計算理論　第３，４回

1. 有限オートマトン
2. 正則表現と正則言語
   - 正則表現
   - 有限オートマトンから正則表現へ
   - 正則表現から有限オートマトンへ
   - 正則表現の代数的法則
3. 正則言語の性質
4. 文脈自由文法と言語
5. プッシュダウン・オートマトン
6. 文脈自由言語の性質
7. チューリングマシン

テキスト
3.1～3.4節

本資料の学習目標

- 正則表現を，例を用いて説明できる
- 正則表現が与えられたとき，その言語を説明できる
- 指定された言語を正則表現で表すことができる
- 有限オートマトンが与えられたとき，その言語を正則表現で表現できる
- 与えられた正則表現が表す言語を受理する有限オートマトンを設計できる
- 正則表現の代数的法則を，例を用いて説明できる

3.1 正則表現

- **正則表現**（**正規表現**，regular expression）
  - 言語（語の集合）を表す
  - 情報科学分野ではよく利用される表現法
    - 検索文字列の指定
    - UNIX 系ツール（grep コマンドなど）
    - コンパイラの字句解析に用いる lex や flex などのコマンド
- **正則表現が表す言語＝有限オートマトンが認識する言語**
  - これは，後で証明する

3.1.1 正則表現の演算

- L, M : Σ 上の言語（語の集合）
- **集合和** $L \cup M$
  - $L \cup M = \{w \in \Sigma^* \mid w \in L$ **または** $w \in M\}$
- **連接** LM
  - $LM = \{\alpha\beta \in \Sigma^* \mid \alpha \in L,\ \beta \in M\}$
  - $L = \{001, 10, 111\}$，$M = \{\epsilon,\ 001\}$ のとき
    $LM = \{001, 10, 111, 001001, 10001, 111001\}$
- **閉包(スター閉包，クリーネ(Kleene)閉包)** L^*
  - $L^* = \bigcup_{i \geq 0} L^i$
    - $L^0 = \{\epsilon\}$，$L^k = LL^{k-1}$　（$k \geq 1$）
  - $L = \{0, 11\}$ のとき，$L^* = \{\epsilon, 0, 11, 00, 011, 110, 1111, \ldots\}$

3.1.2 正則表現の構成 (1)

- **アルファベット Σ 上の正則表現**
  - ϵ, $\emptyset$, (,), $+$, $*$ は Σ に含まれない記号
- **以下の構成法で構成される表現のみが正則表現**
  $L(E)$：正則表現 E が表す言語
  - **基礎**
    1. ϵ, $\emptyset$ は正則表現
       - $L(\epsilon) = \{\epsilon\}$, $L(\emptyset) = \emptyset$
    2. $a \in \Sigma$ に対し，a は正則表現
       - $L(a) = \{a\}$

3.1.2 正則表現の構成 (2)

- **再帰**
  1. E, F が正則表現 $\Rightarrow$ $E + F$ も正則表現
     - $L(E + F) = L(E) \cup L(F)$
  2. E, F が正則表現 $\Rightarrow$ EF も正則表現
     - $L(EF) = L(E)L(F)$
  3. E が正則表現 $\Rightarrow E^*$ も正則表現
     - $L(E^*) = L(E)^*$
  4. E が正則表現 $\Rightarrow (E)$ も正則表現
     - $L((E)) = L(E)$

例3.2 正則表現

- **正則表現**：0, 1
  - $L(0) = \{0\}$, $L(1) = \{1\}$
- **正則表現**：01
  - $L(01) = \{0\}\{1\} = \{01\}$
- **正則表現**：$(01)^*$
  - $L\big((01)^*\big) = \{01\}^* = \{\epsilon, 01, 0101, 010101, 01010101, \dots\}$
- **正則表現**：$(01)^* + (10)^* + 0(10)^* + 1(01)^*$
  - $L\big((01)^* + (10)^* + 0(10)^* + 1(01)^*\big)$
    $= \{w \mid w$ は 0 と 1 が交互に現れる語$\}$
- **正則表現**：$(\epsilon + 1)(01)^*(\epsilon + 0)$
  - 上と同じ言語を表す

3.1.3 正則表現中の演算の順序

- **正則表現の括弧の省略**
  - **優先順位が高い順に**
    - スター演算，連接，集合和
    - $01^* + 1 = (0(1)^*) + 1$
  - **連接は結合法則を満たす**
    - $012 = (01)2 = 0(12)$

3.2 有限オートマトンと正則表現

- **有限オートマトンが受理する言語のクラスと、正則表現で定義できる言語のクラスは一致する**

  (1) **任意の有限オートマトン A に対し，$L(A)$ を表す正則表現 R が存在する（3.2.1節，3.2.2節）**
  - A を DFA と仮定してよい

  (2) **任意の正則表現 R に対し，$L(R)$ を受理する有限オートマトンが存在する（3.2.3節）**
  - A を ϵ-NFA と仮定してよい

3.2.1 DFA から正則表現へ

【定理3.4】DFAの受理集合は正則表現で記述できる．すなわち，DFA A に対して $L = L(A)$ のとき，$L = L(R)$ を満たす正則表現 R がある．

