Recursion in Regex

or

How parse IBM Datastage dsx file?

Today we'll talk about recursion in regular expressions

Inspired by

Parsing JSON with a single regex

brian d _foy

http://blogs.perl.org/users/brian_d_foy
/2013/10/parsing-json-with-a-single-regex.html

and Buddy Burden

http://blogs.perl.org/users/buddy_burden
/2013/06/slideshows-in-vroom-so-noted.html

Actually I am delighted by Brian D Foy as

He plays with regexes

```
use v5.10;
$_ = <<'HERE';
Amelia said "I am a camel"
HERE

say "Matched [$+{said}]!" if m/
           ( ['"] )
           (?<said>.*?)
           ( ['"] )
           /x;
```

Here we know that% $ + this hash contains named

brackets

```
use v5.10;
$_ = <<'HERE';
Amelia said "I am a camel"
HERE

say "Matched [$+{said}]!" if m/
            ( ['"] )
            (?<said>.*?)
            ( \1 )
            /x;
```

```
use v5.10;
$_ = <<'HERE';
Amelia said "I am a camel"
HERE

say "Matched [$+{said}]!" if m/
            ( ['"] )
            (?<said>.*?)
            (?1)
            /x;
```

```
use v5.10;
$_ = <<'HERE';
Amelia said "I am a camel'
HERE

say "Matched [$+{said}]!" if m/
            ( ['"] )
            (?<said>.*?)
            (?1)
            /x;
```

```
use v5.18;
$_ = <<'HERE';
He said 'Amelia said "I am a camel"'
HERE

say "Matched [$+{said}]!" if m/
           (?<said>              #$1
           (?<quote>['"])
                  (?:
                      [^'"]++
                      |
                      (?<said> (?1) )
                  )*
           \g{quote}
           )
           /x;
```

```
use v5.18;
$_ = <<'HERE';
Out "Top 'Middle "Bottom" Middle' Out"
HERE

say "Matched [$+{said}]!" if m/
            (?<said>                #$1
            (?<quote>['"])
                   (?:
                       [^'"]++
                       |
                       (?R)
                   )*
            \g{quote}
            )
            (?{say "Inside regex: $+{said}"})
            /x;
```

```
use v5.18;
$_ = <<'HERE';
Out "Top 'Middle "Bottom" Middle' Out"
HERE


say "Matched [$+{said}]!" if m/
          (?(DEFINE)
              (?<QUOTE> ['"])
              (?<NOT_QUOTE> [^'"])
          )
          (?<said>
          (?<quote>(?&QUOTE))
                  (?:
                     (?&NOT_QUOTE)++
                     |
                     (?R)
                  )*
           \g{quote}
          )
          (?{ say "Inside regex: $+{said}" })
          /x;
```

```
use v5.18;
$_ = <<'HERE';
Out "Top 'Middle "Bottom" Middle' Out"
HERE

my @matches;

say "Matched [$+{said}]!" if m/
          (?(DEFINE)
              (?<QUOTE> ['"])
              (?<NOT_QUOTE> [^'"])
          )
          (?<said>
          (?<quote>(?&QUOTE))
                  (?:
                     (?&NOT_QUOTE)++
                     |
                     (?R)
                  )*
           \g{quote}
          )
          (?{ push @matches, $^N })
          /x;

say "\n".join '|',@matches;
```

```
use v5.18;
$_ = <<'HERE';
Out "Top 'Middle "Bottom" Middle' Out"
HERE

my @matches;

say "Matched!" if m/
        (?(DEFINE)
            (?<QUOTE_MARK> ['"])
            (?<NOT_QUOTE_MARK> [^'"])
            (?<QUOTE>
                (
                     (?<quote>(?&QUOTE_MARK))

                     (?:
                          (?&NOT_QUOTE_MARK)++
                          (?&QUOTE)
                     )*
                     \g{quote}
             )
            (?{ push @matches, $^N })
      )
     )
   (?&QUOTE)
        /x;
```

```
use v5.18;
$_ = <<'HERE';
Out "Top 'Middle "Bottom" Middle' Out"
HERE

my @matches;

say "Matched!" if m/
        (?(DEFINE)
            (?<QUOTE_MARK> ['"])
            (?<NOT_QUOTE_MARK> [^'"])
            (?<QUOTE>
                (
                    (?<quote>(?&QUOTE_MARK))

                    (?:
                        (?&NOT_QUOTE_MARK)++
                        (?&QUOTE)
                    )*
                    \g{quote}
             )
            (?{ [ @{$^R}, $^N ] })
      )
     )
   (?&QUOTE) (?{ @matches=@{ $^R } })
        /x;
```

Slideshows in Vim

* Hate using PowerPoint or HTML Slides for Talks?

Slideshows in Vim

* Hate using PowerPoint or HTML Slides for Talks?
* Use Vroom!

Slideshows in Vim

* Hate using PowerPoint or HTML Slides for Talks?
* Use Vroom!

* You can write you slides in Vim...
* ...and present them in Vim!

Getting Started

* Write a file called 'slides.vroom'.
  * Do this in a new directory.
* Run 'vroom vroom'.
* Voilà!

Navigation

* Hit <SPACE> to move forward.
* Hit <BACKSPACE> to go backwards.
* Hit 'Q' to quit.

```
# This is some Perl code.
# Notice the syntax highlighting.
# Run it with the <RR> vim command.
for my $word (qw(Vroom totally rocks!)) {
    print "$word\n";
}
```

Get Vroom!

* http://search.cpan.org/dist/Vroom/
* http://github.com/ingydotnet/vroom-pm/

Vroom as HTML

* http://ingydotnet.github.com/vroom-pm/

The End