平 成 ２２年 度

名古屋大学大学院情報科学研究科
情報システム学専攻
入 学 試 験 問 題

専　　　　　門

平成２１年８月１０日（月）
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、日本語から母国語への辞書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. 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(１)解析・線形代数、(２)確率・統計、(３)プログラミング、(４)計算機理論、(５)ハードウェア、(６)ソフトウェアの６科目がある。このうち３科目を選択して解答せよ。なお、選択した科目名を解答用紙の指定欄に記入せよ。

6. ITスペシャリストコースへの配属を希望する者は、必ず、「プログラミング」を選択して解答すること。

7. 解答用紙は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。

8. 解答用紙は試験終了後に３枚とも提出せよ。

9. 問題冊子、草稿用紙は試験終了後に持ち帰ってよい。

解析・線形代数

（解の導出過程も書くこと）

［1］次の行列 $\boldsymbol{X}$ について，以下の問いに答えよ．

$$\boldsymbol{X} = \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix}$$

(a) 行列 $\boldsymbol{X}$ の全ての固有値と，各固有値に対応する固有ベクトルを求めよ．ただし，固有ベクトルは大きさが 1 のものを書け．

(b) 行列 $\boldsymbol{X}$ は，ある正則行列 $\boldsymbol{P}$ によって，$\boldsymbol{D} = \boldsymbol{P}^{-1}\boldsymbol{X}\boldsymbol{P}$ と対角化される．このような行列 $\boldsymbol{D}$，$\boldsymbol{P}$，$\boldsymbol{P}^{-1}$ を求めよ．

(c) 任意の自然数 n に関する行列 $\boldsymbol{X}^n$ を求めよ．

［2］次の不等式を満たす複素数 z の範囲を複素平面上に図示せよ．

$$\mathrm{Re}\left(\frac{1}{z}\right) \geq \frac{1}{2}$$

［3］次の条件を満たす実平面上の点 (x, y) について，以下の問いに答えよ．

$$\begin{cases} x = \cos^3\theta \\ y = \sin^3\theta \end{cases} \quad (0 \leq \theta \leq 2\pi)$$

(a) 点 (x, y) の軌跡を図示せよ．

(b) (a) で求めた軌跡の長さを求めよ．

(c) (a) で求めた軌跡に囲まれる領域の面積を求めよ．

Translation of technical terms

行列	matrix
固有値	eigenvalue
固有ベクトル	eigenvector
大きさ	length
正則 行列	regular matrix
対角化	diagonalization
自然数	natural number
不等式	inequality
複素数	complex number
範囲	range
複素平面	complex plane
実平面	real plane
点	point
軌跡	locus
長さ	length
領域	region
面積	area

確率・統計

（解の導出過程も記すこと．数値での解答を求められている問には全て既約分数で答えること）

[1]　2 種のコイン C_1, C_2 があり，それらを投げて表が出る確率は，それぞれ 1/2, 2/3 であるとする．このとき，以下の問に答えよ．

(1) コインを 4 回投げて表が 3 回出る確率を，コインが C_1 の場合と C_2 の場合のそれぞれについて求めよ．

(2) コイン C_1, C_2 をそれぞれ 5/8, 3/8 の割合で含む箱から無作為にコインを 1 枚取り出す．取り出したコインを 4 回投げたところ，表が 3 回出たとする．この結果から，取り出したコインが C_1, C_2 である確率をそれぞれ求めよ．

[2]　確率変数 X の確率密度関数 $f(x)$ が次式で与えられているとする．

$$f(x) = \begin{cases} 3x & (0 \le x \le 2/3) \\ 6(1-x) & (2/3 < x \le 1) \\ 0 & (\text{その他}) \end{cases}$$