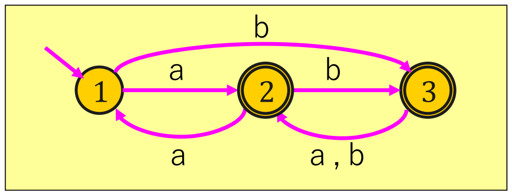
- **動的計画法の考え方で証明**
- DFA $A = (Q, \Sigma, \delta, q_0, F)$
  $Q = \{1, 2, \ldots, n\}$, $q_0 = 1$ とする
- **正則表現** $R_{i,j}^{(k)}$: $L\left(R_{i,j}^{(k)}\right) = \{\alpha = a_1 a_2 \ldots a_m \mid \hat{\delta}(i, \alpha) = j \wedge \hat{\delta}(i, a_1 a_2 \ldots a_h) \le k \ (1 \le h < m)\}$

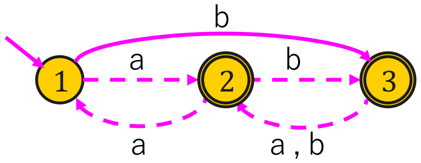
状態 i から j に遷移

途中に経由する状態は$\{1, 2, \ldots, k\}$の状態のみ

DFA から正則表現へ：$R_{i,j}^{k}$ (1)

- **正則表現** $R_{i,j}^{(k)}$: $L\left(R_{i,j}^{(k)}\right) = \{\alpha = a_1 a_2 \ldots a_m \mid \hat{\delta}(i, \alpha) = j \wedge \hat{\delta}(i, a_1 a_2 \ldots a_h) \le k \ (1 \le h < m)\}$

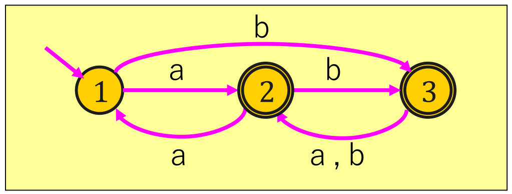


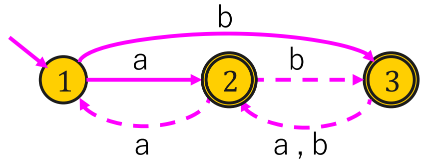


$R_{1,3}^{(0)} = b$ **状態経由なし**

DFA から正則表現へ：$R_{i,j}^{k}$ (2)

- **正則表現** $R_{i,j}^{(k)}$: $L\left(R_{i,j}^{(k)}\right) = \{\alpha = a_1 a_2 \ldots a_m \mid \hat{\delta}(i, \alpha) = j \wedge \hat{\delta}(i, a_1 a_2 \ldots a_h) \le k \ (1 \le h < m)\}$

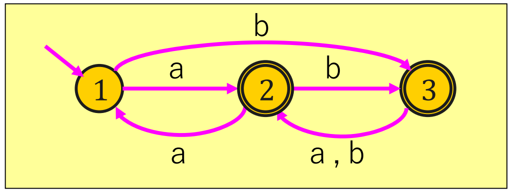


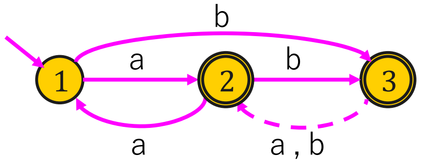


$R_{1,3}^{(1)} = b$ **状態 1 のみ経由可能**

DFA から正則表現へ：$R^{k}_{i,j}$ (3)

- **正則表現** $R^{(k)}_{i,j}$ ：$L\left(R^{(k)}_{i,j}\right) = \{\alpha = a_1a_2\dots a_m \mid \hat{\delta}(i,\ \alpha) = j \wedge$
  $\hat{\delta}(i,\ a_1a_2\dots a_h) \le k\ \ (1 \le h < m)\}$

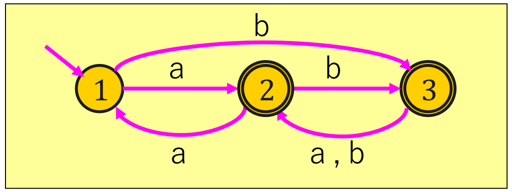


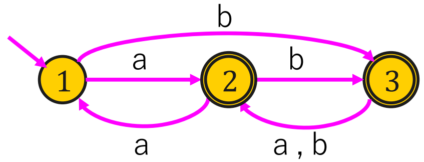


$R^{(2)}_{1,3} = a^*b$ **状態 1, 2 のみ経由可能**

DFA から正則表現へ：$R^{k}_{i,j}$ (4)

- **正則表現** $R^{(k)}_{i,j}$ ：$L\left(R^{(k)}_{i,j}\right) = \{\alpha = a_1a_2\dots a_m \mid \hat{\delta}(i,\ \alpha) = j \wedge$
  $\hat{\delta}(i,\ a_1a_2\dots a_h) \le k\ \ (1 \le h < m)\}$





