

情報工学

予想問題 第2回

(対抗シナリオ)

必須問題：問題 1・2 選択問題：問題 4・6・7 より 2 問選択

試験時間：3 時間 配点：500 点満点

2018～2026 年度（一部 2006～2017 年度）の過去問分析に基づく自作予想問題

作成：Claude (Anthropic)

1 【必須問題】 アルゴリズムとプログラミング

配点：(1-1) 15 点, (1-2) 20 点, (2) 30 点, (3) 25 点, (4) 20 点, (5) 15 点

次ページの図 1 は ANSI C で記述されたプログラムである。大域変数 `count` は 7 行目が実行された回数を数えるカウンタである。図 1 のプログラムに関する以下の各問に答えよ。

(1) 以下の各小問に答えよ。

(1-1) 長さ n_1 と n_2 の 2 つのソート済み区間を `merge` で併合するとき、7 行目が実行される回数の最大値と最小値をそれぞれ n_1, n_2 を用いて示せ。

(1-2) 図 1 のプログラムを実行したときの `funcA` の分割の様子を図 2 に示す。空欄①～④に入る部分配列を答えよ。

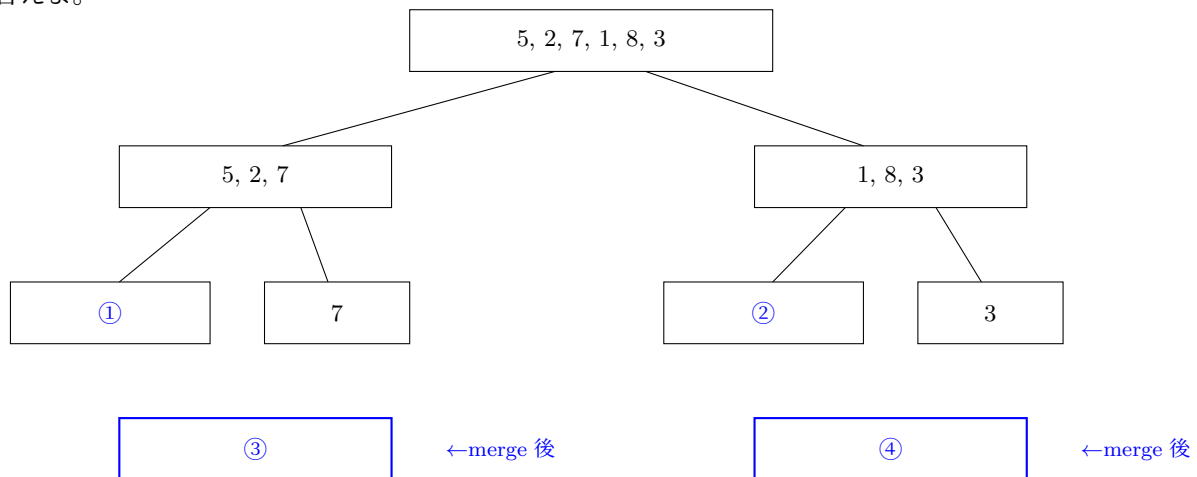


図 2 分割の様子（下段は merge 直後の部分配列。①～④を埋めよ。最上段の merge 結果は省略）

(2) 図 1 のプログラムが開始してから終了するまでに出力される内容をすべて記述せよ。`count` の値については、各 `merge` 呼び出しでの比較回数の内訳も示すこと。

(3) $n = 2^k$ (k は正の整数) のとき、サイズ n の配列に対する `count` の最大値が $n \log_2 n - n + 1$ となることを、各段の `merge` の比較回数を総和して導出せよ。

(4) 8 行目の比較が `A[i] <= A[j]` (等号を含む) であることに着目し、このプログラムが安定ソート (stable sort) である理由を 2～3 行で説明せよ。

