

【Ⅱ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 10 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】 $\mathbb{R}^n$ の部分空間 V に対して、 $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m, \vec{b} \in V$ とし、 $\{\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m\}$ は一次独立とする。このとき、 $\{\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m, \vec{b}\}$ が一次従属であることと、 $\vec{b}$ が $\vec{a}_1, \vec{a}_2, \dots, \vec{a}_m$ の一次結合で表せることが必要十分条件であることを示せ。

【問題2】 以下の問いに答えよ。

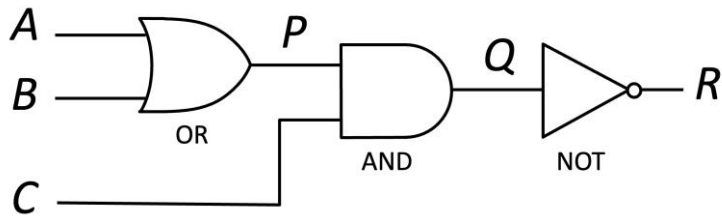
(1) $\lim_{x \rightarrow 0} \frac{\sin(2x) - 2x}{x^3}$ を求めよ。

(2) $D = \{(x, y) \mid 0 \leq x \leq 1, \sqrt{1-x^2} \leq y\}$ とするとき、 $\iint_D \frac{2 \, dx \, dy}{y^2}$ を求めよ。

〔II〕 情報

【問題 1】

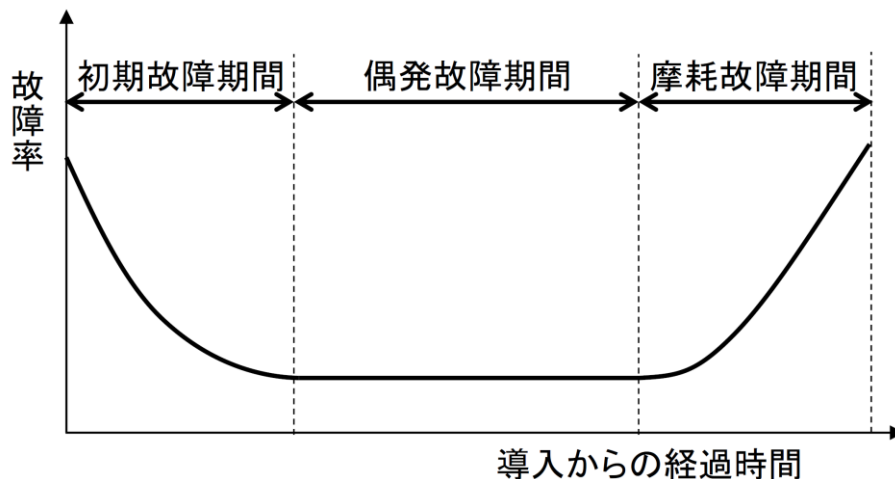
次の論理回路について、各設問に答えよ。



- (1) $A=1, B=0, C=1$ のとき、 P, Q, R の値はそれぞれどうなるか示せ。
- (2) A, B, C を入力として R を出力するような論理式を示せ。

【問題 2】

一般的な工業製品は、故障率と導入からの経過時間の関係が下図のようになると言われている。これは故障率曲線と呼ばれるもので、形状がバスタブに似ていることからバスタブ曲線と呼ばれることもある。故障という観点では製品のライフサイクルは初期故障期間、偶発故障期間、摩耗故障期間に分類される。初期故障期間は導入当初の高い故障率が徐々に下がっていく期間である。偶発故障期間は偶発的な原因によって故障がランダムに発生する期間であるが、故障率が低く安定的に製品を使用できる期間である。また摩耗故障期間は、故障率が高まっていく期間である。たとえば機械や電子部品で構成される装置の故障率は、部品の劣化等の原因からこのような変化をたどると考えられる。



あるソフトウェア製品の故障率がこのような変化をたどったとする。このとき、各期間におけるこのソフトウェア製品の故障の原因と、故障を防ぐために取りうる対策について、次の問いに答えよ。