$R^{(3)}_{1,3} = a^*b\big((a+b)a^*b\big)^*$ **全状態経由可能**

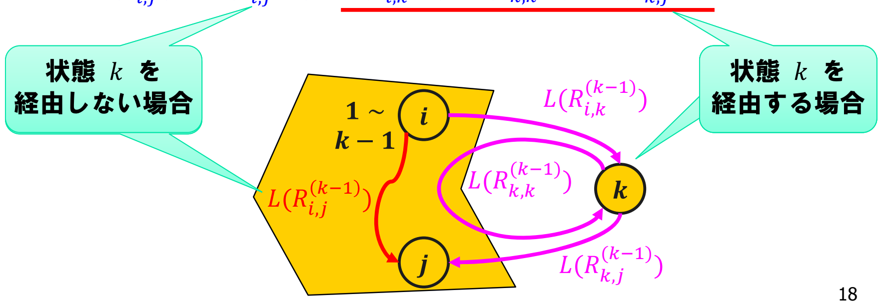
DFA から正則表現へ： $R^{k}_{i,j}$ の帰納的定義

- **正則表現** $R^{(k)}_{i,j}$ ：$L\left(R^{(k)}_{i,j}\right) = \{\alpha = a_1a_2\dots a_m \mid \hat{\delta}(i,\ \alpha) = j \wedge$
  $\hat{\delta}(i,\ a_1a_2\dots a_h) \le k\ \ (1 \le h < m)\}$

$L(R^{(0)}_{i,j}) = \{a \in \Sigma \mid \delta(i,\ a) = j\} \quad (i \ne j)$

$L(R^{(0)}_{i,i}) = \{a \in \Sigma \mid \delta(i,\ a) = i\} \cup \{\epsilon\}$

$L(R^{(k)}_{i,j}) = L(R^{(k-1)}_{i,j}) \cup \underline{L(R^{(k-1)}_{i,k}) \cdot L(R^{(k-1)}_{k,k})^* \cdot L(R^{(k-1)}_{k,j})}$



DFA から正則表現へ

$L(R^{(0)}_{i,j}) = \{a \in \Sigma \mid \delta(i,\ a) = j\} \quad (i \ne j)$

$L(R^{(0)}_{i,i}) = \{a \in \Sigma \mid \delta(i,\ a) = i\} \cup \{\epsilon\}$

$L(R^{(k)}_{i,j}) = L(R^{(k-1)}_{i,j}) \cup L(R^{(k-1)}_{i,k}) \cdot L(R^{(k-1)}_{k,k})^* \cdot L(R^{(k-1)}_{k,j})$

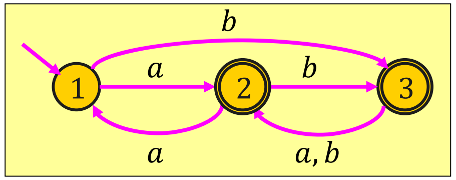
$L(R^{(k)}_{i,j})$**は，正則表現で表現できる**

- $A = (Q,\ \Sigma,\ \delta,\ q_0,\ F) \quad (Q = \{1,\ 2, \dots,\ n\},\ \ q_0 = 1)$
  が認識する言語

  $L(A) = \cup_{p \in F}\ L(R^{(n)}_{1,p})$

  は，正則表現で表現できる

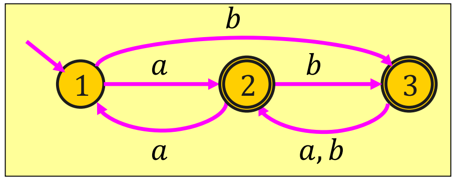
適用例 (1)



$R_{1,1}^{(0)}$ ： ϵ	$R_{1,2}^{(0)}$ ： a	$R_{1,3}^{(0)}$ ： b
$R_{2,1}^{(0)}$ ： a	$R_{2,2}^{(0)}$ ： ϵ	$R_{2,3}^{(0)}$ ： b
$R_{3,1}^{(0)}$ ： $\emptyset$	$R_{3,2}^{(0)}$ ： $a+b$	$R_{3,3}^{(0)}$ ： ϵ
$R_{1,1}^{(1)}$ ： ϵ	$R_{1,2}^{(1)}$ ： a	$R_{1,3}^{(1)}$ ： b
$R_{2,1}^{(1)}$ ： a	$R_{2,2}^{(1)}$ ： $\epsilon + aa$	$R_{2,3}^{(1)}$ ： $b+ab$
$R_{3,1}^{(1)}$ ： $\emptyset$	$R_{3,2}^{(1)}$ ： $a+b$	$R_{3,3}^{(1)}$ ： ϵ

$R_{2,3}^{(0)} +$
$R_{2,1}^{(0)} \cdot R_{1,1}^{(0)*} \cdot R_{1,3}^{(0)}$

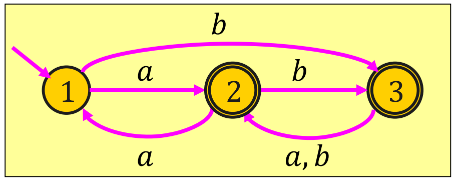
適用例(2)



$R_{1,1}^{(2)}: (aa)^*$	$R_{1,2}^{(2)}: a(aa)^*$	$R_{1,3}^{(2)}: a^*b$
$R_{2,1}^{(2)}: a(aa)^*$	$R_{2,2}^{(2)}: (aa)^*$	$R_{2,3}^{(2)}: a^*b$
$R_{3,1}^{(2)}: (a+b)(aa)^*a$	$R_{3,2}^{(2)}: (a+b)(aa)^*$	
$R_{3,3}^{(2)}: \epsilon + (a+b)a^*b$		

