CS3133

HW#8

DUE: Thursday, October 13

1. (5 points) Show the values of the different $T_{ij}, 1 \le i \le j \le 3$, when using the CKY algorithm to test if the string 111 can be generated by the grammar

$$S \to SS | 0 | AS | AB | AC$$
$$A \to 1 | SS | SA$$
$$B \to SA | 0 | 1$$
$$C \to CA | CC$$

and, if 111 can be generated, show a parse tree.

2. (4 points) Is the grammar

$$S \to SS | 0 | AS | AB | AC$$
$$A \to 1 | SS | SA$$
$$B \to SA | 0 | 1$$
$$C \to CA | CC$$

ambiguous? Justify your answer.

3. (8 points) Is the following question decidable?

INPUT: A finite set $\{G_1 = (N_1, \Sigma, P_1, S_1), \ldots, G_n = (N_n, \Sigma, P_n, S_n)\}$ of CFGs.

QUESTION: Is $\bigcup_{1 \le i \le n} L(G_i) = \varnothing$?

4. (12 points) Let Λ be the set of all regular languages over $\{0,1\}$ which contain the string 01. For example, $0^*1 + (1+0)^* 1110^* \in \Lambda$, but $1^*0^* + 10 \notin \Lambda$.

a Is Λ recursively enumerable? Justify your answer.

b Is Λ recursive? Justify your answer.

CS3133

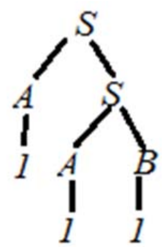
Solutions to HW#8

1.

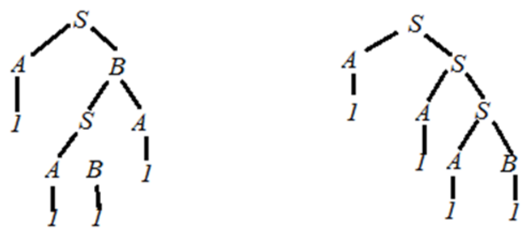
$$T_{11} = \{A, B\} \quad T_{22} = \{A, B\} \quad T_{33} = \{A, B\}$$

$$T_{12} = \{S\} \quad T_{23} = \{S\}$$

$$T_{13} = \{S, A, B\}$$



2. The grammar is ambiguous because 1111 admits the following two parse trees.



3. The question is decidable. The set NULLABLE(G_i) is easily computable for any CFG G_i , and $\left(L\left(G_i\right)=\varnothing\right) \leftrightarrow S_i \in$ NULLABLE(G_i).

EMPTY $\leftarrow$ TRUE

for $i \leftarrow 1$ **to** n **do**

if $S_i \notin$ NULLABLE(G_i) **then** EMPTY $\leftarrow$ FALSE

$$\left(\bigcup_{1 \le i \le n} L\left(G_i\right)=\varnothing\right) \leftrightarrow \text{EMPTY}$$

4. We only need to show that Λ is recursive, because this implies that it's recursively enumerable. We know from the first part of the class that for any regular language L there is a DFA M such that $L=L(M)$, and we know that there is an algorithm to test if $01 \in L\left(M\right)$. So there must be a total Turing Machine to *accept* regular language L if $01 \in L$ and *reject* regular language L if $01 \notin L$.