平 成 ２ ６ 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２５年８月７日（水）
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)

n 次実対称行列 A の固有値の最大値を $\lambda_1(A)$，最小値を $\lambda_n(A)$ とおく．列ベクトル $\mathbf{x} \in \mathbb{R}^n$ の転置を $\mathbf{x}^T$ と表す．A, B を n 次実対称行列として，以下の各問に答えよ．ただし，n 次実対称行列は n 次直交行列で対角化可能である事実を，証明なしに用いてよい．

(1) 以下の等式を証明せよ．

$$\lambda_1(A) = \max\{\mathbf{x}^T A\mathbf{x} \mid \mathbf{x} \in \mathbb{R}^n,\, \mathbf{x}^T\mathbf{x} = 1\},$$
$$\lambda_n(A) = \min\{\mathbf{x}^T A\mathbf{x} \mid \mathbf{x} \in \mathbb{R}^n,\, \mathbf{x}^T\mathbf{x} = 1\}.$$

(2) 以下の不等式を証明せよ．

$$\lambda_1(A+B) \le \lambda_1(A) + \lambda_1(B),$$
$$\lambda_n(A+B) \ge \lambda_n(A) + \lambda_n(B).$$

(3) $\lambda_n(A)$ が正のとき，$\kappa(A) = \lambda_1(A)/\lambda_n(A)$ と定義する．$\kappa(A) = \kappa(A^{-1})$ を証明せよ．

(4) $\lambda_n(A)$, $\lambda_n(B)$ はともに正とする．(3) で定義した κ について，$\kappa(A+B) \le \kappa(A)+\kappa(B)$ を証明せよ．

問題 2. (微分積分)

以下の各問に答えよ．

(1) $\log 1.1$ の値を小数点以下第 4 位まで求めよ．ただし $\log$ は自然対数を表すものとする．

(2) 以下の小問に答えよ．

(i) $D = \{(x,y,z) \mid x,y,z \ge 0,\ x+y+z \le a\}$ において重積分

$$F(a) = \iiint_D \frac{dxdydz}{(1+x+y+z)^2}$$

を求めよ．

(ii) $E = \{(x,y,z) \mid x,y,z \ge 0,\ 1 \le x+y+z \le 2\}$ において重積分

$$\iiint_E \frac{dxdydz}{(1+x+y+z)^2}$$

を求めよ．

問題 3.（離散数学）

以下の各問 (1)，(2)，(3) すべてに答えよ．

(1) 有限群 G とその部分群 H に対して，G の元 a, b の関係 $\sim$ を

$$a \sim b \iff a^{-1}b \in H$$

により定める．G, H の位数をそれぞれ $|G|, |H|$ とおく．以下の各問に答えよ．

(a) $\sim$ が同値関係であることを示せ．

(b) $\sim$ に関する G の同値類の個数を $|G : H|$ とおく．このとき $|G| = |G : H| \cdot |H|$ が成り立つことを示せ．

(2) G を有限集合 Ω 上の対称群の部分群とする．Ω の要素 α に対して，

$$G_\alpha = \{g \in G \mid g(\alpha) = \alpha\}, \quad G(\alpha) = \{g(\alpha) \mid g \in G\}$$

とおく．以下の各問に答えよ．

(a) G_α が G の部分群であることを示せ．

(b) $|G| = |G(\alpha)| \cdot |G_\alpha|$ が成り立つことを示せ．

(3) n を正の整数，p を素数とし，$q = p^n$ とする．標数 p（characteristic p）の体において，以下の (a)，(b) を示せ．

(a) 方程式 $x^q - x = 0$ は重解をもたない．

(b) 方程式 $x^q - x = 0$ の解全体の集合は体をなす．

ただし，$x = a$ が $f(x) = 0$ の重解であることと，$f(a) = f'(a) = 0$ であることは同値である．また解は代数的閉包（algebraic closure）を一つ固定してその中で考えることとする．

問題 4. (グラフ理論)

以下の各問に答えよ.

(1) 無向グラフ (undirected graph) $G=(V,E)$ $(|V|\geq 2)$ に対し，G のタフネス (toughness) $t(G)$ を次式で定義する.

