Complexity theory
Department of Mathematics and Statistics
Fall 2017
Exercise set 11

Read chapters 4.1 – 4.2 of the book.

Exercise 1. Show that $\mathbf{NP} \subseteq \mathbf{PSPACE}$.

Exercise 2. The configurations we looked at in class assume that the machine never looks at the input tape beyond the input (or input+1 to detect its ending). Show that we can make this assumption without loss of generality, i.e., if S is space-constructible and M is a Turing machine running in space $S(n)$, then there is another Turing machine M' (with k extra work tapes for some constant k depending on M) running in space $S(n)$ such that it never moves more than one step beyond the input on the input tape. Hint: show that going beyond the input can be simulated with a clock.

Exercise 3. Prove the existence of a universal Turing machine for space-bounded computation, i.e., prove that there exists a Turing machine $\mathcal{SU}$ such that for every string α, and input x, if M_α halts on x before using t cells of its work tapes, then $\mathcal{SU}(\alpha, t, x) = M_\alpha(x)$, and, moreover, $\mathcal{SU}$ uses at most Ct cells of its work tapes, where C is a constant depending only on M_α.

Exercise 4. Prove the *Space Hierarchy Theorem*: If f, g are space-constructible functions satisfying $f(n) = o(g(n))$, then

$$\mathbf{SPACE}(f(n)) \subsetneq \mathbf{SPACE}(g(n)).$$

Exercise 5. Define **polyL** to be $\bigcup_{c>0} \mathbf{SPACE}(\log^c n)$. Steve's Class $\texttt{SC}$ (named in honour of Steve Cook) is defined to be the set of languages that can be decided by deterministic machines that run in polynomial time and $\log^c n$ space for some $c > 0$.

We will see that $\overline{\texttt{PATH}} \in \mathbf{NL}$, where

$$\texttt{PATH} = \{(G, s, t) : G \text{ is a directed graph in which there is a path from } s \text{ to } t\}.$$

It is an open problem whether $\texttt{PATH} \in \texttt{SC}$. Who does Savitch's Theorem not resolve this question?

Is $\texttt{SC}$ the same as $\mathbf{polyL} \cap \mathbf{P}$?