- (1) このソフトウェア製品の初期故障期間における故障の原因として考えられるものは何か。解答欄 (A) に記せ。またそれを防ぐための対策として有効と考えられるものは何か。解答欄 (B) に記せ。
- (2) このソフトウェア製品の摩耗故障期間における故障の原因として考えられるものは何か。解答欄 (A) に記せ。またそれを防ぐための対策として有効と考えられるものは何か。解答欄 (B) に記せ。

【問題 3】

送信元から宛先まで k ホップの経路で構成されるパケット交換ネットワークがある。このネットワークでは、 p ビットのデータ部と h ビットのヘッダ部を持つ固定長のパケットを送信する。中継装置の転送方式はストアアンドフォワード方式、すなわち中継装置はパケットを一旦すべて受信した後、直ちに次の宛先への転送を開始する方式とする。

回線のビットレートは b ビット/秒で伝播遅延は無視できるものとする。このネットワークについて以下の問いに答えよ。

- (1) $k=2$ の経路で 3 つのパケットを伝送するのに要する時間（送信元ノードが送信を開始してから宛先ノードで受信が完了するまでの時間）を求めよ。
- (2) x バイトのファイルを k ホップの経路で伝送するのに要する時間を求めよ。
- (3) x バイトのファイルの伝送時間が最も小さくなるように p の値を決めたい。 p をどのように決めれば良いか、その考え方や方法を述べよ。ネットワークの規格（例えば Ethernet での最大フレーム長）や通信路における誤り率についても言及すること。なお、 x は p に比べて複数のパケットに分割できる程大きいものとする。

[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。
- 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。
- 配列 `array` に対し、`array.length` は、`array` の要素数を表す変数である。

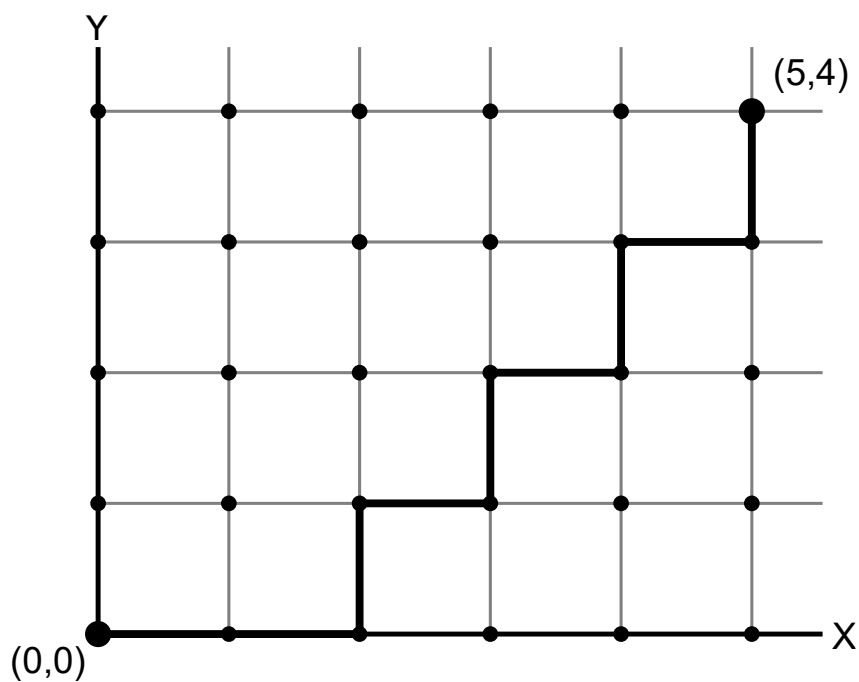
Processing における主な算術演算子

- + 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】

下図のように 2 次元平面上の格子点(X 座標・Y 座標ともに整数である点)のみを通る経路のうち、原点 $(0,0)$ から座標 (m,n) までの最短経路の数を求めたい (ただし $m,n > 0$)。なお、点 (x,y) から移動できるのは点 $(x+1,y)$ か点 $(x,y+1)$ のいずれかとする。下のような関数で最短経路の数を求めるとき、空欄①～④にはそれぞれどのような式が入るか答えよ。

```
int count_paths(int m, int n) {  
    if (m < 1 || n < 1) {  
        return 1;  
    }  
    return count_paths( ① , ② ) + count_paths( ③ , ④ );  
}
```



