主なレポート課題の例 (続き) (12/16 配布)

問 8. 言語 L に対して、語 $w \in \Sigma^*$ の後に連接すると L の元になる語全体の成す集合を $S_L(w)$ とする。(集合・写像などを用いた概念記述の練習)

(1) 関数 $S_L : \Sigma^* \to \mathcal{P}(\Sigma^*)$ を記述せよ。
(2) $a \in \Sigma$ に対し、$\ell_a : \Sigma^* \to \Sigma^*$ (resp. r_a) を、語に左 (resp. 右) から文字 a を連接させる関数とする (これもきちんと記述せよ)。$\ell_a^{-1}(S_L(w)) = S_L(r_a(w))$ を示せ。
(3) $w \in \Sigma^*$ に対し、$\ell_w : \Sigma^* \to \Sigma^*$ (resp. r_w) を、語に左 (resp. 右) から文字列 w を連接させる関数とする (これもきちんと記述せよ)。$\ell_w^{-1}(S_L(v)) = S_L(r_w(v))$ を示せ。
(4) その他、議論に必要または便利と思われる概念を定式化せよ。

問 9. (決定性) 有限オートマトン $M = (Q, \Sigma, \delta, s, F)$ に対し、次のものを記述してみよ。前問の S_M と同様に、

(1) 語 $w \in \Sigma^*$ を読んだ後の状態を与える関数 $\widetilde{\delta} : \Sigma^* \to Q$ (ヒント: 語の長さ $|w|$ に関する帰納的定義を用いよ)
(2) より一般に、状態 $q \in Q$ にいる所から出発して語 $w \in \Sigma^*$ を読んだ後の状態を与える関数 $\widetilde{\delta} : Q \times \Sigma^* \to Q$ (記号の濫用だが、$\widetilde{\delta}(s, w)$ を単に $\widetilde{\delta}(w)$ と書く、ということにして上問との整合性を取ることにする)
(3) M が認識する言語 $L(M)$
(4) より一般に、語 $w \in \Sigma^*$ を読んだ後に、続けて読めば受理される語全体の成す集合 (これも 1 つの言語) を与える関数 $S_M : \Sigma^* \to \mathcal{P}(\Sigma^*)$ (前問との整合性としては $S_M = S_{L(M)}$)
(5) その他、議論に必要または便利と思われる概念を定式化せよ。

問 10. L が有限オートマトンで認識可能 $\iff$ $\mathrm{Im} S_L$ が有限集合 (ヒント: $\Longrightarrow$ 側: $L = L(M)$ とするとき、$\widetilde{\delta}(w_1) = \widetilde{\delta}(w_2) \Longrightarrow S_L(w_1) = S_L(w_2)$. $\Longleftarrow$ 側: $\mathrm{Im} S_L$ を状態集合とする有限オートマトンを構成せよ)

問 11. 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 12. $\Sigma = \{a, b\}$ 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 が偶数個
(4) その他、適当に非自明で興味深い例を考えよ

問 13. $\Sigma = \{a, b\}$ を alphabet とする次の言語について、(a) 生成規則で表せ。(b) 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。(c) 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 14. $\Sigma = \{a, b, +, -, \times, /, (,)\}$ から成る "文法に適っている" 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 15. 例えば、ローマ字仮名変換のように、入力を 1 文字づつ読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 $\Sigma = \{\mathtt{a}, \mathtt{o}, \mathtt{k}, \mathtt{r}, \mathtt{y}\}$
 - 出力文字集合 $\Gamma = \{$ あ, お, か, こ, ら, ろ, や, よ, き, り, ゃ, ょ, っ$\}$

(3) ローマ字仮名変換を行なうプログラムを作成せよ。

問 16. 前問を更に発展させて、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てみよう。

(1) この計算モデルを定式化せよ。
(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、次のような制限下でも良い。
 - 入力文字集合 $\Sigma = \{a, b, +, \times, (,)\}$
 ここに、a, b は変数名で、演算子 $+, \times$ の優先順位は通常通り。

問 17. 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 18. 文脈自由言語 (非決定性プッシュダウンオートマトン) に対する Pumping Lemma (注入補題・反復補題) について述べよ。

問 19. 集合 X が可算 (enumerable) であるとは、自然数全体の集合 $\boldsymbol{N}$ との間に全単射 $X \to \boldsymbol{N}$ が存在することをいう。また、単射 $X \to \boldsymbol{N}$ 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 20. 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 2 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。