平成 19 年 8 月 4 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 20 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 3 題（$\boxed{1}$～$\boxed{3}$），選択問題 8 題（$\boxed{4}$～$\boxed{11}$），合計 11 題である．
  必須問題は 3 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 19 枚である．
- 解答用紙は全部で 7 枚である．
  1 枚目（灰色）の解答用紙には $\boxed{1}$（必須問題）の解答を
  2 枚目（緑色）の解答用紙には $\boxed{2}$（必須問題）(1)の解答を
  3 枚目（緑色）の解答用紙には $\boxed{2}$（必須問題）(2)の解答を
  4 枚目（赤色）の解答用紙には $\boxed{3}$（必須問題）(1)の解答を
  5 枚目（赤色）の解答用紙には $\boxed{3}$（必須問題）(2)の解答を
  6 枚目（白色）の解答用紙には $\boxed{4}$～$\boxed{11}$（選択問題）から選択した 1 題の解答を
  7 枚目（白色）の解答用紙には $\boxed{4}$～$\boxed{11}$（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，「問」の欄には対応する問題番号（1～11 から 1 つ）を記入すること．
  また，選択問題調査票上の，選択した問題の番号（4～11 から 2 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．
- 留学生特別選抜の受験者は，英語で解答することも可能である．その際，字数制限のある問については，同等な適当な長さの英文で解答すること．

1 （必須問題:アルゴリズムとプログラミング）

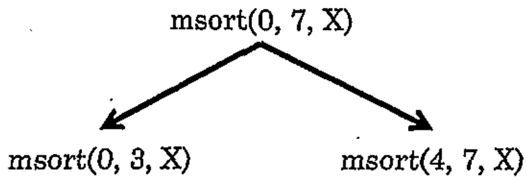
配点：(1) 15 点，(2-1) 20 点，(2-2) 20 点，(2-3) 20 点，(2-4) 25 点

マージソート（merge sort）とは，要素列を再帰的（recursive）に分割し，併合（merge）しながら整列（ソート： sort）するアルゴリズムである．ここでは，整列された 2 本の要素列があるとき，それらの列をまとめて 1 本の整列された列を作ることを併合と呼ぶ．以下の各問に答えよ．

(1) マージソートとクイックソート（quick sort）のアルゴリズムを平均および最悪の時間計算量（time complexity）の観点から比較せよ．

(2) 次ページの図 1 は構造体（structure） Item の配列（array）をメンバ（member） key の値で，マージソートを用いて昇順（ascending order）に整列する C 言語で書かれたプログラムである．なお，各行の行頭の数字およびコロンは行番号を表している．以下の各小問に答えよ．

(2-1) 図 1 の関数 msort は，key の値で配列 A の first 番目から last 番目の要素をマージソートで整列する．サイズ 8 の配列 X が与えられ，msort(0, 7, X) が呼び出されたときに，関数 msort が再帰的に呼び出される様子を，下図に追記する形で解答用紙に完成させよ．



(2-2) 図 1 の関数 merge は，key の値で昇順に整列されたサイズ nA の配列 A と key の値で昇順に整列されたサイズ nB の配列 B を一つの配列に併合する．なお，併合された配列は，配列 C の i 番目以降の要素に格納される．空欄（ア），（イ）を埋めよ．