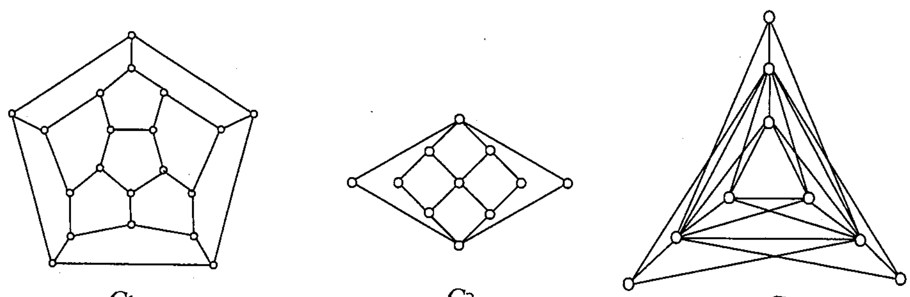
$$t(G)=\min_{S\subset V}\frac{|S|}{c(G-S)}$$

つまり，S を頂点の任意の集合 (ただし，$S\neq\emptyset, V$) とするときの $|S|/c(G-S)$ の最小値のことである．$c(G-S)$ はグラフ G から S の頂点 (vertex) とそこに接続している辺 (edge) を除去してできるグラフの連結成分 (connected component) の個数を表す.

(i) G のすべての頂点をちょうど一度ずつ通る閉路はハミルトン閉路 (Hamilton cycle) と呼ばれる．G がハミルトン閉路を持つなら $t(G)\geq 1$ であることを示せ.

(ii) $G=(V,E)$ が 2 部グラフ (bipartite graph) であるとは，V の 2 分割 ($V=V_1\cup V_2$, $V_1\cap V_2=\emptyset$) が存在し，すべての辺が V_1 の頂点と V_2 の頂点を結ぶようにできるときをいう．$G=(V,E)$ が 2 部グラフなら $t(G)\leq 1$ であることを示せ.

(iii) 次のグラフ G_1,G_2,G_3 はハミルトン閉路を持つか (解答はその根拠を示すこと).



G_1　　G_2　　G_3

(2) 次の定理とその証明を読み，証明の誤りを指摘せよ．なお，δ はグラフの最小次数 (minimum degree) を表す.

[定理] G は頂点数 $n(\geq 3)$ の単純無向グラフ[1]とする．$\delta\geq\frac{n}{2}$ のとき G にはハミルトン閉路が存在する.

（証明）n に関する数学的帰納法で証明する．$n=3$ のとき定理は正しい．なぜなら，各点の次数が 2 なので，G は 3 頂点の完全グラフ（三角形）になるからである．定理が $n=k(\geq 3)$ で成立すると仮定する．G' を k 頂点の単純グラフで $\delta\geq\frac{k}{2}$ とすると，仮定より G' にはハミルトン閉路 C' が存在する．v を新しい頂点とし G' の $(k+1)/2$ 以上の個数の頂点と辺で結んでできる $k+1$ 頂点のグラフを G とする．v は C' 上の連続する 2 点と隣接していることに注意しよう．それらの 2 点を u,w とする．C' の辺 uw を路 (path)uvw で置き換えると G のハミルトン閉路 C を得る．よってこの定理は $n=k+1$ に対して成立する．したがって，帰納法によりこの定理はすべての $n\geq 3$ について成立する.

[1] 自己ループ (self-loop) や並列辺 (parallel edge) がないグラフを単純グラフという.

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，どの問題を選択したのかはっきり分かるように記入せよ．

I. 自然数上の線形算術式 (linear arithmetic expression) を次のように帰納的に定義する．

- 変数および自然数は線形算術式である
- e_1 と e_2 が線形算術式のとき，$(e_1 + e_2)$ と $(e_1 \dot{-} e_2)$ は線形算術式である

ここで，自然数上の引き算 $x \dot{-} y$ は次のように定義されるとする．

$$x \dot{-} y = \begin{cases} x - y & \text{if } x \geq y \\ 0 & \text{if } x < y \end{cases}$$

例えば，関数 f_1 を自然数上の線形算術式 $(1 \dot{-} x)$ を用いて

$$f_1(x) = (1 \dot{-} x)$$

と定義した場合，次の性質を持つ関数となる．

$$f_1(x) = \begin{cases} 0 & \text{if } x \geq 1 \\ 1 & \text{if } x = 0 \end{cases}$$

このとき，以下の問に答えよ．

(1) 自然数上の線形算術式を用いて次のように定義された関数 f_2 はどのような関数か？できるだけ平易に説明せよ．

$$f_2(x, y) = ((x \dot{-} y) + y)$$

(2) 次の性質を持つ関数 f_3, f_4 を自然数上の線形算術式を用いて定義せよ．

(i) $f_3(x) = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$　　(ii) $f_4(x, y) = \begin{cases} 0 & \text{if } x \leq y \\ 1 & \text{if } x > y \end{cases}$

