平成 29 年 7 月 29 日（土）9:00～12:00

大阪大学大学院情報科学研究科

コンピュータサイエンス専攻
情報システム工学専攻
情報ネットワーク学専攻
マルチメディア工学専攻
バイオ情報工学専攻

平成 30 年度　博士前期課程　入試問題

（A）情報工学

【注意事項】

- 問題数は必須問題 2 題（問題 1～2），選択問題 5 題（問題 3～7），合計 7 題である．
  必須問題は 2 題すべて解答すること．また，選択問題は 2 題を選択して解答すること．
- 問題用紙は表紙を含めて 14 ページである．
- 解答用紙は全部で 5 枚ある．
  1 枚目（赤色）の解答用紙には問題 1（必須問題）の解答を
  2 枚目（青色）の解答用紙には問題 2（必須問題）の(1)の解答を
  3 枚目（緑色）の解答用紙には問題 2（必須問題）の(2)の解答を
  4 枚目（白色）の解答用紙には問題 3～7（選択問題）から選択した 1 題の解答を
  5 枚目（白色）の解答用紙には問題 3～7（選択問題）から選択したもう 1 題の解答を
  それぞれ記入すること．
  解答用紙は間違えると採点されないことがあるので注意すること．
- 解答用紙は 5 枚すべてを回収するので，すべての解答用紙に受験番号を記入すること．
- 解答用紙の「試験科目」の欄には解答した問題の科目名（「アルゴリズムとプログラミング」など）を記入すること．
  また，選択問題調査票には，選択した問題の番号（3～7 から二つ）に○をつけること．
- 解答欄が不足した場合は裏面を使用すること．その際，表面末尾に「裏面に続く」と明記しておくこと．解答用紙の追加は認めない．
- 解答用紙には，日本語または英語で解答すること．

1 【必須問題】アルゴリズムとプログラミング （情報工学 1/13）

配点：(1-1) 15 点，(1-2) 10 点，(1-3) 30 点，(2-1) 20 点，(2-2-1) 30 点，(2-2-2) 20 点

図 1 に示す ANSI-C 準拠である C 言語のプログラム (program) は任意の 2 人が同じ組織 (organization) に所属するか判定し，その結果を出力 (output) するものである．人 (構成員 (member) と呼ぶ) はN人 (Nは正の整数 (positive integer)) 存在し，各構成員には直属の上司 (direct supervisor) である構成員が 1 人以下 (1 or less) 存在する．組織は一つ以上存在し，各組織は，各構成員を節点 (ノード (node))，その直属の上司を親 (parent) とする木 (tree) で表される．同じ組織に所属する構成員は，その組織の最上位の上司を根 (root) とする一つの木を構成する．

各構成員にはそれぞれ 0～N-1 の整数 (integer) が構成員番号として重複なく付与されている．配列 (array) p は構成員番号 i である構成員の直属の上司の構成員番号を要素 (element) として p[i]に格納し，直属の上司が存在しない場合は自身の構成員番号を格納する．このデータ構造を用い，関数 same は 2 人の構成員番号を引数とし，同じ組織に所属するかどうかを標準出力に出力する．input.txt，pair.txt という図 2 および図 3 にそれぞれ示すようなフォーマットのファイルが存在するものとし，図 1 のプログラムでそれらを読み込み実行する．input.txt の 1 行目には構成員の総数N (Nは N_MAX 以下の正の整数)，2 行目以降の各行には，構成員とその直属の上司の構成員番号のペア (pair) がこの順で書かれている．pair.txt の各行には，同じ組織に所属するか判定したい構成員のペアの構成員番号が書かれている．以下の各問に答えよ．

(1) 図 1 のプログラムは，図 2 の input.txt と図 3 の pair.txt を読み込み実行する．以下の各小問に答えよ．

(1-1) 21～29 行目で読み込まれる全ての木を示せ．ただし図 4 にならい，丸でノードを，丸の中の数字で構成員番号を，線で枝 (edge) を表すこと．

(1-2) find(x)が意味する内容を，x を用いて説明せよ．

(1-3) 14 行目の空欄 A に当てはまる式を，15 行目の空欄 B に当てはまる条件式をそれぞれ書け．

(2) 非常に大きな数の構成員に対し，多数の無作為に選ばれた構成員のペアのそれぞれが同じ組織に所属するか判定したい．そのため，このようなデータを含む input.txt および pair.txt を図 1 のプログラムで読み込み実行する．ただし，図 1 のプログラム 2 行目の N_MAX の値を適切に変更するものとする．以下の各小問に答えよ．

(2-1) 関数 same を実行する際の，1 回当たりの平均時間計算量 (average time complexity) をオーダ表記 (order notation) で表せ．また理由も答えよ．ただし，ノードの平均の深さ (average depth) をhとする．

(2-2) ここで，関数 find の 9 行目を変更し，最上位の上司を格納するように配列 p を更新することで，実行時間を短縮できる場合がある．以下に答えよ．

(2-2-1) 変更後の 9 行目を一文で書け．

(2-2-2) 関数 same を十分大きな回数実行した場合に漸近する，1 回当たりの平均時間計算量をオーダ表記で表せ．また理由も答えよ．

```
1  #include <stdio.h>
2  #define N_MAX 1000
3  int p[N_MAX];
4
5  int find(int x) {
6     if (p[x] == x)
7        return x;
8     else
9        return find(p[x]);
10 }
11 void same(int x, int y) {
12    int n, m;
13    n = find(x);
14    m = [ A ];
15    if ( [ B ] )    /*同じ組織に所属する*/
16       printf("%d & %d are in the same organization.\n", x, y);
17    else
18       printf("%d & %d are in different organizations.\n", x, y);
19 }
20 int main(void) {
21     FILE *fp;
22     int n, i, j, sv;
23     fp = fopen("input.txt", "r");
24     fscanf(fp, "%d", &n);
25     for (i = 0; i < n; i++)
26         p[i] = i;
27     while (fscanf(fp, "%d %d", &i, &sv) != EOF)
28         p[i] = sv;
29     fclose(fp);
30     fp = fopen("pair.txt", "r");
31     while (fscanf(fp, "%d %d", &i, &j) != EOF)
32         same(i, j);
33     fclose(fp);
34     return 0;
35 }
```