$R_{3,3}^{(1)} + R_{3,2}^{(1)} \cdot R_{2,2}^{(1)*} \cdot R_{2,3}^{(1)}$
$= \epsilon + (\mathrm{a}+\mathrm{b}) \cdot (\epsilon + aa)^* \cdot (b+ab)$
$= \epsilon + (a+b) \cdot (aa)^* \cdot (b+ab) = \epsilon + (a+b)a^*b$

適用例(3)



$R_{1,2}^{(3)}$: $R_{1,2}^{(2)} + R_{1,3}^{(2)} \cdot R_{3,3}^{(2)*} \cdot R_{3,2}^{(2)}$

$a(aa)^* + a^*b(\epsilon + (a+b)a^*b)^*(a+b)(aa)^*$

$= a(aa)^* + \underline{a^*b\big((a+b)a^*b\big)^*}(a+b)(aa)^*$

$R_{1,3}^{(3)}$: $\underline{a^*b\big((a+b)a^*b\big)^*}$

オートマトンが受理する言語： $R_{1,2}^{(3)} + R_{1,3}^{(3)}$

$a(aa)^* + a^*b\big((a+b)a^*b\big)^*(\epsilon + (a+b)(aa)^*)$

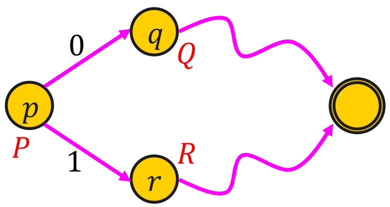
DFA から正則表現へ

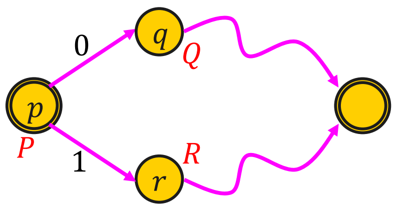
- **動的計画法の考え方**
  - **手間： $O(n^3)$　n ：オートマトンの状態数**
    - $R_{i,j}^{(k)}$ $(1 \le i \le n, 1 \le j \le n, 0 \le k \le n)$ **をすべて求める**
  - **NFA, ϵ-NFA にも適用可能**
- **正則表現の方程式（3.2.2節の代わりに）** テキストにない内容
  状態遷移から作った方程式を解く
  各状態から始まる受理される言語を方程式で表す

テキストにない内容

正則表現の方程式：アイデア

- **各状態 p に対し，p から受理状態に到達する語の集合を表す変数 P**





方程式 $P = 0Q + 1R$　　　**方程式** $P = 0Q + 1R + \epsilon$

テキストにない内容

正則表現の方程式：基本定理１（１）

- $A, B, X : \Sigma$ **上の正則表現**

  ただし， $\epsilon \notin L(A)$ **とする**

  $$X = AX + B \iff X = A^*B$$

- $X = A^*B \Rightarrow X = AX + B$

  $X = A^*B$ **を代入**

  $A^*B = AA^*B + B = A^+B + B = A^*B$

テキストにない内容

正則表現の方程式：基本定理１（２）

- $X = AX + B \Rightarrow X = A^*B$

  上より， $L(A^*B) \subseteq L(X)$ **は明らか**

  $L(X) - L(A^*B) \neq \emptyset$ **と仮定（背理法）**

  $w : L(X) - L(A^*B)$ **の最短の語**

  $w \notin L(A^*B)$ **より，** $w \notin L(B)$

  $X = AX + B$ **より，** $w \in L(AX)$

  $w = \alpha x \quad (\alpha \in L(A),\ x \in L(X))$

  $\alpha \neq \epsilon$ **より，** $|x| < |w|$　　（ $\epsilon \notin L(A)$ **を思い出そう**）

  w **が** $L(X) - L(A^*B)$ **の最短の語なので** $x \in L(A^*B)$

  $w = \alpha x \in L(A) \cdot L(A^*B) \subseteq L(A^*B)$　**矛盾**

テキストにない内容

正則表現の方程式：基本定理１（注）

- $A, B, X : \Sigma$ **上の正則表現**

  $X = AX + B$

  $\epsilon \in L(A)$ **ならば，解は一通りとは限らない**

- $X = (a + \epsilon)X + b$

  $X = a^*b,\quad X = (a + b)^*$ **はいずれも解**

テキストにない内容

正則表現の方程式：基本定理２ (1)

- $A,\ X$：Σ 上の正則表現
  ただし，$\epsilon \notin L(A)$ とする
  $X = AX \iff X = \emptyset$

- $X = \emptyset \Rightarrow X = AX$
  $X = \emptyset$ を代入
  $\emptyset = A\emptyset = \emptyset$

テキストにない内容

正則表現の方程式：基本定理２ (2)

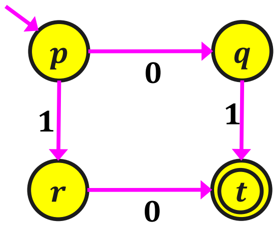
- $X = AX \Rightarrow X = \emptyset$
  $X \neq \emptyset$ を仮定（背理法）
  w ： $L(X)$ に属する最短の語
  $w \in L(X) = L(AX)$ より，
  $w = \alpha x \quad (\alpha \in L(A),\ x \in L(X))$
  $\alpha \neq \epsilon$ より，$|x| < |w|$　（ $\epsilon \notin L(A)$ を思い出そう）
  $x \in L(X)$ より，
  w が $L(X)$ に属する最短の語に矛盾

