Complexity Theory
Lent 2004
Suggested Exercises 3

1. Show that a language L is in $\mathsf{co-NP}$ if, and only if, there is a nondeterministic Turing machine M and a polynomial p such that M halts in time $p(n)$ for all inputs of length x, and L is exactly the set of strings x such that *all* computations of M on input x end in an accepting state.

2. Define a *strong* nondeterministic Turing machine as one where each computation has three possible outcomes: accept, reject or maybe. If M is such a machine, we say that it accepts L, if for every $x \in L$, every computation path of M on x ends in either accept or maybe, with at least one accept *and* for $x \notin L$, every computation path of M on x ends in reject or maybe, with at least one reject.

   Show that if L is decided by a strong nondeterministic Turing machine running in polynomial time, then $L \in \mathsf{NP} \cap \mathsf{co\text{-}NP}$.

3. Consider the algorithm presented in the lecture which establishes that $\mathsf{Reachability}$ is in $\mathsf{SPACE}((\log n)^2)$. What is the time complexity of this algorithm?

   Can you generalise the time bound to the entire complexity class? That is, give a class of functions F, such that

$$\mathsf{SPACE}((\log n)^2) \subseteq \bigcup_{f \in F} \mathsf{TIME}(f)$$

4. Show that, for every nondeterministic machine M which uses $O(\log n)$ work space, there is a machine R with three tapes (`input`, `work` and `output`) which works as follows. On input x, R produces on its output tape a description of the configuration graph for M, x, and R uses $O(\log |x|)$ space on its work tape.

   Explain why this means that if $\mathsf{Reachability}$ is in L, then $\mathsf{L} = \mathsf{NL}$.

5. Consider the language L in the alphabet $\{a, b\}$ given by $L = \{a^n b^n \mid n \in \mathbb{N}\}$. $L \notin \mathsf{SPACE}(c)$ for any constant c. Why?