(3) 変数を高々一つ含む (含む場合には x とする) 線形算術式 e を用いて $g_e(x) = e$ のように定義した関数 g_e に対して，次の性質を満たす自然数 k と n_0 が存在することを，線形算術式の構成に関する帰納法で示せ．

$$\forall n \geq n_0,\ g_e(n) < kn$$

なお，帰納法の仮定を用いた箇所は明示すること．

II. N を自然数全体の集合，つまり $\mathrm{N} = \{0, 1, 2, \ldots\}$ とする．各 $n \in \mathrm{N}$ に対して A_n は N の無限部分集合であるとする．このとき，N の新たな部分集合 X で，どの $n \in \mathrm{N}$ についても $A_n \cap X$ と $A_n \setminus X (= \{i \in \mathrm{N} \mid i \in A_n$ かつ $i \notin X\})$ がともに無限集合であるようなものが存在することを示せ．

問題 6. (確率・統計)

確率変数 X, Y は独立に区間 $(0,1)$ 上の一様分布に従うとする．ここで，区間 $(0,1)$ 上の一様分布の確率密度関数 $p(x)$ は

$$p(x) = \begin{cases} 1, & 0 < x < 1, \\ 0, & \text{その他} \end{cases}$$

で与えられる．確率変数 Z, W を

$$Z = \sqrt{-2\log X}\cos(2\pi Y), \quad W = \sqrt{-2\log X}\sin(2\pi Y)$$

で定める．これは (X, Y) から (Z, W) への 1 対 1 変換である．以下の各問に答えよ．

(1) X を Z, W を用いて表せ．

(2) Z の分散を求めよ．

(3) 2 次元確率変数 (Z, W) の確率密度関数 $h(z, w)$ 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 行列とし，実数 t に対して，$X(t)$ を

$$X(t) = e^{itZ} X e^{-itZ}$$

で定義する．ψ を任意の単位ベクトルとして，以下の問に答えよ．

(1) $U(t) = e^{itZ}$ を I, X, Y, Z の線形結合で表わせ．

(2) $f(t) = \|[U(t), X]\psi\|$ の値を求めよ．ただし，$[\cdot, \cdot]$ は交換子（$[A, B] = AB - BA$）を表わす．

(3) $X(t)$ を I, X, Y, Z の線形結合で表わせ．

(4) $g(t) = \frac{1}{\sqrt{2}}\|[X(t), Y]\psi\|$ の値を求めよ．

(5) $0 \le t \le \frac{\pi}{4}$ ならば $f(t)g(t) < 1$ が成り立つことを示せ．

(6) 任意の t について $f(t) + g(t) \ge 1$ が成り立つことを示せ．

問題 8. (アルゴリズム設計法)

鉄道，バス，飛行機等の公共交通において目的地までの切符を購入するよりも中間地点をいくつか経由して切符を分けて購入する方が料金（円）が安くなることがある．以下では直線状の路線で n 地点よりなるものを考え，出発地を 1，目的地を n として，番号の若い地点への後戻りを含まない経路のみを考察の対象とする．たとえば $1 \to 3 \to 4 \to 5$ はよいが $1 \to 4 \to 2 \to 5$ は許されない．$1 \leq i < j \leq n$ を満たす各対 (i, j) に対して地点 i から j への切符の料金 $c(i, j)$ が与えられたとき，出発地 1 から目的地 n までの後戻りを含まない経路で料金が最小のものを求めたい．以下の表は $n = 5$ の場合のある料金表であり，たとえば地点 1 から 2 への切符代 $c(1, 2)$ は 100 円，地点 1 から 3 への切符代 $c(1, 3)$ は 210 円であることを表している．

	2	3	4	5
1	100	210	340	460
2		130	220	350
3			120	230
4				110

この場合，出発地 1 から目的地 5 まで直行する切符を購入すると $c(1, 5) = 460$ 円かかるが，たとえば経路 $1 \to 4 \to 5$ のように出発地 1 から地点 4 までの切符と地点 4 から目的地 5 への切符に分けて購入すると，料金は $c(1, 4) + c(4, 5) = 340 + 110 = 450$ 円と若干安くなる．以下の各問に答えよ．

(1) 可能な経路を全て列挙してそのおのおのの料金を調べるアルゴリズムを考える．このアルゴリズムの計算量を示せ．

(2) 出発地 1 から地点 j ($j = 2, 3, \ldots, n$) までの（後戻りを含まない）経路の料金の最小値を $f(j)$ とする．また，便宜上 $f(1) = 0$ と定義する．$f(j)$ に対する動的計画法 (dynamic programming) の漸化式を示せ．

