2.5 正規表現と正規集合

- この節で分かること
  - 正規表現の（数学的な）定義と意味づけ
    - 正規表現は文字列処理において重要な概念
    - UNIXシステムやプログラミング言語（Perl、Ruby等）で用いられる正規表現は（実用的に）拡張されている
  - 有限オートマトンと正規表現とが、言語を定義する能力において同等である
    - 任意の正規表現 r に対し、r で定義される言語 L を受理する有限オートマトンが存在する
    - その逆もいえる

Unix等における正規表現

- ファイル名の正規表現
  - > rm ＊.txt
  - > cp Important[0-9].doc
- 検索ツールGrepの正規表現
  - > grep –E “for.+(256|CHAR_SIZE)” ＊.c
- プログラミング言語Perlの正規表現
  - $line = m|^http://.+¥.jp/.+$|

正規表現の定義

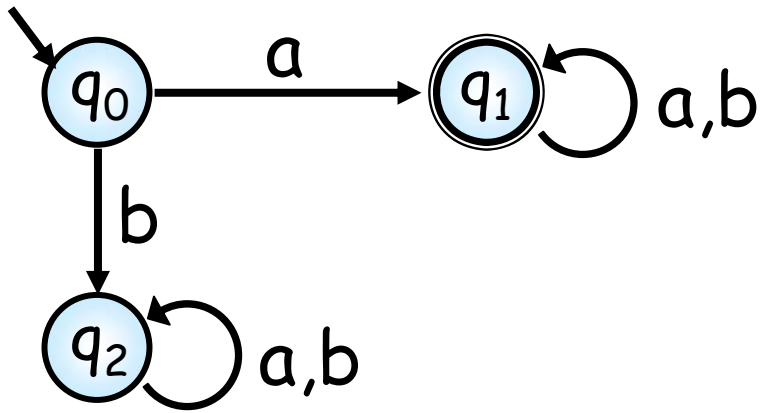
- アルファベットΣ上の正規表現とは
  $A=\{), (, \phi, \cdot, +, *\}$ を用いて次のように定義される。
  - (1) ϕ とΣの要素は正規表現である
  - (2) α と β が正規表現ならば $(\alpha \cdot \beta)$ も正規表現である
  - (3) α と β が正規表現ならば $(\alpha + \beta)$ も正規表現である
  - (4) α が正規表現ならば α^* も正規表現である
  - (5) 上から導かれるものだけが正規表現である

- 例: (a･(a+b)*)

正規表現の意味づけ

- 正規表現を Σ^* の部分集合に写像する
  - (i) $\|\phi\| = \phi$
  - (ii) $a \in \Sigma$ に対して $\|a\| = \{a\}$
  - (iii) 正規表現 α, β に対して $\|(\alpha \cdot \beta)\| = \|\alpha\| \cdot \|\beta\|$
  - (iv) 正規表現 α, β に対して $\|(\alpha + \beta)\| = \|\alpha\| + \|\beta\|$
  - (v) 正規表現 α に対して $\|\alpha^*\| = \|\alpha\|^*$

- 例 :
  - $\|(a \cdot (a+b)^*)\|$
    $= \{ax \mid x \in \{a,b\}^*\}$



2.5節の構成（同等の証明）

- 定理2.10 （正規表現→正規集合）　割と簡単
  - 補題2.2(1)（ L_1,とL_2が正規集合なら和$L_1 \cup L_2$は正規集合）
  - 補題2.6（空集合は正規集合）
  - 補題2.7（任意の$a \in \Sigma$について{a}は正規集合）
  - 補題2.8（ L_1,とL_2が正規集合なら積$L_1 \cdot L_2$は正規集合）
  - 補題2.9（ Lが正規集合なら閉包L*は正規集合）

- 定理2.12 （正規集合→正規表現）　結構たいへん
  - 補題2.11　　$(\| \alpha_{ij}^{(k)} \| = R_{ij}^{(k)})$

正規表現$\alpha_{i,j}^{(k)}$の定義

- 有限オートマトン　$M = (K, \Sigma, \delta, q_1, F)$において
  $K=\{q_1, q_2, \ldots, q_k\}$とし，各i, j, kに対して正規表現$\alpha_{i,j}^{(k)}$を以下のように定める．

$$\alpha_{i,j}^{(0)} = \begin{cases} \sum a & (i \neq j \text{のとき}) \\ \sum a + \phi^* & (i = j \text{のとき}) \end{cases}$$

$$\alpha_{i,j}^{(k)} = \alpha_{i,j}^{(k-1)} + \alpha_{i,k}^{(k-1)} \cdot \left(\alpha_{k,k}^{(k-1)}\right)^* \cdot \alpha_{k,j}^{(k-1)}$$

$\sum a$は$\delta(q_i, a) = q_j$となる$a \in \Sigma$の和

例2.7

- 図2.9の有限オートマトンに対する正規表現
  - $\gamma = \alpha_{11}^{(3)} + \alpha_{13}^{(3)}$

    $\alpha_{11}^{(3)} = \alpha_{11}^{(2)} + \alpha_{13}^{(2)} \cdot (\alpha_{33}^{(2)})^* \cdot \alpha_{31}^{(2)}$

    $\alpha_{11}^{(2)} = \alpha_{11}^{(1)} + \alpha_{12}^{(1)} \cdot (\alpha_{22}^{(1)})^* \cdot \alpha_{21}^{(1)}$

    $\alpha_{11}^{(1)} = \alpha_{11}^{(0)} + \alpha_{11}^{(0)} \cdot (\alpha_{11}^{(0)})^* \cdot \alpha_{11}^{(0)}$

    $=(a+\phi^*)+(a+\phi^*)\cdot(a+\phi^*)^*\cdot(a+\phi^*)$

    $=a^*$

    $\alpha_{12}^{(1)} = \alpha_{12}^{(0)} + \alpha_{11}^{(0)} \cdot (\alpha_{11}^{(0)})^* \cdot \alpha_{12}^{(0)} = b+(a^*\cdot b)$

    $\alpha_{22}^{(1)} = \alpha_{22}^{(0)} + \alpha_{21}^{(0)} \cdot (\alpha_{11}^{(0)})^* \cdot \alpha_{12}^{(0)} = a\cdot a^*\cdot b$

    $\alpha_{21}^{(1)} = \alpha_{21}^{(0)} + \alpha_{21}^{(0)} \cdot (\alpha_{11}^{(0)})^* \cdot \alpha_{11}^{(0)} = a\cdot a^*$

    ・・・
  - $\gamma = a^*+a^*(baa^*)^*+a^*(baa^*)^*bbb^*+\cdots$

言語$R_{i,j}^{(k)}$の定義

- 各i, j, kに対して言語$R_{i,j}^{(k)}$を以下のように定める.

$$R_{i,j}^{(k)} = \left\{ x \in \Sigma^* \;\middle|\; \begin{array}{l} \delta(q_i, x) = q_j \text{かつ} x = yz, y, z \in \Sigma^+ \text{なる} \\ y \text{に対して} \delta(q_i, y) = q_\lambda \text{ならば} \lambda \leq k \end{array} \right\}$$

途中でkより番号の大きい状態を通らずに
状態q_iからq_jまで遷移させるような語の全体

有限オートマトン⇒正規表現

- 補題2.11
  $||\alpha_{ij}^{(k)}|| = R_{ij}^{(k)}$

- 定理2.12
  任意の正規集合Lに対して$||\gamma||$ =Lとなる正規表現γが存在する.