College of Computer & Information Science — Spring 2008
Northeastern University — Handout 13
CSG714: Theory of Computation — 27 February 2008

Problem Set 4 (due Tuesday, March 11)

1. (4 + 4 + 6 = 14 points) Balanced parentheses in logspace

The set of balanced parentheses is the set generated by the context-free grammar

$$S \to (S)|SS|\varepsilon$$

For a given string x, let $\ell(x)$ (resp., $r(x)$) denote the number of left (resp., right) parenthesis symbols in x.

(a) Show that a string x is balanced if and only if $\ell(x) = r(x)$ and $\ell(y) \geq r(y)$ for every prefix y of x.

(b) Prove that the set of balanced parantheses is in L.

Now consider the set of balanced parenthesis of two types, given by the following context-free grammar.

$$S \to (S)|[S]|SS|\varepsilon.$$

(c) Prove that the set of balanced parenthesis of two types is also in L.

(*Hint:* Extend the claim of (a) to the case of two types.)

2. (10 points) The two-player game version of PUZZLE

Problem 8.14 from Sipser's text.

3. (10 points) The cat-and-mouse game

Problem 8.15 from Sipser's text.

4. (3 + 7 = 10 points) k-headed DFA

A k-headed DFA is a one-tape TM with k read-only input heads that can move left or right but cannot move off the input string.

(a) Give a formal definition of a k-headed DFA.

(b) Show that a language A is in L if and only if it is accepted by a k-headed DFA for some k.

5. (10 points) The majority circuit

The majority of n boolean inputs x_i, $1 \leq i \leq n$ is 0 if $\sum x_i < n/2$ and 1 otherwise. Show that the majority can be computed with $O(n)$ size circuits.

(*Hint:* Construct an $O(m)$ size *addition circuit* that takes two m bit numbers and computes their $m+1$-bit addition. Use recursion and this addition circuit to compute the majority. You may want to refer to Problems 9.24-9.26 in Sipser's text.)

6. ($4 \times 4 = 16$ points) Satisfiability and its variants

In class, we have discussed SAT, CIRCUIT-SAT, CVP, and the relationship among them. This problem explores other variants of boolean satisfiability. Define the SVP problem as follows: Given a SAT formula ϕ and an assignment to the variables, determine the truth value of ϕ.

(a) What is the complexity of SVP?

(b) Define CNFSVP as the SVP problem in which ϕ is given in conjunctive normal form. What is the complexity of CNFSVP?

A Boolean decision diagram (BDD) is a directed acyclic graph with a single source and two sinks, one labeled 0 and the other labeled 1. Each non-sink node has two outgoing edges, one labeled x and the other labeled $\overline{x}$ for some Boolean variable x. The value of a BDD on a truth assignment σ is the label of the sink node of the unique σ-enabled path from the source to a sink, where an edge with literal ℓ is enabled if $\sigma(\ell) = 1$. For BDDs, what is the complexity of

(c) determining the value for a given σ?

(d) satisfiability?

(The part on BDDs is taken from Kozen's text.)