Ph.D. Qualification Examination

Programming Languages and Compilers (Apr. 2005)

(1) (15%) Consider the context-free grammar

$$S \rightarrow SS+ \mid SS* \mid a$$

(a) Show how the string $aa+a*$ can be generated by this grammar.
(b) Construct a parse tree for this string.
(c) What language is generated by this grammar? Justify your answer.

(2) (15%) Construct a syntax-directed translator that verifies that the parentheses in an input string are properly balanced.

(3) (25%) In a string of length n, how many of the following are there?
(a) prefixes
(b) suffixes
(c) substrings
(d) proper prefixes
(e) subsequences

(4) (15%) Show that no LL(1) grammar can be ambiguous.

(5) (30%) Consider the following modification of the LR(1) grammar $L \rightarrow Lb \mid a$:
$L \rightarrow MLb \mid a$
$M \rightarrow \epsilon$
(a) What order would a bottom-up parser apply the productions in the parser tree for the input string *abbb*?
(b) Show that the modified grammar is not LR(1).