

# 令和4年度

## 名古屋大学大学院情報学研究科 知能システム学専攻 入学試験問題（解析・線形代数）

令和3年8月5日

### 注意事項

1. 試験開始の合図があるまでは、この問題冊子を開いてはならない。
2. 試験終了まで退出できない。
3. 外国人留学生の志願者は、日本語と日本語以外の1言語間の辞書1冊に限り使用してよい。電子辞書の持ち込みは認めない。
4. 日本語または英語で解答すること。
5. 問題冊子、解答用紙1枚、草稿用紙1枚が配布されていることを確認すること。
6. 解答用紙の所定の欄に受験番号を必ず記入すること。解答用紙に受験者の氏名を記入してはならない。
7. 解答用紙に書ききれない場合は、裏面を使用してもよい。ただし、裏面を使用した場合は、その旨、解答用紙表面右下に明記すること。
8. 解答用紙は試験終了後に提出すること。
9. 問題冊子、草稿用紙は試験終了後に持ち帰ること。

# 解析・線形代数

(解の導出過程も書くこと)

[1] 複素数  $z$  に関する次の方程式を解け。 $i$  は虚数単位を表す。

$$z^2 + 2z + 1 - 2i = 0$$

[2] 次の関数について、極値を求めよ。

$$f(x, y) = x^3 + y^3 - 3xy$$

[3] 次の2次形式について、以下の問いに答えよ。 $a$  は定数である。

$$Q(x_1, x_2) = ax_1^2 + ax_2^2 + 2(a+1)x_1x_2$$

(a) 対称行列  $A$  を用いて、 $Q = {}^t x A x$  と表すとき、 $A$  を示せ。 ${}^t x$  は  $x$  の転置を表す。

(b)  $A$  の固有値を求めよ。

(c) 2次形式  $Q$  が定符号であるための  $a$  の範囲を定めよ。

[4] 次の微分方程式 (\*) について、以下の問いに答えよ。ただし、 $x > 0$  とする。

$$x^2 \frac{d^2 y}{dx^2} - 3x \frac{dy}{dx} + 4y = 0 \quad (*)$$

(a) 微分方程式 (\*) の解の一つは  $y = x^m$  の形である。 $m$  の値を求め、この解を示せ。

(b) (a) で示した微分方程式 (\*) の解と  $x$  の関数  $u(x)$  を用い、 $\frac{du}{dx} = z$  とおいて、微分方程式 (\*) を  $z$  の  $x$  に関する微分方程式に書き換えよ。

(c) (b) で得られた微分方程式を解き、微分方程式 (\*) の一般解を求めよ。

## Translation of technical terms

複素数	complex number
虚数単位	imaginary unit
極値	extreme value
定数	constant
転置	transpose
定符号	definite
解	solution
一般解	general solution

方程式	equation
関数	function
2次形式	quadratic form
対称行列	symmetric matrix
固有値	eigenvalue
微分方程式	differential equation
値	value

令和4年度

名古屋大学大学院情報学研究科  
知能システム学専攻  
入学試験問題（確率・統計）

令和3年8月5日

注意事項

1. 試験開始の合図があるまでは、この問題冊子を開いてはならない。
2. 試験終了まで退出できない。
3. 外国人留学生の志願者は、日本語と日本語以外の1言語間の辞書1冊に限り使用してよい。電子辞書の持ち込みは認めない。
4. 日本語または英語で解答すること。
5. 問題冊子、解答用紙1枚、草稿用紙1枚が配布されていることを確認すること。
6. 解答用紙の所定の欄に受験番号を必ず記入すること。解答用紙に受験者の氏名を記入してはならない。
7. 解答用紙に書ききれない場合は、裏面を使用してもよい。ただし、裏面を使用した場合は、その旨、解答用紙表面右下に明記すること。
8. 解答用紙は試験終了後に提出すること。
9. 問題冊子、草稿用紙は試験終了後に持ち帰ること。

