COMS 3261: Computer Science Theory

Problem Set 4, due Wednesday, 11/13/13, at the beginning of the class

Please follow the Homework Guidelines.

Try to make your answers as precise, succinct, and clear as you can.

Part A: [30 points] Do the problems posted at Gradiance.

Part B:

Problem 1. [15 points] Show that the language $\{0^i1^j2^k \mid i > j > k \geq 0\}$ is not context-free.

Problem 2. [15 points] Show that the complement of the language of Problem 1 is context-free.
Hint: Consider the following three types of strings: (a) strings that are not of the form $0^i1^j2^k$, (b) strings that are of the form $0^i1^j2^k$ where $i \leq j$, (c) strings that are of the form $0^i1^j2^k$ where $j \leq k$. Show that each of the three types of strings is a context-free language, and use this to show that the complement of the language of Problem 1 is context-free.

Problem 3. [20 points] Do Exercise 7.1.5 in HMU (page 277). We repeat here the exercise for convenience. Begin with the following grammar:
$S \rightarrow aAa \mid bBb \mid \varepsilon$
$A \rightarrow C \mid a$
$B \rightarrow C \mid b$
$C \rightarrow CDE \mid \varepsilon$
$D \rightarrow A \mid B \mid ab$

Perform the following steps one by one, and show the grammar that results after each step.

a. Eliminate ε-productions.
b. Eliminate any unit productions in the resulting grammar.
c. Eliminate any useless symbols in the resulting grammar.
d. Put the resulting grammar into Chomsky Normal Form.

Problem 4. [20 points]

1. [5 points] (Exercise 7.4.3, part c in HMU, page 308). Use the CYK algorithm to determine whether the string *aabab* is in the language of the following grammar

$S \rightarrow AB \mid BC$

$A \rightarrow BA \mid a$

$B \rightarrow CC \mid b$

$C \rightarrow AB \mid a$

2. [10 points] (Exercise 7.4.5 in HMU, page 308). Modify the CYK Algorithm to report the number of distinct parse trees for the given input, rather than just reporting membership in the language.

(*Hint:* Modify the CYK algorithm so that it computes for each pair of indices i,j not only the set X_{ij} of variables that can derive the substring $a_i \ldots a_j$, but in addition it computes for every variable A in X_{ij} the number of different parse trees with root A and yield $a_i \ldots a_j$.)

3. [5 points] Apply your modified CYK algorithm to the grammar and input string of part 1.