(2-3) 図 1 の関数 main を実行した際に標準出力に出力される結果を示せ．

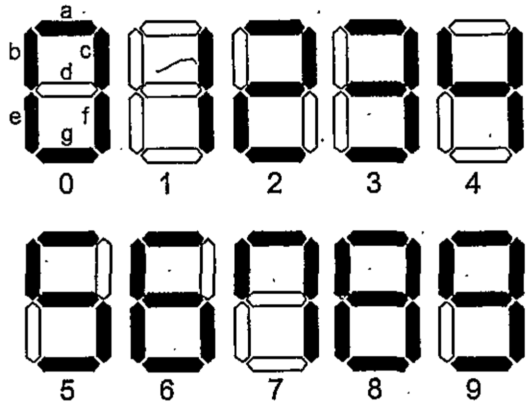
(2-4) 同じ key の値をもつ複数の配列要素がある場合，それらの要素間の整列前の配列におけるインデックス（添え字，index）の大小関係と，整列後の配列におけるインデックスの大小関係が変化しない整列アルゴリズムは安定である (stable) という．図 1 の関数 msort について，安定であるか安定でないかを答えよ．また，その理由をプログラムの行番号を指定して簡潔に答えよ．

```
#include <stdio.h>
#define MAX 32

typedef struct item {
  int key;
  float info;
} Item;

void merge(Item *A, int nA, Item *B, int nB, int i, Item *C) {
  int iA, iB, iC;
  iA = iB = iC = 0;
  while (iC <= nA + nB − 1) {
    if (iA >= nA) C[i + iC] = B[iB++];                   /* A については終了 */
    else if (iB >= nB) C[i+ iC] = A[iA++];              /* B については終了 */
    else if (A[iA].key <= B[iB].key)  [    (ア)    ];
    else                              [    (イ)    ];
    iC = iC + 1;
  }
}

void msort(int first, int last, Item *A) {
  int i, j, middle;
  Item A1[MAX], A2[MAX];

  if (last − first < 1) return;
  middle = (first + last) / 2;                          /* 中央値により分割 */
  msort(first, middle, A);                              /* 前半の整列 */
  msort(middle + 1, last, A);                           /* 後半の整列 */

  i = first; j = 0;
  while (i < middle + 1) A1[j++] = A[i++];              /* A1 は前半の部分配列 */
  i = middle + 1; j = 0;
  while (i < last + 1) A2[j++] = A[i++];                /* A2 は後半の部分配列 */
  merge(A1, middle − first + 1, A2, last − middle, first, A);  /* 併合 */
}

void print_array(int first, int last, Item *A) {
  int k;
  for (k = first; k < last + 1; k++)
    printf("%d: %d, %3.1f¥n", k, A[k].key, A[k].info);
}

Item X[] = {{5, 2.0}, {8, 3.5}, {2, 4.5}, {7, 5.0}, {2, 6.5}, {3, 9.0}, {4, 8.0}, {3, 7.5}};

int main() {
  msort(0, 7, X);
  print_array(0, 7, X);
  return 0;
}
```

図 1：ソートプログラム

(配点：(1-1)10 点，(1-2)18 点，(1-3)10 点，(1-4)12 点，(2-1)10 点，(2-2)20 点，(2-3)20 点)

(1)
下図は，a～g の 7 つのオン(on)・オフ(off)可能な表示素子(display element)からなる表示器(display device)を用いて 1 桁の 10 進数(decimal number)を表示した例である．例えば，「0 (zero)」は a, b, c, e, f, g をオン，d をオフにすることにより表現できる．それぞれの表示素子は，論理値(logical value) 1(0)を入力するとオン(オフ)になるとする．以下の各小問に答えよ．



(1-1) 入力する 1 桁の 10 進数 n を 4 ビット(bit)の 2 進数(binary number) $(x_3x_2x_1x_0)_2$ で表現する．このとき，n ($0 \leqq n \leqq 9$) および論理変数(logical variable) x_3, x_2, x_1, x_0 の値，表示素子 a～g のオン・オフの対応を示す，12 列からなる真理値表(truth table)を作りたい．解答用紙の空欄を埋めて真理値表を完成させせよ．ただし，x_0 を最下位桁(LSB; Least Significant Bit)とする．

(1-2) b, c, g の最小積和形(最簡積和形, minimal sum-of-products expression)をカルノー図(Karnaugh map)を用いてそれぞれ求めよ．ただし，10 進数の入力 n の値が 10～15 に対する出力値はドント・ケア(don't care)としてよい．