図 1 プログラム

10	
2	0
3	1
4	3
6	3
7	1
8	4
9	8

図2 input.txt

9	7
6	4
0	2

図3 pair.txt

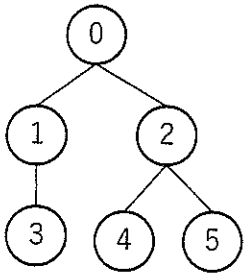


図4 木の表記例

2 【必須問題】計算機システムとシステムプログラム （情報工学 3/13）

配点： (1-1) 8 点，(1-2) 10 点，(1-3-1) 18 点，(1-3-2) 29 点，
(2-1) 18 点，(2-2-1) 27 点，(2-2-2) 10 点，(2-2-3) 5 点

(1) キャッシュメモリ (cache memory) および仮想記憶 (virtual memory) を含む，計算機 (computer) の記憶システム (memory system) に関する以下の各小問に答えよ．解答は全て解答用紙の太線枠内に書くこと．

(1-1) キャッシュメモリや仮想記憶では，(a)「一度アクセス (access) されたアドレス (address) は近いうちに再アクセスされる可能性が高い」，(b)「一度アクセスされたアドレスに近接するアドレスは，近いうちにアクセスされる可能性が高い」というメモリアクセスの性質が一般に利用されている．これらの性質の名称を記せ．

(1-2) (a) キャッシュメモリを導入することによる効果，(b) 仮想記憶を導入することによる効果を，それぞれ一つずつ，簡潔に記せ．

(1-3) キャッシュメモリおよび仮想記憶を有し，以下の箇条書きに示す仕様 (specification) を持つ記憶システムを考える．なお，論理アドレス (logical address) から物理アドレス (physical address) への変換が，CPU とキャッシュメモリの間で行われる場合と，キャッシュメモリと主記憶 (main memory) の間で行われる場合の，二通りの方式がある．前者は物理アドレスキャッシュと呼ばれ，物理アドレスを用いてキャッシュメモリへのアクセスが行われる．後者は論理アドレスキャッシュと呼ばれ，論理アドレスを用いてキャッシュメモリへのアクセスが行われる．ここでは，これら双方の方式について考える．この記憶システムに関し，(1-3-1)，(1-3-2) に答えよ．

- アドレスは 1 [バイト] ([byte]) 毎に付与される．
- キャッシュメモリのマッピング (mapping) 方式は，直接マッピング (direct mapping) 方式．
- キャッシュメモリの 1 [ブロック] ([block]) は 2 [バイト]．
- キャッシュメモリ容量は 8 [バイト]．但し，タグ (tag) 等に必要な容量は，これには含まれない．
- 仮想記憶の制御方式は，ページング (paging) 方式．
- 仮想記憶の 1 [ページ] ([page]) は 8 [バイト]．
- 主記憶容量は 32 [バイト]．但し，実際のシステムでは主記憶上にページングの対象とならない領域が存在するが，ここでは，便宜上，主記憶の全ての領域がページングの対象となるものとする．
- 仮想記憶容量は 256 [バイト]．

(1-3-1) 以下の値 (a)〜(f) を記せ．

(a) 論理アドレスを表すのに必要な最小ビット長 (minimum bit length) [ビット] ([bit]).

(b) キャッシュメモリ内のブロック数．

(c) ページテーブル (page table) のエントリー (entry) 数．

(d) 主記憶内におけるページ番号を表わすのに必要な最小ビット長 [ビット]．

(e) 物理アドレスキャッシュにおける，キャッシュメモリの各タグの最小ビット長 [ビット]．

(f) 論理アドレスキャッシュにおける，キャッシュメモリの各タグの最小ビット長 [ビット]．

(1-3-2) 物理アドレスキャッシュにおいて，サイクル (cycle) 0〜7 に，以下の表に示す順に論理アドレスが参照されたとする．初期状態において，キャッシュメモリ内のブロックは空であり，主記憶内のページ枠 (page frame) 0〜3 には，それぞれ，仮想記憶内のページ 0，1，2，31 が割り当てられているとする．主記憶，仮想記憶，共に，アドレス 0 からページ 0 が始まり，連続してページが配置されるものとする．0〜7 の各サイクルでの，(a) 参照論理アドレスに対応する主記憶内のブロック番号，(b) アクセスされるキャッシュメモリ内のブロック番号，(c) アクセスされるキャッシュメモリブロックのタグの値を記せ．また，(d) サイクル 0〜7 におけるキャッシュヒット率を記せ．

サイクル	0	1	2	3	4	5	6	7
参照論理アドレス	0	1	2	3	4	8	21	255

(2) ファイルシステム (file system) に関する以下の各小問に答えよ．解答は全て解答用紙の太線枠内に書くこと．

(2-1) 以下の文中の空欄 [(ア)] ～ [(カ)] それぞれに最も適切な語を選択肢から一つ選び，その記号を答えよ．ただし，同じ選択肢を複数回用いてはならない．

ファイルシステムは，ファイルをファイル装置 (file device) に割り付ける (allocate) 等のファイル管理機能を提供する．ファイルは，ブロック (block) と呼ばれるファイル装置上の最小構成単位 (minimum configuration unit) でファイル装置に物理的に格納されており，ファイルへのアクセス (access) もブロック単位で行われる．ファイルへのアクセス方式には，ファイルを構成する先頭ブロックから末尾に向かってアクセスする [(ア)] と，ファイルを構成するブロックのいずれにも任意順にアクセスできる [(イ)] がある．

