**COMP2270/6270 – Theory of Computation**
**Eleventh Week**

**School of Electrical Engineering & Computing**
**The University of Newcastle**

**Exercise 1)** Consider the language $L = \{<M> : M$ accepts at least two strings$\}$.

a) Describe in clear English a Turing machine $M$ that semidecides $L$.
b) Suppose we changed the definition of $L$ just a bit. We now consider:

    $L' = \{<M> : M$ accepts *exactly* 2 strings>.

    Can you tweak the Turing machine you described in part a to semidecide $L'$?

**Exercise 2)** Consider the language $L = \{<M> : M$ accepts the binary encodings of the first three prime numbers$\}$.

a) Describe in clear English a Turing machine $M$ that semidecides $L$.
b) Suppose (contrary to fact, as established by Theorem 19.2) that there were a Turing machine *Oracle* that decided H. Using it, describe in clear English a Turing machine $M$ that decides $L$.

**Exercise 3)** Show that the set D (the decidable languages) is closed under:

a) Union
b) Concatenation
c) Kleene star
d) Reverse
e) Intersection

**Exercise 4)** If $L_1$ and $L_3$ are in D and $L_1 \subseteq L_2 \subseteq L_3$, what can we say about whether $L_2$ is in D?

**Exercise 5)** Let $L_1$ and $L_2$ be any two decidable languages. State and prove your answer to each of the following questions:

a) Is it necessarily true that $L_1 - L_2$ is decidable?

b) Is it possible that $L_1 \cup L_2$ is regular?

**Exercise 6)** Construct a standard one-tape Turing machine $M$ to enumerate the language $A^nB^n$. Assume that $M$ starts with its tape equal to $\underline{\square}$. Also assume the existence of the printing subroutine $P$, defined in Section 20.5.1.

**Exercise 7)** If $w$ is an element of $\{0, 1\}^*$, let $\neg w$ be the string that is derived from $w$ by replacing every `0` by `1` and every `1` by `0`. So, for example, $\neg 011 = 100$. Consider an infinite sequence $S$ defined as follows:

$$S_0 = 0.$$
$$S_{n+1} = S_n \neg S_n.$$

The first several elements of $S$ are `0`, `01`, `0110`, `01101001`, `0110100110010110`. Describe a Turing machine $M$ to output $S$. Assume that $M$ starts with its tape equal to $\underline{\square}$. Also assume the existence of the printing subroutine $P$, defined in Section 20.5.1, but now with one small change: if $M$ is a multitape machine, $P$ will output the value of tape 1. (Hint: use two tapes.)

REFERENCES

[1] Elaine Rich, Automata Computatibility and Complexity: Theory and Applications, Pearson, Prentice Hall, 2008.