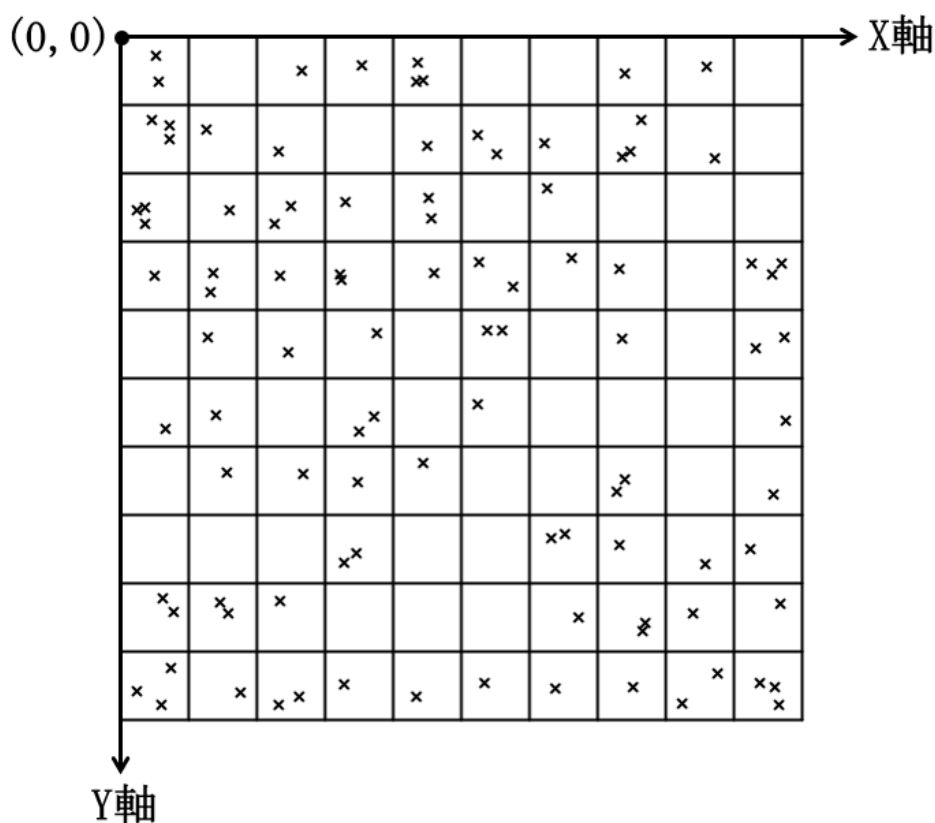
(0,0) から (5,4) までの最短経路の例

【問題 2】

画面上に点 $(x[i], y[i])$ が 100 個定義されている ($0 \leq i < 100$)。 $x[i]$ 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。

ここで下図のように、1 マスのサイズが 50×50 、全体のサイズが 500×500 の格子を $(0, 0)$ を起点に置くことを考える。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。

各マスに含まれるすべての点は、そのマスの上・下・左・右の各辺に対し、最短距離が 5 以上離れている必要がある。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】

以下はある模様が描かれた画像を生成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、map は縦 10 マス・横 16 マスの模様データを格納した 2 次元配列である。また、関数 createPattern() は map の内容をもとに模様が描かれた画像を生成する関数であり、配列要素が 0 のときに白いマス、1 のときに黒いマスを画像上に描画する。(A)～(D)の実行結果の画像ではマスの境界が見やすいように黒い枠線が描かれている。なお、画像の座標系は、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。

