

情報工学

予想問題 第3回 解答

(FA 年・プロセッサ年の再来)

記述式の小問については解答例を示す。同じ内容が伝われば表現は問わない。

作成：Claude (Anthropic)

1 【必須問題】 アルゴリズムとプログラミング**(1) ① 30 ② 40 ③ 10**

キーは 50, 30, 70, 20, 40, 60, 10 の順に挿入される。30 は 50 の左, 70 は 50 の右, 20 は 50 → 30 と辿って 30 の左, 40 は 50 → 30 と辿って 30 の右, 60 は 50 → 70 と辿って 70 の左, 10 は 50 → 30 → 20 と辿って 20 の左に挿入される。

(2) 出力は以下のとおり。

```
10 20 30 40 50 60 70
1 3
```

(1 行目は中間順巡回の結果, 2 行目は `search(root, 40)` が NULL でないことを表す 1 と `count` の値 3 である。)

昇順に出力される理由: 二分探索木ではすべての節点について「左部分木に含まれるキーはすべて自身のキーより小さく, 右部分木に含まれるキーはすべて自身のキーより大きい」という性質が成り立つ。中間順巡回は「左部分木 → 自身 → 右部分木」の順に出力するため, 各節点において自身より小さいキーがすべて先に, 大きいキーがすべて後に出力される。これが再帰的に成り立つので, 全体として昇順となる。

(3) 訪問順に 50, 30, 40 の 3 個。

40 < 50 より左へ, 40 > 30 より右へ進み, 40 = 40 で発見して `return` する。

(4) 昇順に挿入する場合, k 番目に挿入するキーはそれまでに挿入されたとのキーよりも大きい。したがって根から右の子へ進む経路を辿り, 既存の $k-1$ 個の節点すべてを通過する。各節点で 13 行目の条件式 `key < p->key` が 1 回ずつ評価される (いずれも偽となる) ので, k 番目の挿入における評価回数は $k-1$ 回である。

挿入全体での総和は

$$\sum_{k=1}^n (k-1) = \sum_{m=0}^{n-1} m = \frac{n(n-1)}{2}.$$

このとき完成する木は右方向へ一直線に伸びた形 (連結リストと同型) となるので, 根から最も深い節点までの経路上の節点数は n である。

(5) 必要な性質は, 木の高さが $O(\log n)$ に保たれること, すなわち**平衡性 (balance)** である。AVL 木や赤黒木などの平衡二分探索木は, 挿入時に回転操作を行うことでこの性質を維持する。

理由: `search` は根から葉に向かう一本の経路のみを辿り, 各節点で 22 行目の `count++` が 1 回実行される。したがって比較回数は高々「木の高さ +1」である。木の高さが $O(\log n)$ に抑えられていれば, 比較回数も $O(\log n)$ で抑えられる。

2 【必須問題】 計算機システムとシステムプログラム

(1-1) 各命令の実行に要する時間は、必要な段階の所要時間の合計である。表 2 より

演算命令 : $2.0 + 1.0 + 2.0 + 1.0 = 6.0$, ロード命令 : $2.0 + 1.0 + 2.0 + 2.0 + 1.0 = 8.0$,

ストア命令 : $2.0 + 1.0 + 2.0 + 2.0 = 7.0$, 分岐命令 : $2.0 + 1.0 + 2.0 = 5.0$ [ns]

である。単一サイクルプロセッサではすべての命令を 1 クロックサイクルで完了させるため、最も時間のかかる命令に合わせる必要がある。よってクロックサイクル時間は **8.0 [ns]** である。

理由：クロックの立ち上がりから次の立ち上がりまでにすべての命令の処理が完了していなければ正しい結果が得られない。したがって最長の命令（ロード命令）の所要時間がクロックサイクル時間の下限となる。

(1-2) 出現頻度による重み付き平均をとると

$$6.0 \times 0.50 + 8.0 \times 0.25 + 7.0 \times 0.15 + 5.0 \times 0.10 = 3.00 + 2.00 + 1.05 + 0.50 = \mathbf{6.55 \text{ [ns]}}.$$