このとき，以下の問に答えよ．

(1) 確率密度関数 $f(x)$ の分布に従う標本が 2 個得られたとき，それらの値が 2 個とも 1/3 以下となる確率を求めよ．

(2) 確率密度関数 $f(x)$ の分布に従う標本が 2 個得られたとき，それらの値の合計が 1/3 以下となる確率を求めよ．

(3) 確率変数 X の平均 μ と分散 σ^2 を求めよ．

(4) 互いに独立な n 個の確率変数 $X_1, X_2, \cdots, X_n$ がすべて確率密度関数 $f(x)$ の分布に従うとする．ここで，新たな確率変数 Y を

$$Y = \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$$

で定義する．上式で n を大きくしたとき，確率変数 Y は，確率密度関数 $g(y)$ で表されるある分布に近づくことが知られている．この分布の名称を答えよ．

(5) 上記確率密度関数 $g(y)$ を，n，μ，σ を用いて表せ．

【専門用語の英訳】

既約分数 irreducible fraction，確率 probability，無作為 random，確率変数 random variable，密度関数 density function，標本 sample，平均 mean，分散 variance，独立 independent，分布 distribution

プログラミング

最小値 N、最大値 M の n 個 (n ＞ 0)の整数を昇順に整列して格納した一次元配列がある。N≦k≦M を満たす整数 k がこの配列の中に存在するかどうかを探索する C 言語の関数 search を以下のように書いた。この関数は k が存在する場合は 1、そうでない場合は 0 を返す。また、そのテストのための main 関数を以下のように書いた。なお左端の番号は行番号を示すものでプログラムの一部ではない。

```
 1  #include <stdio.h>
 2
 3  int search( (a) n, (b) , (c) k ){
 4     int i,j,p;
 5     i = (d) ;
 6     j = (e) ;
 7     while( i <= j ){
 8        p = (i + j) / 2;
 9        if ( b[p] <= k ){
10           i = p + 1;
11        } else {
12           j = p - 1;
13        }
14     }
15     if( (f) ){
16        return 1;
17     } else {
18        return 0;
19     }
20  }
21
22  int main(){
23     int a[]={ 3,6,7,10,13,15,19 };
24     if( search( 7, a, 14 )){
25        printf( "14 is found in array a[].\n" );
26     } else {
27        printf( "14 is not found in array a[].\n" );
28     }
29     return 0;
30  }
```

このプログラムについて以下の問いに答えよ。

（1）$\boxed{\text{(a)}}$ から $\boxed{\text{(f)}}$ を埋めてプログラムを完成せよ。

（2）この探索法はどのような名前で呼ばれるか。

（3）このプログラムの実行において、行番号 8 は複数回実行される。それぞれ実行した後の i, j, p はどのような値になるか示せ。

（4）行番号 7 の while 文の b, i, j, k に関するループ不変式(loop invariants)を示せ。
（ヒント：最小値 N の左隣に-∞、最大値 M の右隣に+∞が存在するものと考える。）

（5）k が配列要素中の最小値から最大値の間の値であるという前提がない場合にはどのような問題が生じるか。また、それに対処するには $\boxed{\text{(f)}}$ をどのように変更すれば良いかを示せ。

（6）この探索法の計算量のオーダーを示せ。

（7）高速な探索法としてハッシュ探索がある。これはどのようなものか 300 文字以内(or in 100 English words)で説明せよ。

計算機理論

[1] $m_1, m_2, \ldots, m_n$ を互いに素な正整数 (relatively prime positive integers) とし、$a_1, a_2, \ldots, a_n$ を整数とする。整数変数 x に関する連立方程式 (system of equations)

$$x \equiv a_i \pmod{m_i} \qquad (i = 1, 2, \ldots, n) \qquad (\star)$$

を考える。以下の問に答えよ。

(1) $M = \prod_{i=1}^{n} m_i$ とする。$i = 1, 2, \ldots, n$ について

$$\frac{y_i M}{m_i} \equiv 1 \pmod{m_i}$$

を満たす整数 y_i が存在することを証明せよ。

(2) 式 $(\star)$ の解の 1 つを a_i, m_i, M, y_i $(i = 1, 2, \ldots, n)$ を用いて表せ。

[2] 下記の状態遷移表 (state transition table) で示される非決定性有限オートマトン (nondeterministic finite automaton) M について、以下の問に答えよ。

現状態	次状態 入力 0	次状態 入力 1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$
$* \ q_1$	$\emptyset$	$\{q_0, q_2\}$
q_2	$\{q_0\}$	$\{q_0, q_3\}$
$* \ q_3$	$\emptyset$	$\emptyset$

("$\rightarrow$" : 初期状態 (initial state) / "$*$" : 受理状態 (accepting states))

