Fall 2024 COT 4210 Homework #4, 5: Turing Machines, Decidability, Halting Problem

1) Give the sequence of configurations that the Turing machine M_2 in chapter 3 that decides membership in the language L = $\{0^{2^n} \mid n \geq 0\}$ of the text (and shown in class) goes through while reading in the three following strings:

a) 00
b) 00000
c) 0000000000

2) Go to this website: https://morphett.info/turing/turing.html, which is linked off the course web page and create a machine that accepts the following language over the alphabet {0, 1}:

L = { $0^n1^n0^n$ | n > 0 }

Note that epsilon is not a part of the language. The shortest string that belongs to L is 010.

Just copy the exact text of your Turing Machine code into your homework submission and briefly explain your design conceptually.

3) Consider a Turning machine Model, the 23TM which is like a regular Turing Machine except that the transition function has the form:

$$\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{2L, 3R\}$$

If δ(q, a) = (r, b, 2L), when the machine is in state q reading an a, the machine's head moves to the left by 2 squares (or goes all the way to the left end of the time if this isn't possible.) Similarly, if δ(q, a) = (r, b, 3R), when the machine is in state q reading an a, the machine's head moves to the right by 3 squares. Show that this model, the 23TM of a Turing machine is equivalent to the original model. Namely, show that any regular TM can be emulated with a 23TM AND also show that a 23TM can emulate any regular machine.

4) Let SS_{DFA} = { <A, B> | A and B are DFAs such that L(A) ⊆ L(B)}. Show that SS_{DFA} is decidable. You may use any of the results from class of other related decidable languages.

5) Let a 2-PDA be a pushdown automata with access to 2 stacks. (In each transition, we can read the top of both stacks and push something on top of both stacks, if we choose.)

a) Give an example of a language that is NOT context free that can be accepted by a 2-PDA. Briefly describe in words how this 2-PDA would operate to accept that language.

b) Show that a standard Turing Machine can be implemented using a 2-PDA.

6) Prove that the number of subsets of the positive integers is not countable.

7) Let $LONG_{CFG}$ = {<G, n> | G_1 is a CFG such that no strings in L(G) have n or fewer characters.} Determine, with proof, whether $LONG_{CFG}$ is decidable or undecidable.

8) Let L = { <M, n> | M is a Turing Machine such that L(M) contains n or fewer strings}. Prove that L is undecidable via a reduction from $HALT_{TM}$ to L. (Thus, using a decider to L, show how to build a decider for $HALT_{TM}$.)