(1-3) 速度向上率は

$$\frac{8.0}{6.55} \approx \mathbf{1.22 \text{ 倍}}.$$

採用されにくい理由：命令ごとにクロックサイクル時間を変えるには可変周期のクロックを生成する回路が必要となり、制御が複雑になるうえタイミング設計が困難になる。実際のプロセッサでは、クロック周期を一定に保ったままステージを分割するパイプライン方式によって性能を高める方が現実的である。

(2-1) 各ステージの処理時間は等しく $T/5$ となるので、クロックサイクル時間は **$T/5$** である。十分多数の命令を実行する場合、理想的には 1 サイクルごとに 1 命令が完了するので、単位時間あたりの命令数は $1/T$ から $5/T$ となり、スループットの向上率は **5 倍** である。

(2-2) $1w$ の結果 $r1$ は $I1$ の WB（第 5 サイクル）で書き込まれ、同一サイクルの ID で読める。 $I2$ は $I1$ の WB と同じ第 5 サイクルまで ID を進められない。同様に $I3$ は $I2$ の WB（第 8 サイクル）まで ID を進められない。

サイクル	1	2	3	4	5	6	7	8	9	10	11
$I1$	IF	ID	EX	MEM	WB						
$I2$		IF	—	—	ID	EX	MEM	WB			
$I3$			IF	—	—	—	—	ID	EX	MEM	WB

総サイクル数は **11**。

(2-3) フォワーディングにより、 $I2$ は $I1$ の MEM 段の出力を MEM/WB ステージ間レジスタ経由で EX 段の入力に受け取れる。しかし $1w$ の結果は MEM 段が終了する第 4 サイクル末まで得られないため、 $I2$ の EX を第 4 サイクルに置くことはできず 1 サイクルのストールが必要となる。 $I3$ は $I2$ の EX 段の結果を EX/MEM 経由で受け取れるのでストールは不要である。

サイクル	1	2	3	4	5	6	7	8
$I1$	IF	ID	EX	MEM	WB			
$I2$		IF	ID	—	EX	MEM	WB	
$I3$			IF	—	ID	EX	MEM	WB

総サイクル数は **8**。

ストールが残る理由： $I1$ は $1w$ であり、その結果はメモリアクセス（MEM 段）が完了するまで確定しない。一方 $I2$ がその値を必要とするのは EX 段の入力時点である。パイプライン上で $I1$ の MEM 段と $I2$ の

EX 段は本来同じサイクルに位置するため、フォワーディングをもってしても値が間に合わず 1 サイクル分の遅延が不可避となる。この依存関係をロードユース (load-use) ハザードという。

(2-4) 最初の命令の完了に k サイクル要し、以後は 1 サイクルごとに 1 命令が完了するので

$$k + (N - 1) = k + N - 1 \text{ サイクル.}$$

非パイプライン実行では kN サイクル要するので、速度向上率は

$$\frac{kN}{k + N - 1} \xrightarrow{N \rightarrow \infty} k$$

となり、ステージ数 k に等しくなる。

分岐命令によるハザードの名称は**制御ハザード** (分岐ハザード, control hazard) である。

軽減手法の例：**分岐予測 (branch prediction)**。分岐の成立・不成立を予測して後続の命令を投機的にフェッチ・実行する。予測が当たればストールなしに実行が続き、外れた場合はパイプライン中の投機実行した命令を破棄して正しい命令から再開する。予測精度が高ければ分岐による損失を大幅に削減できる。

4 【選択問題】 計算理論

(1-1) 受理状態の集合 $\{q_5\}$ とそれ以外による 2 分割から始めて細分化する。

- **分割 1** : $\{q_5\}$, $\{q_0, q_1, q_2, q_3, q_4\}$
- **分割 2** : q_3, q_4 は入力 a, b のいずれに対しても q_5 (受理側のブロック) へ遷移するのに対し, q_0, q_1, q_2 はいずれも非受理側のブロックへ遷移する。よって

$$\{q_5\}, \quad \{q_3, q_4\}, \quad \{q_0, q_1, q_2\}$$

- **分割 3** : q_1 は (q_3, q_4) へ, q_2 は (q_4, q_3) へ遷移し, いずれもブロック $\{q_3, q_4\}$ に入る。一方 q_0 は (q_1, q_2) へ遷移し, これはブロック $\{q_0, q_1, q_2\}$ に入る。よって