(5) このプログラムがクイックソートと比べて不利な点を、記憶領域の観点から 1 つ挙げ、図 1 のどの変数がその原因かを明示して理由とともに述べよ。

```
1 #include <stdio.h>
2 int count = 0;
3
4 void merge(int A[], int left, int mid, int right, int tmp[]) {
5     int i = left, j = mid + 1, k = left;
6     while (i <= mid && j <= right) {
7         count++;
8         if (A[i] <= A[j]) tmp[k++] = A[i++];
9         else tmp[k++] = A[j++];
10    }
11    while (i <= mid) tmp[k++] = A[i++];
12    while (j <= right) tmp[k++] = A[j++];
13    for (k = left; k <= right; k++) A[k] = tmp[k];
14 }
15
16 void funcA(int A[], int left, int right, int tmp[]) {
17     if (left < right) {
18         int mid = (left + right) / 2;
19         funcA(A, left, mid, tmp);
20         funcA(A, mid + 1, right, tmp);
21         merge(A, left, mid, right, tmp);
22     }
23 }
24
25 int main(void) {
26     int A[6] = {5, 2, 7, 1, 8, 3};
27     int tmp[6];
28     funcA(A, 0, 5, tmp);
29     for (int k = 0; k < 6; k++) printf("%d ", A[k]);
30     printf("\n");
31     printf("%d\n", count);
32     return 0;
33 }
```

図1 プログラム

2 【必須問題】 計算機システムとシステムプログラム

配点：(1-1) 15 点, (1-2) 25 点, (1-3) 20 点, (2) 25 点, (3-1) 25 点, (3-2) 15 点

単一プロセッサ上で動作する並行プロセスを考える。共有変数の更新は直ちに他プロセスから見えるものとする。セマフォ変数 s に対する操作を以下のように定義する。

- $P(s)$: $s > 0$ なら s を 1 減らして続行。 $s = 0$ なら $s > 0$ になるまで待つ。
- $V(s)$: s を 1 増やす。

(1) サイズ 1 のバッファ `buf` を介して、生産者プロセスが書き込んだ値を消費者プロセスが読み出す。セマフォ `empty` (初期値 1) と `full` (初期値 0) を用いる。

生産者 (10回繰り返す)	消費者 (10回繰り返す)
<code>for (i = 0; i < 10; i++) {</code>	<code>for (j = 0; j < 10; j++) {</code>
[(a)];	[(c)];
<code>buf = i;</code>	<code>x = buf;</code>
[(b)];	[(d)];
<code>}</code>	<code>printf("%d\n", x);</code>
	<code>}</code>

(1-1) $P \cdot V$ 操作が不可分 (atomic) に実行される必要がある理由を 2~3 行で説明せよ。

(1-2) 0 から 9 がすべて重複・欠落なくこの順で出力されるように、空欄 (a)~(d) に入る P/V 操作を答えよ。

(1-3) 空欄 (a) と (b) の操作を入れ替えた場合に何が起こるか。プロセスの実行順の具体例を挙げて 2~3 行で説明せよ。

(2) デッドロックが発生するための 4 つの必要条件をすべて挙げよ。また、そのうち 1 つを崩してデッドロックを予防する具体的な方策を 1 つ述べよ。

(3) 二つのセマフォ s_1 (初期値 1) と s_2 (初期値 0) を使い、書き込みプロセス (関数 g_1) と読み出しプロセス (関数 g_2) を並行実行して、共有変数 n 経由で 0 から 9 を必ず昇順に出力したい。

<code>void g1(void) {</code>	<code>void g2(void) {</code>
<code>int i;</code>	<code>int j, x;</code>
<code>for (i = 0; i < 10; i++) {</code>	<code>for (j = 0; j < 10; j++) {</code>
[(a)];	[(c)];
<code>n = i;</code>	<code>x = n;</code>
[(b)];	[(d)];
<code>}</code>	<code>printf("%d\n", x);</code>
<code>}</code>	<code>}</code>

