CARLETON UNIVERSITY
SCHOOL OF COMPUTER SCIENCE
WINTER 2015
COMP. 3803
INTRODUCTION TO THEORY OF COMPUTATION
ASSIGNMENT IV
DUE: MONDAY APR. 6, 2015 (11:30 AM)

Assignment Policy: Late assignments will **not** be accepted. You are expected to work on the assignments on your own. Past experience has shown conclusively that those who do not put adequate effort into the assignments do not learn the material and have a probability near 1 of doing poorly on the exams.

Important note: When writing your solutions, you must follow the guidelines below.

- Please write your name and student number clearly. Your last name must be in Upper Case.
- The answers should be concise, clear and neat. Make sure that your TA can read your solutions.
- Please submit the solutions *in the order of the problems*, the solution to Problem 1, then to Problem 2 and so on.
- When presenting proofs, every step should be justified so as to get partial credit.
- Assignments should be stapled (or in an unsealed envelope) *with your name and student number*.

Substantial departures from the above guidelines will not be graded.

1. Prove, using the Pumping Lemma for CFL, that the following are not context free:
   - $L = \{a^N b^{2N} c^M d^{2N} \mid M,N \geq 0\}$
   - $L = \{a^k b^m c^n \mid \text{where } k = \text{Max}(m, n)\}$.

2. Construct (deterministic or non-deterministic) pushdown automata that accept the following languages:
   - $L = \{ c^K a^N b^M \mid K \geq 0, N \geq M \geq 0, \}$, where $\Sigma = \{a, b,c\}$
   - $L = \{a^N b^{2M} c^M \mid M,N \geq 0\}$, where $\Sigma = \{a, b, c\}$
   - $L = \{c^K w w^R \mid \Sigma = \{a, b, c\}, w \in \{a,b\}^*$, and where w^R is the reverse of w; $K \geq 0\}$

   In each case, please add some sentences explaining the *rationale* for the transitions. This will make the grading easier, in case your design is wrong.

3. Construct a **one-tape** Turing machine that accepts the language $L = \{a^{2N} b^N \mid N \geq 0\}$, where the alphabet $\Sigma = \{a, b\}$. You have to give the **informal** description, describing the states you are going to use, their meaning, and what kind of transitions it will have. This will give you partial credit. But you have to also work out all the details of the actual transitions. You can assume that the head of the machine is at the start of the string.

4. Construct a **one-tape** Turing machine that accepts the language $L = \{w \mid w$ has more 1's than 0's$\}$, where the alphabet $\Sigma = \{0, 1\}$. You have to give the **informal** description, describing the states you are going to use, their meaning and what kind of transitions it will have. This will give you partial credit. But you have to also work out all the details of the actual transitions. You can assume that the head of the machine is at the start of the string.

5. Construct a **two-tape** Turing machine that accepts the language $L = \{a^{2N} b^N \mid N \geq 0\}$, where the alphabet $\Sigma = \{a, b\}$. You have to give the **informal** description, describing the states you are going to use, their meaning and what kind of transitions it will have. But you do **not** have to also work out all the details of the actual transitions. You can assume that the head of the machine is at the start of the string. *Observe that you must use both tapes for reading and writing, and not just ignore the second tape and resort to the one-tape solution of Question 3.*