
BNF Grammar Writing Exercise #2: SL (Sign Language)

Consider language SL to consist of the set of all strings of one or more delimited plus-strings or delimited minus-strings, where a delimited plus string consists of a left delimiter followed by any number of pluses followed by a matching right delimiter, and a delimited minus string consists of a left delimiter followed by any number of minuses followed by a matching right delimiter, and matching delimiters are either matching parentheses, or matching square brackets, or matching braces. Since English descriptions of languages like this invariably leave something to be desired, examples are generally provided in order to clarify the English specification. For SL, these examples should serve the purpose:

1. `()`
2. `[-----]`
3. `{++}`
4. `() [-] () (+) {-} (--)[+++][++++] () {}`
5. `(++) [+++++](+) (+++++){--}`
6. `(-)(+)(--)(+)(-----)(+)(+++++)`

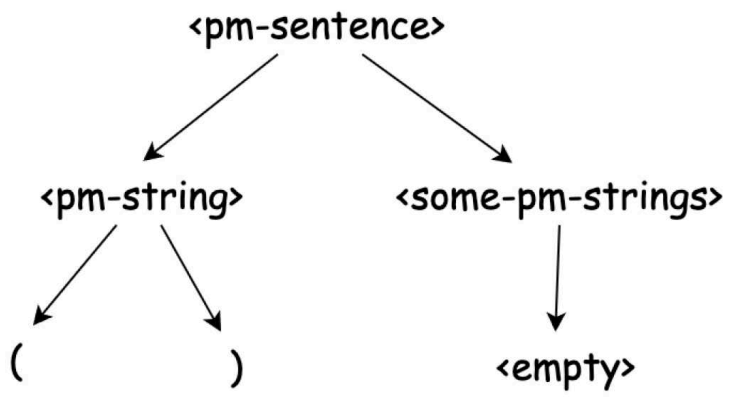
Your tasks ...

1. Write a BNF grammar for this language.
2. Draw a parse tree for the following sentence: `()`
3. Draw a parse tree for the following sentence: `[--]{+++}`

The BNF Grammar

```
<pm-sentence> ::= <pm-string> <some-pm-strings>
<pm-string> ::= <p-string> | <m-string> | ( ) | [ ] | { }
<some-pm-strings> ::= <empty> | <pm-string> <some-pm-strings>
<p-string> ::= ( <plusses> ) | [ <plusses> ] | { <plusses> }
<m-string> ::= ( <minusses> ) | [ <minusses> ] | { <minusses> }
<plusses> ::= <empty> | + <plusses>
<minusses> ::= <empty> | - <minusses>
```

First parse tree



Second parse tree