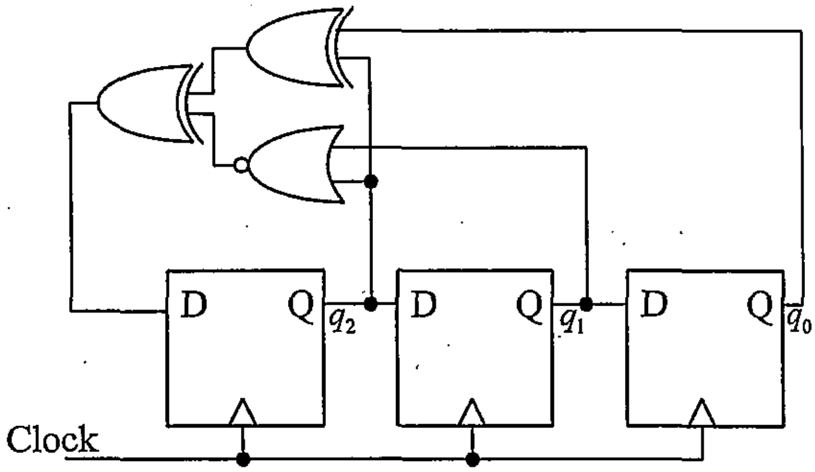
(1-3) (1-2) の結果に基づいて，表示素子 c への適切な入力を与える論理回路(logical circuit)を，NAND ゲート 3 つのみを用いて構成し図示せよ．ただし，入力として x_3, x_2, x_1, x_0 に加えてそれらの否定も利用できるものとする．

(1-4) 表示素子 f への適切な入力を与える論理回路を $\boxed{\mathrm{F}}$ で表す．入力として x_3, x_2, x_1, x_0 に加えてそれらの否定も利用できるものとする．このとき，表示素子 b へ適切な入力を与える論理回路は AND ゲート，OR ゲート，および論理回路 $\boxed{\mathrm{F}}$ のみを用いて構成できる．用いるゲート数が最小の構成を図示せよ．論理回路 $\boxed{\mathrm{F}}$ への入力は図示しなくてよい．

(2)

3 個のエッジトリガ型 D フリップフロップ(flip flop)を用いたカウンタ(counter)の設計について各小問に答えよ．

(2-1)　以下の図で表される順序回路(sequential circuit)について考える．この回路はある周期(sequence length)をもつカウンタとして動作する．カウンタの初期状態(initial state)は (q_2, q_1, q_0)=(0, 0, 0)である．カウンタの周期を答えよ．



(2-2)　1 ビットの入力 x を持つカウンタを新たに設計する． $x=0$ のとき，(2-1)で示したカウンタ動作を， $x=1$ のとき， (q_2, q_1, q_0) は(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1), (0, 0, 0), (0, 0, 1) …とカウントアップ動作をする．このカウンタの状態遷移表(state transition table)を完成させよ．

(2-3)　q_2, q_1, q_0 に対応する D フリップフロップの入力を d_2, d_1, d_0 とする．(2-2)の状態遷移表を元にカルノー図を利用して， d_2, d_1, d_0 をそれぞれ x, q_2, q_1, q_0 の論理関数で表せ．結果は最小積和形で示すこと．なお，設計の過程がわかるように解答すること．

3 （必須問題：計算機システムとシステムプログラム）

配点：(1-1) 21 点，(1-2) 25 点，(1-3) 12 点，(2-1) 16 点，(2-2) 16 点，(2-3) 10 点

(1) 図 1-1，図 1-2 は，4 ビット×4 ビット乗算器 (multiplier) の構成例を示している．これらの乗算器に関する以下の説明を読んで，小問 (1-1)～(1-3) に答えよ．

