

テキスト処理 第8回 (2007-06-26) 正規表現エンジン動作レポート説明

田中哲

産業技術総合研究所
情報技術研究部門

`akr@isc.senshu-u.ac.jp`

`http://staff.aist.go.jp/tanaka-
akira/textprocess-2007/`

前々回のレポートについて

- String#each があるので、obj.respond_to? :each では Array と String の区別がつかない
- String#each は行単位で分割して String を yield するので、each で分解して再帰すると無限再帰になる
- Ruby 1.9 では String#each が除去されていて、普段そっちを使っているのが気がつかなかった。ごめんなさい

レポート: 正規表現エンジンの使用

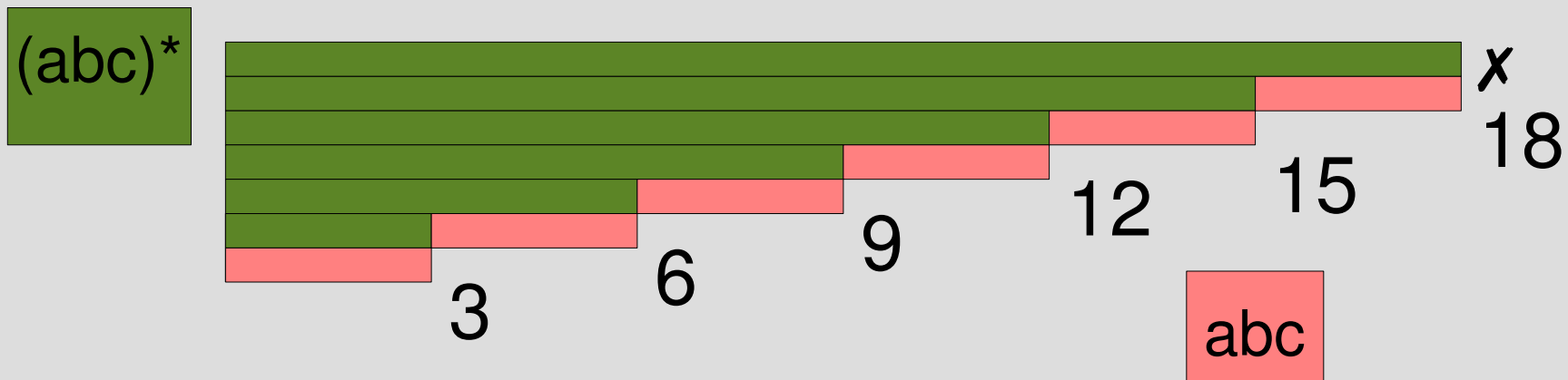
- 以下の正規表現を抽象構文木に変換し、今回の正規表現エンジンを用いて動かして動作を解説せよ
 - `/(abc)*abc/`
 - `/abc(abc)*`
 - `/(a|b)*abc/`
- 切 2007-06-26 16:20
- HIPLUS
- 拡張子が `txt` なテキストファイルにしてほしい

/(abc)*abc/の抽象構文木

```
[:cat,  
  [:rep,  
    [:cat, [:lit, "a"],  
      [:cat, [:lit, "b"],  
        [:lit, "c"]]]],  
  [:cat, [:lit, "a"],  
    [:cat, [:lit, "b"],  
      [:lit, "c"]]]]
```

/(abc)*abc/の動作

abcabcabcabcabcabc



matchstr(..., "abc"*6) #=> [18,15,12,9,6,3]