(3-1) 空欄 (a)~(d) に入る P/V 操作を答えよ。

(3-2) s_1 と s_2 の初期値を両方とも 0 にした場合に何が起こるか、理由とともに 2~3 行で述べよ。

4 【選択問題】 計算理論

配点：(1-1) 35 点, (1-2) 20 点, (2) 15 点, (3) 15 点, (4) 40 点

文脈自由文法 (context-free grammar) $G = (V, T, P, S)$ は、変数 (variable) の有限集合 V 、終端記号 (terminal symbol) の有限集合 T 、生成規則 (production rule) の有限集合 P 、および開始記号 (start symbol) S により定められる。変数は非終端記号 (nonterminal symbol) と呼ばれることもある。頭部 (head) が同じ生成規則は、本体 (body) を「|」で区切って列挙できる。生成規則の形を $X \rightarrow YZ$, $X \rightarrow a$ に限定した文脈自由文法を、Chomsky 標準形 (Chomsky normal form) と呼ぶ。ただし、 X, Y, Z は変数、 a は終端記号である。

Chomsky 標準形で与えられる文脈自由文法 $G = (V, T, P, S)$ および終端記号列 $w = a_1a_2 \dots a_n$ ($n \geq 1$ とし、各 a_i は T に属する) に対し、 w が G によって生成されるかどうかを判定する以下のアルゴリズムを考える。

- 1: $w = a_1a_2 \dots a_n$ に対し、2次元配列の要素である集合 $Q[i][j]$ ($1 \leq i \leq j \leq n$) をすべて空集合に初期化。
- 2: **for** $i = 1$ **to** n **do**
- 3: $(X \rightarrow a_i) \in P$ となるすべての $X \in V$ を $Q[i][i]$ に追加。
- 4: **end for**
- 5: **for** $l = 2$ **to** n **do**
- 6: **for** $i = 1$ **to** $n - l + 1$ **do**
- 7: j を $i + l - 1$ とする。
- 8: **for** $k = i$ **to** $j - 1$ **do**
- 9: $Y \in Q[i][k]$, $Z \in Q[k+1][j]$ であるすべての Y, Z に対し、 $(X \rightarrow YZ) \in P$ であるすべての $X \in V$ を $Q[i][j]$ に追加。
- 10: **end for**
- 11: **end for**
- 12: **end for**
- 13: **if** [空欄] **then** 「Yes」を出力。
- 14: **else** 「No」を出力。

- (1) 文法 $G_1 = (V_1, T_1, P_1, S)$ を、 $V_1 = \{S, A\}$, $T_1 = \{a, b\}$,

$$P_1 = \{ S \rightarrow AbA, \quad A \rightarrow a \mid \varepsilon \}$$

と定める。ただし ε は空列 (empty string) を表す。

(1-1) $L(G)$ を、文法 G により生成される言語とする。 $L(G_2) = L(G_1) - \{\varepsilon\}$ を満たし、無用な (useless) 記号を含まない Chomsky 標準形の文法 G_2 を示せ。 ε 規則の除去 $\rightarrow$ 終端記号の変数化 $\rightarrow$ 長さ 3 以上の規則の分解、の各段階の結果を示すこと。

(1-2) (1-1) で得た文法 G_2 および終端記号列 $w = aba$ に対して、上記アルゴリズム 13 行目の開始時点での各集合 $Q[i][j]$ ($1 \leq i \leq j \leq 3$) を、表 1 の形式で示せ。さらに、その結果に基づき $w = aba \in L(G_2)$ かどうかを判定し、根拠を述べよ。

$i \backslash j$	1	2	3
1			
2			
3			

表 1 $Q[i][j]$ ($w = aba$, $w_1 = a, w_2 = b, w_3 = a$). 灰色セルは $i > j$ のため使用しない

- (2) i, j を $1 \leq i \leq j \leq n$ を満たす任意の整数とする。上記アルゴリズム 13 行目の開始時点で集合 $Q[i][j]$ に含まれる変数はある特定の性質を持つ。 $w = a_1 a_2 \dots a_n$ として、この性質を $a_i \dots a_j$ を用いて説明せよ。
- (3) 上記アルゴリズム 13 行目の空欄は、 w が G によって生成されるか否かを判定するための条件である。空欄を埋めよ。
- (4) Chomsky 標準形の任意の文法 G の構文木 (parse tree) について考える。 n 個の葉 (leaf node) を持つ構文木が、ちょうど $2n - 1$ 個の頂点 (node) を持つことを証明せよ。

6 【選択問題】電子回路と論理設計

配点：(1-1) 10 点, (1-2) 15 点, (1-3) 15 点, (1-4) 20 点, (2-1) 15 点, (2-2) 15 点, (2-3) 35 点