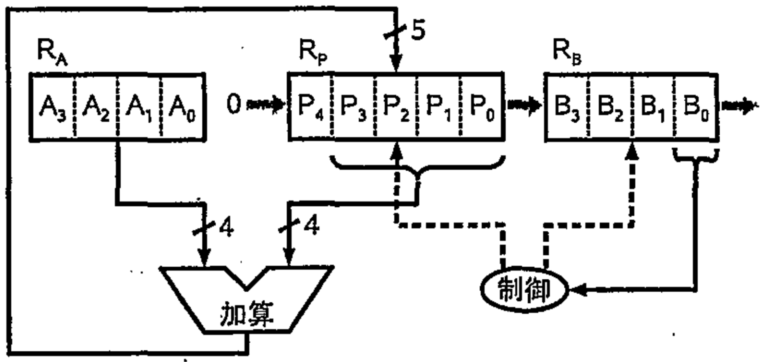


図 1-1: 繰り返し型 (逐次型) 乗算法による乗算器構成例

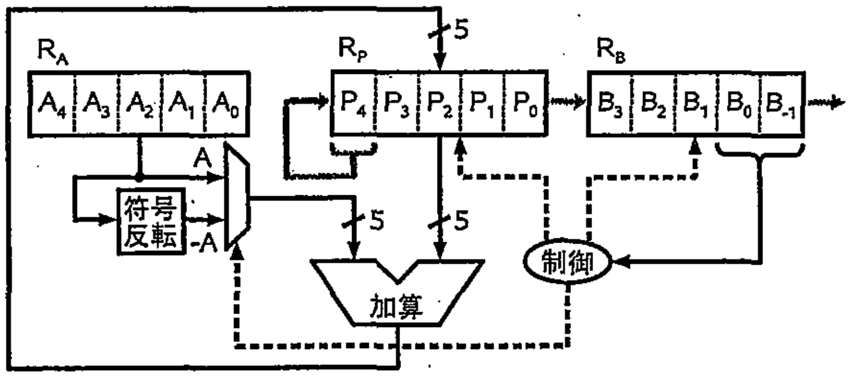


図 1-2: Booth 法による乗算器構成例

図中で R_A，R_B，R_P はレジスタ (register)，実線矢印はデータの流れ，点線矢印は制御信号の流れをそれぞれ示し，実線矢印に斜め線と共に示されている数値は，ビット幅を表している．R_B と R_P はシフト機能も有し，灰色矢印はシフト動作時のデータの流れを示している．R_A に被乗数 (multiplicand) A，R_B に乗数 (multiplier) B を格納し，R_P を 0 に初期化して乗算器を動作させると，乗算結果の下位 4 ビットが B_0～B_3 に，上位 4 ビットが P_0～P_3 に格納される．

図 1-1 は繰り返し型 (逐次型) 乗算法 (sequential multiplication) による符号無し数の乗算器構成例を示している．この乗算器では，R_B の最下位桁 (least significant bit) B_0 の値に応じて，加算を行うか否かが選択され，次のクロックサイクルで R_P と R_B の内容が 1 ビット論理右シフト (logical shift right) される (左から 0 が詰められる)．これらの処理を繰り返して乗算が行われる．加算は，4 ビット符号無し加算器により行われ，最上位桁 (most significant bit) の桁上げ (carry) も含めて 5 ビットが R_P に格納される．

図 1-2 は，条件付きで加算回数の削減が可能であり，かつ，符号付き数を統一的に取り扱い可能な乗算法として知られる Booth 法 (Booth's algorithm) による符号付き数の乗算器構成例を示している．この乗算器では，符号付き数は 2 の補数 (2's complement) 表現により表される．被乗数 A には符号拡張 (sign-extension) により A_4 が付加され，乗数 B には初期値 0 の B_{-1} が付加される．B_0，B_{-1} の値に応じて表 1 に従って加算処理が行われ，次のクロックサイクルで R_P と R_B の内容が 1 ビット算術右シフト (arithmetic shift right) される (左から符号ビット P_4 が詰められる)．これらの処理を繰り返して乗算が行われる．加算は 5 ビット符号無し加算器により行われ，最上位桁の桁上げは無視される．

