Backus–Naur Form (BNF) & Regular Expression

By Wally

Backus–Naur Form

- A notation technique for context-free grammars
- <symbol> ::= __expression__
- <symbol> is a nonterminal
- The __expression__ consists of one or more sequences of symbols
- More sequences are separated by the vertical bar "|", indicating a choice
- Symbols that never appear on a left side are terminals

A Simple Int Addition Using BNF

```
Val1 + Val2
```

A Simple Int Addition Using BNF

```
Val1 + Val2
<add> ::= num add-op num
```

A Simple Int Addition Using BNF

```
Val1 + Val2 + Val3 + ...
```

A Simple Int Addition Using BNF

```
        Val1 + Val2 + Val3 + ...
<add> ::=
        num add-op <add> |
        num add-op num
```

What about num? —— How do define num w/ BNF?

Number:

```
                  123.456...
<num> ::=
           <digits> |
           <digits> . <digits>
```

Number:

```
            123.456...
<digits> ::=
      <digit> | <digit><digits>
<digit> ::= 0|1|2|...|8|9
```

Regular Expression?

Some people, when confronted with a problem, think "I know, I'll use regular expressions." Now they have two problems.

Windows/DOS wildcard:

“*”: “*.docx”

Regular Expression:

- Basic string: hi
- * (0 or more times): 0*42 - 42, 042, 0042
- | (select): gray|grey
- () (priority): gr(a|e)y

Regular Expression:

- + (1 or more times): goo+gle - google, gooogle, goooogle
- . (0 or 1 times): colou?r - color, colour
- [] (character class): pl[abc]in - plain
- [^] (exclude from character class):

Java Identifier

- Can contain only “A-Z”, “a-z”, “0-9”, “_”, “$”
- Must not start with “0-9”

Java Identifier

- `[$_A-Za-z][$_A-Za-z0-9]*`

Thank you!