Context-free Grammars: In-class Exercise

(1) Consider the CFG G with S' as the start symbol:

$$
\begin{array}{rcl}
S' & \rightarrow & S \mid \epsilon \\
S & \rightarrow & T \mid (\; N \; , \; C \;) \\
C & \rightarrow & C \; , \; S \mid S \\
T & \rightarrow & a \mid b \mid c \\
N & \rightarrow & x \mid y \mid z
\end{array}
$$

a. List the set of terminal symbols and the set of non-terminal symbols in G.

Answer:

$$
\begin{array}{rcl}
T & = & \{a, b, c, x, y, z, \backslash, , (,)\} \\
N & = & \{S', S, C, T, N\}
\end{array}
$$

b. For each of the following strings, write down `true` if the string is in the language $L(G)$ generated by G, `false` otherwise.

1. `y`
2. `c`
3. `(x)`
4. `(x,y)`
5. `(z,a,b,a,b,c)`
6. `(x,a,(y,b),c)`
7. `(x,(y,a),(z,b))`
8. `(x,(x,(x,(x,a))`

Answer:

1. `y` : false
2. `c` : true
3. `(x)` : false
4. `(x,y)` : false
5. `(z,a,b,a,b,c)` : true
6. `(x,a,(y,b),c)` : true
7. `(x,(y,a),(z,b))` : true
8. `(x,(x,(x,(x,a))` : false

(2) Consider the family of CFGs G_k with S as the start symbol and k is some arbitrary non-zero positive integer such that $G_1, G_2, G_3, \ldots$ are individual CFGs with the rules:

$$\begin{array}{lcl} S & \rightarrow & A\ B \\ B & \rightarrow & C\ A\ A \\ C & \rightarrow & c \\ A & \rightarrow & a_i \quad \text{defines } i \text{ rules, where } i \in [1, k] \end{array}$$

For example, in G_3 the rules with A as left-hand side are: $A \rightarrow a_1 \mid a_2 \mid a_3$ with three terminal symbols.

a. Provide the number of terminal symbols in a grammar G_k.

Answer: $k + 1$

b. If the string $a_4ca_3a_2$ is accepted by grammar G_3 then provide a derivation for it.

Answer: a_4 does not exist as a terminal in G_3.

c. If the string $a_4ca_3a_2$ is accepted by grammar G_4 then provide a derivation for it.

Answer: $S \Rightarrow A\ B \Rightarrow a_3\ B \Rightarrow a_3\ C\ A\ A \Rightarrow a_3\ c\ A\ A \Rightarrow a_3\ c\ a_1\ A \Rightarrow a_3\ c\ a_1\ a_2$

d. Provide the total number of strings that can be generated for a grammar G_k.

Answer: k^3

(3) One of the rules in the CFG below is redundant: any sentence that can be generated using this rule can already be generated by a combination of other rules. Write down the redundant rule.

S	→	NP VP	IV	→	runs	N	→	John
NP	→	N	IV	→	sits	N	→	he
NP	→	D N	TV	→	chases	N	→	Mary
VP	→	VP PP	TV	→	eats	N	→	dog
VP	→	VP CONJ VP	TV	→	catches	N	→	tree
VP	→	IV	TV	→	tells	N	→	squirrel
VP	→	IV PP	TV	→	sees	D	→	the
VP	→	TV NP	CONJ	→	and			
VP	→	TV C S	C	→	that			
NP	→	NP CONJ NP	P	→	in			
PP	→	P	P	→	away			
PP	→	P NP						

Answer: $VP \rightarrow IV\ PP$ can be generated using $VP \rightarrow IV$ and $VP \rightarrow V\ PP$.