(1) M を決定性有限オートマトン (deterministic finite automaton) に変換し、その状態遷移図 (state transition graph) を示せ。

(2) M が受理する言語を長さ 15 以内の正規表現 (regular expression) で記述せよ。
アルファベット $\{0, 1\}$ に対する正規表現 R は以下の拡張 BNF (extended BNF) で定義されるものとする。

$$R ::= \varepsilon \mid \emptyset \mid 0 \mid 1 \mid R \cdot R \mid R + R \mid (R) \mid R^*$$

ここで、$\cdot$ は連接 (concatenation)、$+$ は和 (set union)、$*$ はクリーネ閉包 (Kleene closure) を表し、演算子の結合の強さは、クリーネ閉包 > 連接 > 和の順とする。ε は空列 (empty string)、$\emptyset$ は空集合 (empty set) を表す。また、省略しても演算順序に影響がない括弧 (parenthesis)、および、連接記号は省略してよい。省略した記号は正規表現の長さに含めない。

[3] 下記の各言語 L_1, L_2, L_3 に対する文脈自由文法 (context free grammar) を記述せよ。
ただし、各問で用いる非終端記号 (nonterminals) は S, T の 2 つのみとし、S を開始記号 (start symbol) とせよ。

(1) $L_1 = \{a^i b^j c^k \mid i = j + k\}$

(2) $L_2 = \{a^i b^j c^k \mid i = j + k, \ k > 0\}$

(3) $L_3 = \{a^i b^j c^k \mid i = j + k, \ i > 0\}$

ハードウェア

［1］ Nビットの符号なし整数の加算器について，以下の問に答えよ．なお，iビット目の被加数を a_i，加数を b_i，和を s_i，キャリーインを c_i，キャリーアウトを c_{i+1} とする($0 \leq i \leq N-1$).

(1) 図 1 に示すように，1 ビットの全加算器 FA を 3 つ直列に接続することにより，3 ビットの順次桁上げ加算器 RCA3 を構成する．FA の 2 つの出力 s_i，c_{i+1} について，最小積和形の論理式を a_i，b_i，c_i を用いてそれぞれ書け．

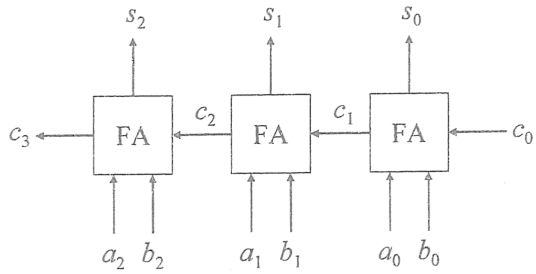


図 1. 3 ビットの順次桁上げ加算器 RCA3

(2) 問(1)の順次桁上げ加算器 RCA3 において，FA を積和形の 2 段論理回路で実現したとき，RCA3 の最大遅延を求めよ．ここで，AND（論理積）ゲートおよび OR（論理和）ゲートの遅延を T とし，NOT（否定）ゲートおよび配線の遅延は 0 とする．

(3) 図 2 に示すように，3 ビットの桁上げ先見加算器 CLA3 を構成する．FA2 は s_i を計算する回路であり，CL は桁上げ先見を行う回路である．CL の 3 つの出力 c_1，c_2，c_3 について，積和形の論理式を g_0，g_1，g_2，p_0，p_1，p_2，c_0 を用いてそれぞれ書け．

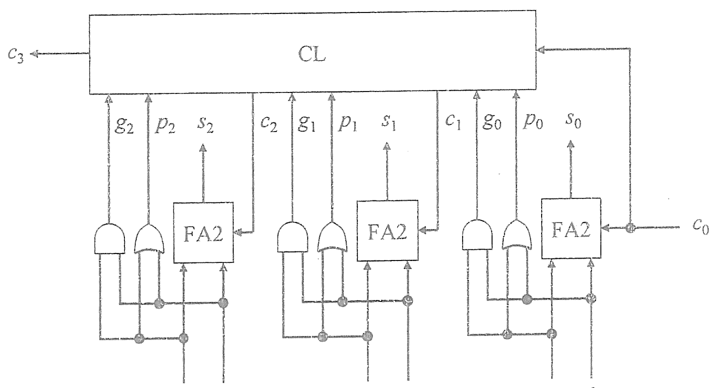


図 2. 3 ビットの桁上げ先見加算器 CLA3

(4) 問(3)の桁上げ先見加算器 CLA3 において，FA2 と CL を積和形の 2 段論理回路で実現したとき，CLA3 の最大遅延を求めよ．ただし，各ゲートや配線の遅延は問(2)と同様とする．