(3) 動的計画法に基づいて出発地 1 から目的地 n までの経路の料金の最小値を求めるアルゴリズムを示せ．また，その計算量を評価せよ．漸化式の $f(1), \ldots, f(n)$ の値を格納する配列を `F` とし，最終的に全ての j に対して $f(j)$ の値が `F[j]` に格納されている状態にするために必要な計算を簡潔に示すこと．再帰呼び出しを用いてはならない．

(4) 計算済みの配列 `F` を用いて料金最小の経路を得る方法を簡潔に説明せよ．

(5) 上の表の料金に対して動的計画法を適用したときの配列 `F` を求めよ．

問題 9. (オートマトン理論)

アルファベット $\Sigma = \{0, 1\}$ について，以下の各問に答えよ.

(1) 言語 $L_1 = \{w \mid |w| \geq 2, w$ の最後から 2 番目が $0\}$ を次のそれぞれの形式で表せ.

(i) 正規表現 (regular expression)

(ii) 非決定性有限オートマトン (non-deterministic finite automaton)(図示せよ)

(iii) 決定性有限オートマトン (deterministic finite automaton)(図示せよ)

(iv) 正規文法 (regular grammar)(もしくは文脈自由文法 (context free grammar))

(2) 自然数 n の 2 進数表現を $b(n)$ で表す．すなわち，$b(0) = 0$，$b(1) = 1$，$b(2) = 10$，... である．アルファベット $\Gamma = \left\{\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}\right\}$ とし，Σ 上の文字列 w_1, w_2 の Γ 上の文字列へのコーディング $B(w_1, w_2)$ を次のように与える.

$$
\begin{aligned}
&B(\varepsilon, \varepsilon) = \varepsilon \\
&B(wa, \varepsilon) = \begin{pmatrix} a \\ 0 \end{pmatrix} \cdot B(w, \varepsilon) && (w \in \Sigma^*, a \in \Sigma) \\
&B(\varepsilon, wa) = \begin{pmatrix} 0 \\ a \end{pmatrix} \cdot B(\varepsilon, w) && (w \in \Sigma^*, a \in \Sigma) \\
&B(w_1 a_1, w_2 a_2) = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \cdot B(w_1, w_2) && (w_1, w_2 \in \Sigma^*, a_1, a_2 \in \Sigma)
\end{aligned}
$$

このとき以下の問に答えよ.

(i) $B(110, 1100)$ を示せ.

(ii) B はどのような関数か？　できるだけ平易に説明せよ.

(iii) $L_2 = \{\varepsilon\} \cup \{B(b(n), b(2n)) \mid n$ は自然数$\}$ を認識する有限オートマトンを図示せよ.

問題10.（プログラミング）

与えられた正整数 (positive integer) N に対し，１以上 N 以下の正整数を要素 (element)とし，それらが０個以上重複なくつながれた双方向リスト (doubly linked list) を考える．この双方向リストを，名前を list，要素数 (array size) を N+2 とする構造体の配列 (array of structures) により表現する．双方向リスト上の要素 i （1≦i≦N）を配列の要素 list[i]に対応させる．また，list[0]は双方向リストの先頭要素を指すために，list[N+1]は末尾要素を指すために使用される．これに対応して HEAD=0，TAIL=N+1 と定義する．構造体のメンバ (member) は next と prev とし，next は双方向リスト上の次の要素または TAIL，prev は双方向リスト上の前の要素または HEAD を保持し，以下の条件１を満足する．

条件１：メンバ next を使うことにより，HEAD から順方向 (forward direction) に双方向リスト上のすべての要素をたどることができ，最終的に TAIL に到達する．また，メンバ prev を使うことにより，TAIL から逆方向 (backward direction) に双方向リスト上のすべての要素をたどることができ，最終的に HEAD に到達する．

表１は，N=8 として要素 1，5，3 がこの順でつながれた双方向リストを表現したときの構造体配列 list の各要素のメンバ next，prev の値である．要素 2，4，6，7，8 は未使用 (unused) であり，表中の「－」は任意の値である．また，プログラムリスト１は N=8 として双方向リストの操作を C 言語で記述したプログラムである．表２に main 関数を除く各関数(function)の機能 (functionality) を示す．

表１．N=8 として要素 1，5，3 がこの順でつながれた双方向リストを表現したときの構造体配列 list の各要素のメンバ next, prev の値

要素	0(=HEAD)	1	2	3	4	5	6	7	8	9(=TAIL)
next	1	5	－	9	－	3	－	－	－	－
prev	－	0	－	5	－	1	－	－	－	3

表２．プログラムリスト１における関数の機能

関数名	関数の機能
init_list()	初期化(initialization)
insert_elem(elem)	要素 elem を双方向リストの先頭要素となるように挿入 (insertion)
delete_elem(elem)	要素 elem を双方向リストから削除 (deletion)
print_list()	双方向リスト上の要素の出力 (output)

これに対して以下の問いに答えよ.

