

2025 年度大学院博士前期課程入学試験

大阪大学大学院工学研究科 電気電子情報通信工学専攻

専門科目試験問題 (情報通信工学コース)

(実施時間 14:00 ~ 16:00)

【注 意 事 項】

1. 問題用紙はこの表紙を除いて13ページある。解答開始の指示があるまで開いてはいけない。解答開始後、落丁や不鮮明な箇所等があった場合は、手を挙げて監督者にその旨を伝えること。
2. 試験問題は、「通信ネットワーク」、「情報理論」、「信号処理」、「論理回路と計算機システム」、「データ構造とアルゴリズム」、及び、「情報セキュリティ」の全部で6題あり、この順番に綴じられている。このうち、3題を選択し解答すること。
3. 解答開始前に、別紙の「専門科目試験問題選択票」に記載の注意事項も読んでおくこと。
4. 問題用紙は持ち帰ってもよい。

【通信ネットワーク】 解答は、黄色の解答用紙に記入すること.

容量が 1 のバッファを有する送信ノードから基地局へパケットを伝送する状況を, $M/M/1/2$ 待ち行列としてモデル化し解析することを考える. 送信ノードは率 λ ($\lambda > 0$) のポワソン過程に従ってパケットを生成する. パケット生成時点において送信ノードが他のパケットを伝送中でない場合, 生成されたパケットの伝送が即座に開始される. 一方, パケット生成時点において送信ノードが他のパケットを伝送中である場合, バッファが空であれば生成されたパケットはバッファに格納され, バッファが空でなければ生成されたパケットは破棄される. 一つのパケットの伝送開始から受信完了までに要する時間は, 平均 $1/\mu$ ($\mu > 0$) の指数分布に従って独立かつ同一に分布すると仮定する.

このとき, 以下の問いに答えよ.

- (i) 時刻 t において送信ノードのバッファに格納されているパケットの数を $B(t)$ とし, また, 時刻 t において送信ノードが伝送中であるパケットの数を $S(t)$ とする. さらに, 送信ノード内のパケット総数を $L(t) = B(t) + S(t)$ と定義する. $L(t)$ が従う出生死滅過程の状態遷移速度図を示せ.
- (ii) $L(t) = k$ である確率を $p_k(t)$ とし, 第 k 要素 ($k = 0, 1, 2$) が $p_k(t)$ である行ベクトルを $\pi(t)$ とする.

$$\pi(t) = [p_0(t) \quad p_1(t) \quad p_2(t)]$$

また, 行列 Q を次式で定義する.

$$Q = \begin{bmatrix} -\lambda & \lambda & 0 \\ \mu & -\lambda - \mu & \lambda \\ 0 & \mu & -\mu \end{bmatrix}$$

このとき, 次式が成り立つことを示せ.

$$\frac{d}{dt} \pi(t) = \pi(t)Q$$

- (iii) 前問 (ii) で定義した $\pi(t)$ が次式を満たすとする.

$$\frac{d}{dt} \pi(t) = \mathbf{0}$$

ただし, $\mathbf{0} = [0 \quad 0 \quad 0]$ である. このとき, $\pi(t)$ を求めよ.

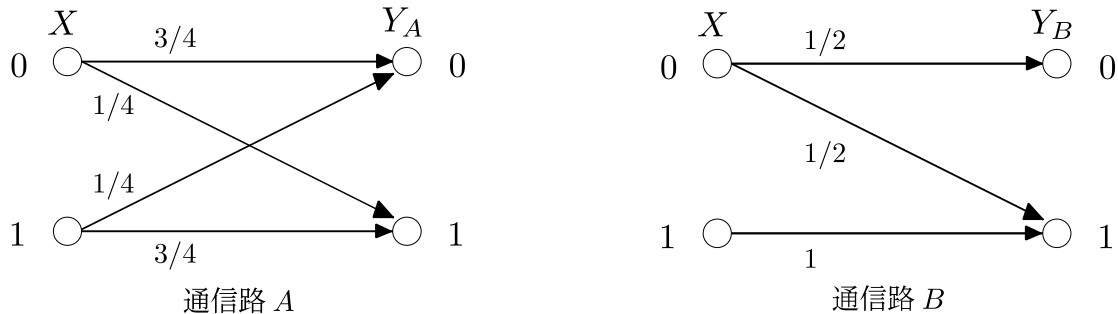
- (iv) 行列 Q の固有値はすべて実数かつ 0 以下であることを示せ.
- (v) 行列 Q の最大の固有値に対応する左固有ベクトルを求め, 要素の和が 1 となるように正規化した形で示せ. ただし, 固有値 γ に対応する左固有ベクトルとは $\mathbf{x}Q = \gamma\mathbf{x}$ を満たし, かつ $\mathbf{x} \neq \mathbf{0}$ である行ベクトル $\mathbf{x}$ のことをいう.

