

平成 27 年度

名古屋大学大学院情報科学研究科
計算機数理科学専攻
入 学 試 験 問 題

専 門

平成 26 年 8 月 7 日 (木)
12 : 30 ~ 15 : 30

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. 外国人留学生は、英語で解答してもよい。さらに、電子辞書以外の辞書 (1冊) を持ち込んでもよい。
4. 問題冊子、解答用紙 3 枚、草稿用紙 3 枚が配布されていることを確認せよ。
5. 問題は、線形代数、微分積分、離散数学、グラフ理論、数理論理学、確率・統計、量子力学、アルゴリズム設計法、オートマトン理論、プログラミングの 10 題からなる。このうち **3 題を選択**して解答せよ。

選択した問題名または問題番号を解答用紙の指定欄に記入せよ。

ただし、数理論理学は選択問題であり、問題はIとIIからなる。数理論理学を選択する場合は、IまたはIIの一方のみを答えよ。

6. 解答用紙の指定欄に受験番号を必ず記入せよ。解答用紙に受験者の氏名を記入してはならない。
7. 解答用紙は試験終了後に 3 枚とも提出せよ。
8. 問題冊子、草稿用紙は試験終了後に持ち帰ってよい。

問題 1. (線形代数)

以下の各問に答えよ.

(1) 次の行列の行列式 (determinant) を求めよ.

$$A = \begin{pmatrix} 1 & 2 & 0 & 0 & 0 \\ 3 & 4 & 0 & 0 & 0 \\ 5 & 6 & 7 & 0 & 0 \\ 8 & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & 17 \end{pmatrix}$$

(2)

$$A = \begin{pmatrix} 1 & -1 & 0 & -1 \\ -2 & 1 & -2 & 2 \\ 0 & 1 & 1 & 1 \\ 2 & 0 & 2 & -1 \end{pmatrix}$$

とする. 以下に答えよ.

- (i) $AB = B$ をみたす階数 (rank) 2 の行列 B を 1 つ求めよ.
- (ii) $AB = B$ をみたす階数 3 の行列 B は存在しないことを示せ.

問題 2. (微分積分)

以下の各問に答えよ.

- (1) 数列 $\{a_n\}_{n=1}^{\infty}$ を

$$a_n = -\log n + \sum_{k=1}^n \frac{1}{k}$$

によって定める.

- (i) すべての n に対して $a_{n+1} \leq a_n$ が成り立つことを示せ.
- (ii) 極限值 $\lim_{n \rightarrow \infty} a_n$ が存在することを示せ.
- (2) c が関数 f の不動点とは, $f(c) = c$ を満たすことである. 関数 f が開区間 I で微分可能で, I 中のすべての点 x において $f'(x) < 1$ となっているとき, f は I において高々 1 つの不動点しか持てないことを示せ.
- (3) \mathbb{R}^3 における 2 つの円柱 (cylinder)

$$x^2 + z^2 \leq a^2, \quad y^2 + z^2 \leq a^2$$

の共通部分の表面積を求めよ.

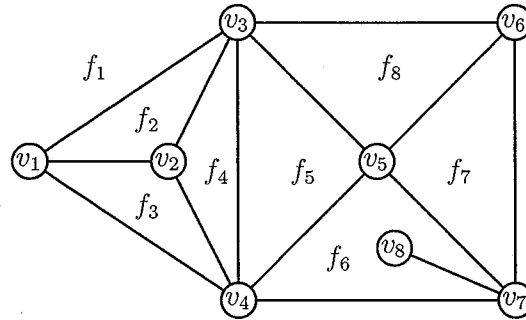
問題 3. (離散数学)

G を群とする. $x, y \in G$ に対して, $[x, y] = x^{-1}y^{-1}xy$ と定め, すべての $[x, y]$ によって生成される G の部分群 (subgroup) を D で表す. 以下の各問に答えよ.

