in NP.

Hint: Assume that D can be decided in time $2^{O(n^c)}$. What happens when D is given input $\langle D, w\rangle$, where w is a sufficiently long string?

Problem 4

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 $\phi = (x_1 \vee x_2) \wedge (\overline{x}_1 \vee x_3) \wedge (x_2 \vee x_3)$, 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 $\langle \phi \rangle$, 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 $\phi \in 3SAT$.

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