

2023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

事項

試験時間は90分です。

外国人留学生志願者のみ、英語による解答を認めます。

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。

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。

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。

問題用紙は10ページあります（表紙は含まず）。

解答は、解答用紙に記入しなさい。

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

1期

2期

2022

1期

2期

2022

1期

2期

(I) 数学

【問題1】行列 A を

$$A = \begin{pmatrix} 1 & 1 & 2 \\ 1 & 2 & 1 \\ 2 & 1 & 1 \end{pmatrix}$$

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、その理由を述べよ。

【問題2】

(1) 次の定積分の値を求めよ。

$$\int_0^1 \frac{dx}{(x^2+1)^2}$$

(2) $D = \{(x, y) \mid x \geq 0, y \geq 0, x^2 + y^2 \leq 4, (x-1)^2 + y^2 \geq 1\}$ のとき、次の2重積分の値を求めよ。

$$\iint_D \frac{dx dy}{y^2}$$

【問題3】3頂点あるグラフで、どの2つの頂点も1本の辺で結ばれているグラフを K_3 と表す。この各頂点を赤、青、黄のいずれかの色で塗るとき、何通りの塗り方があるか。ただし、グラフの型で移りあえるものは同じとみなす。

(II) 情報

問題1】

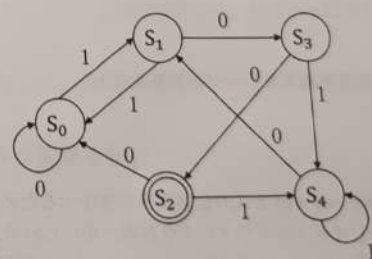
記数法に関する以下の問いに答えよ。なお、下付きの数値は基数を表す。

- 2の補数表現された8ビットの値 $A6_{(16)}$ を符号付き10進数で表せ。
- 8ビットの整数 $X = -76_{(10)}$ について、 $X \gg 1$ (右へ1ビット算術シフト) を求め、16進数で表せ。
- 整数部4ビットが $B_{(16)}$ 、小数部4ビットが $9_{(16)}$ の固定小数点数 $B.9_{(16)}$ を符号無し10進数で表せ。

問題2】

有限オートマトンに関する以下の問いに答えよ。

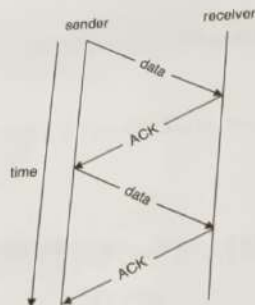
- 以下の状態遷移図の有限オートマトンにおいて受理状態の集合 $F = \{S_2\}$ である場合、初期状態に依存せず受理される文字列を1つ挙げよ。



- 0と1のみから構成される文字列を対象に、2個以上かつ偶数個の0を含む文字列のみを受理する有限オートマトンの状態遷移図を示せ。

【問題3】

ACKによる受信確認を用いた通信方法の一つにstop and wait方式がある。stop and wait方式による通信例を下図に示す。図のようにstop and wait方式では、1つのパケットを送信し、それに対するACKが返るまで次のパケットの送信を行わない。



stop and wait方式の通信において帯域が B [ビット/秒]、RTT (Round Trip Time) が R [秒]、1パケットの大きさが L [バイト]であるとき、以下の問いに答えよ。なお、通信の遅延は往復とも等しいとし、ACKパケットの大きさおよび送受信側のコンピュータでの内部処理に要する時間は無視できる程度に小さいものとする。また、途中でのパケットの損失はないものとする。

- (1) 1パケットのデータを送信し始めてから、受信確認が終わるまでに要する時間を求めよ。
- (2) (1)の転送を行う際の伝送効率とスループットを求めよ。
- (3) 遅延が大きな通信路ではstop and wait方式の伝送効率が悪化するため、受信確認を行い、かつ伝送効率を高めるためにスライディングウィンドウ方式が用いられる。非常に大きな帯域の通信路では、スライディングウィンドウ方式のスループットは何にほぼ比例し、何にほぼ反比例するか、それぞれ示せ。なお、ウィンドウサイズは W [バイト]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。
- 関数 `draw()` は画面を描画するために定期的に呼び出される。
- 関数 `mousePressed()` はプログラム実行時に画面内をマウスクリックされたときに呼び出される。
- 変数 `mouseX` は、その時点でのマウスのX座標の値が格納されている。
- 変数 `mouseY` は、その時点でのマウスのY座標の値が格納されている。
- 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 `circle(x, y, d)` は座標 (x, y) を中心として直径 d の正円を描画する。
- 関数 `noFill()` を利用すると、図形を描画する際に塗りつぶさなくなる。
- 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。
- 関数 `sqrt(x)` は x の平方根を求める。
- 関数 `dist(x1, y1, x2, y2)` は座標 $(x1, y1)$ から座標 $(x2, y2)$ までの距離を求める。
- 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。