- (1) $[x, y]^{-1} = [y, x]$ を示せ.
- (2) G が可換群 (commutative group) であることと, $D = \{e\}$ は同値 (equivalent) であることを示せ. ただし, e は G の単位元 (identity) である.
- (3) D は G の正規部分群 (normal subgroup) であること, すなわち, 任意の $x \in D$ および任意の $y \in G$ に対して, $y^{-1}xy \in D$ であることを示せ.
- (4) G/D は可換群であることを示せ.
- (5) G の正規部分群 N に対して, G/N が可換群ならば, $D \subset N$ であることを示せ.

問題 4. (グラフ理論)

頂点集合 V , 辺集合 E をもつ無向グラフ $G = (V, E)$ を平面上に辺の交差なく描画できるとき, そのように描画したものを平面グラフ (plane graph) と呼び, 辺によって分割された領域のそれぞれを面 (face) と呼ぶ. 平面グラフの外側の領域も面の1つである. たとえば以下の平面グラフには f_1 から f_8 までの8つの面がある.



2つの面が境界の辺を共有するとき隣接する (adjacent) という. たとえば上のグラフの面 f_1 と f_2 は隣接しているが, 面 f_2 と f_5 は隣接していない. 隣接する2面の色が異なるように全ての面を k 色で彩色できるとき, k -面彩色可能であるという. 以下の各問に答えよ.

- (1) 上のグラフにおいて面 f_8 と隣接している面を全て挙げよ.
- (2) 出来るだけ小さい色数 k で上のグラフの k -面彩色を与えよ.
- (3) 平面グラフの各辺と高々1回交差する閉曲線でどの頂点とも交わらないものを閉路と呼び, 閉路 C の内部に含まれる頂点集合を $S(C)$ と記す. また, 頂点 v の次数 (degree) を $d(v)$ で表す. 任意の閉路 C に対し, $\sum_{v \in S(C)} d(v)$ の偶奇と, 閉路 C と交差する辺数 $|\{(u, v) \in E \mid u \in S(C), v \notin S(C)\}|$ の偶奇が一致することを示せ.
- (4) どの辺も相異なる2面に接している (つまり図中の辺 (v_7, v_8) のように辺の両側が同じ面であるような辺がない) 平面グラフ G に関して, 以下の2つの性質を証明せよ.

性質 1. G が2面彩色可能であれば, すべての頂点 v において $d(v)$ は偶数である.

性質 2. すべての頂点 v において $d(v)$ が偶数であれば, G は2面彩色可能である.

問題 5. (数理論理学)

数理論理学は選択問題である. 次の I, II の いずれか一つを選択して 答えよ. 解答用紙の指定欄に, どちらの問題を選択したのかはつきりわかるように記入せよ.

I. 以下の各問に答えよ.

- (1) 実数全体の集合 \mathbb{R} を定義域にもつ実数値 (real-valued) 単調増加関数 (monotonically increasing function) は不連続点を高々可算個しかもたないことを示せ.
- (2) \mathbb{R} を定義域とする実数値単調増加関数の全体はちょうど連続濃度をもつことを示せ.

II. 一階言語 (language of first order) $\mathcal{L} = \{\times\}$ を考える. \mathcal{L} の原子論理式 (atomic formula) で用いてよい記号は, 関数記号 \times , 等号 $=$ と変数記号のみである. 実数全体の集合 \mathbb{R} を領域 (domain) とし, \times をふつうの乗法として解釈した \mathcal{L} 構造を \mathcal{R} で表す. このとき, 以下の各問に答えよ.

- (1) \mathcal{R} での解釈が『 $x = 0$ である』と同値になる \mathcal{L} の論理式 $\text{zero}(x)$ を与えよ.
- (2) \mathcal{R} での解釈が『 $x = 1$ である』と同値になる \mathcal{L} の論理式 $\text{one}(x)$ を与えよ.
- (3) \mathcal{R} での解釈が『 $x > 0$ である』と同値になる \mathcal{L} の論理式 $\text{positive}(x)$ を与えよ.
- (4) \mathcal{R} での解釈が『 $x > 1$ である』と同値になる \mathcal{L} の論理式は存在しないことを示せ.

