Homework 8

With this homework, we will practice constructing Turing machines and also deepen our understanding of automata by solving a number of challenging problems. To make these problems more approachable, there are hints included at the bottom of the page.

1. Give transition state diagrams for Turing machines recognizing the following languages:

- **a.** binary strings in which every 0 is immediately followed by a 1;
- **b.** binary strings of the form 0^n10^n, where $n \geq 0$;
- **c.** binary strings with equally many 0s and 1s.

2. Let $\Sigma = \{0,1,2, \ldots ,9\}$ be the decimal alphabet. For a fixed positive integer k, define $A_k \subseteq \Sigma^*$ to be the language of positive integer multiples of k. For example, $A_3 = \{3, 6, 9, 12, 15, 18, \ldots\}$ and $A_{11} = \{11, 22, 33, 44, 55, 66, \ldots\}$. Prove that A_k is regular for every k.

3. Let N be a given NFA over an alphabet Σ. Consider the following languages:

$A = \{w \in \Sigma^*$: **at least one** computation of N on w results in acceptance$\}$,
$B = \{w \in \Sigma^*$: **every** computation of N on w results in acceptance$\}$.

In general, A and B need not be equal. We already know from lecture that A is regular. Prove that B is regular as well.

1.63

- **a.** Let A be an infinite regular language. Prove that A can be split into two infinite disjoint regular subsets.
- **b.** Let B and D be two languages. Write $B \Subset D$ if $B \subseteq D$ and D contains infinitely many strings that are not in B. Show that if B and D are two regular languages where $B \Subset D$, then we can find a regular language C where $B \Subset C \Subset D$.

1.64 Let N be an NFA with k states that recognizes some language A.

- **a.** Show that if A is nonempty, A contains some string of length at most k.
- **b.** Show, by giving an example, that part (a) is not necessarily true if you replace both A's by $\overline{A}$.
- **c.** Show that if $\overline{A}$ is nonempty, $\overline{A}$ contains some string of length at most 2^k.

Hints

2 A finite automaton obviously can't memorize the entire integer, but luckily it doesn't need to! Since we are interested in divisibility by k, it's OK to reduce intermediate computations modulo k.

3 Prove regularity for $\overline{B}$ and then invoke the closure properties.

1.63(a) To get you started on (a), use the pumping mechanism of the pumping lemma to construct an infinite subset of strings in A.

1.63(b) Derive part (b) from (a) by considering the language $A = D \setminus B$.

1.64(a) Fix any string w accepted by A. If w has length at most k, we are done. If w has length greater than k, show that we can obtain from w a shorter string that is still accepted by A.

1.64(c) Obtain from A a DFA of size 2^k for the *complementary* language, and apply part (a).