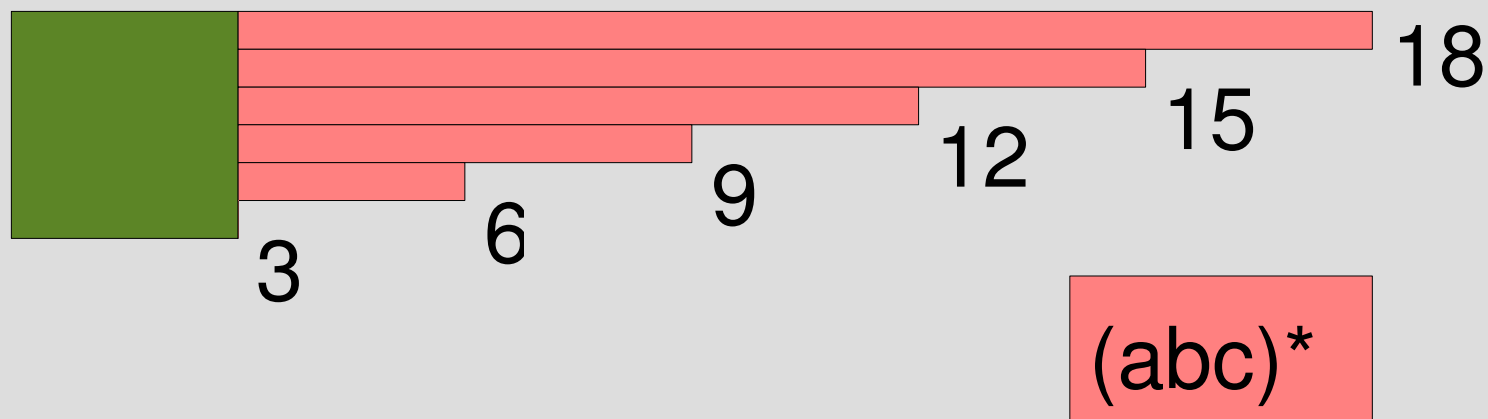
/abc(abc)* / の抽象構文木

```
[:cat,  
  [:cat, [:lit, "a"], [:cat, [:lit, "b"],  
    [:lit, "c"]]],  
[:rep,  
  [:cat, [:lit, "a"], [:cat, [:lit, "b"],  
    [:lit, "c"]]]]]
```

/abc(abc)*の動作

abcabcabcabcabcabc

abc



matchstr(..., "abc"*6) #=> [18,15,12,9,6,3]

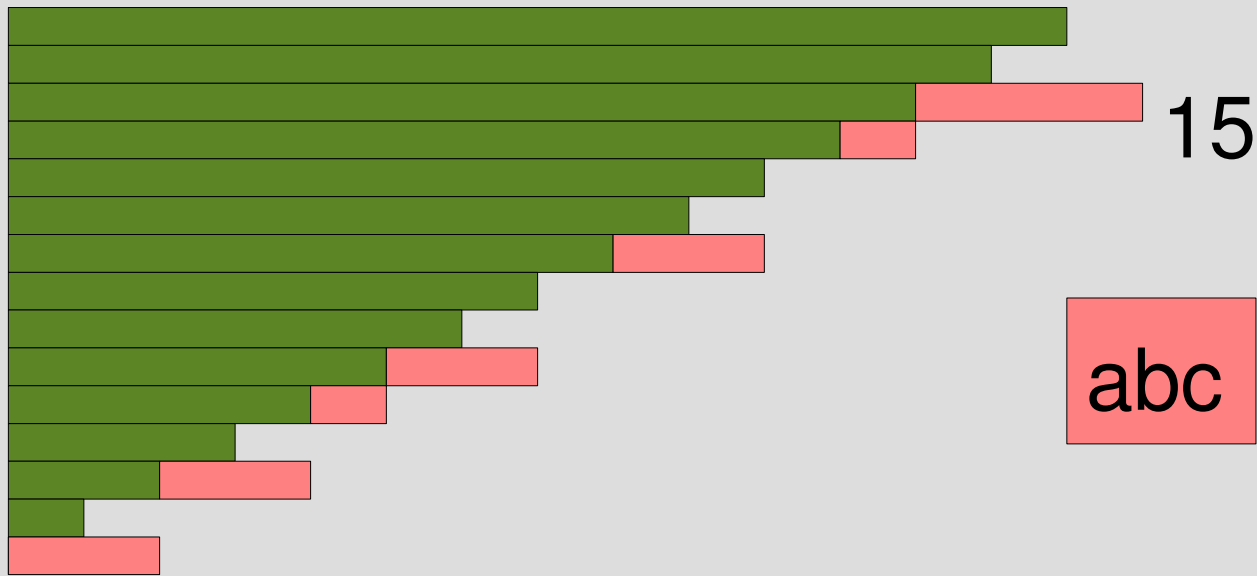
$/(a|b)^*abc/$ の抽象構文木

```
[ :cat,  
  [ :rep, [ :alt, [ :lit, "a" ], [ :lit, "b" ] ] ],  
  [ :cat, [ :lit, "a" ], [ :cat, [ :lit, "b" ],  
                                [ :lit, "c" ] ] ] ] ]
```


/(a|b)*abc/の動作

ababaabbabbaabcbaa

(a|b)*



matchstr(..., "...") #=> [15]

ざっと眺めた結果

- 解説がないもののがいくらかあった
 - 動作を追いかけるというのが意図
- 文章で解説しているものの中でも再帰の構造が不正確なものが見られた