専門用語の英訳

容量	: capacity
待ち行列	: queue
率	: rate
ポワソン過程	: Poisson process
指数分布	: exponential distribution
独立かつ同一に分布	: independent and identically distributed
出生死滅過程	: birth-and-death process
状態遷移速度図	: state transition rate diagram
行ベクトル	: row vector
固有値	: eigenvalue
左固有ベクトル	: left eigenvector

【情報理論】 解答は、桃色の解答用紙に記入すること．

次の通信路線図で表される 2 つの通信路 A および通信路 B について考える．どちらも送信側はビット 0 を確率 p_0 で独立に発生する無記憶情報源とし，その送信ビットを X で表す．また，通信路 A および通信路 B における受信ビットをそれぞれ Y_A および Y_B で表し，各矢印の値は遷移確率を表す．



次に，2 元エントロピー関数を次式で定義する．

$$h_2(p) \triangleq -p \log_2 p - (1-p) \log_2 (1-p), \quad p \in [0, 1]$$

ここで $p \log_2 p$ は $p = 0$ についてはその極限值 $\lim_{p \rightarrow +0} p \log_2 p$ で定義される．

以下の問いに答えよ．なお，各問いにおいて導出の過程も示すこと．

- 通信路 A および通信路 B のそれぞれについて， Y_A および Y_B が 0 である確率 $P(Y_A = 0)$ および $P(Y_B = 0)$ をそれぞれ p_0 を用いた式で表現せよ．
- Y_A および Y_B のエントロピー $H(Y_A)$ および $H(Y_B)$ をそれぞれ求め， p_0 と 2 元エントロピー関数 $h_2(\cdot)$ を用いた式で表現せよ．
- X が与えられた場合の Y_A および Y_B の条件付きエントロピー $H(Y_A|X)$ および $H(Y_B|X)$ をそれぞれ求め， p_0 と 2 元エントロピー関数 $h_2(\cdot)$ を用いた式で表現せよ．
- X と Y_A の相互情報量 $I(X; Y_A)$ および X と Y_B の相互情報量 $I(X; Y_B)$ をそれぞれ求め， p_0 と 2 元エントロピー関数 $h_2(\cdot)$ を用いた式で表現せよ．
- 通信路 A および通信路 B のそれぞれについて，通信路容量 C_A および C_B を数値で表せ．この結果から，どちらの通信路がより多くの情報を伝送できるかを答えよ．なお，数値計算においては次の $h_2(p)$ に対する数値表を用いてよい．

p	$\frac{1}{8}$	$\frac{1}{7}$	$\frac{1}{6}$	$\frac{1}{5}$	$\frac{1}{4}$	$\frac{2}{7}$	$\frac{1}{3}$	$\frac{3}{8}$	$\frac{2}{5}$	$\frac{3}{7}$
$h_2(p)$	0.54	0.59	0.65	0.72	0.81	0.86	0.92	0.95	0.97	0.99

専門用語の英訳

通信路線図	channel diagram
無記憶情報源	memoryless source
遷移確率	transition probability
エントロピー	entropy
2 元エントロピー関数	binary entropy function
条件付きエントロピー	conditional entropy
相互情報量	mutual information
通信路容量	channel capacity

【信号処理】 解答は, だいたい色の解答用紙に記入すること.

1. 連続時間信号 $x(t)$ に対し, 連続時間信号 $y(t)$ を

$$y(t) = x(t) e^{j\omega_c t} \sin(\omega_c t)$$

と与える. ここで, t は連続的な時刻を表す実数, ω_c は角周波数を表す正の実数である. 以下の問いに答えよ.

- (i) $\sin(\omega_c t)$ のフーリエ変換を求めよ. ただし, 以下の定義に従うデルタ関数 $\delta(t)$ を用いても良いとする.

$$\int_{-\infty}^{\infty} f(t) \delta(t - t_0) dt = f(t_0)$$

ここで, t_0 は時刻を表す. また, $f(t)$ のフーリエ変換 $\mathcal{F}[f(t)]$ を $F(\omega)$ としたとき (ω は角周波数を表す), フーリエ変換の双対性 $\mathcal{F}[F(t)] = 2\pi f(-\omega)$ を適宜用いて良い.