[(イ)] の場合のファイル割り付け方式を考える．ファイル装置上の連続したブロックに格納する連続ファイル割り付け (contiguous file allocation) を用いると，ファイル生成時に [(ウ)] を見積もる必要があるという問題や，ファイル装置上の未使用領域が [(エ)] した状態になるといった問題が生じる．これらの問題を回避するため，各ブロックへのポインタ配列 (array of pointers) を索引ブロックに格納しておく [(オ)] や，各ブロックに次のブロックへのポインタ（アドレス）を持たせておく [(カ)] 等が用いられる．

選択肢

(A) インデックスファイル割り付け（索引ブロックを用いた割り付け, indexed file allocation), (B) 直接アクセス (direct access), (C) 非断片化 (defragmentation), (D) プロセス割り付け (process allocation), (E) 最大ファイルサイズ (maximum file size), (F) 順アクセス（逐次アクセス，sequential access), (G) 断片化 (fragmentation), (H) アクセス時間 (access time), (I) リンクファイル割り付け（連結リスト割り付け, linked file allocation)

(2-2) ブロックサイズ (block size) が b [バイト], ブロックのアドレスの長さが a [バイト] で構築されたファイルシステムに，以下の三つの方式それぞれによってファイルを割り付けることを考える．

(i) 連続ファイル割り付け

(ii) リンクファイル割り付け

(iii) インデックスファイル割り付け

なお，方式 (ii) において，各ブロックには，次のブロックのアドレスを格納する容量が必要となることに注意せよ．また，方式 (iii) において，各ファイルの索引は一つの索引ブロックに収まるものとする．解答に際しては，床関数 (floor function) floor(x)（実数 x 以下の最大の整数を返す関数）や，天井関数 (ceiling function) ceil(x)（実数 x 以上の最小の整数を返す関数）を用いて良い．次の (2-2-1)～(2-2-3) に答えよ．

(2-2-1) ファイルサイズ s [バイト] のファイルを，方式 (i)～(iii) のそれぞれによってファイルシステムに割り付けた場合を考え，以下の (a)～(c) を求めよ．

なお，(a), (b) を求める際には，次の二点に注意せよ．

・各ブロックへの事前アクセスは無く，初めてアクセスするものとする．

・同一ブロックへ複数回アクセスする場合であっても，アクセスするブロック数は 1 である．

(a) ファイルの n ($1 \leq n \leq s$) [バイト] 目にアクセスする場合に，アクセスするブロック数

(b) ファイルの先頭から n ($1 \leq n \leq s$) [バイト] 目まで順にアクセスする場合に，アクセスするブロック数

(c) ファイルを格納するのに必要となるファイル装置上の最小サイズ p（ブロックサイズの整数倍）[バイト]

(2-2-2) ファイル装置上の最小サイズ p に対するファイルサイズ s の割合を利用効率 e ($= s/p$) として定義する．また，ブロックサイズに対するファイルサイズの比を r ($= s/b$) として定義する．(2-2-1) の結果を踏まえて，方式 (i) と (iii) のそれぞれについて，r に対する利用効率 e の変化を，$0 < r \leq 3$ の範囲で図示せよ．

(2-2-3) (2-2-2) の結果を踏まえて，r が大きくなるにつれて，方式 (i) の利用効率に対する方式 (iii) の利用効率の比がどのように変化するか，簡潔に述べよ．

配点:(1-1)~(1-4) 各 6 点，(1-5-1) 16 点，(1-5-2) 25 点，(2-1-1) 15 点，(2-1-2) 15 点，(2-2-1) 15 点，(2-2-2) 15 点

(1) 一階述語論理（first-order predicate logic）に関する以下の各小問に答えよ．ただし，論理式（logic formula）の記述には以下の記号を用いる．$\forall$, $\exists$ はそれぞれ全称作用素（universal quantifier），存在作用素（existential quantifier）であり，$\Leftrightarrow$, $\rightarrow$, $\wedge$, $\vee$, $\neg$ はそれぞれ等価（equivalence），含意（implication），論理積（conjunction, and），論理和（disjunction, or），否定（negation, not）を表す論理演算子である．また，必要に応じて $\bigwedge_{1\leq i\leq n} T_i$（あるいは $\bigvee_{1\leq i\leq n} T_i$）の表記を用いる．これは論理式 T_i の i を 1 から n（n は正整数）まで順次変えながら論理積（あるいは論理和）で結合した式を意味している．

本問では，要素（element）の列であるリスト（list）を扱う．先頭の要素が H であり，残りの要素により構成されるリストが T であるとき，このリストを $[H|T]$ で表す．また，空のリストを $[\,]$ で表し，要素数が 1 のリスト $[H|[\,]]$ は簡略化して $[H]$ と表す．以降，X, Y, Z はリストを表す変数とし，A はリストの要素を表す変数とする．リスト X とリスト Y をこの順に接合したリストがリスト Z であることを表す述語（predicate）を $append(X,Y,Z)$ とし，リスト X の要素の順を逆にしたリストが Y であることを表す述語を $reverse(X,Y)$ とする．以下の各小問に答えよ．

(1-1) 「任意のリスト X に対して，空のリスト $[\,]$ とリスト X をこの順に接合したリストがリスト X である」ことを表す閉論理式（closed formula）を示せ．

(1-2) 「$append(X,Y,Z)$ が成り立つ任意のリスト X，Y，Z に対して，リスト X の先頭に任意の要素 A を追加したリストとリスト Y とをこの順に接合したリストは，リスト Z の先頭に要素 A を追加したリストである」ことを表す閉論理式を示せ．