(5) 問(3)の桁上げ先見加算器 CLA3 を基に，図 3 に示すように，9 ビットの桁上げ先見加算器 CLA9 を構成する．FA2 と CL は，問(3)のものと同じ回路である．CL2 は，CL から最上位ビットのキャリーアウトを計算する回路を省いた回路である．G_0 と P_0 の積和形の論理式を g_0，g_1，g_2，p_0，p_1，p_2 を用いてそれぞれ書け．

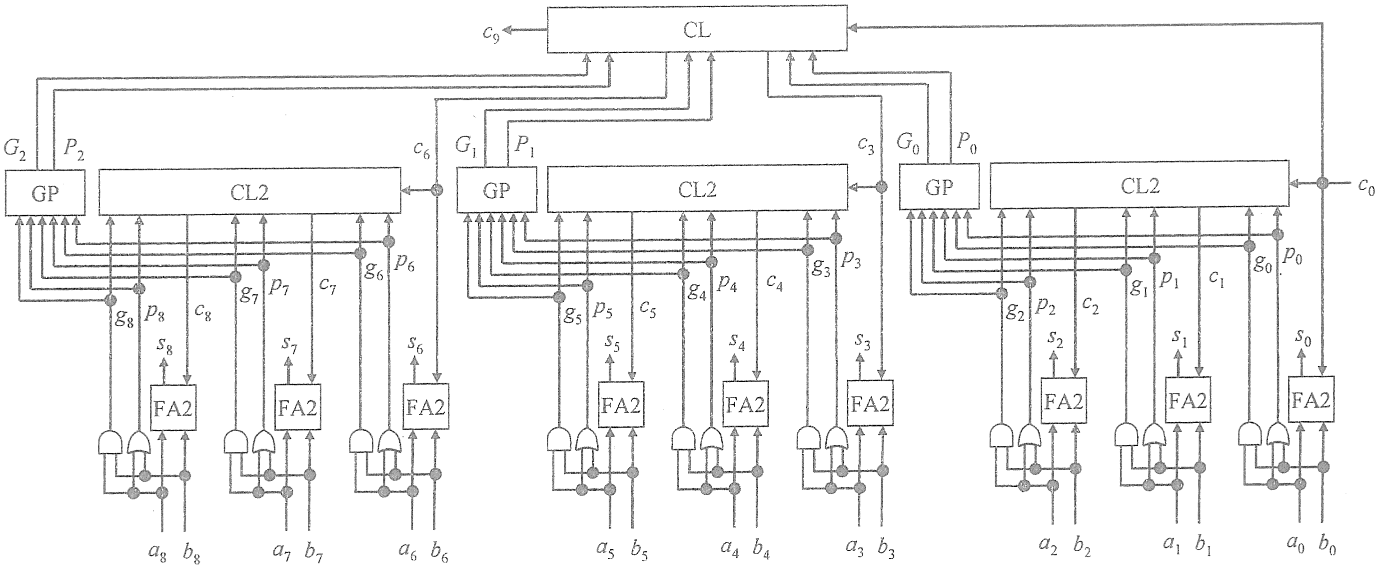


図 3.　9 ビットの桁上げ先見加算器 CLA9

(6) 問(5)の桁上げ先見加算器 CLA9 において，FA2，CL，CL2，GP を積和形の 2 段論理回路でそれぞれ実現したとき，CLA9 の最大遅延を求めよ．ただし，各ゲートや配線の遅延は問(2)と同様とする．