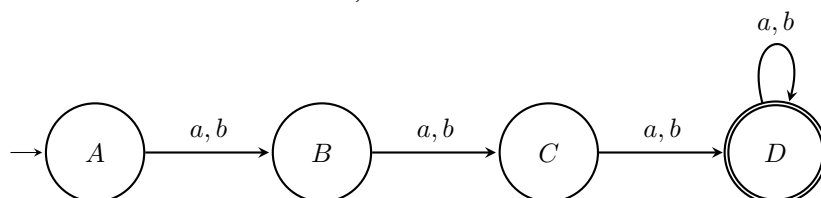
$$\{q_5\}, \quad \{q_3, q_4\}, \quad \{q_0\}, \quad \{q_1, q_2\}$$

- **分割 4** : これ以上細分化されないので終了。

最小 DFA M' は 4 状態であり、各状態と M の状態の対応は

$$A = \{q_0\}, \quad B = \{q_1, q_2\}, \quad C = \{q_3, q_4\}, \quad D = \{q_5\}$$

である。状態遷移図は次のとおり (A が開始状態, D が受理状態)。



(1-2) $L(M)$ は, $\Sigma = \{a, b\}$ 上の長さ 3 以上のすべての語からなる言語である。(開始状態から 3 回遷移すると必ず q_5 に到達し, q_5 からは q_5 以外へ遷移しないため。)

(2) 空欄の答えは以下のとおり。

空欄	答え
(a)	$[r_1, r_2]$
(b)	$[r_1, r_2]$
(c)	$[\delta_1(r_1, a), \delta_2(r_2, a)]$
(d)	$[q_1, q_2]$
(e)	$[r_1, r_2]$
(f)	$r_1 \in F_1$ かつ $r_2 \in F_2$

すなわち

$$Q = Q_1 \times Q_2, \quad \delta([r_1, r_2], a) = [\delta_1(r_1, a), \delta_2(r_2, a)], \quad q = [q_1, q_2], \quad F = F_1 \times F_2.$$

M は M_1 と M_2 を同じ入力に対して同時に走らせ、その状態の組を自身の状態とする。積集合を認識するためには、入力を読み終えた時点で**両方**が受理状態にある必要があるので $F = F_1 \times F_2$ とする。(和集合の場合は少なくとも一方が受理状態であればよいので $F = (F_1 \times Q_2) \cup (Q_1 \times F_2)$ となり、この点だけが異なる。)

(3-1) $L_0 = L(0^*1^*)$ は「0 がすべて 1 より前に並んだ語」全体である。 L は 0 と 1 の個数が等しい語全体であるから、

$$L \cap L_0 = \{0^n 1^n \mid n \geq 0\}.$$

(3-2) L が正則言語であると仮定する。 $L_0 = L(0^*1^*)$ は正則表現で表されるので正則言語である。正則言語は積集合について閉じているから、 $L \cap L_0$ も正則言語となる。

しかし小問 (3-1) より $L \cap L_0 = \{0^n 1^n \mid n \geq 0\}$ であり、これは正則言語でないことが証明済みである。これは矛盾である。

したがって仮定が誤りであり、 L は正則言語ではない。☒

6 【選択問題】電子回路と論理設計

(1-1) pMOS トランジスタを 2 個並列に接続して電源 V_{DD} 側（プルアップ網）に、nMOS トランジスタを 2 個直列に接続して接地側（プルダウン網）に配置する。計 4 個のトランジスタを用いる。

二つの入力のうち少なくとも一方が 0 のとき、対応する pMOS が導通して出力は V_{DD} （論理 1）となる。両方の入力が 1 のときのみ直列の nMOS がともに導通して出力は接地電位（論理 0）となる。これは NAND の真理値表と一致する。

