Name________

**Homework 8**

**Worked with: ____________**

**Consulted: _______________**

**#1. Show the following are decidable or undecidable. Show why.**

a) **L = { (M,w) | M is a dfa and on input *w* visits each one of its states}**

**decidable**

b) **L = { (M,w) | M is a TM and on input *w* makes an odd number of transitions}**

**L is not decidable. To prove this, we will show that if L is decidable, then HALT is decidable.**

**(Since we know HALT is not decidable, we will conclude that L is not either.)**

**Assume that L is decidable, and let $M_L$ be a TM that decides it. Construct the following TM $M_H$ (which will ultimately decide HALT) :**

**$M_H$ on input M,w:**

**Construct a TM M' by modifying M so that M' makes an extra transition at the start. (Do this by adding a new start state s0, and make each transition from s0 go to the old start state without changing the tape.) Note that M halts on w iff (M,w) OR (M'w) are in L (i.e., M makes an odd number of transitions on w or M' does)**

c) **L= { n | n is prime}**

**It may seem that this is undecidable since they are always discovering new primes, but there are plenty of algorithms for deciding if a number is prime. Maybe n is so large, the program doesn't finish, but that doesn't mean it's indecidable.**

**#2. Given 2 dfa's $M_1$ and $M_2$, Consider the question $EQ_{dfa}$: "Is $L(M_1) = L(M_2)$?"**

a) **State this as a language problem**

$EQ_{dfa} = \{(M_1, M_2) \mid M_1 \text{ and } M_2 \text{ are dfa's and } (L(M_1) \cap \sim L(M_2)) \cup (\sim L(M_1) \cap L(M_2)) = \varnothing\}$

b) **Show $EQ_{dfa}$ is decidable or undecidable**

**It is certainly decidable whether $M_1$ ,and $M_2$ are dfa's and all the operations are decidable.**

**#3. Prove or disprove: Given a grammar in CNF and a string w ε L(G) with derivation tree T, *if depth(T) = n, then* $|w| \leq 2n\text{-}1$**

**#4. Given an arbitrary turing machine, M, and an arbitrary state q e M, show that it is undecidable whether M ever enters state q. Do not use Rice's Theorem.**

**#5. A property, P, of re languages is a mapping:**

**P: {re languages over S*} → {T,F}**

**A property P is *trivial* if it is true of *all* re languages or *no* re language**

a) **Name a non-trivial property of re languages**
b) **Name a trivial property of re languages**

**(Note: You may research this: just say where you got your answer from)**

a) **$P_{regular}$: {L | L is regular}**
  **$P_{empty}$: {L | L is empty} (not to be confused with the empty property)**

**b) There are exactly 2 trivial properties: $P_{none}$: {L | P false for no re language L} and $P_{all}$: {L| P true for all re languages}.**

**i.e. $P_{none}$: $\phi$**
**and $P_{all}$: {L| L is an re language}**

**The TM for $P_{none}$ immediately rejects**
**The TM for $P_{all}$ decides whether the input is a legal encoding of some TM.**