(1) 図 1 に示す RLC 直列回路について、フェーザ表示を用いて以下に答えよ。

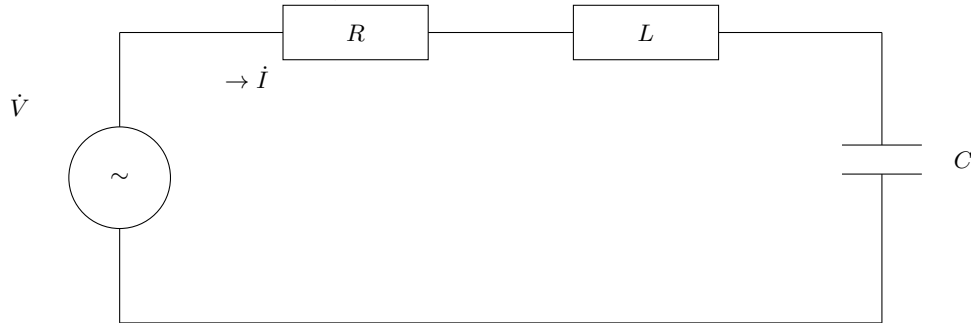


図 1 RLC 直列回路（電源は角周波数 ω ，実効値 V の正弦波）

(1-1) 合成インピーダンス $\dot{Z}$ を R, L, C, ω で示せ。

(1-2) 電流の大きさが最大となる共振角周波数 ω_0 を導出せよ。

(1-3) $\omega < \omega_0$ と $\omega > \omega_0$ のそれぞれで、電流の位相が電源電圧に対して進むか遅れるか、理由とともに答えよ。

(1-4) 共振時のコンデンサ両端電圧の大きさが $V_C = QV$ ($Q = \omega_0 L/R$) となることを導出せよ。

(2) 論理設計に関する以下の各小問に答えよ。

(2-1) 4 変数論理関数 (logic function) $f(a, b, c, d) = \sum m(0, 2, 5, 7, 8, 10, 13, 15)$ の最簡積和形（最小積和形, minimum sum-of-products form）を、カルノー図 (Karnaugh map) を用いて求めよ。用いたカルノー図も示すこと。

(2-2) 小問 (2-1) で求めた最簡積和形を、2 入力 NAND ゲートのみを用いて実現する回路を示せ。また、使用するゲートの個数を答えよ。ただし、NOT は 2 入力 NAND の二つの入力を束ねることで実現してよい。

(2-3) 入力系列に部分列 (substring) 101 が現れた時点で出力を 1 とする Mealy 型同期式順序回路 (Mealy synchronous sequential circuit) を設計する。入力を x ，出力を z とし、部分列の重なりを許す（例えば入力系列 10101 に対しては 2 回検出する）。この回路は二つの D フリップフロップ (D flip-flop) を持ち、状態数は 3 である。現在の状態 (current state) を (y_1, y_0) ，次状態 (next state) を (D_1, D_0) と表す。以下の各小問に答えよ。

(a) この回路の状態遷移図を示せ。ただし、各状態には、その状態に至るまでの入力系列について何を記憶しているかの説明を併記すること。また、遷移に付与するラベルは「入力／出力」すなわち x/z の形式で表すこと。

(b) 状態割り当て (state assignment) を $S_0 = (y_1, y_0) = (0, 0)$, $S_1 = (0, 1)$, $S_2 = (1, 0)$ とするとき、 D_1, D_0, z の論理式の最簡積和形をそれぞれ求めよ。ここで、解答の各積項のリテラル (literal) は x, y_1, y_0 の順で記載すること。また、未使用状態 (1, 1) に対する出力および次状態はドントケア (don't care) として扱ってよい。

7 【選択問題】 数学解析と信号処理

配点：(1-1) 15 点, (1-2) 25 点, (2-1) 25 点, (2-2) 20 点, (3-1) 10 点, (3-2) 15 点, (3-3) 15 点

