CS 3120 Quiz Day 4

This packet contains the quizzes for this quiz day. This **cover sheet** is here to provide instructions, and to cover the questions until the quiz begins. **do not open this quiz packet** until your proctor instructs you to do so.

You will have the 1 hour to complete these quizzes. Each quiz is two pages (front and back of one sheet of paper) worth of questions. Make sure to **write your name and computing id at the top of each individual quiz**.

When you are finished, simply submit this packet at the front of the classroom.

This quiz is CLOSED text book, closed-notes, closed-calculator, closed-cell phone, closed-computer, closed-neighbor, etc. Questions are worth different amounts, so be sure to look over all the questions and plan your time accordingly. Please sign the honor pledge below.

In theory, there is no difference between theory and practice.
But, in practice, there is.

THIS COVER SHEET WILL NOT BE SUBMITTED FOR GRADING. DO NOT PUT WORK YOU WANT GRADED ON THIS PAGE

Quiz - Module 4: Turing Machines

Name ____________________

1. [10 points] Answer the following True/False questions.

A *Turing Machine* contains a finite set of states Q, but an infinitely sized tape	**True**	**False**
For language R and string w, a *Recognizer* will always *accept* when $w \in R$ but might loop forever when $w \notin R$	**True**	**False**
A *Turing Machine* makes a decision on what transition to follow based on the current state and the symbol currently on the tape at the head position	**True**	**False**
*NTM*s can recognize functions that *DTM*s cannot	**True**	**False**
Multi-tape Turing Machines can recognize more functions than *Deterministic Turing Machines* if they have *n tapes*, where n is the size of the input	**True**	**False**
Detecting whether the language of two *DFA*s is equivalent is *Decidable*	**True**	**False**
Suppose I alter *TM*s so that they can move the head up to *five times* per step. This *TM* would be *MORE* powerful than a standard *TM*	**True**	**False**
Recognizers will always *loop forever* on *NO* instances	**True**	**False**
If a problem A is *decidable*, then $\bar{A}$ is *recognizable*	**True**	**False**
If a problem A is *recognizable* but *undecidable*, then $\bar{A}$ is *unrecognizable*	**True**	**False**

2. [2 points] Suppose I have two recognizers for languages R_1 and R_2. Is it possible to construct a *recognizer* for $R_1 \cap R_2$? Briefly explain why or why not.

3. [2 points] Consider the following problem, which is related to *Hilbert's 10th Problem*: Given a polynomial P over n variables $\{x_1, x_2, ..., x_n\}$, return true iff there are exist no integer roots for P (i.e., all integer values over $\{x_1, x_2, ...x_n\}$ do NOT evaluate P to 0). Is this problem *co-recognizable*? Explain why or why not.

The problems on this page all involve the same problem. Consider the function $TM_{eq}(M_1, M_2)$, which returns *True* if and only if $\mathcal{L}(M_1) = \mathcal{L}(M_2)$.

4. [6 points] Show that TM_{eq} is undecidable by providing a reduction from $A_{TM}(M, w)$. We will split this reduction into two parts:

First, given input to A_{TM} (i.e., a machine M and string w), describe (write the pseudocode for) how to construct two machines M_1 and M_2 (one machine per box). You should do this such that $\mathcal{L}(M_1) = \mathcal{L}(M_2)$ if and only if M accepts w. *HINT: One of your two machines will be trivial (it will either always reject or always accept, the other is a function of M and w)*

Now, assuming the construction above works, describe the pseudocode steps for a decider to the *undecidable problem A_{TM}*. You may invoke your construction from the previous problem by simply stating *Construct M_1 and M_2*. Assume the construction works even if your answer to the previous problem isn't correct.

We provide the beginning of the proof here, which leads into the description of your decider: *Assume that TM_{eq} is decidable, thus a decider M_{eq} exists that decides it. We can construct a working decider for A_{tm} as follows:*