問題 6. (確率・統計)

1次元確率変数 X, Y, Z は独立に同一の確率分布 (probability distribution) にしたがう。その分布関数 (distribution function) を $F(x)$, $x \in \mathbb{R}$ とする。以下の各問に答えよ。

- (1) $X < Y$ となる確率 $P(X < Y)$ は $1/2$ 以下であることを示せ。
- (2) $X < Y < Z$ となる確率 $P(X < Y < Z)$ は $1/6$ 以下であることを示せ。
- (3) $a < b$ を満たす定数 a, b に対して、

$$\min\{X, Y, Z\} \leq a < \max\{X, Y, Z\} \leq b$$

となる確率を求めよ。

- (4) (3) で求めた確率を最大にする分布関数 $F(x)$ に対して、 $F(a)$ と $F(b)$ の値を求めよ。

問題 7. (量子力学)

I, X, Y, Z を

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

で定義される 2×2 行列とする. 実数 w, x, y, z に対して, S, W を

$$S = xX + yY + zZ, \quad W = wI + S$$

で定まる 2×2 行列とする. 以下の各問に答えよ.

- (1) S および W は, ともにエルミート行列であることを示せ.
- (2) S および W の重複度を込めた固有値の和は, それぞれ 0 および $2w$ であることを示せ.
- (3) $S^2 = (x^2 + y^2 + z^2)I$ となることを示せ.
- (4) W の固有値を w, x, y, z で表せ.
- (5) W が密度行列 (トレースが 1 の半正定値行列) であるための w, x, y, z の条件を求めよ.
- (6) W が射影行列 (冪等なエルミート行列) であるための w, x, y, z の条件を求めよ.

(注) エルミート行列: Hermitian matrix; 重複度: multiplicity; 固有値: eigenvalue; 密度行列: density matrix; 半正定値行列: positive-semidefinite matrix; 射影行列: projection matrix; 冪等: idempotent

問題 8. (アルゴリズム設計法)

以下の各問に答えよ.

(1) 1次元配列 $A[1..n]$ に n 個の整数が与えられているとする.

(i) 配列 A の **プリフィックスサム** (prefix sum) とは配列 $B[1..n]$ で, $B[i] = \sum_{j=1}^i A[j]$ ($1 \leq i \leq n$) であるものをいう. 下に例を示す.

$$A \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline & 1 & & & & & & & & & & n \\ \hline 5 & -7 & 4 & 6 & -11 & 2 & 6 & -7 & 10 & -1 & -7 & 3 \\ \hline \end{array}$$

$$B \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline & 1 & & & & & & & & & & n \\ \hline 5 & -2 & 2 & 8 & -3 & -1 & 5 & -2 & 8 & 7 & 0 & 3 \\ \hline \end{array}$$

配列 A が与えられたときにそのプリフィックスサムを $O(n)$ 時間で計算するアルゴリズムを書け.

(ii) 配列 A の **最大部分配列** (maximum sub-array) とは A の連続した部分でその要素の和が最大になるものをいう. 上の配列 A では $A[6..9]$ が最大部分配列である.

$$A \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|} \hline & 1 & & & & & & & & & & n \\ \hline 5 & -7 & 4 & 6 & -11 & 2 & 6 & -7 & 10 & -1 & -7 & 3 \\ \hline \end{array}$$

1次元配列 A が与えられたとき, A の最大部分配列を見つける $O(n)$ 時間のアルゴリズムを書け.

(2) A は m 行 n 列の2次元配列でその要素は整数とする. 2次元配列 A の最大部分配列とは, A の連続した行と連続した列からなる2次元配列で, その要素の和が最大になるもののことである. A の **行固定の最大部分配列** とは, A のすべての行と連続した列からなる2次元配列で, その要素の和が最大のものを用いる.