(1-2) nMOS はソースに低電位を与えたときには劣化なく論理 0 を伝えられるが、高電位を伝えようとするとき値電圧分だけ低下した電位しか出力できない。pMOS はその逆で、論理 1 を劣化なく伝えられるが論理 0 の伝達には向かない。したがって出力を高電位に引き上げるプルアップ網には pMOS を、低電位に引き下げるプルダウン網には nMOS を用いることで、出力を常に V_{DD} または接地電位までフルスイングさせることができる。

(2-1)

$$s = a \oplus b \oplus c_{in}, \quad c_{out} = ab + bc_{in} + ac_{in}.$$

($c_{out} = ab + c_{in}(a \oplus b)$ と書いてもよい。)

(2-2) 半加算器 (HA) は 2 入力 p, q に対し和 $p \oplus q$ と桁上げ pq を出力する。

- HA1 に a と b を入力し、和 $s_1 = a \oplus b$ 、桁上げ $c_1 = ab$ を得る。
- HA2 に s_1 と c_{in} を入力し、和 $s = s_1 \oplus c_{in} = a \oplus b \oplus c_{in}$ 、桁上げ $c_2 = s_1 c_{in} = (a \oplus b)c_{in}$ を得る。
- OR ゲートで $c_{out} = c_1 + c_2 = ab + (a \oplus b)c_{in}$ を得る。

(3) 最下位の桁上げ入力の変化は、1 段目の全加算器を通して桁上げ出力に現れ、これが 2 段目、3 段目、4 段目へと順に伝搬する。各段で τ の遅延が生じるので、最上位の桁上げ出力が確定するまでの最悪遅延は 4τ である。

改善する方式：桁上げ先見加算器（キャリールックアヘッド加算器，carry look-ahead adder）。

(4-1) 状態遷移表は以下のとおり。

u	y_1	y_0	D_1	D_0	次状態
1	0	0	0	1	01
1	0	1	1	0	10
1	1	0	1	1	11
1	1	1	0	0	00
0	0	0	1	1	11
0	0	1	0	0	00
0	1	0	0	1	01
0	1	1	1	0	10

(4-2) D_0 について： u によらず、 y_0 が反転する。よって

$$D_0 = \overline{y_0}.$$

D_1 について： $D_1 = 1$ となるのは $(u, y_1, y_0) = (1, 0, 1), (1, 1, 0), (0, 0, 0), (0, 1, 1)$ の 4 通りである。カルノー図に記入すると、これらは市松模様に配置され、隣接する 1 のマスの組が存在しない。したがって簡単化できず、最簡積和形は

$$D_1 = \bar{u} \bar{y}_1 \bar{y}_0 + \bar{u} y_1 y_0 + u \bar{y}_1 y_0 + u y_1 \bar{y}_0$$

の 4 項からなる。

排他的論理和を用いると、これは「 u, y_1, y_0 のうち 1 の個数が偶数のとき 1」を表すので

$$D_1 = \overline{u \oplus y_1 \oplus y_0}$$

と簡潔に書ける。

7 【選択問題】 数学解析と信号処理

(1-1) 微分方程式の両辺をラプラス変換する。 $y(0) = 0$ より $\mathcal{L}[dy/dt] = sY(s)$ であるから

$$sY(s) + 3Y(s) = X(s) \implies (s+3)Y(s) = X(s).$$

よって

$$H(s) = \frac{Y(s)}{X(s)} = \frac{1}{s+3}.$$

(1-2) 単位ステップ入力のラプラス変換は $X(s) = 1/s$ であるから

$$Y(s) = H(s)X(s) = \frac{1}{s(s+3)}.$$

部分分数分解すると

$$\frac{1}{s(s+3)} = \frac{1}{3} \left(\frac{1}{s} - \frac{1}{s+3} \right)$$

であるから

$$y(t) = \frac{1}{3} (1 - e^{-3t}) \quad (t \geq 0).$$

$t \rightarrow \infty$ では $e^{-3t} \rightarrow 0$ であるから

$$\lim_{t \rightarrow \infty} y(t) = \frac{1}{3}.$$

(最終値の定理 $\lim_{s \rightarrow 0} sY(s) = \lim_{s \rightarrow 0} \frac{1}{s+3} = \frac{1}{3}$ と一致する。 $sY(s)$ の極 $s = -3$ は左半平面にあるので定理が適用できる。)