(1-3) 「空リストを逆順にしたものは空リストである」ことを表す論理式を述語 *reverse* を用いて示せ．

(1-4) 述語 *reverse* に対して成り立つ以下の論理式について，空欄 α を埋めよ．

$$\forall A\ \forall X\ \forall Y\ \forall Z((reverse(X,Y) \wedge append(Y,[A],Z)) \rightarrow reverse(\boxed{\ \alpha\ }, Z))$$

(1-5) 論理プログラミングでは以下の論理式の恒真性（validity）を問うことで解を探索する．

$$\left(\bigwedge_{1\leq i\leq m} (\forall x_1 \cdots \forall x_h((P_{i1} \wedge \cdots \wedge P_{in_i}) \rightarrow Q_i))\right) \rightarrow \exists x_1 \cdots \exists x_h(R_1 \wedge \cdots \wedge R_k) \quad (1)$$

ただし，すべての P_{i1}, $\cdots$, P_{in_i}, Q_i, R_j $(1 \leq i \leq m, 1 \leq j \leq k)$ は原子論理式（atomic formula）であり，x_1, $\cdots$, x_h はこれらの原子論理式中に現れる変数で，以降の問では $h \geq 1$ とする．また，$n_1, \cdots, n_m \geq 0$ である．この論理式の恒真性を調べるには，この論理式の否定である式 (2) の充足不能性（unsatisfiability）を調べればよい．

(1-5-1) 式 (1) の否定をとると式 (2) となる．空欄 β および空欄 γ を埋めよ．ただし，解答には含意（$\rightarrow$）は含めないこと．

$$\forall x_1 \cdots \forall x_h(\boxed{\ \beta\ } \wedge \boxed{\ \gamma\ }) \quad (2)$$

(1-5-2) 式 (1) の否定が式 (2) であるという事実と小問 (1-1)~(1-4) で得られた論理式を利用して，導出原理（resolution principle）により，$reverse([a|[b]], W)$ の解が存在することと，そのときの W への代入を示せ．解の探索過程も記すこと．ただし，a，b はリストの要素を表す定数，W はリストを表す変数とする．

(2) 直立した m 本の棒（rod）1, 2, ..., m と，中心に穴があいた n 枚の円盤（disk）がある．円盤は大きさがすべて異なり，小さい順に 1, 2, ..., n で表す．各円盤 i はその穴によって，必ずいずれかの棒に挿されている．i から j $(1 \leq i \leq j \leq n)$ までの $j-i+1$ 枚の円盤が棒 k $(1 \leq k \leq m)$ に挿されているとき，かつそのときに限り真（true）となる述語（predicate）を $S_k(i,j)$ で表す．

$S_1(1,n)$ が成り立つ状態（state）から，$S_m(1,n)$ が成り立つ状態に移行するために，円盤を 1 枚ずつ移動させる手順（移動手順）を考える．ただし，いかなる状態においても，ある円盤の上にそれより大きい円盤があってはならず，ある円盤はその上に円盤がないときに限り移動できる．図 1 は，$m=3$ および $n=3$ の場合の，$S_1(1,n)$ が成り立つ状態（左図），その状態から円盤 1 を棒 2 に移動した状態（中図），$S_m(1,n)$ が成り立つ状態（右図），を表している．

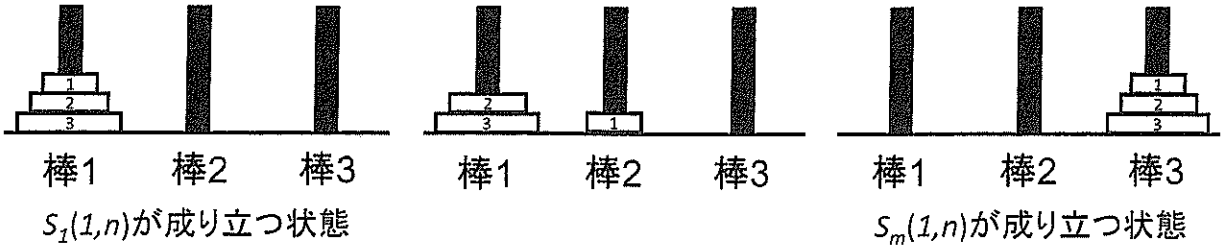


図 1: $m=3$ および $n=3$ の場合の状態の例

以下の各小問に答えよ．

(2-1) $m=3$ とする．このとき，$S_1(1,n)$ が成り立つ状態から，$S_3(1,n)$ が成り立つ状態へ移行する移動手順のうち，円盤の移動回数が最小のものを F_n で表し，F_n の円盤の移動回数を f_n で表す．以下の (2-1-1) および (2-1-2) に答えよ．

(2-1-1) 漸化式（recurrence relation）を用いた求め方により，f_n を n の式で表せ．

(2-1-2) F_n において，各円盤 i $(1 \leq i \leq n)$ が棒 1 から初めて移動されるとき，どの棒に移動されるかを答えよ．

(2-2) $m=4$ とし，ある円盤 d $(1 \leq d \leq n)$ に着目する．$S_1(1,n)$ が成り立つ状態から，$S_3(1,d)$ が成り立つ状態を少なくとも 1 つ経て，$S_4(1,n)$ が成り立つ状態に移行する移動手順を考える．それらの移動手順のうち，いかなる状態においても以下の (条件 1) および (条件 2) を満たし，かつ，円盤の移動回数が最小であるものを G_n で表し，G_n の円盤の移動回数を g_n で表す．

（条件 1）円盤 d およびそれより小さいどの円盤も棒 2 に挿されていてはならない．

（条件 2）円盤 d より大きいどの円盤も棒 3 に挿されていてはならない．

以下の (2-2-1) および (2-2-2) に答えよ．

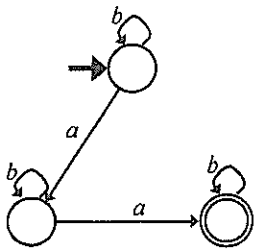
(2-2-1) g_n を n と d の式で表せ．求め方も述べよ．

(2-2-2) $f_n = g_n$ となる d の値を n で表せ．求め方も述べよ．

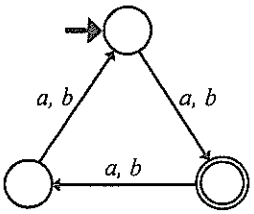
（配点：(1-1) 20 点, (1-2-1) 20 点, (1-2-2) 20 点, (2-1) 15 点,
(2-2) 5 点, (2-3) 10 点, (2-4) 10 点, (2-5) 25 点）