- (ii) $x(t)$ の周波数スペクトルを $X(\omega)$ としたとき, $X(\omega)$ を用いて, $y(t)$ の周波数スペクトル $Y(\omega)$ を表せ.
- (iii) $x(t)$ を, 角周波数 $-\omega_M$ から ω_M の範囲で帯域制限したとする. つまり, $x(t)$ の周波数スペクトル $X(\omega)$ は $|\omega| \geq \omega_M$ で 0 である. また, ω_M は正の実数であり, $\omega_M < \omega_c$ である. 周波数応答,

$$H(\omega) = \begin{cases} 1 & (|\omega| \leq \omega_L) \\ 0 & (|\omega| > \omega_L) \end{cases}$$

を持つ理想的なローパスフィルタによって, $y(t)$ から $ax(t)$ を抽出可能な ω_L の範囲を求めよ. ここで, 係数 a は任意の複素数であり, 求める必要はない.

2. システム L へ入力した離散時間信号 $x[n]$ とその出力である離散時間信号 $y[n]$ には以下の関係がある.

$$y[n] = L[x[n]] = \frac{1}{M} \sum_{m=0}^{M-1} x[n-m]$$

ここで, n は離散的な時刻を表す整数, M は 2 以上の整数である. 以下の問いに答えよ. また, 解答には, 以下の定義に従う離散時間信号 $g[n]$ の z 変換 $G(z)$ および離散時間フーリエ変換 $G(\Omega)$ を用いて良い.

$$G(z) = \sum_{n=-\infty}^{\infty} g[n] z^{-n}, \quad G(\Omega) = \sum_{n=-\infty}^{\infty} g[n] e^{-j\Omega n}$$

- (i) z 変換により得られるシステム L の伝達関数を $H(z)$ とするとき, システム L の周波数応答 $H(\Omega)$ の $0 \leq \Omega \leq \pi$ における振幅特性および位相特性を求めよ. ただし, Ω は正規化角周波数である.
- (ii) システム L に離散時間信号 $x[n] = u[n]u[-n+2]$ を入力する. $M=2$ としたとき, システム L

の出力である離散時間信号 $y[n]$ を, $u[n]$ を用いず表せ. ただし, $u[n]$ は以下のように定義される.

$$u[n] = \begin{cases} 1 & (n \geq 0) \\ 0 & (n < 0) \end{cases}$$

- (iii) システムを設計する際には, 入出力差分方程式 $\sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k]$ の係数 a_k および b_k を所望の周波数応答 $H(\Omega)$ が得られるように定める. このようなシステム設計を行うとき, 一般的には z 変換を行い, 信号解析を行う. しかし, 最終的には周波数応答を求めるにも関わらず, 離散時間フーリエ変換を直接用いずに z 変換を介するのはなぜか述べよ.

専門用語の英訳	
角周波数	angular frequency
フーリエ変換	Fourier transform
デルタ関数	delta function
双対性	duality
入出力差分方程式	input-output difference equation
z 変換	z transform
伝達関数	transfer function
正規化角周波数	normalized angular frequency
離散時間フーリエ変換	discrete time Fourier transform

【論理回路と計算機システム】 解答は、水色の解答用紙に記入すること。

1. 離散時間の時系列信号に対する移動平均フィルタを論理回路で実装することを考える。ここで扱う移動平均フィルタは、現在時刻 t ($t > 0$) から過去 $(L - 1)$ 時刻前までの入力信号値 $X_t, X_{t-1}, \dots, X_{t-L+1}$ の算術平均値 $Y_t = (X_t + X_{t-1} + \dots + X_{t-L+1})/L$ を出力する。入力信号値 X_t は $0, 1, 2, 3$ のいずれかの値を取るものとし、時刻 0 以前 ($t \leq 0$) では $X_t = 0$ であるとする。

まず、現在時刻の入力値 $X (= X_t)$ を 2 ビットの 2 進数で符号化する。つまり、 0 または 1 の値を取る 2 値変数 x_0, x_1 を用いて、 $X = (x_1 x_0)_2$ と表す。ここで、 $()_2$ は 2 進数表記であることを示す。