## 確率・統計

解の導出過程も書くこと。問題を解く上で必要なら表に示す対数の値を用いてもよい。

[1] 長さ1の線分上の無作為に選ばれた位置に点を配置し、この点で線分を切断する。切断後の全ての線分の長さは0より大きいとする。このとき、以下の問いに答えよ。

- (1) 点を1つ配置し、線分を切断して2つに分ける。切断後の線分の中で少なくとも1つの長さが0.7より長くなる確率を求めよ。
- (2) 点を2つ配置し、線分を切断して3つに分ける。切断後の線分の中で少なくとも1つの長さが0.5より長くなる確率を求めよ。

[2] 確率変数 $X$ は区間 $[1,3]$ における連続一様分布に従う。

- (1)  $X$ の確率密度関数を書け。
- (2)  $Y = X^2 + 2$ とおくとき  $Y$ の確率密度関数を求めよ。

[3] 1から6の目が全て同一の確率で出るサイコロがある。

- (1) サイコロを $n$ 回振ったとき1の目が出る回数を $X$ とおく。1の目が $X$ 回出る確率 $P(X)$ を書け。
- (2) 1の目が全く出ない確率を0.4以上とするためには、サイコロを何回まで振ることができるか。
- (3) サイコロを10回振ったとき、1の目が出る回数が1回以下となる確率は0.5以上になるか。理由を添えて答えよ。

表 対数  $\log_e x$  の値

$\log_e 2$	$\log_e 3$	$\log_e 4$	$\log_e 5$	$\log_e 6$	$\log_e 7$	$\log_e 8$	$\log_e 9$
0.6931	1.0986	1.3863	1.6094	1.7918	1.9459	2.0794	2.1972

### Translation of technical terms

対数: logarithm, 線分: line segment, 無作為: random, 点: point, 確率: probability,

確率変数: random variable, 連続一様分布: continuous uniform distribution,

確率密度関数: probability density function, サイコロ: dice,

令和4年度

名古屋大学大学院情報学研究科  
知能システム学専攻  
入学試験問題（プログラミング）

令和3年8月5日

注意事項

1. 試験開始の合図があるまでは、この問題冊子を開いてはならない。
2. 試験終了まで退出できない。
3. 外国人留学生の志願者は、日本語と日本語以外の1言語間の辞書1冊に限り使用してよい。電子辞書の持ち込みは認めない。
4. 日本語または英語で解答すること。
5. 問題冊子、解答用紙1枚、草稿用紙1枚が配布されていることを確認すること。
6. 解答用紙の所定の欄に受験番号を必ず記入すること。解答用紙に受験者の氏名を記入してはならない。
7. 解答用紙に書ききれない場合は、裏面を使用してもよい。ただし、裏面を使用した場合は、その旨、解答用紙表面右下に明記すること。
8. 解答用紙は試験終了後に提出すること。
9. 問題冊子、草稿用紙は試験終了後に持ち帰ること。

## プログラミング

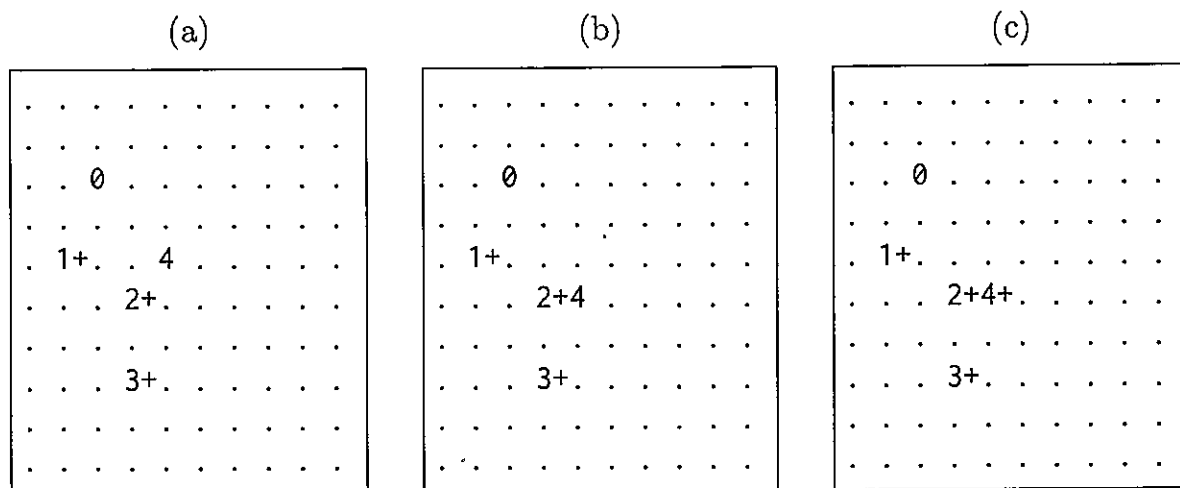
3 ページ目からのプログラムは、ヒトの移動による感染症の広がりをシミュレーションするプログラムである。なお、プログラム中の「」は空白記号である。

