

主なレポート課題の例 (07/09 配布)

問1. L が有限オートマトンで認識可能 $\iff \text{Im}S_L$ が有限集合 (ヒント: 演習問題での記号 $S_L(w) := \{v \in \Sigma^* | wv \in L\}$ を用いると、 \implies 側: $L = L(M)$ とし、 M が語 $w \in \Sigma^*$ を読んだ後にいる状態を $\tilde{\delta}_0(w)$ とすると、 $\tilde{\delta}_0(w_1) = \tilde{\delta}_0(w_2) \implies S_L(w_1) = S_L(w_2)$.) \impliedby 側: $\text{Im}S_L$ を状態集合とする有限オートマトンを構成せよ)

問2. 正規言語が非決定性有限オートマトンで認識可能であることを示すために、次の言語を認識する非決定性有限オートマトンを形式的に (状態遷移の概念図ではなく、5つ組 $M = (Q, \Sigma, \delta, s, F)$ を与えることによって) 構成せよ。

- (1) $L(\emptyset) = \emptyset, L(\varepsilon) = \{\varepsilon\}, L(a) = \{a\} \ (a \in \Sigma)$
- (2) 非決定性有限オートマトン $M_i = (Q_i, \Sigma, \delta_i, s_i, F_i)$ が認識する言語 $L_i = L(M_i)$ に対し、 $L_1 \cup L_2, L_1L_2, L_1^*$

問3. Σ を alphabet とする正規言語 $A, B \in \mathcal{P}(\Sigma^*)$ に対し、 $A \cap B$ も再び正規言語であることを示せ。(ヒント: 正規表現の定義から示すのは煩わしい。 A, B を認識する有限オートマトン M_A, M_B が存在するので、これを用いて $A \cap B$ を認識する有限オートマトンを構成せよ。)

問4. $\Sigma = \{a, b\}$ を alphabet とする次の言語について、(a) 正規表現で表せ。(b) 生成規則で表せ。(c) 受理する非決定性有限オートマトンを構成せよ。(d) 受理する決定性有限オートマトンを構成せよ。(e) 上記の有限オートマトンを模倣するプログラムを作成せよ (言語は何でも良い)。

- (1) a で始まり b で終わる
- (2) a, b の片方しか現われない
- (3) a が 3 つ以上続かない
- (4) a が偶数個
- (5) その他、適当に非自明で興味深い例を考えよ。

問5. $\Sigma = \{a, b\}$ を alphabet とする次の言語について、(a) 正規言語でないことを示せ (正規言語に対する Pumping Lemma (注入補題・反復補題) または部屋割り論法を用いよ)。(b) 生成規則で表せ。(c) 受理する (非決定性) プッシュダウンオートマトンを構成せよ。(d) 上記のプッシュダウンオートマトンを模倣するプログラムを作成せよ (言語は何でも良い)。

- (1) $\{a^n b^n | n \geq 0\}$
- (2) 偶数文字の回文
- (3) a, b が同数現われる
- (4) その他、特に正規言語でないような、適当に非自明で興味深い例を考えよ。

問6. $\Sigma = \{a, b, +, -, \times, /, (,)\}$ から成る “文法に合っている” 数式を、

- (1) 生成規則で表せ。
- (2) 受理する (非決定性) プッシュダウンオートマトンを構成せよ。
- (3) 上記のプッシュダウンオートマトンを模倣するプログラムを作成せよ。

但し、簡単のため、考える上で多少の制約・許容を行なって良い (+, - を単項演算子としては使わないとか、普通なら付けない括弧が付いても気にしないとか)。

問7. 例えば、ローマ字仮名変換のように、入力を 1 文字ずつ読み込みながら適宜出力をしてゆくような処理を定式化するために、“出力付き有限オートマトン” とでも言うべき有限状態変換器 (finite state transducer) と呼ばれる計算モデル (処理モデル) を考えてみよう。

- (1) この計算モデルを定式化せよ。
- (2) その下で、ローマ字仮名変換を実装してみよ (これを実行する “出力付き有限オートマトン” を構成せよ)。例えば、次のような制限下でも良い。
 - 入力文字集合 $\Sigma = \{a, o, k, r, y\}$
 - 出力文字集合 $\Gamma = \{\text{あ, お, か, こ, ら, ろ, や, よ, き, り, や, よ, つ}\}$
- (3) ローマ字仮名変換を行なうプログラムを作成せよ。

