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

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

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

2. (4 points) Is the grammar

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

ambiguous? Justify your answer.

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

**INPUT:** A finite set $\{G_i = (N_i, \Sigma, P_i, S_i), \ldots, G_n = (N_n, \Sigma, P_n, S_n)\}$ of CFGs.

**QUESTION:** Is $\bigcup_{i=s}^{n} L(G_i) = \emptyset$?

4. (12 points) Let $\Lambda$ 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 $\Lambda$ recursively enumerable? Justify your answer.

**b** Is $\Lambda$ recursive? Justify your answer.
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 \( \text{NULLABLE}(G_i) \) is easily computable for any CFG \( G_i \), and \( (L(G_i) = \emptyset) \leftrightarrow S_i \in \text{NULLABLE}(G_i) \).

   \[
   \text{EMPTY} \leftarrow \text{TRUE} \\
   \text{for } i \leftarrow 1 \text{ to } n \text{ do} \\
   \quad \text{if } S_i \notin \text{NULLABLE}(G_i) \text{ then } \text{EMPTY} \leftarrow \text{FALSE} \\
   \quad \left( \bigcup_{i \in L(G_i)} \emptyset \right) \leftrightarrow \text{EMPTY}
   \]

4. We only need to show that \( \Lambda \) 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 \( 01 \in L(M) \). So there must be a total Turing Machine to accept regular language \( L \) if \( 01 \in L \) and reject regular language \( L \) if \( 01 \notin L \).