次に、目的の論理回路を図 1 のようにシフトレジスタ部と演算部に分けて設計する。シフトレジスタ部は、 K ($K \geq 1$) 時刻前に入力された値を $S = (s_1 s_0)_2$ として出力する論理回路である。ここで、 s_0, s_1 は 0 または 1 の値を取る 2 値変数とする。この回路は 2 つのシフトレジスタ R_i ($i = 0, 1$) で構成され、シフトレジスタ R_i は x_i を入力とし、 s_i を出力する。演算部は、現在時刻の入力値 X とシフトレジスタ部の出力値 S を用いて算術平均値を計算し、 Y として出力する。なお、シフトレジスタの初期状態は予め 0 が入力されているように設定する。

最初に、 $L = 2$ とした回路を設計することを考える (図 1)。シフトレジスタ部では $K = 1$ とし、現在時刻における回路の出力値 $Y (= Y_t)$ を 3 ビットで表現する。つまり、 0 または 1 の値を取る 2 値変数 y_0, y_1, y_2 を用いて、 $Y = (y_2 y_1 y_0)_2$ と符号化する (表 1)。「 $.$ 」を小数点として、 $y_2 y_1$ は整数部、 y_0 は小数部である。この論理回路に関し、以下の問いに答えよ。

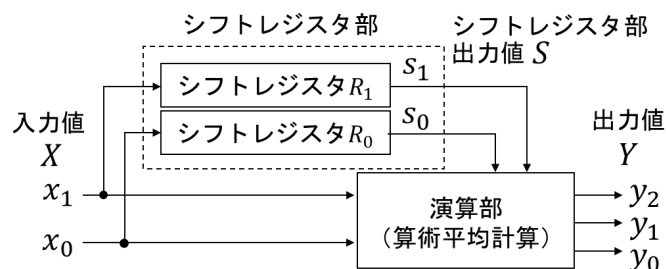


図 1 論理回路の構成

表 1 入力値と出力値の符号化

X	(x_1, x_0)	Y	(y_2, y_1, y_0)
0	(0, 0)	0.0	(0, 0, 0)
1	(0, 1)	0.5	(0, 0, 1)
2	(1, 0)	1.0	(0, 1, 0)
3	(1, 1)	1.5	(0, 1, 1)
		2.0	(1, 0, 0)
		2.5	(1, 0, 1)
		3.0	(1, 1, 0)

- (i) シフトレジスタ部の出力 $S = (s_1 s_0)_2$ と現在時刻の入力 $X = (x_1 x_0)_2$ の算術平均値 $Y = (S + X)/2$ を考える。この Y の各ビット y_0, y_1, y_2 に関する論理関数のカルノー図を、 x_0, x_1, s_0, s_1 を用いてそれぞれ書け。
- (ii) 問い(i)で解答したカルノー図に基づき、 y_0, y_1, y_2 の論理式を最小積和形でそれぞれ示せ。

続いて、移動平均フィルタの時刻範囲を $L > 2$ とする。この場合の演算部における算術平均値の計算手続き、および、そのための論理回路実装法について考える。

- (iii) 次の説明文は、算術平均値の計算に必要な加算回数を削減するための工夫について述べている。5 つの空欄 (a) から (e) にあてはまる適切な値または式を答えよ。

まず、算術平均値の計算手続きについて考える。単純に計算すると、信号値の和の計算部分で (a) 回の加算が必要となる。これに対し、時刻 t での信号値の和 $Z_t = X_t + X_{t-1} + \dots + X_{t-L+1}$ の再帰性に着目すると、 Z_t は X_t, Z_{t-1}, X_{t-L} を用いて、 $Z_t =$ (b) と表せる。このとき、加算は (c) 回、減算は (d) 回だけ必要となる。 Y_t は Z_t を L で割ることで計算できる。

次に、 Y_t を計算するための論理回路の構成について考える。まず、 Z_t の値を 1 時刻分記憶する論理回路を新たに設置することで、 Z_{t-1} の値が得られる。続いて、図 1 中のシフトレジスタ部における K の値を (e) に設定することで、 X_{t-L} の値が得られる。現在時刻の入力値 X_t と併せて Z_t を計算し、その結果を L で割る演算回路をそれぞれ構成すれば、算術平均値を計算できる。

