Parsing

Exercise 1:
Consider grammar specified below:

1. Remove the left recursion in the following grammar.

2. Why are left recursions bad, and for what type of parser?

\[
\begin{align*}
\text{<sentence>} & \ ::= \ <\text{wordList}>. \\
\text{<wordList>} & \ ::= \ <\text{wordList}>\langle\text{word}\rangle \mid \langle\text{word}\rangle
\end{align*}
\]

Exercise 2:
Extend the grammar from Exercise 1 so it can support questions (sentences ending in a question mark), exclamations (sentences ending in an exclamation mark), complex sentences (parts are divided by a comma), and the notion that the first word of a sentence must begin with a capital letter. Also, any other word in the sentence can begin with a capital letter. Assume that \text{<capitalWord>} is a word with a capital first letter.

Good will task (not graded): write regular expressions for \text{<capitalWord>} and \text{<word>}.

Exercise 3:
Write a grammar for parsing a URI.

```
https://some.thing.com/path/pa/path/?param=val&paparm=valal
```

\text{Scheme} is optional and should accept http, https, and ftp. \text{Domain} should accept any domain name starting from second level (two words separated by a dot). \text{Path} is optional and can contain multiple segments separated by slashes. The last slash of \text{path} is optional and double slashes are not allowed. Finally, \text{query} is also optional and may contain many key-value pairs. Assume that the parts of the URI are composed of letters and you have \text{<letter>} at your disposal.

Examples:

- google.com
- http://hc.ebinu.scg.unibe.ch/
- ftp://bo.net/users/joe/posts?tag=news
- www.cambridge.org/us/catalogue/catalogue/?isbn=alongnumber&ss=res

Good will task (not graded): make sure that you don’t have left recursion in your grammar.