[例] 下の2次元配列の最大部分配列は網掛け部分 (要素の和は18) である. 行固定の最大部分配列は□で囲まれた部分 (要素の和は9) である.

$$\begin{pmatrix} -2 & 2 & -3 & 5 & -12 & -3 & 3 & -3 \\ 2 & -5 & -6 & -8 & 2 & -4 & 4 & -2 \\ 3 & -2 & 9 & -9 & -1 & 13 & -5 & -4 \\ 1 & -3 & 5 & -7 & 8 & -2 & 2 & -6 \\ 2 & -5 & -6 & -8 & -2 & -3 & 4 & 2 \end{pmatrix}$$

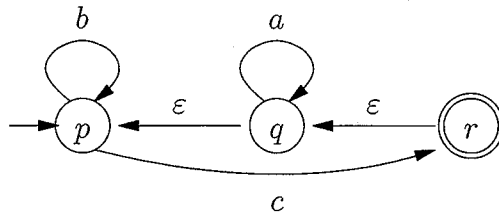
(i) m 行 n 列の2次元配列 A が与えられたとき, 行固定の最大部分配列を見つける $O(mn)$ 時間のアルゴリズムを書け.

(ii) m 行 n 列の2次元配列 A が与えられたとき, $O(m^2n)$ 時間で A の最大部分配列を見つけるアルゴリズムを書け.

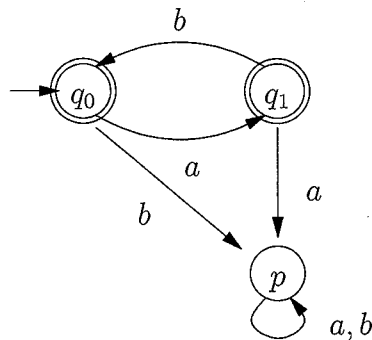
問題 9. (オートマトン理論)

以下の各問に答えよ.

- (1) 正規表現 (regular expression) $(a + b)^*a(a + b)a$ と等価な決定性有限オートマトン (deterministic finite automaton) を図示せよ.
- (2) 次の有限オートマトン M_1 に対して、各問に答えよ.



- (i) 各状態 (state) に対して ϵ 閉包 (ϵ -closure) を求めよ.
 - (ii) M_1 と等価で ϵ 遷移 (ϵ -transition) を持たない有限オートマトンを図示せよ.
- (3) $\Sigma = \{a, b\}$ をアルファベットとし、次の決定性有限オートマトン M_2 を考える.



$\alpha, \gamma \in \Sigma^*$ に対して $\alpha\beta = \gamma$ を満たす $\beta \in \Sigma^*$ が存在するとき、 $\alpha \preceq \gamma$ と書く. $\alpha \in \Sigma^*$ に出現する a の数から b の数を引いた値を $\#(\alpha)$ で表す.

M_2 の遷移関数 (transition function) を δ で表すとき、 $i = 0$ と $i = 1$ のいずれについても、任意の $\alpha \in \Sigma^*$ に対して以下の (A) と (B) が同値であることを α の長さに関する同時帰納法 (simultaneous induction) で証明せよ.

(A) $\delta(q_0, \alpha) = q_i$

(B) $\#(\alpha) = i$, かつ, $\forall \beta \in \Sigma^* (\beta \preceq \alpha \Rightarrow 0 \leq \#(\beta) \leq 1)$

問題 10. (プログラミング)

ハッシュを用いて文字列を検索するプログラムを実装したい。プログラムPはこれを実現するためのC言語プログラムである。構造体listは1つの文字列に対応する情報を格納し、hash_tableは検索対象の文字列を記憶する配列である。このプログラムについて以下の問いに答えよ。なお、プログラム中の演算子、関数、アスキーコードの説明を問いの後に示す。

- (1) 空欄ア～エを埋めよ。
- (2) 標準出力に出力されるこのプログラムの実行結果を書け。
- (3) 66行目のinit_hash_table()を実行した直後のhash_table[HASHSIZE] (14行目)が保持する文字列wordの内容を下の形式で書け。nextにNULL以外の値が入る場合、該当するwordを「→」を使って連結せよ。

例:

添え字	word
0	red
1	NULL
2	black → blue → green
...	...