プログラムにおいて112行目から116行目が各ヒトの初期設定を示している。5項組で示される値は、順番にヒトの位置のx座標、y座標、x座標の移動方向、y座標の移動方向、感染の有無を示している。例えば113行目の{1, 2, 0, +1, True}は、ヒト1が最初に(1, 2)にいて、次の時刻には(1+0, 2+1)へ移動し、現時点で感染していることを示している。ある時刻において、感染しているヒトと8方向いずれかで隣接するヒトは感染する。その後、設定された移動方向に従ってヒトは移動する。このとき、SIZE×SIZEの範囲外には移動しない。移動できない場合はその場に留まり、次の時刻からは逆方向に移動する。

次ページは出力結果である。この出力結果において、各数字はヒトを表わし、数字の横の+はそのヒトが感染していることを表わす。time 0は初期状態を出力したものであり、time 9までの結果が出力されている。また最後にヒト0の感染経路が表示されている。

このとき以下の問いに答えよ。

- (1) プログラム中の空欄A, B, Cを埋めてプログラムを完成させよ。
- (2) time 2の出力結果として正しいものを以下から選べ。



- (3) 48行目からの関数 update() の目的を説明せよ。
- (4) コメントアウトされた96行目から97行目はデバッグ用の記述である。コメントアウトを解除し、この行を実行すると、例えばtime 8の状態を示す部分の表示は右のような結果になる。time 9の状態を同様の形式で記述せよ。

```

8: 0(8,8) (+1,+1)
8: 1(1,9) (+0,-1)
8: 2(9,9) (+1,+1)
8: 3(3,6) (+0,-1)
8: 4(4,9) (+0,+1)

```

- (5) 出力結果のtime 9において0が表示されていない。その理由を説明せよ。

### Translation of technical terms

感染症	infectious disease	シミュレーション	simulation
初期設定	initial setting	5項組	5-tuple
		座標	coordinate
感染	infection	隣接する	neighbor
		経路	path
関数	function	コメントアウト	comment out
		デバッグ	debug

出力結果

```

time 0
0 . . . . .
. 1+ . . . . .
. . 4 . . . . .
. . 2 . . . . .
. . 3+ . . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
. . . . .
time 1
. . . . .
. 0 . . . . .
. 1+ . . . . .
. . 2+ 4 . . . . .
. . . 3+ . . . . .
. . . . .
. . . . .
. . . . .
time 2
[Redacted]
time 3
. . . . .
. . . 0 . . . . .
. 1+ . 4+ . . . . .
. . . 2+ . . . . .
. . . 3+ . . . . .
. . . . .
time 4
. . . . .
. . . 0 . . . . .
. . 4+ . . . . .
. 1+ . . . . .
. . . 2+ . . . . .
. . . 3+ . . . . .

```

```

time 5
. . . . .
. . . . .
. . . . .
. . . 0+ . . . . .
. . . 4+ . . . . .
. 1+ . . . . .
. . . 2+ . . . . .
. . . 3+ . . . . .
time 6
. . . . .
. . . . .
. . . . .
. . . 0+ . . . . .
. . . 4+ . . . . .
. 1+ 3+ . . . . .
. . . 2+ . . . . .
time 7
. . . . .
. . . . .
. . . . .
. . . . .
. . . 3+ . 0+ . . . . .
. . . 4+ . . . . .
. 1+ . . . . 2+ . . . . .
time 8
. . . . .
. . . . .
. . . . .
. . . 3+ . . . . .
. . . . .
. . . 0+ . . . . .
. 1+ . 4+ . . . 2+ . . . . .
time 9
. . . . .
. . . . .
. . . . .
. . . 3+ . . . . .
. . . . .
. 1+ . . . . .
. . . 4+ . . . 2+ . . . . .
infection path:
3 -> 2 -> 4 -> 0

