CS 3120 Exam 2

This packet contains exam 1. 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 1 hour and 45 minutes to complete this exam. 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 page**.

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.

A crash reduces
Your expensive computer
To a simple stone.

Module 3: Context-Free Grammars

Name _____

1. [6 points] Answer the following True/False questions.

*PDA*s are allowed to use *non-determinism*	**True**	**False**
It is possible for a *context-free grammar* to generate only the empty string.	**True**	**False**
It is possible to write a working *context-free grammar* for the language Σ^*	**True**	**False**
$a^x b^y c^y d^x$ is a *context-free language*	**True**	**False**
A *PDA*'s stack can overflow, and in this case the machine always *rejects*	**True**	**False**
A *PDA* that has access to a *queue* is computationally equivalent to a *PDA* that has access to a *stack*	**True**	**False**

2. [2 points] Draw a *PDA* for the following language: $C = \{ww^R \mid w \in \Sigma^*\}$

3. [2 points] Write out a *context-free grammar* for the following language: $D = 1^*0^n1^n0^*$

Module 4: Turing Machines

Name _____

4. [6 points] Answer the following True/False questions.

A *Turing Machine* with a two-dimensional (infinite in both dimensions) tape is computationally more powerful than a *single-tape Turing Machine*	**True**	**False**
Some *recognizers* never *halt and reject* on any input	**True**	**False**
If a language A is *decidable*, then the complement $\bar{A}$ is also *decidable*	**True**	**False**
$a^n b^n c^n$ is *turing-decidable*	**True**	**False**
Turing Machines can be run (simulated) on other *Turing Machines* to see how they behave on various input strings	**True**	**False**
The *turing-recognizable* languages are closed under *complement*	**True**	**False**

5. [2 points] Suppose you have a *recognizer* for the problem A (call it R) and a *recognizer* for the complement $\bar{A}$ (call it R'). Describe an algorithm that *decides* A.

6. [2 points] In your own words, explain how we were able to reduce A_{TM} to the *Halting Problem*. Describe the reduction informally (or formally if you prefer) in your answer.

Miscellaneous

7. [3 points] In class, we discussed how *detecting properties of Turing Machines* was often undecidable (e.g., emptiness testing) but *detecting properties of DFAs* was always decidable. At an intuitive level, why is this the case?

8. [3 points] Suppose I have a *grammar* for a language A with a start variable S_A and another grammar for a language B with a start variable S_B. Describe how I can combine these grammars by adding one new variable and one new rule to create the grammar for $C = AB$ (that's A concatenated with B). *HINT: your one new variable will be the start variable of the new union language.*

9. [3 points] Show that $a^n b^n c^n$ is not context-free. You can use any technique you would like.