(1-3) $H(s) = \frac{1}{s+3}$ の極は $s = -3$ である。これは複素平面の左半平面（実部が負）にあるので、このシステムは安定である。実際、インパルス応答 $h(t) = e^{-3t}$ は $t \rightarrow \infty$ で 0 に収束する。

(2-1) $h[k]$ は $0 \leq k \leq N-1$ でのみ $1/N$ をとるから

$$y[n] = \sum_{k=0}^{N-1} \frac{1}{N} x[n-k] = \frac{1}{N} (x[n] + x[n-1] + \cdots + x[n-N+1]).$$

FIR である理由：インパルス応答 $h[n]$ が $0 \leq n \leq N-1$ の有限個の n に対してのみ非零であり、それ以外ではすべて 0 である。すなわちインパルス応答の長さが有限であるため、FIR（有限インパルス応答）フィルタである。

(2-2) 定義より

$$H(e^{j\omega}) = \sum_{n=0}^{N-1} \frac{1}{N} e^{-j\omega n} = \frac{1}{N} \sum_{n=0}^{N-1} (e^{-j\omega})^n = \frac{1}{N} \cdot \frac{1 - e^{-j\omega N}}{1 - e^{-j\omega}}.$$

分子・分母をそれぞれ半角の指数でくくると

$$1 - e^{-j\omega N} = e^{-j\omega N/2} \cdot 2j \sin \frac{N\omega}{2}, \quad 1 - e^{-j\omega} = e^{-j\omega/2} \cdot 2j \sin \frac{\omega}{2}$$

であるから

$$H(e^{j\omega}) = \frac{1}{N} \cdot \frac{e^{-j\omega N/2} \cdot 2j \sin(N\omega/2)}{e^{-j\omega/2} \cdot 2j \sin(\omega/2)} = \frac{1}{N} e^{-j\omega(N-1)/2} \frac{\sin(N\omega/2)}{\sin(\omega/2)}.$$

(2-3) $|H(e^{j\omega})| = \frac{1}{N} \left| \frac{\sin(N\omega/2)}{\sin(\omega/2)} \right|$ が 0 となるのは、 $0 < \omega \leq \pi$ で分母が 0 でないことから $\sin(N\omega/2) = 0$ すなわち

$$\omega = \frac{2\pi k}{N} \quad \left(k = 1, 2, \dots, \left\lfloor \frac{N}{2} \right\rfloor \right).$$

$\omega \rightarrow 0$ では $\sin x \approx x$ より

$$\lim_{\omega \rightarrow 0} |H(e^{j\omega})| = \frac{1}{N} \cdot \frac{N\omega/2}{\omega/2} = 1.$$

直流成分での利得が 1 で最大であり、 ω が増大すると $\omega = 2\pi/N$ で最初の零点をとり以降は振動しながら減衰する。すなわち低周波成分をそのまま通し高周波成分を減衰させるので、低域通過フィルタとして機能する。

(2-4) 小問 (2-2) より位相は $\theta(\omega) = -\frac{N-1}{2}\omega$ であり (sin 比の符号が変わる点で $\pm\pi$ の跳びが生じるが各区間内では)、 ω の一次式である。

利点：群遅延 (group delay) $-d\theta/d\omega = (N-1)/2$ が周波数によらず一定となるため、通過帯域内のすべての周波数成分が同じサンプル数だけ遅延する。したがって成分間の相対的な時間関係が保たれ、出力波形は入力波形を単に時間シフトしたものとなって位相歪みによる波形の崩れが生じない。

(2-5) $N = 4$ のとき $y[n] = \frac{1}{4}(x[n] + x[n-1] + x[n-2] + x[n-3])$ である。 $x[0] = x[1] = 1$ 、それ以外は 0 であるから

$$y[0] = \frac{1}{4}, \quad y[1] = y[2] = y[3] = \frac{1}{2}, \quad y[4] = \frac{1}{4}, \quad \text{その他の } n \text{ では } y[n] = 0.$$