|  |
| --- |
| CS 440 |
| Programming Languages and Translators |
|  |
| MCL: Grammar report |
| 02/26/2017 |

|  |
| --- |
| Cai Chen  zdthm007@gmail.com  Yuyang Luo  yluo41@hawk.iit.edu |

line → assignment\_list | init\_list | exit\_command | print\_exp | line assignment\_list

| line init\_list | line print exp | line exit\_command

Init\_list → Matrix init\_lm | Int init\_li

init\_lm → init\_m | init\_m ',' init\_lm

init\_m → identifier | identifier '=' exp | identifier '[' exp ',' exp ']' '=' array

init\_li → init\_i | init\_i ',' init\_li

init\_i → identifier | identifier '=' exp

assignment\_list → assignment | assignment ',' assignment\_list

assignment → identifier '=' exp | identifier '[' exp ',' exp ']' '=' exp

exp → exp1 | exp '+' exp1 | exp '-' exp1

exp1 → term | exp1 '\*' term | exp1 '/' term

term → number | matr | '(' exp ')' | identifier | '-' term

| identifier '[' exp ',' exp ']'

Conclusion:

This grammar is unambiguous since I don’t want to cause any trouble made by incorrect order of operations. For example, in the expression 5\*3-5, if minus operation works first and multiple operation executes late, the result will be -10. So I use unambiguous grammar to give multiple operation a high precedence than minus operation.