テキストにない内容

正則表現の方程式：例１ (1)

	0	1
$\rightarrow p$	q	r
q	−	t
r	t	−
$*\ t$	−	−

$P = 0Q + 1R$
$Q = 1T$
$R = 0T$
$T = \epsilon$　（受理状態）



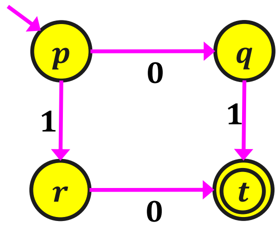
P：状態 p から受理状態に到達する語の集合を表す正則表現
$L(P) = \{\alpha \in \Sigma^* \mid \hat{\delta}(p,\alpha) \cap F \neq \emptyset\}$
$Q,\ R,\ T$ も同様

テキストにない内容

正則表現の方程式：例１ (2)

	0	1
$\rightarrow p$	q	r
q	−	t
r	t	−
$*\ t$	−	−

$P = 0Q + 1R$
$Q = 1T$
$R = 0T$
$T = \epsilon$　（受理状態）



$T = \epsilon$ を $Q,\ R$ の方程式に代入： $Q = 1,\ R = 0$
$Q = 1,\ R = 0$ を P の方程式に代入： $P = 01 + 10$

テキストにない内容

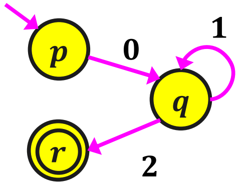
正則表現の方程式：例２

	0	1	2
$\to p$	q	−	−
q	−	q	r
$*\ r$	−	−	−

$P = 0Q$

$Q = 1Q + 2R$

$R = \epsilon$ （**受理状態**）



$R = \epsilon$ を Q の方程式に代入： $Q = 1Q + 2$

基本定理１： $Q = 1^*2$

$Q = 1^*2$ を P の方程式に代入： $P = 01^*2$

テキストにない内容

正則表現の方程式：例３

	0	1
$\to * p$	s	q
q	q	r
r	s	p
s	s	s

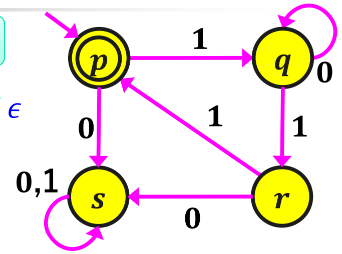
受理状態

$P = 0S + 1Q + \epsilon$

$Q = 0Q + 1R$

$R = 0S + 1P$

$S = 0S + 1S = (0 + 1)S$



基本定理２： $S = \emptyset$

$S = \emptyset$ を R の方程式に代入： $R = 1P$

$R = 1P$ を Q の方程式に代入： $Q = 0Q + 11P$

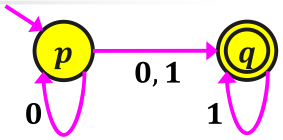
基本定理１： $Q = 0^*11P$

$S = \emptyset$，$Q = 0^*11P$ を P の方程式に代入： $P = 10^*11P + \epsilon$

基本定理１： $P = (10^*11)^*$

テキストにない内容

正則表現の方程式：例４



	0	1
$\to p$	p, q	q
$*\ q$	−	q

$P = 0P + 0Q + 1Q$

$Q = 1Q + \epsilon$ （**受理状態**）

基本定理１： $Q = 1^*$

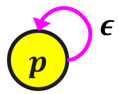
$Q = 1^*$ を P の方程式に代入： $P = 0P + (0 + 1)1^*$

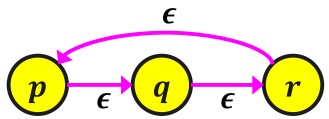
基本定理１： $P = 0^*(0 + 1)1^*$

テキストにない内容

基本定理を思い出してみると

- **基本定理１**
  - A, B, X：Σ 上の正則表現．ただし，$\epsilon \notin L(A)$ とする．
  - $X = AX + B \iff X = A^*B$
- **基本定理２**
  - A, X：Σ 上の正則表現．ただし，$\epsilon \notin L(A)$ とする．
  - $X = AX \iff X = \emptyset$
- **条件** $\epsilon \notin L(A)$
  - ϵ 動作だけで元の状態に戻らなければ満たす





本資料の学習目標

目標を達成できたか確認してみよう（復習も含めて）

- 正則表現を，例を用いて説明できる
- 正則表現が与えられたとき，その言語を説明できる
- 指定された言語を正則表現で表すことができる
- 有限オートマトンが与えられたとき，その言語を正則表現で表現できる
- 与えられた正則表現が表す言語を受理する有限オートマトンを設計できる
- 正則表現の代数的法則を，例を用いて説明できる

計算理論　第３，４回

1. 有限オートマトン
2. 正則表現と正則言語
   - 正則表現
   - 有限オートマトンから正則表現へ
   - 正則表現から有限オートマトンへ
   - 正則表現の代数的法則
3. 正則言語の性質
4. 文脈自由文法と言語
5. プッシュダウン・オートマトン
6. 文脈自由言語の性質
7. チューリングマシン

