COMP4141 Homework 6 (2016)
Time Complexity

Due date: Wed 27-04-2016, 18:05

Exercise 1 Which of the following languages are decidable? Give either a proof of undecidability or describe an algorithm that decides the language.

1. $\{\langle M, w\rangle \mid M$ is a TM that halts in at most $2^{|w|}$ steps on input $w\}$
2. $\{\langle M\rangle \mid M$ is a TM with $L(M) \in \mathbf{TIME}(2^n)$ $\}$

Exercise 2 Show that **P** is closed under concatenation. More precisely, suppose that $L_1 \in \mathbf{TIME}(n^k)$ and $L_2 \in \mathbf{TIME}(n^l)$, for constants k and l, then there exists a constant m such that $L_1L_2 \in \mathbf{TIME}(n^m)$. What is the lowest value of m for which you can prove this?

Exercise 3 In this exercise we consider another machine model that turns out not to make a difference as far as the complexity class **P** is concerned, and which brings us quite close to a model resembling standard computers.

Define a *deterministic infinite store Turing machine* to be like a deterministic 2-tape Turing machine with an infinite memory attached. Rather than having to move a head over the memory, the machine will be able to directly access a memory location by giving its position as binary number. Tape 1 is an input and work tape, as usual. Tape 2 has tape alphabet $\{0, 1\}$, with the contents of the tape representing binary numbers. A state of the memory is a function *store* that associates to each natural number n (represented in binary) a symbol $store(n)$ from the Tape 1 tape alphabet. Initially, $store(n)$ is the blank symbol for all n.

A transition of the machine is determined based on the current control state, and the symbols being read on the two tapes. The action in response to these inputs is one of the the following three possibilities:

- a standard transition for a 2-tape Turing machine, that writes a symbol to the current head locations and moves the heads left or right
- a memory *read* operation that treats the string on Tape 2 as a number n in binary and writes symbol $store(n)$ to the current head location on Tape 1. (Assume that the contents of Tape 2 is always a single contiguous block of 0's and 1's surrounded by blanks, so that it can always be interpreted as a binary number.)
- a memory *write* operation that treats the string on Tape 2 as a number n in binary and modifies *store* so that $store(n)$ contains the symbol at the current head location on Tape 1.

Each of these operations is treated as taking a single unit of time for the purposes of complexity analysis. Note that to construct a binary number of length k on tape 2, the machine still needs at least k steps of computation, but the contents of this tape are understood in one large gulp by the read and write operations.

In no more than 1 page, explain why any language that can be decided in polynomial time on a deterministic infinite store Turing machine can be decided in polynomial time on a standard multi-tape Turing machine. Describe any data representations that the standard machine uses.