［2］　命令パイプライン処理を行うプロセッサについて，以下の問に答えよ．

(1) 構造ハザード，データ・ハザード，制御ハザード（分岐ハザードともいう）とは何か，各 50 文字程度で説明せよ．

(2) 制御ハザードを解消または軽減する手法を 2 つ，各 100 文字程度で説明せよ．

English translations of technical terms

加算器	:	adder
キャリーイン	:	carry-in
キャリーアウト	:	carry-out
全加算器	:	full adder
順次桁上げ加算器	:	ripple carry adder
最小積和形	:	minimum sum-of-product form
2 段論理回路	:	two-level logic circuit
桁上げ先見加算器	:	carry lookahead adder
命令パイプライン処理	:	instruction pipelining
構造ハザード	:	structural hazard
データ・ハザード	:	data hazard
制御ハザード	:	control hazard
分岐ハザード	:	branch hazard

ソフトウェア

[1] コンパイラ (compiler) に関する以下の語句についてそれぞれ 150 字程度で説明せよ．英語の場合は 50 語程度 (about 50 words in English) で説明せよ．

(1) フローグラフ (flow graph) とフローグラフを用いた最適化 (optimization)

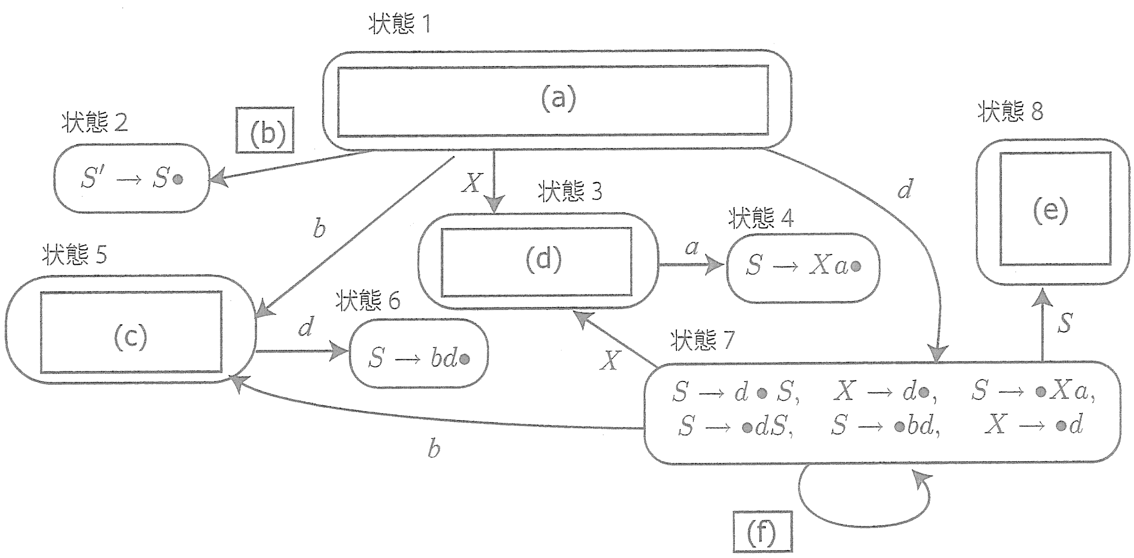
(2) 参照呼び (call by reference)

[2] 文脈自由文法 (context free grammar) G を (N, T, P, S') で定める．ここで，非終端記号集合 (non-terminal symbols) $N = \{S', S, X\}$，終端記号集合 (terminal symbols) $T = \{a, b, d\}$，開始記号 (start symbol) を S' とし，生成規則集合 (production rules)P は以下の 5 つの規則集合からなるとする．(ルールの前のコロンつき数字は規則につけた番号である．)

$$1: \ S' \to S \quad 2: \ S \to Xa \quad 3: \ S \to dS \quad 4: \ S \to bd \quad 5: \ X \to d$$

G に対する SLR(1) 構文解析 (SLR(1) parsing) について以下の問に答えよ．入力の終わりを示す記号 (termination symbol) は \$ を用いること．

(1) 以下に示す G に対する LR(0) オートマトン (LR(0) automaton) の (a) ~ (f) にあてはまる適当な LR(0) 項集合 (LR(0) terms)，終端記号，または，非終端記号を示せ．



(2) N の各要素に対して FOLLOW 集合を求めよ．

(3) SLR(1) 構文解析のための LR-構文解析表 (LR parsing table) を構成せよ．状態番号は，(1) の LR(0) オートマトンの状態番号を用いること．

(4) G が SLR(1) 構文解析可能か否か，理由をつけて答えよ．