**Scanner**:

Tokens:

* number => e.g. 3257
* sign => + | - | \* | / | ^
* identifier => x

Regular Expressions:

* identifier = x
* operation = + | - | \* | / | ^
* digit = [0-9]
* number = digit digit\* //one or more digit

State Machine for scanning tokens:

+ or – or \* or / or ^

x

[other]

digit

digit

other

**Parser:**

Grammar BNF

* addsubexp -> addsubexp addop muldivexp | muldivexp
* addop -> + | -
* muldivexp -> muldivexp mulop powerexp | powerexp
* mulop -> \* | /
* powerexp -> exp powerop powerexp | exp
* powerop -> ^
* exp -> (addsubexp ) | identifier | number

Grammar EBNF

* addsubexp -> muldivexp { addop muldivexp }
* muldivexp -> powerexp { mulop powerexp }
* powerexp -> { exp powerop } exp
* exp -> (addsubexp) | identifier | number