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】

以下に示すプログラムを実行すると、標準出力に 3 行文字列が出力される。
 それぞれの行に何が表示されるか解答用紙に記入せよ。

```
void setup() {
  int x = 25;
  int y = 5;
  println( func1(x) + func1(y) );
  println( func2(x, y) );
  println( func3(y) );
}

int func1(int a) {
  return (a % 10) * 10 + (a / 10);
}

int func2(int a, int b) {
  while (b > 0) {
    a = a + b;
    b = b - 1;
  }
  return a;
}

int func3(int a) {
  if (a > 1) {
    return a * func3(a - 1);
  }
  return 1;
}
```

【問題 2】

横 600 ピクセル、縦 400 ピクセルのウィンドウに中心が(200, 200)で直径 300
 ピクセルの円 A、中心が(400, 200)で直径 200 ピクセルの円 B が描かれている。

このウィンドウ上の任意の点をクリックしたとき、円 A と円 B の交わる領域
 であれば「A and B」、円 A 内で円 B に含まれない領域であれば「A only」、円 B
 内で円 A に含まれない領域であれば「B only」、円 A にも円 B にも含まれない
 領域であれば「not A or B」と標準出力するプログラムを作るため、関数 judge()
 を完成させたい。解答用紙の空白部分を埋めプログラムを完成させよ。

```
void setup() {
  size(600, 400);
  noFill();
}

void draw() {
  circle(200, 200, 300); // 円 A
  circle(400, 200, 200); // 円 B
}

void mousePressed() {
  judge(mouseX, mouseY);
}

void judge(int x, int y) {

}
```


【問題3】

(1) 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。プログラム中に示される空欄に(A)~(F)の条件式を当てはめた際の実行結果を(ア)~(ク)の中から選べ。

なお、cell 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 cell 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1のときに黒いマスを、0のときに白いマスを描画する。

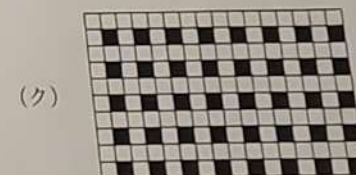
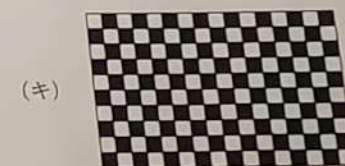
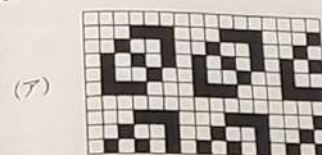
```
int WHITE = 0, BLACK = 1;
int[][] cell = new int[10][16];

void setup() {
  for (int y = 0; y < 10; y++) {
    for (int x = 0; x < 16; x++) {
      if (  ) {
        cell[y][x] = BLACK;
      } else {
        cell[y][x] = WHITE;
      }
    }
  }
  createPattern();
}
```

条件式

- | | |
|-----|--|
| (A) | $y \% 2 > 0$ |
| (B) | $x \% 3 > 0$ |
| (C) | $(x + y) \% 3 > 0$ |
| (D) | $(x * y) \% 2 > 0$ |
| (E) | $(x + y * 2) \% 5 > 0$ |
| (F) | $((x * y) \% 2 + (x * y) \% 3) \% 2 > 0$ |

実行結果



1 万までの友愛数を求めるプログラムを作成したい。ここで友愛数とは、異なる 2 つの自然数の組で、自分自身を除いた約数の和が、互いに他方と等しくなるもののことである。例えば、220 の自分自身を除いた約数は 1, 2, 4, 5, 10, 11, 20, 22, 44, 55, 110 で和は 284 であり、284 の自分自身を除いた約数は 1, 2, 4, 71, 142 で和は 220 となり、220 と 284 が友愛数となる。

次に、その calcSumDivisor 関数を使いながら友愛数のペアを求め、左側に小さい値、右側に大きい値を並べて **(220, 284)** のように 1 行ずつ標準出力するプログラムを setup() の解答用紙の空白部分を埋め完成させよ。

10