

テキスト処理 第8回 (2008-06-10)

rx_exact と rx_include 解説

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-2008/`

レポート

- try を使って ematch_exact と同じ動作をする rx_exact を書く
- try を使って ematch_include と同じ動作をする rx_include を書く
- ユニットテストを配布するので確かめること
- ✖切 2008-06-10 12:00
- RENANDI
- 拡張子が txt なプレーンテキストで

関連資料

- `try` の説明は第6回の資料に説明がある
- `ematch_exact`, `ematch_include` は第5回の資料に説明がある

rx_exact の実行例

- `rx_exact("a", "a") #=> true`
- `rx_exact("a", "b") #=> false`
- `rx_exact(
 [:alt, [:cat, "c", [:cat, "a", "t"]],
 [:cat, "d", [:cat, "o", "g"]]], "dog")
#=> true`

rx_exact の実行例 (2)

- try を使うので繰り返しも扱える
- rx_exact([:rep, "a"], "aaaa") #=> true
- rx_exact([:rep, "a"], "aaaab") #=> false
- rx_exact([:cat, [:rep, "a"], "b"], "b") #=> true

String#split による文字列の分割

- rx_exact, rx_include の第2引数は文字列
- try の第2引数は文字の配列
- 文字列を文字単位に分割して配列にするには String#split を使う
- str.split(//)
- "apple".split(//) #=> ["a","p","p","l","e"]

rx_include の実行例

- `rx_include("a", "abc") #=> true`
- `rx_include("z", "abc") #=> false`
- `rx_include(
 [:alt, [:cat, "c", [:cat, "a", "t"]],
 [:cat, "d", [:cat, "o", "g"]]], "education")
#=> true`

rx_include の実行例 (2)

- try を使うので繰り返しも扱える
- rx_exact([:rep, "a"], "aaaa") #=> true
- rx_exact([:rep, "a"], "aaaab") #=> true
- rx_exact([:cat, [:rep, "a"], "b"], "b") #=> true

注意

- enumre は使わない
- 可能なら、途中で結果が分かったら中断する
- 日本語による解説もつけること

rx_exact(r, str)

```
def rx_exact(r, str)
  str = str.split('//')
  try(r, str, 0) {|e|
    return true if str.length == e
  }
  false
end
```

- str の最初 (0) からマッチ
- str の終わり (str.length) でちょうどマッチしたら成功
- ひとつもそういうマッチがなければ失敗

rx_include(r, str)

```
def rx_include(r, str)
  str = str.split("//)
  0.upto(str.length) {|b|
    try(r, str, b) {|e|
      return true
    }
  }
  false
end
```

- 開始位置を 0 から str.length まで順に変えてマッチを試す
- マッチが一回でも成功したら成功
- 結局成功しなかったら失敗

ざっとレポートを見た結果

- 0 や空文字列の特別扱い
rx_exactやrx_includeで特別扱いする理由はない
- rx_exact では開始位置を変えながら試す必要はない
文字列全体にマッチするということはいつも文字列
先頭からマッチが始まる