CSE 431 Fall 2019
Assignment #5

Due: Tuesday, November 12, 2019, 8:59 PM

Reading assignment: Read Chapter 7 of Sipser's text up to the start of the proof of the Cook-Levin Theorem. Then read section 9.3 which gives the proof of the Cook-Levin Theorem that we will do.

Problems:

1. (20 points) Prove that EQ_{DFA} is in P.

2. (20 points) Prove that P is closed under intersection, concatenation, and star. (You previously proved that the sets of decidable and Turing-recognizable languages are closed under these same operations.) Hint: Use ideas similar to those for the CKY algorithm when needed.

3. (20 points) Define GRAPH-HOMOMORPHISM=$\{\langle G, H\rangle \mid G$ and H are directed graphs and there is a mapping φ from the vertices of G to the vertices of H such that:

   - (u, v) is an edge in G if and only if $(\varphi(u), \varphi(v))$ is an edge in H and
   - for every vertex w in H there is a vertex v in G such that $\varphi(v) = w\}$.

   Prove that GRAPH-HOMOMORPHISM is in NP.

4. (20 points) Prove that NP is closed under intersection, concatenation, and star.

5. (20 points) Prove that A_{TM} is NP-hard.

6. (Extra Credit) In this question you will show that if an ordinary 1-tape TM M has running time $o(n \log n)$ then $L(M)$ must be regular.

   A *crossing-sequence* is the sequence of states on which, and directions from which, a boundary between two cells is crossed during the course of a computation.

   (a) Show that if the lengths of all the crossing sequences for a TM are bounded by some constant k (independent of the input length) then $L(M)$ is regular. Do this by building an NFA N to recognize $L(M)$.

   (b) Use a pigeonhole argument to argue that for any TM running in $o(n \log n)$ time on any sufficiently long input, there must exist two different cell boundaries for cells that originally contained the input that have precisely the same crossing sequence in the computation on that input.

(c) Show that if a 1-tape TM M has crossing sequences of arbitrarily large size then it cannot run in $o(n \log n)$ time. To do this, consider a minimal-length string that produces a long crossing sequence when M is run on it and use part (b) to derive a contradiction by splicing out a piece of the input string using the repeated crossing sequence.

(d) Finally, put the pieces together to produce the claimed result.

7. (Extra Credit) For a PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$ we say that a string $\alpha \in \Gamma^*$ is a *possible stack of* M if there is some input and some choice of moves of M such that α appears as M's stack contents during its computation. Prove that the languge $L \subseteq \Gamma^*$ of possible stacks is regular. (This fact is actually important for certain software verification systems since it allows one to consider the set of possible call stacks using only a finite state machine.)