

【I 期入試】

2026 年度 大学院入学試験問題

先端数理科学研究科 先端メディアサイエンス専攻（博士前期課程）

科目：専門科目

注意事項

- 1 試験時間は 90 分です。
- 2 外国人留学生志願者のみ、英語による解答を認めます。
- 3 問題は、〔I〕数学、〔II〕情報、〔III〕プログラミングの 3 分野から出題されています。3 分野のうち 2 分野を選択し解答しなさい。
- 4 選択した 2 分野の解答用紙の選択欄に「○」を、選択しなかった 1 分野の解答用紙の選択欄に「×」を、それぞれ必ず記入しなさい。選択欄にこれら以外の記載がある場合や、空欄がある場合、すべての答案を無効とします。
- 5 選択しなかった分野を含め、すべての解答用紙および問題用紙に受験番号と氏名を記入しなさい。
- 6 問題用紙は 11 ページあります（表紙は含まず）。
- 7 解答は、解答用紙に記入しなさい。
- 8 問題用紙、解答用紙は試験終了後すべて回収します。

受験番号 _____

氏 名 _____

(I) 数学

【問題 1】 行列 A を

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

とする。ある正則行列 P に対して $P^{-1}AP$ が対角行列となるとき、そのときの P と $P^{-1}AP$ を求めよ。もしいかなる正則行列 P に対しても $P^{-1}AP$ が対角行列とはならないならば、その理由を説明せよ。

【問題 2】 $\sinh x = \frac{e^x - e^{-x}}{2}$ とする。以下の問いに答えよ。

- (1) $y = \sinh x$ とするとき、 x を y の式で表せ。ただし、 $\operatorname{arcsinh}$ や $\sinh^{-1}$ は用いないものとする。
- (2) $\sinh x$ の逆関数を $\operatorname{arcsinh} x$ とするとき、 $(\operatorname{arcsinh} x)'$ を求めよ。
- (3) 不定積分 $\int \frac{dx}{\sqrt{x^2 + 3}}$ を求めよ。

〔II〕 情報

【問題 1】

次に示す手順は、列中の少なくとも一つは 1 であるビット列が与えられたとき、最も右にある 1 を残し、他のビットをすべて 0 にするアルゴリズムである。例えば、00110000 が与えられたとき、00010000 が求まる。

手順 1 与えられたビット列 A を符号なしの 2 進数と見なし、A から 1 を引き、結果を B とする。
手順 2 A と B の排他的論理和 (XOR) を求め、結果を C とする。
手順 3 A と C の 空欄 を求め、結果のビット列とする。

- (1) この手順に、「01010000」が与えられたときについて、手順 1 によって求めた結果のビット列 B を示せ。
- (2) (1) で求めたビット列 B を使い、手順 2 によって求めた結果のビット列 C を示せ。
- (3) 空欄 に入る演算名を答えよ。

【問題 2】

- (1) すべての命令が 6 サイクルで完了するパイプライン制御のプロセッサで、25 命令を実行するには何サイクル必要となるか答えよ。ただし、すべての命令は途中で停止することなく実行されるものとする。
- (2) パイプライン方式のプロセッサにおいて、パイプラインが分岐先の命令を取得するときに起こるハザードはどれか。
- (a) 制御ハザード
 - (b) データハザード
 - (c) 資源ハザード
 - (d) 構造ハザード
- (3) パイプライン処理を有効に機能させるプログラミング方法は次のうちどれか。
- (a) CASE 文を減らす。
 - (b) 関数の個数をできるだけ多くする。
 - (c) 分岐命令を多用する。
 - (d) メモリアクセス命令を少なくする。

【問題 3】

CSMA/CD は Carrier Sense Multiple Access with Collision Detection の略であり、複数のノードが共有する半二重通信路において、データの衝突を検出し、その衝突を回避して効率的な通信を行うための媒体アクセス制御 (MAC) 方式である。CSMA/CD について以下の問いに答えよ。

