# Homework 7

## CptS 317, Spring 2021

## <u>Due Date:</u> April 23, 2021 by 11:59 PM Pacific.

## To be submitted on Canvas.

---

This homework has eight problems, which are **not** equally weighted. The weight (points) of each problem is indicated next to its description.

---

1. (15 pts) Show that the collection of decidable languages is closed under the operation of:

   a) Complementation

   b) Concatenation

   c) Star

2. (15 pts) Let $ALL_{\text{DFA}} = \{\langle A \rangle |\ A \text{ is a DFA and } L(A) = \Sigma^*\}$. Show that $ALL_{\text{DFA}}$ is decidable.

3. (15 pts) Let $\mathcal{B}$ be the set of all infinite sequences over $\{0, 1\}$. Show that $\mathcal{B}$ is uncountable using a proof by diagonalization.

4. (15 pts) Let $A$ and $B$ be two disjoint languages. Say that $C$ **separates** $A$ and $B$ if $A \subseteq C$ and $B \subseteq \overline{C}$. Show that any two disjoint co-Turing-recognizable languages are separable by some decidable language.

5. (15 pts) A **useless state** in a pushdown automaton is never entered on any input string. Consider the problem of determining whether a pushdown automaton has any useless states. Formulate this problem as a language and show that it is decidable.

6. (15 pts) If $A \leq_M B$ and $B$ is a regular language, does that imply that $A$ is a regular language? Why or why not?

7. (10 pts) Use Rice's Theorem[1] to prove the undecidability of the following languages.

   a) $\{\langle M \rangle | M \text{ is a TM and } L(M) \text{ is an infinite language}\}$.

   b) $ALL_{\text{TM}} = \{\langle M \rangle | M \text{ is a TM and } L(M) = \Sigma^*\}$.

8. (**Extra credit**, 10 pts) Cellular automata are a type of automata defined by a grid of cells, each of which may exist in any of some finite number of states; and a set of rules to transform the grid's state based on its previous state.

   Conway's Game of Life is an example of a cellular automaton. It uses a 2D grid of square cells, each of which is either "live" or "dead". The rules for computing the next version of a grid based on the previous grid are defined based on the neighbors of each cell (note: on a square grid, a cell has eight neighbors, which includes the diagonals):

   (a) Any live cell with fewer than two live neighbours dies, as if by underpopulation.

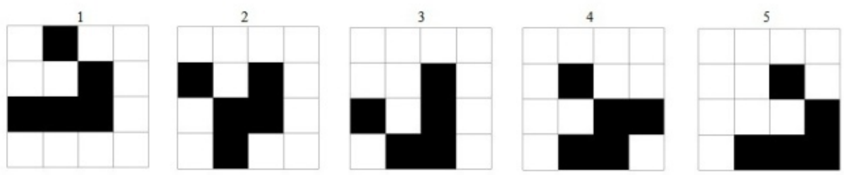
   (b) Any live cell with two or three live neighbours lives on to the next generation.

   (c) Any live cell with more than three live neighbours dies, as if by overpopulation.

   (d) Any dead cell with exactly three live neighbours becomes a live cell, as if by reproduction.

[1] Rice's theorem states that every non-trivial, semantic property of a Turing machine is undecidable. Here, a semantic property is one that refers to a TM's behavior (e.g. does it halt), and a non-trivial property is one that is neither universally true nor universally false for computable functions

In Conway's Game of Life, a number of patterns have been discovered whereby a group of live cells will, after some number of iterations, appear in the same configuration as they were initially (although perhaps in a different location). These patterns are known as "gliders", named after the first such pattern discovered, which is depicted below (note: the live cells are colored black, while the dead cells are colored white):




It has also been discovered that it is possible to simulate a Turing Machine in Conway's Game of Life. A paper describing this process can be found here. With this information in mind, is it possible for a Turing Machine to decide if, given the positions of every live cell in an instance of Conway's Game of Life, those cells comprise a glider? If this problem is undecidable, how might it be reduced to an existing undecidable problem?