2. 計算機の仮想記憶方式により、限られた主記憶容量で大規模なプログラムを実行したり、複数の仮想マシンが効率よく主記憶を共有したりできる。この方式では、個々のプログラムが持つアドレス空間を物理アドレスに変換する機能や、他の仮想マシンのアドレス空間を保護する機能が必要である。この際、ページと呼ばれる固定長のブロックを単位とする方式をページ方式、可変長のブロックを単位とする方式をセグメント方式という。またプロセッサからアクセスされた内容が主記憶上に存在しない場合をページフォールトと呼ぶ。以下の各問いに答えよ。

- (i) 仮想記憶が有効になる前提として参照の局所性がある。時間的局所性と空間的局所性についてそれぞれ説明せよ。
- (ii) セグメント方式と比較した際の、ページ方式の利点と欠点をそれぞれ答えよ。
- (iii) ページ方式において、主記憶が 4 ページからなる仮想記憶システムを考える。この上で、ページ 0 からページ 9 までの 10 ページからなるプログラムを実行したとき、以下の順にページ参照があったとする。

ページ参照：0→1→2→3→0→1→7→0→1→3→7→9→3→7→5

このとき、ページの置換えに LRU (least recently used) 法を使った場合のページフォールトの回数を答えよ。ただし、プログラムの実行前には主記憶は初期化されており、初期化された主記憶にページを読み込む回数もページフォールトとしてカウントするものとする。

- (iv) 記憶階層におけるデータの書き込み方式として、ライト・スルー (write-through) 方式と、ライト・バック (write-back) 方式がある。このそれぞれについて説明せよ。また、主記憶とディスクとの間の仮想記憶方式ではどちらが用いられるかを理由とともに答えよ。

専門用語の英訳

移動平均フィルタ	moving average filter
シフトレジスタ	shift register
状態	state
カルノー図	Karnaugh map
論理関数	logical function
論理式	logical formula, logical expression
最小積和形	minimum sum-of-products form
主記憶	main memory
仮想記憶	virtual memory
アドレス空間	address space
物理アドレス	physical address
保護	protection
固定長	fixed length
可変長	variable length
ページ方式	paging
セグメント方式	segmentation
プロセッサ	processor
ページフォールト	page fault
参照の局所性	locality of references
時間局所性	temporal locality
空間局所性	spatial locality
記憶階層	memory hierarchy, storage hierarchy

【データ構造とアルゴリズム】 解答は、青色の用紙に記入すること。

1. N 個の整数を並べた系列 $A_N := (a_1, \dots, a_N)$ と整数 $n \geq 2$ が与えられたとき、 A_N を $n \times n$ の行列の各要素に対応付ける手順を考える。以下の問いに答えよ。ただし、導出過程を明記すること。
 - (i) $N = n^2 - 1$ の場合、行列の要素数 n^2 を満たさないため、要素を 1 つ追加しなければならない。まず、 A_N の要素が長さ $N + 1$ の一次元配列のインデックス 0 から $N - 1$ に順に格納されているとする。新たに整数値を配列に挿入するとき、すでに存在している要素を後ろへ 1 つずつ動かす必要がある。挿入位置を一様ランダムに決める場合、この挿入操作に必要な平均計算量を n の関数として答えよ。ただし、要素 1 つの移動に必要な計算量は 1 であるとする。
 - (ii) 問い(i)の平均計算量の漸近的計算量を答えよ。ただし、 n に対して O -記法を使うこと。
 - (iii) 問い(ii)に続き、今度は配列ではなく N 個のセルを順番につなぎ合わせた連結リストによって A_N が格納されている場合を考える。セルのアドレスからポインタ部とデータ部は計算量 $O(1)$ で参照できるとする。挿入位置を一様ランダムに決め、連結リストの先頭セルと、新たに追加したいセルそれぞれのアドレスだけが与えられたとき、ポインタを付け替えてセルを挿入する手続きの漸近的平均計算量を n に対する O -記法で示せ。
 - (iv) $N = n^2$ のとき、系列 A_N の要素を列優先で対応付ける。つまり、行列の 1 列目は上から $(a_1, \dots, a_n)$ 、2 列目は上から $(a_{n+1}, \dots, a_{2n})$ 、という順序である。 A_N が長さ N の一次元配列のインデックス 0 から $N - 1$ へ順に格納されている場合、行列の「角」(左上, 左下, 右上, 右下の計 4 箇所) に対応する配列内のインデックスを列挙せよ。
 - (v) 問い(iv)に続き、 A_N は 1 から n^2 までの整数をそれぞれ一つずつ含むとする。行列の第 i 列の要素の和を C_i と表記する。 $C_1 = C_2 = \dots = C_n$ が成り立つ場合、 C_1 を n の関数として求めよ。
2. 任意の整数 $n > 2$ に対して、 $n \times n$ の行列の各要素に相異なる正の整数を割り当て、各々の行と列と対角の n 個の要素の和がすべて等しくなるとき、その行列は「 n 次の魔方陣」と呼ばれる。以下のプログラム 1 では、奇数の $n > 2$ (現状は $n = 3$) に対して、魔方陣を作る手順を示している。