(1) プログラムリスト1の関数 delete_elem(elem)は, 双方向リストにおける要素 elem の次（要素または TAIL）と前（要素または HEAD）を接続することにより実現できる. プログラムリスト1の(a)の部分に C 言語の代入文 (assignment statement)を 5 文以内で追加し, プログラムを完成させよ. なお, ここでは引数 (argument) elem は双方向リスト上にある要素であることを仮定する.

(2) プログラムリスト1の 45 行目終了時の list[8].next と list[8].prev, 48 行目終了時の list[HEAD].next の値をそれぞれ書け.

(3) プログラムリスト1の 46 行目, 51 行目の print_list() 関数によって出力される文字列をそれぞれ書け.

プログラムリスト1の関数 main()内にある各関数の実行終了後において, 構造体配列 list は上記条件1を満足する. 以下の問いでは, 関数の実行終了後において上記条件1を満足しない場合を不具合 (*bug*) とよぶ.

(4) プログラムリスト1では, 双方向リスト上に存在する要素 e に対して関数 insert_elem(e) を実行すると不具合を生じる場合がある. このような不具合が生じる場合について, 具体的な例を用いて一つ説明せよ.

(5) プログラムリスト1では, 双方向リスト上に存在しない要素 f に対して関数 delete_elem(f) を実行すると不具合を生じる場合がある. このような不具合が生じる場合について, 具体的な例を用いて一つ説明せよ.

(6) 上記(4)(5)の不具合を防ぐため, 双方向リスト上に存在する要素 e に対する関数 insert_elem(e)の実行, ならびに双方向リスト上に存在しない要素 f に対する関数 delete_elem(f)の実行に対して, 双方向リストを変更することなく関数を終了するようにしたい. そのためのプログラムリスト1に対する変更方針を以下の4項目に分け, それぞれ簡潔に説明せよ. 変更する必要がない項目には「変更なし」と記載せよ.
   (a) データ構造 (data structure)や格納値の意味 (meanings of stored values) の変更,
   (b) 関数 insert_elem(elem)内の変更,
   (c) 関数 delete_elem(elem)内の変更,
   (d) その他の関数の変更.

(7) プログラムリスト1のメンバ prev を使用せずに実現した単方向リスト(singly linked list) と比較したときの双方向リストのメリットとデメリットをそれぞれ簡潔に説明せよ. ただし, 日本語で解答するならばそれぞれ 40 文字以内、英語で解答するならばそれぞれ 30 words 以内で記述せよ.

プログラムリスト１（行頭の数字は行番号を表す）

```
1: #include <stdio.h>

2: #define N 8
3: #define HEAD 0
4: #define TAIL ((N)+1)

5: typedef struct _list {
6:     int prev;
7:     int next;
8: } DLLIST;
9: DLLIST list[N+2];

10: void init_list(void) {
11:     list[HEAD].next = TAIL;
12:     list[TAIL].prev = HEAD;
13: }

14: void insert_elem(int elem) {
15:     int next;
16:     if((elem <= HEAD) || (elem >= TAIL)) return;
17:     next = list[HEAD].next;
18:     list[elem].next = next;
19:     list[next].prev = elem;
20:     list[HEAD].next = elem;
21:     list[elem].prev = HEAD;
22: }

23: void delete_elem(int elem) {
24:     int next, prev;
25:     if((elem <= HEAD) || (elem >= TAIL)) return;
                        (a)
26: }

27: void print_list(void) {
28:     int i;
29:     i = list[HEAD].next;
30:     if(i == TAIL) {
31:             printf("EMPTY¥n");
32:     } else {
33:             printf("%d", i);
                                                    /* 次頁に続く */
34:             for(i=list[i].next; i!=TAIL; i=list[i].next) {
35:                     printf("->%d", i);
```

```
        }
        printf("¥n");
    }
}

int main(void) {
    init_list();

    insert_elem(3); print_list();
    insert_elem(5); print_list();
    insert_elem(8); print_list();
    delete_elem(5); print_list();
    insert_elem(7); print_list();
    delete_elem(3); print_list();
    delete_elem(7); print_list();
    insert_elem(1); print_list();
    delete_elem(1); print_list();
    delete_elem(8); print_list();

    return 0;
}
```