Complexity Theory
Lent 2003
Suggested Exercises 2

1. In the lecture, a proof of the Time Hierarchy Theorem was sketched. Give a similar argument for the following Space Hierarchy Theorem:

> **Space Hierarchy.** For every constructible function f, there is a language in $\mathsf{SPACE}(f(n) \cdot \log f(n))$ that is not in $\mathsf{SPACE}(f(n))$.

Could you replace the factor of $\log f(n)$ in this statement with something even smaller?

2. 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

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

3. Show that, if $\mathsf{SPACE}((\log n)^2) \subseteq \mathsf{P}$, then $\mathsf{L} \neq \mathsf{P}$. (Hint: use the Space Hierarchy Theorem from Exercise 1.)

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. 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.