

## 主なレポート課題の例

問1. 言語  $L \subset \Sigma^*$  に対し、 $S_L : \Sigma^* \rightarrow \mathcal{P}(\Sigma^*)$  を  $S_L(w) := \{v \in \Sigma^* | wv \in L\}$  で定める。このとき、 $L$  が有限オートマトンで認識可能  $\iff \text{Im} S_L$  が有限集合。

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、 $L_1 \cup L_2, L_1 L_2, L_1^*$

問3.  $\Sigma$  を alphabet とする正規言語  $A \in \mathcal{P}(\Sigma^*)$  に対し、 $A$  に属さない語全体から成る言語  $\Sigma^* \setminus A$  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。 $A$  を認識する有限オートマトン  $M_A$  が存在するので、これを用いて  $\Sigma^* \setminus A$  を認識する有限オートマトンを構成せよ。）

問4.  $\Sigma$  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  $M_A, M_B$  を用いて  $A \cap B$  を認識する有限オートマトンを構成するか、前問と de Morgan の法則を用いよ。）

問5.  $\Sigma = \{a, b\}$  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6.  $\Sigma = \{a, b\}$  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（言語は何でも可）。

- (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7.  $\Sigma = \{a, b, +, -, \times, /, (, )\}$  から成る“文法に適合している”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8. 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の定式化のために、“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 9. 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 10. 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 11. 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、 $\Sigma$  を終端記号(alphabet)の集合、 $S \in V$  を初期変数として、

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

問 12. 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  $\ell$  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 13.  $\Sigma = \{a, b\}$  を alphabet とする次の言語について、(a) 文脈自由言語でないことを示せ(文脈自由言語に対する Pumping Lemma を用いよ)。(b) 文脈依存文法で記述することが出来るか。(c) 受理する(決定性または非決定性)チューリングマシンを構成せよ。(d) 上記のチューリングマシンを模倣するプログラムを作成せよ(言語は何でも可)。

- (1) 同じ文字列 2 回の繰返しから成る文字列全体  $\{w^2 | w \in \Sigma^*\}$
- (2)  $\{a^n b^n a^n | n \geq 0\}$
- (3) その他、特に文脈自由言語でないような、適当に非自明で興味深い例を考えよ。

問 14. 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 15. 有限集合  $\Sigma$  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  $\Sigma$  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  $\Sigma$  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。

問 16. 或るデータ処理の計算量について考える。データを半分に分けてそれぞれについて処理し、それを合わせて結果を得ることが出来るが、合わせる時にデータ数に比例した計算量が必要だとする。このとき、 $n$  個のデータに対する計算量は、 $O(n \log n)$  であることを示せ。

問 17.  $n \times n$  行列の行列式を求める計算の計算量は、 $n$  について如何程か考える。但し、各成分の大きさについては考慮する必要はなく、四則演算をそれぞれ全て 1 回として数えて良いとする。(勿論、各成分の大きさも考慮しても良い。)

- (1) 「全ての置換に互って  $n$  個の成分の積に置換の符号を掛けて足す」という明示公式を用いるとどうか。
- (2) しかし上の方法は計算量が大き過ぎて適切でない。掃出し法を用いると計算量を減らすことが出来る。掃出し法による計算量は如何程か。(ヒント：多項式時間)

問 18. 十進  $n$  桁の整数  $a, b$  の最大公約数  $d := \gcd(a, b)$  を互除法で計算するとき、必要な割算の回数は  $O(n)$  であることを示し、 $O$ -constant を適切に評価せよ。即ち、或る定数  $C > 0$  が存在して  $Cn$  回以内で済むことを示し、 $C$  が実際にはどの程度小さく取れるか評価せよ。(ヒント： $k$  回の割算を必要とする整数の組のうちの最小のものを考えよ。)

問 19. 自然数  $e$  の二進展開を  $e = e_0 + e_1 \cdot 2 + e_2 \cdot 2^2 + \dots + e_k \cdot 2^k$  ( $e_i = 0, 1$ ) とする。

- (1) 自然数  $m, N$  に対し、 $m^e \bmod N$  を高速に計算するアルゴリズムを記述し、何回の掛算および  $N$  で割った余りの計算で行なえるか考察せよ。
- (2) そのアルゴリズムを実装せよ。

問 20. 複数テープチューリングマシンと単テープチューリングマシンとの計算量の違いについて考える。

- (1)  $f: \mathbf{N} \rightarrow \mathbf{R}_{>0}$  を  $f(n) \geq n$  であるような関数とする。言語  $A$  を計算量  $O(f)$  で判定する決定性 2 テープチューリングマシンが存在するならば、言語  $A$  を計算量  $O(f^2)$  で判定する決定性単テープチューリングマシンが存在することを示せ。
- (2) 言語  $A$  が決定性 2 テープチューリングマシンで多項式時間で判定されることと、言語  $A$  が決定性単テープチューリングマシンで多項式時間で判定されることは、同値である。(即ち、問題の計算量が多項式時間であることは、計算モデルとして用いるチューリングマシンが 2 テープであるか単テープであるかには依らない。)

問 21.  $\Sigma = \{a, b\}$  を alphabet とする言語  $A = \{a^k b^k \mid k \in \mathbf{N}\}$  を考える。

- (1) データ長  $n$  に対し線型時間  $O(n)$  の計算量で判定する決定性 2 テープチューリングマシンを構成せよ。
- (2) データ長  $n$  に対し  $n^2$  より真に小さいオーダー (即ち  $o(n^2)$ ) の計算量で判定する決定性単テープチューリングマシンを構成せよ。

問 22. 互いに素な 2 整数  $a, b$  に対し、 $ax + by = 1$  となる  $x, y \in \mathbf{Z}$  を求めるアルゴリズム (Euclid の互除法拡張版) を実装せよ (プログラムを作成せよ)。

問 23. 多くの数値データを大きさの順に並べ替える操作 (並べ替え・ソート) のアルゴリズムについて調べ、その計算量などについて論ぜよ。

問 24. 長桁乗算に関する高速フーリエ変換 (Fast Fourier Transform) について調べ、その計算量などについて論ぜよ。

問 25. 素数判定のアルゴリズムについて調べ、その計算量などについて論ぜよ。

問 26. 素因数分解のアルゴリズムの二次篩法 (Quadratic Sieve) について調べよ。(他のアルゴリズムでも良いが、これが原理が一番簡単。)

問 27. その他、具体的な数理論題について、それを解く効率的なアルゴリズムやその計算量、およびその実装について論ぜよ。

## レポート提出について

- 締切：2019 年 8 月 5 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。