(1) オートマトン (automaton) に関する以下の各小問に答えよ．

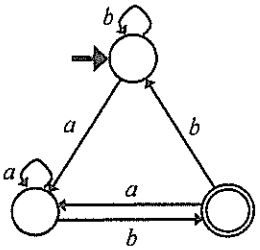
(1-1) 下の (i), (ii), (iii), (iv) は，有限オートマトン (finite automaton) の状態遷移図 (state transition diagram) を表している．太い矢印 (thick arrow) が指し示している状態が開始状態 (start state)，二重丸で表されている状態が最終状態 (final state) である．これらのオートマトンそれぞれについて，そのオートマトンが受理する言語 (language) を表している正則表現 (regular expression) を，1〜8 の中から一つ選べ．



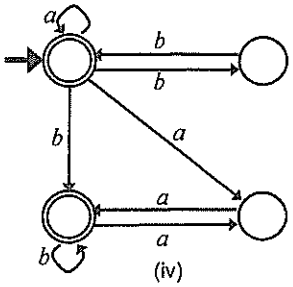
(i)



(ii)



(iii)



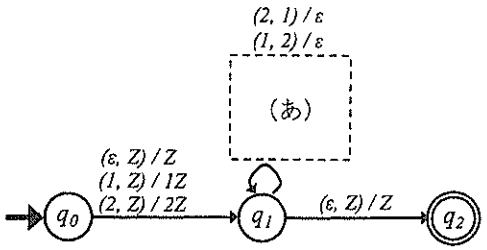
(iv)

1. $(a+b)a^*b^*$　2. $b^*a+aa^*(a+b)$　3. $(a+bb)^*(aa+b)^*$　4. $b^*+a^*+b^*a^*$
5. $(a+b)^*ab$　6. $b^*a^*b(a+b)^*$　7. $b^*ab^*ab^*$　8. $((a+b)(a+b)(a+b))^*(a+b)$

(1-2)10 進数 (decimal) で表現された正の整数 (positive integer) のうち，3 で割り切れるものを受理するオートマトンを考える．なお，10 進数で表現された正の整数を 3 で割った余りは，その正の整数を構成する各桁の数の和を 3 で割った余りと等しくなることを利用して良い．例えば，122 における各桁の数の和は 1+2+2=5 であるので，122 を 3 で割った余りは 2 となる．(1-2-1) および (1-2-2) に答えよ．

(1-2-1) 決定性有限オートマトン (deterministic finite automaton) は $(Q, \Sigma, \delta, p_0, F)$ で表される．ただし，Q は状態 (state) の有限集合，Σ は入力アルファベット (input alphabet)，δ は遷移関数 (transition function)，p_0 は開始状態，F は最終状態の集合である．決定性有限オートマトン $A_D = (\{p_0, p_1, p_2\}, \Sigma_D, \delta_D, p_0, \{p_0\})$ を用いて実現する場合，A_D の状態遷移図を書け．開始状態には太い矢印を付与すること．最終状態は二重丸で表現すること．簡単化のため，入力アルファベット Σ_D は Σ_D={1,2} とする．例えば，1221 や 222111 は受理される語 (word) であるが，1211 や 2222 は受理されない語である．

(1-2-2) プッシュダウンオートマトン (pushdown automaton) は $(Q, \Sigma, \Gamma, \delta, q_0, Z, F)$ で表される．ただし，Q は状態の有限集合，Σ は入力アルファベット，Γ はスタックアルファベット (stack alphabet)，δ は遷移関数，q_0 は開始状態，Z はスタックの開始記号，F は最終状態の集合である．最終状態による受理 (acceptance by final state) を行うプッシュダウンオートマトン $A_P = (\{q_0, q_1, q_2\}, \Sigma_P, \Gamma_P, \delta_P, q_0, Z, \{q_2\})$ を用いて実現する場合，以下の状態遷移図の「(あ)」の部分に必要な動作を全て答えよ．なお，「(あ)」の上部にすでに記述されている $(2,1)/\varepsilon$ および $(1,2)/\varepsilon$ は解答用紙に記述する必要はない．ε は空列を表す．スタックアルファベット Γ_P は $\Gamma_P = \{Z, 1, 2\}$ とする．簡単化のため，入力アルファベット Σ_P は Σ_P={1,2} とする．



(2) 文脈自由文法 (context-free grammar) は一般に $G = (V, T, P, S)$ で定められる．ただし V は変数 (variable; 非終端記号 nonterminal symbol) の集合，T は終端記号 (terminal symbol) の集合，P は生成規則 (production rule) の集合，S は出発記号 (start symbol) であり，$S \in V$ である．文脈自由文法 G_1 を以下の通り定め，G_1 により生成される言語を L_1 で表す．以下の各小問に答えよ．

$G_1 = (V_1, T_1, P_1, S_1)$

- $V_1 = \{A\}$
- $T_1 = \{(,)\}$
- $P_1 = \{A \to AA,\ A \to (A),\ A \to ()\}$
- $S_1 = A$

なお，L_1 に含まれる列 (sequence; 語 word) は，左括弧 (left parenthesis) と右括弧 (right parenthesis) のバランス (balance) がとれているものである．列のバランスがとれているとは，その列の中の列 () の削除を 1 回以上可能な限り繰り返すと，最終的に空列 ε が得られるとき，およびその時のみをいう．例えば列 ((((())))) や列 (())()() はバランスがとれているが，列)((()) や列 (())(() はバランスがとれていない．

