

Name: Jing Teng UTEID: jt23322 Class: CS 378 (Software Design)
 Assignment: Homework 3 (MetaModel)

Backus-Naur (Normal) Form of Grammars

```

/* Start Symbol, Grammar */
<!-- in example Grammar      => IntExp -->
<!-- in example Production => INT or INT "+" or INT "x" or "(" or ")" -->
<!-- in example Pattern      => INT -->
<Grammar>
    := <Production> |
       <Production> <Grammar>
    ;
<Production>
    := <list_whitespaces> <Production_name>
       <list_whitespaces> <expression>
       <line_ending>
<Production_name> := "'" <matching_punctuation> "'" |
                   [a-zA-Z]+
Z
                                <!-- matching for "(", ")" and etc.. -->
                                <!-- regular expression form for matching chars
                                suppose to match any chars between a-z, A-
                                -->
                                ;
<matching_punctuation> := '(' | ')' | '[' | ']' | '{' | '}'
<list_whitespaces> := " " <list_whitespaces> | ""
<expression>      := <verbs> | <verbs> "|" <expression>
<line_ending>      := <list_whitespaces> <EOL> |
                   <line_ending> <line_ending>
                                <!-- "" is empty string, i.e. no whitespace -->
                                <!-- "+" -->
                                ;
<verbs>
    := "'" <action> "'" |
       "'" <action> "'" <Production_name>
                                <!-- + is matched for action -->
                                ;
<action>
    := <text> |
       <text> <list_whitespaces> <action>
                                ;
<text>
    := [0-9a-zA-Z]+
.
                                <!-- Regular expression form for matching chars
                                These are terminals; they terminate the
                                production process -->
<list_whitespaces> := " " |
                   " " <list_whitespaces>
                                ;
<EOL>
    := '\n';
e
                                <!-- can be others depending on specific languag
                                and etc.. -->

```

*Note:

- some infos are inferred from Wikipedia.com
- the EBNF and BNF grammar are used since the assignment used them in an interchangeable way