問 8. 前問を発展させ、プッシュダウンスタックを持った有限状態変換器を考えよう。

- (1) この計算モデルを定式化せよ。(実装を念頭に、決定性のモデルとしてもよい。)
- (2) その下で、中置記法の式を入力して後置記法(逆ポーランド記法)に変換して出力するプッシュダウンスタック付き有限状態変換器を構成せよ。例えば、入力文字集合を $\Sigma = \{a, b, +, \times, (,)\}$ (ここに、 a, b は変数名、演算子 $+, \times$ の優先順位は通常通り) とするような制限下でも良い。

問 9. 後置記法(逆ポーランド記法)で書かれた数式を入力して計算結果を返す“逆ポーランド電卓”を実装せよ。演算子は $+, -, \times, /$ などで、全て二項演算子としてよい。入力数値としては、

- 一桁の自然数 (0~9) に限定 (初級)
- 自然数に限定、二桁以上も可 (中級)
- 実数に対応し、小数点付きの数値も受付ける (上級)

など適当な仕様を選べ。(尚、前問と繋げれば、中置記法の数式を計算する電卓にもなる。)

問 10. 文脈自由言語は、次の形の生成規則の幾つかで生成されることを示せ (Chomsky の標準形という) :

V を変数の集合、 Σ を終端記号 (alphabet) の集合、 $S \in V$ を初期変数として、

- $S \rightarrow \varepsilon$
- $A \rightarrow BC$ ($A, B, C \in V$)
- $A \rightarrow a$ ($A \in V, a \in \Sigma$)

問 11. 文脈自由言語 L に対する Pumping Lemma (注入補題・反復補題) を考える。

- (1) 文脈自由言語 L に対する Pumping Lemma を述べよ。
- (2) L の生成規則を上問の Chomsky の標準形に取ったとする。長さ ℓ の語 $w \in L$ は生成規則を何回適用して得られるか。
- (3) 語 $w \in L$ の構文解析木 T に於いて、 w の各文字に至る道の長さが n 以下であるとき、 w の長さ $|w|$ を上から評価せよ。
- (4) L の生成規則の個数を N とする。語 $w \in L$ の構文解析木に於いて、どの文字に至る道でも同じ生成規則を高々一度しか用いないとすると、 w の長さ $|w|$ を上から評価せよ。
- (5) 文脈自由言語 L に対する Pumping Lemma を示せ。但し、pumping 長 p を N を用いて明示的に与えよ。(ヒント: 語 $w \in L$ の構文解析木に於いて、或る文字に至る道で同じ生成規則を 2 回使っていると、そこから Pumping Lemma の主張する語の分割が構成できる。)

問 12. 集合 X が可算 (enumerable) であるとは、自然数全体の集合 N との間に全単射 $X \rightarrow N$ が存在することをいう。また、単射 $X \rightarrow N$ が存在するとき、 X が高々可算であると言う。

- (1) 高々可算集合の高々可算個の合併集合が再び高々可算であることを示せ。
- (2) 高々可算集合の有限部分集合全体の成す集合が再び高々可算であることを示せ。
- (3) 可算集合の部分集合全体の成す集合 (冪集合) が可算でないことを示せ。(ヒント: 対角線論法)

問 13. 有限集合 Σ を alphabet とする言語で、チューリングマシンで認識できないものが存在することを、以下の手順で示せ。

- (1) チューリングマシンの厳密な定式化を記述せよ。
- (2) チューリングマシンの総数が可算個であることを示せ。
- (3) 有限集合 Σ を alphabet とする言語の総数が可算でないことを示せ。
- (4) 有限集合 Σ を alphabet とする言語で、チューリングマシンで認識できないものが存在することを示せ。

レポート提出について

- 締切: 2012 年 8 月 6 日 (月) 20 時頃まで
- 内容: 配布プリントのレポート課題の例のような内容、及び授業に関連する内容で、授業内容の理解または発展的な取組みをアピールできるようなもの
- 分量: プリントのレポート課題を全部提出する必要はなく、問題の重さによって適宜判断して数問取り組めば良い。内容に関して意欲的な取組みを望む。