(2-1) L_1 に含まれる長さ (length) が 6 以下の列を全て記せ．

(2-2) 列 (())()() に対する構文木 (parse tree) で，互いに異なるもの全てを図示せよ．

(2-3) 列 ()()()() に対する構文木で互いに異なるものは全部で何個あるか．数を答えよ．求め方は示さなくてよい．

(2-4) 列 ()()()()() に対する構文木で互いに異なるものは全部で何個あるか．数を答えよ．求め方は示さなくてよい．

(2-5) バランスのとれている任意の列が L_1 に含まれることを，列の長さに関する帰納法 (induction) で証明したい．証明を完成させるために，空欄（ア）～（エ）を適切な内容で埋めよ．

ただし，文形式 (sentential form) α に対して生成規則を 1 回適用して文形式 β が得られるとき，$\alpha \Rightarrow \beta$ と表記する．また，文形式 α に対して生成規則を 0 回以上適用して文形式 β が得られるとき，$\alpha \overset{*}{\Rightarrow} \beta$ と表記する．

証明: バランスのとれている列の長さは正の偶数 (even number) である．

- 長さ 2 のバランスがとれている列，すなわち列 () の場合，導出 (derivation) $A \Rightarrow$ [　（ア）　] により，この列は L_1 に含まれる．
- 長さ k 以下（ただし k は 2 以上の偶数）のバランスがとれている任意の列は L_1 に含まれると仮定する．
  長さ $k+2$ のバランスがとれている任意の列を x とおく．バランスがとれている二つの列 v と w に対し，$x = vw$ の形に分解 (decomposition) できる場合と，できない場合の 2 通りがある．
  - $x = vw$ の形に分解できる場合：
    v, w はそれぞれ長さが k 以下のバランスがとれている列なので，帰納法の仮定から v, w はともに L_1 に含まれる．すなわち $A \overset{*}{\Rightarrow} v$ かつ $A \overset{*}{\Rightarrow} w$ である．従って，導出 $A \Rightarrow$ [　（イ）　] $= x$ により，列 x は L_1 に含まれる．
  - $x = vw$ の形に分解できない場合：
    このとき，長さ k のバランスのとれている列 y が存在して，$x =$ [　（ウ）　] の形をしている．y の長さは k なので，帰納法の仮定から y は L_1 に含まれる．すなわち $A \overset{*}{\Rightarrow} y$ である．従って，導出 $A \Rightarrow$ [　（エ）　] $= x$ により，列 x は L_1 に含まれる．　(証明終)

配点: (1-1) (あ)〜(う) 各 2 点，(え)〜(け) 各 4 点，(1-2) 10 点，(1-3) 10 点
(2-1) (a)〜(c) 各 6 点，(d) 12 点，(2-2) 25 点，(2-3) 20 点

(1) 2 元ハミング符号 (binary Hamming code) に関する以下の文章について，各小問に答えよ．

次の行列を検査行列 (parity check matrix) とする 2 元ハミング符号を考える．

$$\begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

この符号の符号長 (code length) は [(あ)] であり，情報記号数 (the number of information symbols) は [(い)] である．したがって，符号化率 (code rate) は [(う)] となる．次の行列はこの符号の生成行列 (generator matrix) である．

$$\begin{pmatrix} 1 & 0 & 0 & 0 & \boxed{\text{(え)}} \\ 0 & 1 & 0 & 0 & \\ 0 & 0 & 1 & 0 & \\ 0 & 0 & 0 & 1 & \end{pmatrix}$$

一般に 2 元ハミング符号では，最小距離 (minimum distance) d は [(お)] であり，冗長記号数 (the number of redundancy symbols) を m とするとき，その符号長は [(か)] となる．また，その検査行列は，どの列ベクトル (列要素) (column vector) も零ベクトルではなく，各列ベクトルは全て相異なるという性質をもつ．したがって，冗長記号数が同じと仮定すると，2元ハミング符号より符号化率が高い 2 元ブロック符号の最小距離は [(お)] より小さくなる(i) ことがわかる．

さて，2 元対称通信路 (binary symmetric channel) で，2 元ハミング符号に対して限界距離 (bounded distance) $\lfloor (d-1)/2 \rfloor$ の復号 (decoding) を行う場合，受信語中のビット誤りの個数が [(き)] 個以下の場合は正しく復号 (訂正) できるが，それより多い場合は必ず誤って復号する(ii)．なお，限界距離復号を行う方法として，受信語と [(く)] 行列から求められる [(け)] を用いる方法がよく知られている．

注．限界距離 t の復号法は，受信語から距離 t 以内に符号語が存在すればその符号語に復号し，そうでなければ復号に失敗する．

(1-1) 空欄 (あ) 〜 (け) を埋めよ．

(1-2) 下線部 (i) が成り立つ理由を述べよ．

(1-3) 下線部 (ii) が成り立つ理由について，検査行列の性質 (二重下線部) を用いて述べよ．

(2) イーサネット (Ethernet) で使用される多重アクセス制御方式 (multiple access) の一つである CSMA/CD (Carrier Sense Multiple Access / Collision Detection) 方式に関する説明文を読み，以下の各小問に答えよ．

ブロードキャスト型ネットワーク (broadcast network) において，複数のホスト (host) が同時にフレーム (frame) を送出すると，フレームが回線上で衝突する．CSMA/CD 方式においては，各ホストはフレームを送出する前に搬送波検知 (carrier sense) を行い，他のホストがすでにフレームを送出していることが検知された場合にはフレーム送出を遅らせ，そうでない場合は直ちにフレームを送出する．また，フレーム送出中にも衝突検出 (collision detection) を行い，フレーム送出中に衝突を検出すると送出を中止する．