テキスト 3.1～3.4節

3.2 有限オートマトンと正則表現

- 有限オートマトンが受理する言語のクラスと、正則表現で定義できる言語のクラスは一致する

(1) 任意の有限オートマトン A に対し，$L(A)$ を表す正則表現 R が存在する（3.2.1節，3.2.2節）
   - A を DFA と仮定してよい

前回の内容

(2) 任意の正則表現 R に対し，$L(R)$ を受理する有限オートマトンが存在する（3.2.3節）
   - A を ϵ-NFA と仮定してよい

3.2.3 正則表現からオートマトンへの変換

【定理3.7】正則表現で表される言語はすべて有限オートマトンの受理集合である．

証明の方針

- 正則表現 R の構造に関する帰納法
- 任意の正則表現 R に対して，$L(E) = L(R)$ となる ϵ-NFA E を構成する

復習：正則表現の構成 (1)

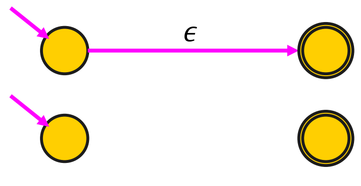
- アルファベット Σ 上の正則表現
  - ϵ, $\emptyset$, (,), $+$, $*$ は Σ に含まれない記号
- 以下の構成法で構成される表現のみが正則表現
  $L(E)$：正則表現 E が表す言語
  - 基礎
    1. ϵ, $\emptyset$ は正則表現
       - $L(\epsilon) = \{\epsilon\}$, $L(\emptyset) = \emptyset$
    2. $a \in \Sigma$ に対し，a は正則表現
       - $L(a) = \{a\}$

復習：正則表現の構成 (2)

- 再帰
  1. E, F が正則表現 $\Rightarrow$ $E+F$ も正則表現
     - $L(E+F) = L(E) \cup L(F)$
  2. E, F が正則表現 $\Rightarrow$ EF も正則表現
     - $L(EF) = L(E)L(F)$
  3. E が正則表現 $\Rightarrow$ E^* も正則表現
     - $L(E^*) = L(E)^*$
  4. E が正則表現 $\Rightarrow$ (E) も正則表現
     - $L((E)) = L(E)$

正則表現から ϵ-NFA への変換：基礎

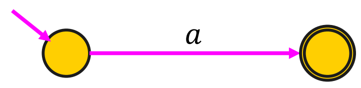
- 帰納法の基礎　＊本変換法では受理状態は一つだけ
  1. ϵ, $\emptyset$ は正則表現
     - ϵ
     - $\emptyset$



$\{\epsilon\}$ を受理するϵ-NFA

$\emptyset$ を受理するϵ-NFA

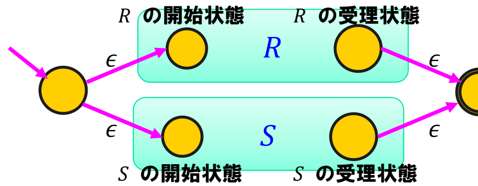
  2. $a \in \Sigma$ に対し，a は正則表現



$\{a\}$ を受理するϵ-NFA

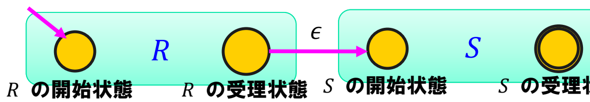
正則表現から ϵ-NFA への変換：帰納 (1)

- 帰納　＊本変換法では受理状態は一つだけ
  1. R, S が正則表現 $\Rightarrow$ $R+S$ も正則表現



$L(R) \cup L(S)$ を受理するϵ-NFA

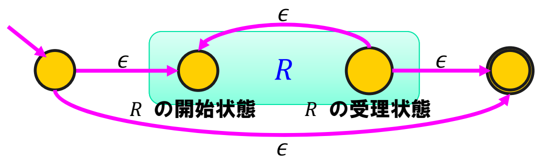
  2. R, S が正則表現 $\Rightarrow$ RS も正則表現



$L(R) \cdot L(S)$ を受理するϵ-NFA

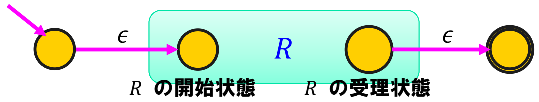
正則表現から ϵ-NFA への変換：帰納(2)

- 帰納
  3. R が正則表現 ⇒ R^* も正則表現



$L(R)^*$ を受理するϵ-NFA

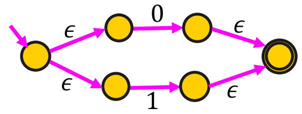
  4. R が正則表現 ⇒(R) も正則表現



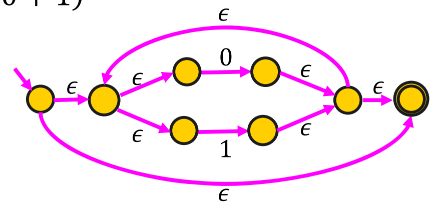
$L(R)$ を受理するϵ-NFA

正則表現から ϵ-NFA への変換：例3.8(1)