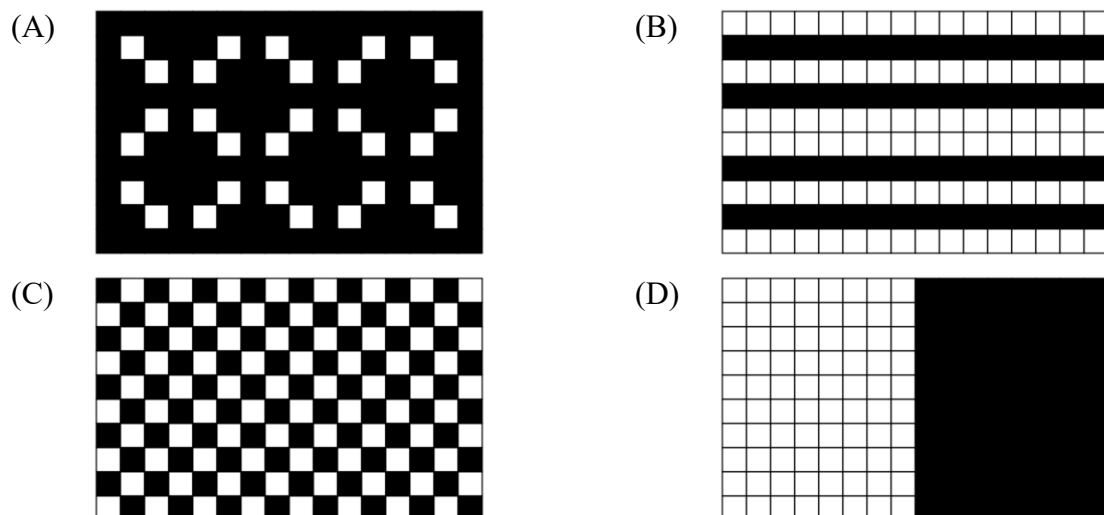
```
int WHITE = 0;
int BLACK = 1;
int[][] map = new int[10][16];

void setup() {
  for (int y = 0; y < 10; y++) {
    for (int x = 0; x < 16; x++) {
      処理①
    }
  }

  for (int y = 0; y < 10; y++) {
    for (int x = 0; x < 16; x++) {
      処理②
    }
  }

  createPattern();
}
```

実行結果



処理①の選択肢群

(a)	<code>map[y][x] = (x + y) % 2;</code>
(b)	<code>map[y][x] = x % 2;</code>
(c)	<code>map[y][x] = y % 2;</code>
(d)	<pre>if ((x * y) % 3 == 0) { map[y][x] = BLACK; } else { map[y][x] = WHITE; }</pre>
(e)	<code>map[y][x] = WHITE;</code>

処理②の選択肢群

(f)	<code>if (x > 7) map[y][x] = 1 - map[y][x];</code>
(g)	<code>if (y > 4) map[y][x] = 1 - map[y][x];</code>
(h)	<code>if (x % 2 == 1) map[y][x] = WHITE;</code>
(i)	<code>if ((x + y) % 2 == 1) map[y][x] = BLACK;</code>
(j)	<code>map[y][x] = 1 - map[y][x];</code>

【問題 4】

いま、 n 次多項式 $p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ の各係数 a_i は、配列 a の i 番目の要素 ($a[i]$) で与えられているものとする。 $p(x)$ を求めるとき、まず $a_n x^n$ を計算し、その結果に $a_{n-1} x^{n-1}$ を足し…と繰り返して計算すると乗算の回数が多くなってしまう。ここで、 $p(x)$ を以下のように変形すると、乗算の回数を削減することができる。この計算方法は **Horner 法** と呼ばれる。

$$p(x) = (\cdots ((a_n x + a_{n-1})x + a_{n-2})x + \cdots + a_1)x + a_0$$

- (1) $a_n x^n$ をまず計算し、その結果に $a_{n-1} x^{n-1}$ の計算結果を足し…と繰り返して計算した場合の乗算回数 $f(n)$ と、**Horner 法** を用いた時の乗算回数 $g(n)$ を、それぞれ n を用いた式で表せ。
- (2) 次数 n 、係数を収めた配列 a 、および変数 x を引数として受け取り、**Horner 法** を用いて n 次多項式の計算結果を返す関数 `eval_polynomial` を作成したい。解答用紙の空白部分を埋め完成させよ。