(2-1) 以下は，イーサネットで使用される CSMA/CD 方式においてフレームの衝突を検出した後に用いられるバックオフアルゴリズム (backoff algorithm) に関する説明文である．文中 (a) から (c) の空欄を適切な用語または数式で埋めよ．また，下線部 (d) の処理を行う理由を述べよ．

衝突を検出してフレーム送出を中止したホストは，時間 t だけ待った後に当該フレームの再送を試みる．t は各ホストで [(a)] に選択される．また，連続して衝突する回数が増えるに従って t の平均が大きくなるようにする(d)．

CSMA/CD 方式において用いられる [(b)] バックオフアルゴリズムでは，連続して衝突する回数が増えるに従って再送待ち時間 (retransmission time) を指数的に増大させる．当該フレームの n 回目の再送においては，$0 \leq r \leq X$ を満たす整数 r を [(a)] に選択し，最大往復伝播遅延時間 (slot time) を D としたとき $t =$ [(c)] 後にフレームを再送する．このとき，X は以下のように決定される．なお，15 回を超える再送は行わない．

$$X = \begin{cases} 2^n - 1 & (1 \leq n \leq 10) \\ 2^{10} - 1 & (10 < n \leq 15) \end{cases}$$

(2-2) イーサネットにおいて 10BASE-5 と呼ばれる規格では，回線容量が 10 [Mbps] であり，同軸ケーブル (coaxial cable) を使用している．また，最小フレーム長は 64 [byte]，最大フレーム長は 1518 [byte] である．

10BASE-5 においては，ホスト間の最大距離は 2.5 [km] (リピータ使用時) と規定されている．衝突検出の観点から，この規定が適切であることを最小フレーム長との関係に基づいて説明せよ．ただし，同軸ケーブルの信号伝播速度は 2×10^5 [km/s] とする．

(2-3) CSMA/CD 方式とは別の多重アクセス制御方式として Pure ALOHA 方式がある．Pure ALOHA 方式においては，CSMA/CD 方式のような搬送波検知や衝突検出を行わず，送信側ホストはフレームをすぐに送出する．一方，受信側ホストはフレームを受信すると ACK (確認応答) を送信側ホストに通知する．送信側ホストは，ある時間以内に ACK を受け取らなければフレームを再送する．

回線の混雑度合いが低い時と高い時のそれぞれにおける，Pure ALOHA 方式を用いた場合の回線のスループット (throughput, 単位時間当たりの伝送成功フレーム数) について，CSMA/CD 方式と比較してその高低を理由とともに説明せよ．ただし，フレーム長は全て等しいものとする．また，搬送波検知にかかる時間は無視できるものとする．

6 【選択問題】電子回路と論理設計 （情報工学 11/13）

配点：(1-1) 20 点，(1-2) 20 点，(1-3) 20 点，(1-4) 20 点，(2-1) 20 点，(2-2) 25 点

(1) 図 1 に示す手順で 2 数の差 (difference) の絶対値 (absolute value) を計算する組合せ論理回路 (combinational logic circuit)を考える．ただし，入力を 2 の補数表現 (two's complement expression) された 3bit の整数 $A=(a_2, a_1, a_0)$ および $B=(b_2, b_1, b_0)$，出力を 3bit の符号なし整数 $F=(f_2, f_1, f_0)$ とする．a_2, b_2, f_2 はそれぞれ A, B, F の MSB (most significant bit) を表している．まず，演算結果がオーバーフローしないように，3bit の $A=(a_2, a_1, a_0)$, $B=(b_2, b_1, b_0)$ を，それぞれ同値を表現する 2 の補数表現された 4bit の $A'=(a_3, a_2, a_1, a_0)$, $B'=(b_3, b_2, b_1, b_0)$ に拡張する．次に，A' から B' を減じた結果 $T=(t_3, t_2, t_1, t_0)$ を計算する．t_3 は T の MSB である．Tが非負であれば求めたい値が t_2, t_1, t_0 に格納されているが，Tが負であれば値の補正を行う．以下の各小問に答えよ．

(1-1) $A=(0,1,1)$, $B=(0,1,0)$の時の A', B', T, Fをそれぞれ求めよ．また，$A=(1,0,1)$, $B=(0,1,0)$の時の A', B', T, Fをそれぞれ求めよ．

(1-2) A'，B' の a_3および b_3 を，a_i, b_i (i=0, 1, 2) および 0, 1 を用いた論理式 (logical expression) で表せ．

(1-3) Tの t_i ($i=0, 1, 2, 3$) は，図 2 に示す 4 個の全加算器 (Full Adder: FA) に論理ゲートを追加して計算することができる．全加算器 FA_iの入力 x_i, y_i, z_i ($i=0, 1, 2, 3$) を a_i, b_i ($i=0, 1, 2, 3$), s_j, c_j (j=0, 1, 2 ただし $j<i$) および 0, 1 を用いた論理式で表せ．s_j, c_j は，それぞれ各全加算器の和出力および桁上げ出力である．ただし，$t_i=s_i$ ($i=0, 1, 2, 3$)となるようにせよ．

(1-4) t_3, t_2, t_1, t_0 を変数とする f_2, f_1, f_0 の最簡積和形(最小積和形, minimal sum-of-products expression) の論理式を導出せよ．