- $(0+1)^*1(0+1)$の ϵ-NFA への変換
  - $0+1$

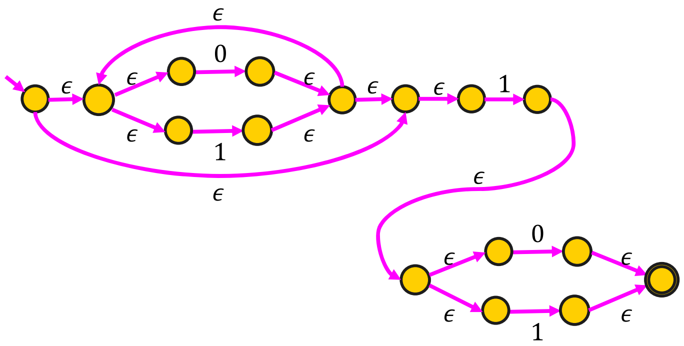


  - $(0+1)^*$



正則表現から ϵ-NFA への変換：例3.8(2)

- $(0+1)^*1(0+1)$の ϵ-NFA への変換



3.4 正則表現の代数的法則
3.4.1 結合法則と交換法則

3.3 正則表現の応用
は読んでおくこと

正則表現の簡単化のために利用できる法則

- L, M, N：任意の正則表現
- **集合和に関する交換法則**

  $L + M = M + L$
- **集合和に関する結合法則**

  $(L+M)+N = L+(M+N)$
- **連接に関する結合法則**

  $(LM)N = L\,(MN)$
- **連接に関する交換法則は成立しない**

  $LM = ML$ は一般には成立しない

3.4.2 単位元と零元

- L：任意の正則表現
- 集合和に関する単位元

  $\emptyset + L = L + \emptyset = L$
- 連接に関する単位元

  $\epsilon L = L\epsilon = L$
- 連接に関する零元

  $\emptyset L = L\emptyset = \emptyset$

3.4.3 分配法則(1)

- L, M, N：任意の正則表現
- 集合和に対する連接の左分配法則（定理3.11）

  $L(M+N) = LM + LN$
- 集合和に対する連接の右分配法則

  $(M+N)L = ML + NL$

3.4.3 分配法則(2)

- 集合和に対する連接の左分配法則（定理3.11）

  $L(M+N) = LM + LN$

証明

- $L(M+N) \subseteq LM + LN$ の証明
  - $w \in L(M+N)$ ならば，$w = xy\ (x \in L, y \in M+N)$ と書ける
    $y \in M$ ならば，$w = xy \in LM$
    $y \in N$ ならば，$w = xy \in LN$
    従って，$w \in LM + LN$
- $LM + LN \subseteq L(M+N)$ の証明
  - $w \in LM + LN$ ならば，$w \in LM$ または $w \in LN$
    $w \in LM$ ならば，　$w = xy\ (x \in L, y \in M)$ と書け，$w \in L(M+N)$
    $w \in LN$ ならば，　$w = xy\ (x \in L, y \in N)$ と書け，$w \in L(M+N)$

3.4.3 分配法則(3)

- 例3.12　$0 + 01^*$ の簡単化
  - $0 + 01^* = 0\epsilon + 01^*$　（ϵ が連接に関する単位元）
    $= 0(\epsilon + 1^*)$　（左分配法則）
    $= 01^*$　（$\epsilon \in 1^*$ なので，$\epsilon + 1^* = 1^*$）

3.4.4 冪等法則

- L：任意の正則表現
- 集合和に関する冪等法則

  $L + L = L$

3.4.5 閉包に関する法則

- L：任意の正則表現
- $(L^*)^* = L^*$
- $\emptyset^* = \epsilon$
- $\epsilon^* = \epsilon$
- $L^+ = LL^* = L^*L$
  - $L^+ = L + LL + LLL + LLLL + \cdots$
- $L^* = L^+ + \epsilon$　　$L^+ = L^* - \epsilon$ は必ずしも成立しない
- $L? = \epsilon + L$　（$L?$ の定義）

3.4.6 正則表現に関する法則の発見 (1)

定理3.13

E：変数 $L_1, L_2, \dots, L_m$ を含む正則表現

C：E 中の各 L_j を記号 a_j で置換して得られる正則表現

$L(E)$：E 中の $L_1, L_2, \dots, L_m$ をそれぞれ具体的な言語と見たときに E が表す言語

$L(E)$ は，$L(C)$ の要素（例えば，$a_{j_1}a_{j_2}\cdots a_{j_k}$）の各記号 a_{j_i} を対応する言語 L_{j_i} の任意の要素で置換することによって得られる列全体に等しい

＊記号 a_{j_i} の異なる出現を L_{j_i} の異なる要素で置換してよい

3.4.6 正則表現に関する法則の発見 (2)

定理3.13：例

E：変数 $L_1, L_2, \dots, L_m$ を含む正則表現
C：E 中の各 L_j を記号 a_j で置換して得られる正則表現

$E = A^*(A + B)$

$C = a^*(a + b)$

$L(E)$：$\mathrm{A} = (01)^*$，$\mathrm{B} = (101)^*$ としたときに E が表す言語

$L(E)$ は，$L(C)$ の要素の記号 a, b それぞれを言語 A, B それぞれの任意の要素で置換することによって得られる列全体に等しい

$aab \in L(C)$

$a \leftarrow 01,\ a \leftarrow \epsilon,\ b \leftarrow 101 \ \Rightarrow\ 01101 \in L(E)$