- (1) CSMA/CD を用いて通信するノードの動作を説明した以下の文中の空欄を埋めよ。

送信前には回線が使用中かどうかを検知し、回線が空いている場合にのみ送信を開始する。しかし、同時に複数のノードが送信を開始することでデータが衝突した場合、それを検知し、

空欄

した後に再送を試みる。

- (2) ノード A とノード B が 10Mbps のリンクで接続され、2 つのノード間の伝播遅延が 32.5×10^{-6} 秒であるとする。このネットワーク上で、512 ビットのメッセージに 64 ビットの制御情報が付加された固定長のフレームを用いて通信を行う。

時刻 $t = 0$ においてノード A がフレームの送信を開始することを考える。ここでは、A の送信が完了する前にノード B もフレームの送信を開始し、A は自身のフレームの送信を完了する前に衝突を検出できない状況が起こりうる。この状況が生じる t の条件を示した上で、その理由を簡潔に説明せよ。

なお、各ノードにおける処理のオーバーヘッドは無視できるほど小さく、遅延は揺らぎがなく一定とし、他のトラフィックは存在しないものとする。

- (3) CSMA/CD ではフレームサイズの下限を設けている。(2) の考え方を元にフレームサイズの下限をどのように設定すべきか簡潔に説明せよ。

[III] プログラミング

この問題では Processing という言語でプログラムが記述されている。
Processing は Java を基にしたプログラミング言語である。以下のプログラムの仕様を前提に各設問に答えよ。

- 関数 `setup()` はプログラム実行開始時に一度だけ呼び出される。
- 関数 `size(w, h)` は画面サイズを横 `w` ピクセル、縦 `h` ピクセルの大きさに設定する。
- 関数 `background(255)` は画面全体を白色で塗りつぶす。
- 関数 `line(x1, y1, x2, y2)` は座標 `(x1, y1)` と座標 `(x2, y2)` とを結ぶ線を描画する。
- 関数 `print(x)` は引数 `x` で指定された内容を標準出力（コンソール）に表示する。
- 関数 `println(x)` は引数 `x` で指定された内容を標準出力（コンソール）に表示した後、改行する。
- 整数同士の演算では、その計算結果の小数点以下の値は切り捨てられる。
- 関数 `int(x)` は引数 `x` で指定された値を小数点以下切り捨てで整数にしたものを返す。
- 関数 `random(m, n)` は `m` 以上 `n` 未満の実数の乱数を出力する。
- 配列 `array` に対し、`array.length` は、`array` の要素数を表す変数である。