```

## プログラム

```
1 #include <stdio.h>
2 #define True 1
3 #define False 0
4 #define N 5
5 #define SIZE 10
6 #define PERIOD 10
7
8 typedef struct {
9     int x, y, stepx, stepy, infected, source;
10 } Person;
11
12 Person persons[N];
13 int map[PERIOD][SIZE][SIZE];
14
15 Person initialize(int state[5]) {
16     Person person;
17
18     person.x = state[0];
19     person.y = state[1];
20     person.stepx = state[2];
21     person.stepy = state[3];
22     person.infected = state[4];
23     person.source = -1;
24
25     return person;
26 }
27
28 void print_map(int t) {
29     int i, x, y;
30
31     for (y = 0; y < SIZE; y++) {
32         for (x = 0; x < SIZE; x++) {
33             i = map[t][x][y];
34             if (i >= 0) {
35                 printf("%d", i);
36                 if (persons[i].infected)
37                     printf("+");
38                 else
39                     printf("_");
40             }
41             else
42                 printf("._");
43         }
44         printf("\n");
45     }
46 }
47
```



```

48 int update(int d, int step) {
49     d = d + step;
50     if (d < 0)
51         d = 0;
52     if (d >= SIZE)
53         d = A;
54
55     return d;
56 }
57
58 void infect(int t) {
59     int i, x, y, xx, yy, n;
60
61     for (i = 0; i < N; i++) {
62         if (!persons[i].infected)
63             continue;
64         for (xx = -1; xx <= 1; xx++)
65             for (yy = -1; yy <= 1; yy++) {
66                 x = update(persons[i].x, xx);
67                 y = update(persons[i].y, yy);
68                 n = map[t][x][y];
69                 if (n >= 0 && !persons[n].infected) {
70                     persons[n].infected = True;
71                     persons[n].source = i;
72                 }
73             }
74     }
75 }
76
77 void move(int t) {
78     int i, x, y;
79
80     for (i = 0; i < N; i++) {
81         x = update(persons[i].x, persons[i].stepx);
82         y = update(persons[i].y, persons[i].stepy);
83
84         if (map[t][x][y] >= 0) {
85             x = persons[i].x;
86             y = persons[i].y;
87         }
88         if (persons[i].x == x && persons[i].y == y) {
89             persons[i].stepx = -persons[i].stepx;
90             persons[i].stepy = -persons[i].stepy;
91         }
92         map[t][x][y] = B;
93         persons[i].x = x;
94         persons[i].y = y;
95
96         /* printf("%d: %d(%d,%d) (%+d,%+d)\n",
97            t, i, x, y, persons[i].stepx, persons[i].stepy); */
98     }
99 }
100

```

```

101 void print_path(int i) {
102     if (persons[i].source > 0) {
103         c;
104         printf("_->_%d", i);
105     }
106     else
107         printf("%d", i);
108 }
109
110 int main(void) {
111     int initial_states[N][5] = {
112         {0, 0, +1, +1, False},
113         {1, 2, 0, +1, True},
114         {2, 4, -1, -1, False},
115         {3, 5, 0, +1, True},
116         {4, 3, 0, +1, False}};
117     int i, t, x, y;
118
119     for (t = 0; t < PERIOD; t++)
120         for (x = 0; x < SIZE; x++)
121             for (y = 0; y < SIZE; y++)
122                 map[t][x][y] = -1;
123
124     for (i = 0; i < N; i++) {
125         persons[i] = initialize(initial_states[i]);
126         map[0][persons[i].x][persons[i].y] = i;
127     }
128     printf("time_0\n");
129     print_map(0);
130     for (t = 0; t < PERIOD - 1; t++) {
131         infect(t);
132         move(t + 1);
133         printf("time_%d\n", t + 1);
134         print_map(t + 1);
135     }
136     printf("infection_path:\n");
137     print_path(0);
138     printf("\n");
139
140     return 0;
141 }

```