【プログラム 1】 魔方陣を C 言語で作る例

```
1  #include <stdio.h>
2  int main() {
3      int n = 3; // 行列のサイズ (奇数であることが前提)
4      int B[n][n]; // 二次元配列
5      for (int i = 0; i < n; i++) // 配列の全要素を 0 に初期化
6          for (int j = 0; j < n; j++)
7              B[i][j] = 0;
8      int r = 0, c = (n-1)/2, r_new, c_new;
9      B[r][c] = 1; // まず 1 行目の中央に最初の要素 1 を配置する
10     for (int t = 1; t < n*n; t++) { // 「配置するマスの決定」と「整数値の配置」を繰り返す
11         r_new = (r-1+n) % n; // 枠からはみ出るときは反対側の行へ回る (剰余演算)
12         c_new = (c+1) % n; // 枠からはみ出るときは反対側の列へ回る (剰余演算)
13         if (B[r_new][c_new] != 0) { // 右上のマスが処理済みの場合は同列の下のマスに行く
14             r_new = (r+1) % n;
15             c_new = c;
16         }
17         B[r_new][c_new] = B[r][c] + 1; // 整数値を新たに配置する
18         r = r_new;
19         c = c_new;
20     }
21     return 0;
22 }
```

プログラム 1 を実行したときに作られる行列（配列 B に相当）を図 1 の (a)～(h) でステップごとに図示している（初期値の 0 は空欄として表示）．左上と右下のマスはそれぞれ $B[0][0]$ と $B[n-1][n-1]$ に相当する．図 1(h) では各々の行と列と対角の 3 個の要素の和はいずれも 15 である．上述の手順を前提として、以下の問いに答えよ．

	1	
		2

(a) $t = 1$

	1	
3		
		2

(b) $t = 2$

	1	
3		
4		2

(c) $t = 3$

	1	
3	5	
4		2

(d) $t = 4$

	1	6
3	5	
4		2

(e) $t = 5$

	1	6
3	5	7
4		2

(f) $t = 6$

8	1	6
3	5	7
4		2

(g) $t = 7$

8	1	6
3	5	7
4	9	2

(h) $t = 8$

図 1: プログラム 1 によって作られる 3 次の魔方陣をステップごとに図示．

- (i) プログラム 1 内の「 $n = 3$ 」を「 $n = 5$ 」に書き換えて実行した場合、5 次の魔方陣が作成される．結果として得られる行列を示せ．なお、大問 1 の問い(v)の結果は検算に利用できる．
- (ii) 問い(i)で得た行列の左端から 1 番目と 2 番目の列の要素（計 10 個）をもとに、根に最小値を持つヒープ（左詰め of 広義の完全二分木の一種）としての表現を考える．これらの要素を木の節点の値として、実際に木を描いて答えよ．ただし、ヒープ条件を維持しつつ、最深レベルの葉の和が最小となるように作ること．
- (iii) 問い(ii)で作成した木を深さ優先（通りがけ順）で一巡し、各節点の値を立ち寄り順に列挙せよ．
- (iv) 問い(ii)と同じ 10 個の要素を前提に、根に最小値を持つヒープの条件が満たされる木は複数考えられる．その中で、最大値の要素が入りうる最も浅い節点のレベルを理由と共に答えよ．ただし、根のレベルを 0 とする．
- (v) ヒープを一次元配列によって実現する場合、幅優先順にヒープの各節点に番号 $0, 1, 2, \dots$ を付け、それらを配列のインデックスと対応付ける．問い(ii)と同じ 10 個の要素が配列内で昇順に格納されるようにヒープを作成せよ．木を描いて答えること．
- (vi) プログラム 1 は整数列 $(1, 2, \dots, 9)$ の要素を順番に配置する手順である．基本的な手順をそのままに、1 から 9 までの整数ではなく、降順に並ぶ 9 個の偶数 $(18, 16, \dots, 2)$ をその順番に配置するようにコードを変更すること．変更すべきコードの行番号と変更内容を明記すると共に、変更後のプログラムの実行結果の行列（3 次の魔方陣）を示せ．