Processing における主な算術演算子

- + 足す（加算）
- 引く（減算）
- * 掛ける（乗算）
- / 割る（除算）
- % 割った数の余り（剰余）

【問題 1】

以下のプログラムを実行した際に、標準出力に表示される数値を答えよ。

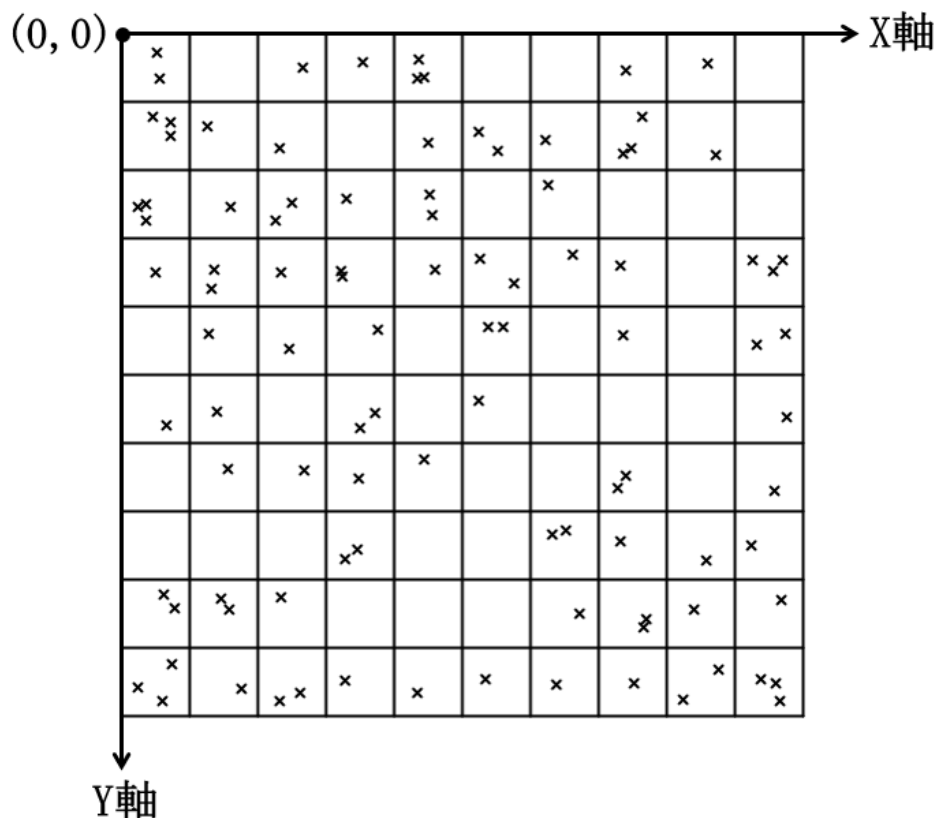
```
int check(int a, int b) {  
    if (a > b) {  
        return a;  
    }  
    return b;  
}  
  
void setup() {  
    int n = 0;  
    for (int i = 1; i < 5; i = i + 2) {  
        for (int j = 1; j < 4; j = j + 1) {  
            n = n + check(i, j);  
        }  
    }  
    println(n);  
}
```

【問題 2】

下図のように、1 マスのサイズが 50×50 、全体のサイズが 500×500 の格子が $(0, 0)$ を起点に置かれている。格子上のすべての辺は X 軸、Y 軸のいずれかに平行であるものとする。

この格子内に点 $(x[i], y[i])$ が 100 個存在する ($i = 0 \sim 99$)。 $x[i]$ と $y[i]$ は対応する点の X 座標と Y 座標をそれぞれ格納した float 型配列であり、座標値の範囲は $0 < x[i] < 500$ 、 $0 < y[i] < 500$ とする。また、すべての点はいかなる格子の辺とも重ならない座標値とする。

各マスの中にある点の数は 5 個以下である必要がある。すべてのマスにおいてこの条件が満たされていれば true を、そうでなければ false を返す関数 `judge()` を作成したい。解答用紙の空白部分を埋め完成させよ。



【問題 3】

以下はある波形を画面に描画するプログラムである。(A)～(D)の実行結果に対応する「処理①と処理②の組み合わせ」を、処理①の選択肢群(a)～(e)と処理②の選択肢群(f)～(j)の中から選べ。