関数 (function) $f(x; \theta)$ は, θ をパラメータ (parameter) とする確率密度関数 (probability density function) を表すことにする。データ集合 $\mathcal{D} = \{x_i \mid i = 1, \dots, n\}$ (n は正の整数 (positive integer)) が与えられたとき, 関数 $L_f(\theta) = \prod_{i=1}^n f(x_i; \theta)$ は尤度関数 (likelihood function) であり, その対数 $\log L_f(\theta)$ を対数尤度 (log likelihood) と呼ぶ。対数尤度を最大化する θ を求める問題は最尤推定 (maximum likelihood estimation) と呼ばれる。以下の各問に答えよ。ただし, 対数の底 (base of logarithm) は e とし, 導出の過程も示すこと。

(1) パラメータ $\lambda > 0$ の指数分布 (exponential distribution) の確率密度関数は, 以下のように表される。

$$f(x; \lambda) = \lambda e^{-\lambda x} \quad (x \geq 0) \quad (1)$$

$x_i \geq 0$ ($i = 1, \dots, n$) を満たすデータ集合 $\mathcal{D}$ が式 (1) の指数分布に従うとして, 以下の各小問に答えよ。

(1-1) 対数尤度 $\log L_f(\lambda)$ が以下の形になることを示せ。また, このときの α, β を答えよ。

$$\log L_f(\lambda) = \alpha \log \lambda - \beta \sum_{i=1}^n x_i \quad (2)$$

(1-2) 式 (2) を λ について偏微分 (partial differentiation) することにより, 対数尤度を最大化する $\hat{\lambda}$ を求めよ。ただし $\hat{\lambda}$ は対数尤度を最大化する λ を表す。また, 求めた値が極大値 (local maximum) を与えることを 2 階微分 (second derivative) により確認せよ。

(2) 最小二乗法 (least square method) は, 観測値 y_i とモデルによる出力 $\hat{y}_i$ の間の残差 (residual) の二乗和 (square sum) を最小化 (minimize) するモデルパラメータを求める方法である。ここでは原点を通る直線モデル $\hat{y}_i = \phi x_i$ により, データ $(x_1, y_1), \dots, (x_n, y_n)$ を近似することを考える。

(2-1) 残差二乗和 $E(\phi) = \sum_{i=1}^n (y_i - \phi x_i)^2$ を最小にする $\hat{\phi}$ を求めよ。

(2-2) 残差 $r_i = y_i - \phi x_i$ ($i = 1, \dots, n$) が互いに独立 (independent) であり, 平均 (mean) 0, 分散 (variance) σ^2 の正規分布 (normal distribution)

$$N(r; 0, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{r^2}{2\sigma^2}\right)$$

に従うとする。このときモデルパラメータ ϕ の最尤推定が, 小問 (2-1) の最小二乗法による解と一致することを示せ。

(3) 周波数 $f_0 = 3$ [Hz] の正弦波 (sine wave) である連続時間信号 (continuous-time signal) $y^*(t) = \sin(2\pi f_0 t)$ を, サンプル周波数 (sampling frequency) $f_s = 4$ [Hz] で標本化 (sampling) して, 図 1 の ● で示す離散時間信号 (discrete-time signal) を得た。

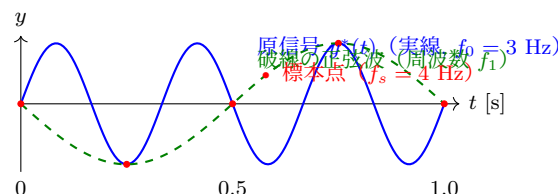


図 1 原信号 (original signal) (実線), 標本点 (●), および標本点をすべて通る低い周波数の正弦波 (破線)

(3-1) 原信号を標本点から一意に復元 (reconstruct) するためにサンプル周波数が満たすべき条件を示せ。また, この条件を与える定理の名称を答えよ。

(3-2) 図 1 の破線が示すように, 標本点のみからは原信号と別の低い周波数 f_1 の正弦波とを区別できない。 f_1 の値を求めよ。また, この現象の名称を答えよ。

(3-3) $f_s = 4$ [Hz] のまま標本化の前処理によって (3-2) の現象の影響を防ぐ方法を, フィルタの種類と遮断周波数の条件を明示して 2~3 行で述べよ。