Instructions

- Classroom Problems C7.1–C7.3 will be discussed and solved at the tutorial session on Wed 4 Mar, 14–16, Room T4 (A238). No credit is given for these problems.
- Homework Problems H7.1–H7.3 you should solve on your own, and submit your solutions via the MyCourses interface by the deadline of Tue 10 Mar, 23:59, These problems will be individually graded on a scale of 0–2 points per problem.
- In preparing your solutions to the Homework Problems:
  1. Justify your solutions, be precise, and provide sufficient detail so that it is easy to follow your reasoning.
  2. Submit your solutions as an easily readable, single pdf file, which is either typeset or written in full sentences and clean handwriting.
  3. **[Code of Conduct]** You can discuss the problems with your colleagues and the course's teaching staff, but you must write the presentations of your solutions *independently* and *individually*, without any notes from such discussions.

Classroom Problems

C7.1 Show that if a language L has (nonuniform) polynomial-size circuits, then there is a polynomially balanced, polynomial-time decidable binary relation R and a sequence of *advice strings* $w_0, w_1, w_2, \ldots$, such that for each $n \geq 0$:

$$x \in L \;\equiv\; R(x, w_n) \quad \text{holds for all } x \text{ with } |x| \leq n.$$

(*Hint:* Concatenate the circuits for dealing with inputs up to each given length n.) Because of this characterisation, the family of languages with polynomial-size circuits is commonly denoted $\mathbf{P/poly}$, suggesting "$\mathbf{P}$ with polynomial-length (nonuniform) advice".

C7.2 In Problem C7.1, the advice strings were essentially of polynomial length, because the relation R was constrained to be polynomially balanced. More generally, we can consider sequences of advice strings whose length is bounded by any given function $f(n)$, or family of functions $\mathbf{f}$, in order to define an advice complexity class $\mathbf{P/f}$. Here we consider the class of languages that can be decided with *logarithmic* (nonuniform) advice: $\mathbf{P/log}$, where $\mathbf{log} = \{c \log n \mid c > 0\}$.

(a) Show that if SAT $\in$ **P/log** then SAT $\in$ **P** and hence **P** = **NP**. (*Hint:* Run through all possible advice strings, and find the correct one using the fact that, given a Boolean formula ϕ, it is easy to compute[1] two formulas ϕ_0, ϕ_1 such that $|\phi_0|, |\phi_1| < |\phi|$ and $\phi \in$ SAT if and only if $\phi_0 \in$ SAT or $\phi_1 \in$ SAT.)

(b) Show that **P/log** contains undecidable languages.

C7.3 (a) Show that if $L_1, L_2 \in$ **RP** then $L_1 \cap L_2 \in$ **RP**. (*Hint:* Recall that the probability of acceptance need not be $\frac{1}{2}$.)

(b) Show that **PP** $\subseteq$ **PSPACE** by first arguing that MAJSAT is **PP**-complete, and then describing an algorithm that solves MAJSAT using polynomial space.

Homework Problems

H7.1 Show that if $L \in$ **ZPP** = **RP**$\cap$**coRP**, then there is a polynomial-time probabilistic Turing machine M such that M always outputs 0, 1 or ?, and for $x \in \{0,1\}^*$ we have

- if $x \in L$, then $\Pr[M(x) = 0] = 0$,
- if $x \notin L$, then $\Pr[M(x) = 1] = 0$, and
- $\Pr[M(x) =?] \leq 1/3$.

Also establish as a corollary that if $L \in$ **ZPP**, then there is a PTM M' with a polynomial *expected* running time, such that whenever M' halts on input $x \in \{0,1\}^*$, then $M'(x) = 1$ if and only if $x \in L$. (2 points)

H7.2 Let $L \subseteq \{0,1\}^*$ be a language and assume that for some constant $c > 1$, there is a polynomial-time PTM M such that for every $x \in \{0,1\}^*$, we have:

- if $x \in L$, then $\Pr[M(x) = 1] \geq |x|^{-c}$, and
- if $x \notin L$, then $\Pr[M(x) = 1] = 0$.

Prove that then for every constant $d > 0$, there is a polynomial-time PTM M' such that for every $x \in \{0,1\}^*$, we have

- if $x \in L$, then $\Pr[M'(x) = 1] \geq 1 - 2^{-|x|^d}$, and
- if $x \notin L$, then $\Pr[M'(x) = 1] = 0$.

(*Hint:* Simulate the machine M on input $x \in L$ for k times for sufficiently large k. The results of the different runs are independent, so you can directly compute the probability that all the runs fail to accept.) (2 points)

[1] By fixing the value of the first available free variable and performing logical simplifications if possible. This property of SAT is called *self-reducibility*.

H7.3 In this exercise we derive the intriguing result that if SAT $\in$ **P/poly** then $\Sigma_2^p = \Pi_2^p$, i.e. the polynomial time hierarchy collapses to its second level [R. Karp & R. Lipton 1982].

(i) Assume that SAT $\in$ **P/poly**, and let R be the corresponding "advice-augmented" decision method described in Problem C7.1. Consider the set of "correct SAT-witnesses" $\mathrm{OK}_{\mathrm{SAT}}$ defined as:

$$\langle w, 1^n \rangle \in \mathrm{OK}_{\mathrm{SAT}} \quad \equiv \quad [\mathrm{SAT}(\phi) \leftrightarrow R(\phi, w) \text{ for all } \phi \text{ with } |\phi| \leq n].$$

Show that $\mathrm{OK}_{\mathrm{SAT}} \in \Pi_1^p =$ **coNP**. (*Hint:* Apply the self-reducibility property of SAT discussed in Problem C7.2.)

(ii) Consider now the language $\overline{\mathrm{QSAT}}_2$, which is Π_2^p-complete by Problem H6.3(ii). Show that if SAT $\in$ **P/poly**, then $\overline{\mathrm{QSAT}}_2 \in \Sigma_2^p$, thus establishing that $\Pi_2^p = \Sigma_2^p$. (*Hint:* Observe that for a formula $\psi = \psi(Y, Z)$,

$$\psi \in \overline{\mathrm{QSAT}}_2 \quad \equiv \quad \forall Y [\neg \psi^Y \in \mathrm{SAT}],$$

where $\neg\psi^Y$ is $\neg\psi$ with the Y-variables set to the given values. Use an existential quantifier to "guess" a SAT-advice string w, the set $\mathrm{OK}_{\mathrm{SAT}}$ to verify its correctness, and replace the condition $[\neg\psi^X \in \mathrm{SAT}]$ by the corresponding w-augmented deterministic decision method.)

(2 points)