Q1. Loco Lists

Q1.1. Write a CFG to represent a list consisting of positive integers and other lists. These lists (and sublists) can have arbitrarily large dimensions.

Notes:

- An empty list is also a valid list.
- You can use \( n \) to denote a positive integer in the CFG. You don't have to worry about representing multidigit numbers since \( n \) encapsulates them all.
- To represent \( \epsilon \) in the CFG, you can either write the word epsilon or just type the letter e.

Examples of Valid Lists:

\[
[] \\
[[[]]] \\
[1,[[1,2],[4],56],[[[564]]]] \\
[1,2,3]
\]

\[
S \rightarrow [S] \mid [T] \mid [ ] \\
T \rightarrow U,T \mid U \\
U \rightarrow n \mid S
\]

Q1.2. Is the language defined by the grammar given above regular? Yes/No

Q2. Context-Free Grammars

My friend Hamza with a peculiar lexicon likes modifying common acronyms in his speech and texting as defined by the following CFG:

\[
S \rightarrow A \mid B \\
A \rightarrow loL \\
L \rightarrow l \mid l\text{ out loud} \mid A \\
B \rightarrow smH \\
H \rightarrow h \mid h\text{ my head} \mid B
\]

Can the grammar above be parsed by a LL1 recursive descent parser (like Project 4)? Justify your answer.

No. The first sets are not disjoint.

Q3. Ambiguity

Prove that the following grammar is ambiguous:

\[
S \rightarrow bS \mid Sb \mid T \\
T \rightarrow Sa \mid Sb \mid Sc \mid \epsilon \\
S \rightarrow bS \rightarrow bT \rightarrow b \\
S \rightarrow Sb \rightarrow Tb \rightarrow b
\]

Any other valid ambiguous productions i.e., same string generated with different paths are correct.
Q4. Operational Semantics

Using the gives rules, fill in the blanks the complete the derivation below:

\[
\begin{align*}
&\quad \frac{A; n \Rightarrow n}{A; x \Rightarrow v} \\
&\quad \frac{A; e_1 \Rightarrow v_1 \quad A, x : v_1; e_2 \Rightarrow v_2}{A; \text{let } x = e_1 \text{ in } e_2 \Rightarrow v_2} \\
&\quad \frac{A; e_1 \Rightarrow v_1 \quad A; e_2 \Rightarrow v_2 \quad v_3 \text{ is } v_1 \land v_2}{A; e_1 \land e_2 \Rightarrow v_3}
\end{align*}
\]

Blank #1: let \( x = \text{"cmsc"} \)

Blank #2: \( A, x:\text{"cmsc"} ; \text{"330"} \Rightarrow \text{"330"} \)

Blank #3: \( x \land y \)

Blank #4: \( A, x:\text{"cmsc"} , y:\text{"330"}(x) = \text{"cmsc"} \)

Blank #5: \( A, x:\text{"cmsc"} , y:\text{"330"}(y) = \text{"330"} \)

Blank #6: \( \text{"cm\text{sc}330"} \text{ is } \text{"cmsc"} \land \text{"330"} \)