$a \leftarrow 0101,\ a \leftarrow 01,\ b \leftarrow 101101 \ \Rightarrow\ 010101101101 \in L(E)$

3.4.6 正則表現に関する法則の発見 (3)

E：変数 $L_1, L_2, \ldots, L_m$ を含む正則表現
C：E 中の各 L_j を記号 a_j で置換して得られる正則表現

定理3.13 の証明

- 正則表現 E の構成に関する帰納法
- 基礎
  1. $E = \epsilon$ または $E = \emptyset$ のとき
     - 変数がないので $E = C$ となり明らか
  2. $E = A$（変数単独）のとき
     - $C = a$ となる
     - a を $L(A)$ の要素で置換して得られる列全体が $L(A)$ に等しいのは明らか

3.4.6 正則表現に関する法則の発見 (4)

E：変数 $L_1, L_2, \ldots, L_m$ を含む正則表現
C：E 中の各 L_j を記号 a_j で置換して得られる正則表現

定理3.13 の証明

- 帰納
  1. $E = F + G$ のとき
     - C：F 中の変数を記号に置換して得られる正則表現
     - D：G 中の変数を記号に置換して得られる正則表現
       - F, G 中の同じ変数は同じ記号で置換しておく
     - E から置換で得られる正則表現は $C + D$
     - E 中の変数を具体的な言語とみたとき，
       $w \in L(E)$ ならば，$w \in L(F)$ または $w \in L(G)$
       - 帰納仮定より，$L(F)$ は $L(C)$ の要素の記号を対応する言語の要素に置換して得られる列の集合に一致する
       - $L(G)$ についても同様
  2. $E = FG$，$E = F^*$ の場合も同様

3.4.7 正則表現に関する法則の検証 (1)

- E, F：変数を含む正則表現
- $E = F$ かどうかを検証したい
  - E, F の変数をどのような言語と見ても $L(E) = L(F)$ が成立するかどうかを検証したい

1. E, F に現れる各変数を記号に置換して得られる正則表現を C, D とする
2. $L(C) = L(D)$ かどうかを調べる
   - $L(C) = L(D)$ なら $E = F$ が成立
   - $L(C) \neq L(D)$ なら $E = F$ が不成立

3.4.7 正則表現に関する法則の検証 (2)

定理3.14　この判定法により，$E = F$ か否かを判定できる

証明：$E = F \iff L(C) = L(D)$ を示す

- $E = F \Rightarrow L(C) = L(D)$
  - E, F の変数に任意の言語を代入しても $L(E) = L(F)$
  - E, F から C, D を作るときに，変数 A に記号 a を代入したなら，A に言語 $\{a\}$ を代入する
    - $L(E) = L(C)$，$L(F) = L(D)$ となる
    - $L(E) = L(F)$ より，$L(C) = L(D)$
- $L(C) = L(D) \Rightarrow E = F$
  - $L(E), L(F)$：E, F の変数に任意の言語を代入して得られる言語
  - $L(E)$ は，$L(C)$ の記号を対応する言語の要素で置換して得られる列の集合に一致する（定理3.13）．$L(F)$ についても同様．
  - $L(C) = L(D)$ より，$L(E) = L(F)$

3.4.7 正則表現に関する法則の検証(3)

例3.15(1)　$(L+M)^* = (L^*M^*)^*$

- L を記号 a で，M を記号 b で置換する
  - 左辺＝$(a+b)^*$，右辺＝$(a^*b^*)^*$
  - 左辺，右辺とも，a, b からなる列全体の集合
  - 従って，$(L+M)^* = (L^*M^*)^*$ が成立

3.4.7 正則表現に関する法則の検証(4)

例3.15(2)　$L^* = L^*L^*$

- L を記号 a で置換する
  - 左辺＝a^*，右辺＝a^*a^*
  - 左辺，右辺とも，a からなる列全体の集合
  - 従って，$L^* = L^*L^*$ が成立

3.4.7 正則表現に関する法則の検証(5)

例3.15(3)　$L + ML = (L+M)L$

- L を記号 a で，M を記号 b で置換する
  - 左辺＝$a+ba$，右辺＝$(a+b)a$
  - 左辺，右辺は等しくない
    - $aa \notin L(a+ba)$，$aa \in L((a+b)a)$
  - 従って，$L + ML = (L+M)L$ は成立しない

本資料の講義のまとめ

1. 有限オートマトン
2. 正則表現と正則言語
   - 正則表現
   - 有限オートマトンから正則表現へ
   - 正則表現から有限オートマトンへ
   - 正則表現の代数的法則
3. 正則言語の性質
4. 文脈自由文法と言語
5. プッシュダウン・オートマトン
6. 文脈自由言語の性質
7. チューリングマシン

テキスト
3.1〜3.4節

本資料の学習目標

目標を達成できたか確認してみよう（復習も含めて）

- 正則表現を，例を用いて説明できる
- 正則表現が与えられたとき，その言語を説明できる
- 指定された言語を正則表現で表すことができる
- 有限オートマトンが与えられたとき，その言語を正則表現で表現できる
- 与えられた正則表現が表す言語を受理する有限オートマトンを設計できる
- 正則表現の代数的法則を，例を用いて説明できる