- (4) 7行目のN_WORDSの値を4から5に変更し、15行目をchar colors[N_WORDS][MAX_LEN] = {"red", "blue", "green", "yellow", "pink"};とする。66行目のinit_hash_table()を実行した直後のhash_table[HASHSIZE] (14行目)の内容を問い(3)の形式で書け。
- (5) 以下の空欄を埋める形式で45行目のmy_strcpyの機能を満たすよう回答せよ(ただし他の関数を呼び出さないこと)。

```
void my_strcpy(char* a, const char* b)
{
    int i = 0;
    while(  ) {
        ;
        i++;
    }
    ;
}
```

次頁に続く

演算子, 関数, アスキーコードの説明

24 行目の演算子「%」は剰余演算子^{じょうよさんざんし}であり「x % y」は x を y で割った余りである。

28 行目の関数 `find_word(char * key)` は文字列 `key` が `hash_table` に存在するとき 1, 存在しないとき 0 を返す。

32 行目の関数 `strcmp(const char* a, const char* b)` は文字列 `a`, `b` が一致するとき 0 を, 一致しないときは 0 以外を返す関数である。

45 行目の関数 `my_strcpy(char* a, const char* b)` は文字列 `a` に文字列 `b` をコピーする関数である。

`char` 配列の各文字はアスキーコードとして保持される。アルファベットに対応する 10 進数の値は以下の表のとおりである。

アルファベット	a	b	c	d	e	f	g	h	i	j	k
アスキーコード	97	98	99	100	101	102	103	104	105	106	107
アルファベット	l	m	n	o	p	q	r	s	t	u	v
アスキーコード	108	109	110	111	112	113	114	115	116	117	118
アルファベット	w	x	y	z							
アスキーコード	119	120	121	122							

Translation of technical terms

ハッシュ	hash	演算子	operator
文字列	string	関数	function
プログラム	program	アスキーコード	ASCII code
C 言語	C language	剰余演算子	modulus operator
構造体	structure	メモリ	memory
配列	array	ポインタ	pointer

プログラムP

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4
5 #define HASHSIZE 10
6 #define MAX_LEN 64
7 #define N_WORDS 4
8
9 struct list {                               /* 1文字列用の構造体 */
10     char word[MAX_LEN];                     /* 文字列を記憶しておく場所 */
11     struct list *next;                      /* 次の list へのポインタ */
12 };
13
14 struct list *hash_table[HASHSIZE];          /* ハッシュテーブル */
15 char colors[N_WORDS][MAX_LEN] = {"red", "blue", "green", "yellow"};
16
17 int hash(char *key)
18 {
19     int hashval = 0;
20     while (*key != '\0') {                   /* 与えられた文字列のハッシュ値を求める */
21         hashval += *key;
22         key++;
23     }
24     return (hashval % HASHSIZE);             /* ハッシュ値を返す */
25 }
26
27 /* find_word は key が hash_table に存在するとき 1、存在しないとき 0 を返す */
28 int find_word(char *key)
29 {
30     struct list *p;
31     for (p = hash_table[hash(key)]; p != NULL; ア)
32         if (strcmp(key, p->word) == 0) return 1;
33     return 0;
34 }
```

```

35 void init_hash_table()
36 {
37     int i, hashvalue;
38     struct list *p, *q;
39     for (i = 0; i < [   イ   ]; i++) { hash_table[i] = NULL; }
40     for (i = 0; i < N_WORDS; i++) {
41         if ((find_word(colors[i])) == [   ウ   ]) {
42             /* ポインタ p に新しい文字列用のメモリを割り当てる */
43             p = (struct list *)malloc(sizeof(struct list));
44
45             my_strcpy(p->word, colors[i]); /*割り当てたメモリに文字列をコピーする*/
46             hashvalue = hash(colors[i]);
47
48             if (hash_table[hashvalue] == NULL) {
49                 /* 文字列がなければ新しい単語をそのまま追加 */
50                 hash_table[hashvalue] = p;
51                 [   エ   ];
52             } else { /* 文字列がすでにあれば現在のリストの末尾に新しい単語を追加 */
53                 q = hash_table[hashvalue];
54                 while (q->next != NULL) q = q->next;
55                 q->next = p;
56                 [   エ   ];
57             }
58         }
59     }
60 }
61
62
63 void main(void)
64 {
65     /* 配列 colors の文字列を登録する */
66     init_hash_table();
67     /* "red"が登録した文字列に含まれるか確認する */
68     printf ("result = %d¥n", find_word("red"));
69 }

```