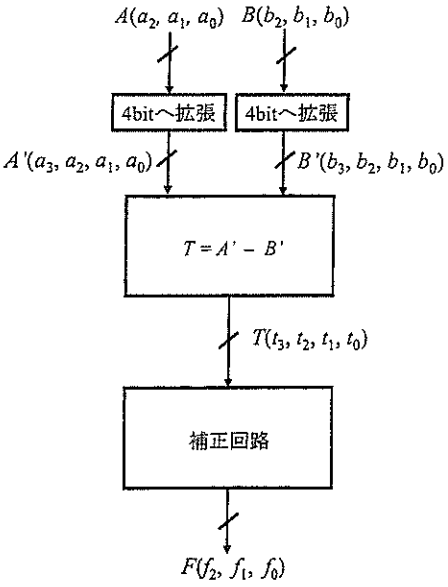


図 1

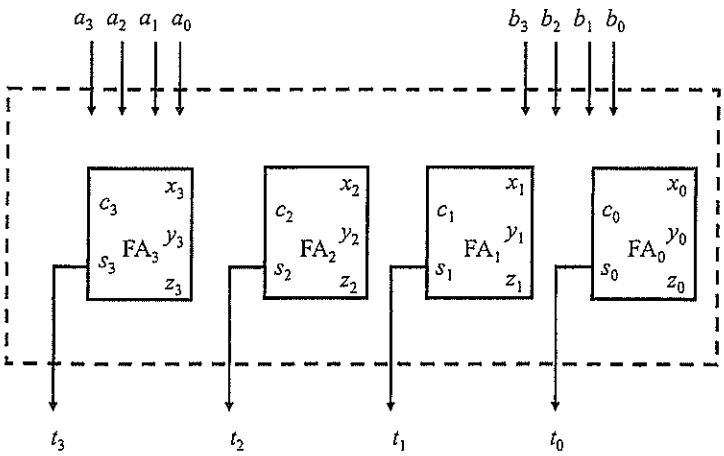


図 2

(2) 図 3 に示すような MOSFET を使った 2 入力 1 出力の CMOS 型 NOR 回路において，以下の小問に答えよ．ただし，$V_{DD}(>0)$ は電源電圧，C_L は負荷容量である．また，x_0，x_1 を入力，y を出力とし，x_0，x_1 は V_{DD} もしくは 0 の値しか取らないとする．

(2-1) 図 3 の A～D において，pMOS と nMOS のどちらが入るかをそれぞれ示せ．

(2-2) すべての MOSFET のドレイン（drain）・ソース（source）間の抵抗値（resistance）はオンにするときには瞬時に R_{MOS} となり，オフするときには瞬時に∞になると仮定する．図 3 の CMOS 型 NOR 回路において，十分に長い時間 V_{DD} であった入力 x_0，x_1 が 0 に変化してから，y における出力電圧が 0.5 V_{DD} になるまでの時間を式で示せ．

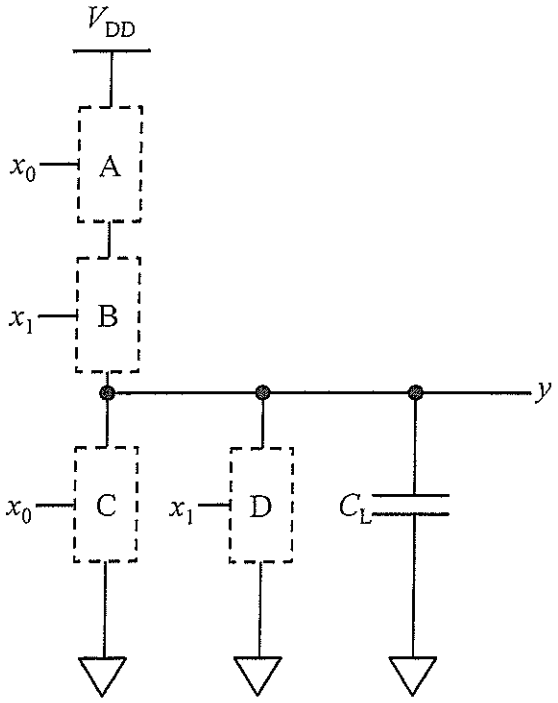


図 3

配点: (1-1)20 点,(1-2)40 点,(2-1)20 点,(2-2-1)25 点,(2-2-2)20 点

(1) 関数 $f(t)$ のラプラス変換 (Laplace transform) を $\mathcal{L}[f(t)] = F(s)$ と定義する.

以下の各小問に答えよ.

(1-1) $\frac{d}{ds}\mathcal{L}[f(t)] = -\mathcal{L}[tf(t)]$ であることを証明せよ.

(1-2) 上記の結果および $\mathcal{L}[f(t)'] = s\mathcal{L}[f(t)] - f(0)$ であることを用いて，次の微分方程式 (differential equation) を満たす $f(t)$ を求めよ.

$$tf(t)'' - (6t-1)f(t)' + 3(3t-1)f(t) = 0$$

ただし $f(t)' = \frac{df(t)}{dt}$, $f(t)'' = \frac{d^2f(t)}{dt^2}$ であり，$f''(0) = 9$ とする.

(2) 関数 $g(u)$ のフーリエ変換 (Fourier transform) を $\mathcal{F}[g(u)](\xi) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} g(u)e^{-i\xi u}du$ と定義する．ただし i は虚数単位（imaginary unit）である.

以下の各小問に答えよ.

(2-1) 下記の $g(u)$ について $\mathcal{F}[g(u)](\xi)$ を示せ.

$$g(u) = \begin{cases} u+2 & (-2 \le u < 0) \\ -u+2 & (0 \le u \le 2) \\ 0 & (otherwise) \end{cases}$$

(2-2) 下記の信号波形 (signal wave) を考える.

$$x(t) = y(t) = \begin{cases} 1 & (-1 \le t \le 1) \\ 0 & (otherwise) \end{cases}$$

(2-2-1) 畳み込み積分 (convolution integral) $x(t)*y(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau$ を求めよ.

(2-2-2) 畳み込み積分のフーリエ変換公式 (Fourier transform formula of convolution integral) $\mathcal{F}[x(t)*y(t)](\xi) = \sqrt{2\pi}\mathcal{F}[x(t)](\xi)\cdot\mathcal{F}[y(t)](\xi)$ について，(2) の下線の定義を用いて左辺と右辺をそれぞれ計算し，公式が成立することを証明せよ.