Ex. 1

To account for the syntax of expressions like the ones in (1-a), the following assumptions are made: a noun phrase is either a proper noun or a noun phrase followed by a conjunction phrase; a conjunction phrase is a coordinating word (a comma or the word *and*) followed by a noun phrase.

1. Write the grammar G for noun phrases following these assumptions.
2. Provide the two possible derivation trees that G associates with (1-a). Which of these two analyses seem the most appropriate to you?
3. Is G able to offer an analysis for (2)? If not, propose a grammar G' which can. Give the corresponding syntactic tree.
4. G generates the variants (3) of the expression (1-a). Propose a grammar G'' which generates (1-a), as well as (1-b), but excludes these variants. In other words, G'' would allow at most one occurrence of *and*, before the last conjunct. Note that it is not asked that G'' generates the embedded form (2).

(1) a. Paul, Marc and André
    b. Paul, Marc, Zoé and André

(2) Paul, Marc and Léa, and Luc

(3) a. Paul, Marc, André
    b. Paul and Marc and André

Ex. 2

A context-free grammar $G = \langle \Sigma, N, S, P \rangle$ is called *simple* if it verifies the two following conditions:

- $P \subset N \times \Sigma N^*$
- $\forall A \in V, \forall x \in \Sigma, \forall u, u' \in (\Sigma \cup N)^*, (A \to xu) \in P \land (A \to xu') \in P \Rightarrow (u = u')$

In words, (1) right hand side parts of the rules start with a terminal letter, followed by an arbitrary number of non-terminal letters (possibly none), and (2) it's not possible to have two differents rules from the same non-terminal whose right hand side part start with the same (terminal) letter.
A context-free language is a *simple language* if there exists a simple grammar that generates it.

1. Find a simple grammar for the language $\{a^n b^{n+1}, n \geq 0\}$
2. Find a simple grammar for the language $\{a^n b^n, n > 0\}$
3. Let L be the language generated by: $S \longrightarrow aSS \mid b$. Build a context-free grammar that generates the language Lc^*d.
4. Show that the product of two simple languages is a simple language. Provide a rigourous explanation, not necessarily a mathematical proof.