なお、画面内の座標系は、画面左上の点を原点 (0, 0) とし、横方向を X 軸（右方向が正）、縦方向を Y 軸（下方向が正）とする。

```
float[] signal = new float[400];

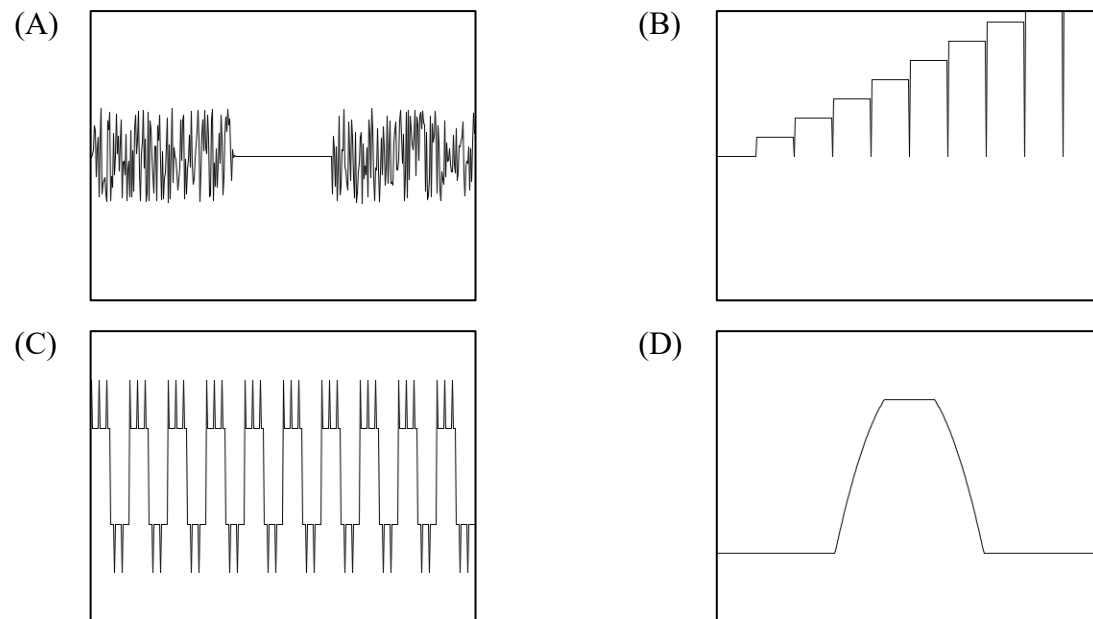
void setup() {
    size(400, 300);

    for (int i = 0; i < 400; i++) {
        処理①
    }
    for (int i = 0; i < 400; i++) {
        処理②
    }

    // 画面全体を白色で塗りつぶす
    background(255);

    // 波形の描画
    for (int i = 0; i < signal.length - 1; i++) {
        line(i, 150 - signal[i], i + 1, 150 - signal[i+1]);
    }
}
```

実行結果



処理①の選択肢群

(a)	<code>signal[i] = i * 0.5;</code>
(b)	<code>signal[i] = -0.03 * (i - 200) * (i - 200) + 100;</code>
(c)	<pre>if (i % 40 < 20) { signal[i] = 50; } else { signal[i] = -50; }</pre>
(d)	<code>signal[i] = random(-50, 50);</code>
(e)	<code>signal[i] = (i / 40) * 20;</code>

処理②の選択肢群

(f)	<code>signal[i] = -signal[i];</code>
(g)	<code>if (i >= 150 && i <= 250) signal[i] = 0;</code>
(h)	<pre>if (signal[i] > 80) signal[i] = 80; if (signal[i] < -80) signal[i] = -80;</pre>
(i)	<code>if (i % 40 == 0) signal[i] = 0;</code>
(j)	<code>if (i % 8 == 0) signal[i] = 2 * signal[i];</code>

【問題 4】

下図に示すプログラムは、正の整数が複数個格納された配列 **a** の最大値を出力するものである。なお、配列 **a** の宣言および配列 **a** への値の代入を行う部分は下図には含まれていない。

```
int m = -1;

for (int i = 0; i < a.length; i++) {
    if (a[i] > m) {
        m = a[i];
    }
}

println(m);
```

図 1

- (1) 上記のコードにならい、配列 **a** の要素のうち、一番目に大きい要素および二番目に大きい要素の値をそれぞれ出力するプログラムを示せ。例えば、配列 **a** の中身が 4, 6, 5, 7, 3 であった場合は、期待される出力は図 2 となる。ただし、同値の要素であっても別々に数えるものとする。例えば、4, 6, 5, 7, 3, 7 の場合は、期待される出力は図 3 となる。

7
6

図 2

7
7

図 3

- (2) (1)で解答したプログラムを修正し、上位 n 個の要素の値を出力するよ
うに、改めたものを示せ。 n は変数としてあらかじめ定められているもの
とする。また n の値は、配列 a の長さに比べて小さいことを前提として
よい。
- (3) (2)の処理について、 n を大きくした際にどのような問題がありうるか、
 n と計算量との関係を示しつつ論じよ。