専門用語の英訳

行	row
列	column
整数	integer
配列	array
格納する	to store
昇順	ascending order
要素	element
計算量	computational complexity
漸近的計算量	asymptotic complexity
平均計算量	average-case complexity
連結リスト	linked list
列優先	column-major
魔方陣	magic square
ヒープ	heap
完全二分木	complete binary tree
深さ優先	depth-first
通りがけ順	in-order
幅優先	breadth-first

【情報セキュリティ】 解答は、緑色の解答用紙に記入すること。

1. RSA デジタル署名方式では、異なるランダムな2つの素数 p, q に対し、 $n = pq$ 及び $(p-1)$ と $(q-1)$ の最小公倍数 L に対して $1 < e < L$ かつ L と互いに素な整数 e のペア $\{e, n\}$ を公開鍵とする。また、 $ed \equiv 1 \pmod{L}$ かつ $1 < d < L$ を満たす d を生成し、これを秘密鍵とする。ここで、 $x \equiv y \pmod{L}$ は x, y を L で割ったときの余りが等しいことを意味する。一般には RSA デジタル署名方式ではハッシュ関数を利用するが、ここではハッシュ関数を利用しない方式を考える。つまり、署名生成者は、平文 m ($1 < m < n$) と秘密鍵 d を用いて、 $\sigma \equiv m^d \pmod{n}$ を満たす $1 < \sigma < n$ を署名 σ として生成する。一方、署名検証者は、署名 σ 、平文 m と公開鍵 n 及び e を用いて、 $m \equiv \sigma^e \pmod{n}$ を満たすとき、署名 σ は平文 m の正しい署名と判断する。RSA デジタル署名方式について以下の問いに答えよ。
 - (i) $p = 5, q = 17$ を用いて、公開鍵を構成する n を求めよ。
 - (ii) $p = 5, q = 17$ を用いて、問い(i)で求めた公開鍵 n と公開鍵 $e = 3$ に対する秘密鍵 d を求めよ。
 - (iii) 問い(ii)で求めた秘密鍵 d を用いて、平文 $m = 6$ の署名 σ を生成せよ。
 - (iv) 署名検証者が問い(i)で求めた公開鍵 n と $e = 3$ に対する平文 $m = 7$ とその署名 $\sigma = 3$ を入手したとき、署名 σ が平文 m の正しい署名であるか否かを示せ。
2. 問い1で述べた RSA デジタル署名の偽造とは、秘密鍵 d を持たない攻撃者が、平文 m^* に対する署名 σ^* を生成することを言う。ここで、公開鍵が $n = 85, e = 3$ であるとき、以下の問いに答えよ。
 - (i) 一般的選択平文攻撃とは、攻撃者が入手した平文と署名のペア (m, σ) を用いて、平文 $m^* \neq m$ に対する署名 σ^* を偽造することである。ここで、攻撃者は平文と署名のペア $(m, \sigma) = (8, 2)$ を入手したとする。 $(m, \sigma) = (8, 2)$ を用いて、 $m^* \neq 8$ である平文とその署名の偽造をせよ。なお、偽造の方法も記述せよ。
 - (ii) 適応的選択平文攻撃とは、攻撃者が偽造したい平文 m^* に対して、 $m \neq m^*$ の任意の平文 m に対する署名 σ を入手できる状況において、平文 m^* の署名 σ^* を偽造することである。適応的選択平文攻撃で平文 $m^* = 3$ を偽造するとする。このとき、 $2 \equiv 8^3 \pmod{85}$ であることから、平文 $m = 6$ の署名 $\sigma = 56$ を入手したとする。 $(m, \sigma) = (6, 56)$ を用いて、平文 $m^* = 3$ の署名を偽造をせよ。なお、偽造の方法も記述せよ。

専門用語の英訳

デジタル署名方式	digital signature scheme
素数	prime number
最小公倍数	least common multiple
互いに素	prime to each other
公開鍵	public key
秘密鍵	secret key
ハッシュ関数	hash function
平文	plaintext
署名	signature
偽造	forgery
一般的選択平文攻撃	generic chosen message attack
適応的選択平文攻撃	adaptive chosen message attack