表 1: Booth 法における加算処理

B_0	B_{-1}	加算処理
0	0	加算しない
0	1	A を加算
1	0	−A を加算
1	1	加算しない

(1-1) 図 1-1 の乗算器において，被乗数 A が 6，乗数 B が 7 の場合の，R_A の内容 (2 進数)，各クロックサイクルの処理内容，各クロックサイクルの処理完了後の R_P，R_B の内容 (2 進数) を示せ．処理内容欄には，下記の選択肢から選んで記号を記入せよ．

(選択肢) (a) 1 ビット右シフト，(b) A を加算

(1-2) 図 1-2 の乗算器において，被乗数 A が −6，乗数 B が 7 の場合の，R_A の内容 (2 進数)，各クロックサイクルにおける処理内容，各クロックサイクルの処理完了後の R_P，R_B の内容 (2 進数) を示せ．処理内容欄には，下記の選択肢から選んで記号を記入せよ．

(選択肢) (a) 1 ビット右シフト，(b) A を加算，(c) −A を加算

(1-3) 図 1-2 の乗算器において，加算回数が最小および最大となるのは乗数 B がどのような場合か示せ．

(2) 以下の各小問に答えよ.

(2-1) 磁気ディスク(magnetic disk)装置に関する次の説明文の空欄（a）～（h）に当てはまる適切な語句を下記の選択肢から1つ選び，その番号を答えよ.

図2に示すように，磁気ディスク装置の内部には，両面に（a）の塗られた円盤 (ディスク：disk) が複数枚あり，それらが回転軸を共有し，モータ(motor)によって高速回転される．データの記録がディスク上に同心円状になされる場合，その同心円を（b）と言う．（b）は，複数個の（c）から構成されており，データは（c）単位で読み書きされる．全てのディスク上の同じ半径を持つ（b）をまとめて（d）と呼ぶ．同じ（d）上に存在するデータは，ディスクが異なっていてもアクセスする際に（e）を移動させる必要がないため，高速アクセスが可能である．

この装置の性能を決定する指標には，目的の(d)まで(e)が移動するための時間である(f)，ディスクが回転してアクセス対象となる（c）の先頭が（e）のある場所に到達するための時間である（g），および実際にデータを読み書きするための時間である（h）が用いられる．また，（f），（g），（h）の和を一般にアクセス時間と呼ぶ.

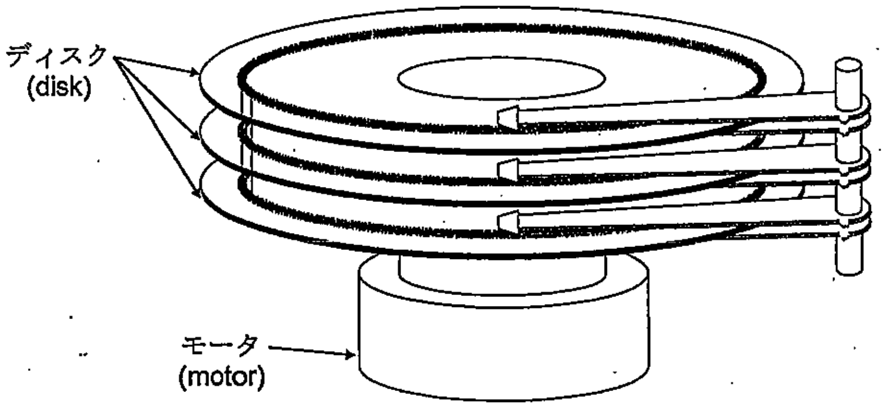


図2：磁気ディスク装置

選択肢

①計算時間 (calculation time)　②パーティション (partition)　③セクタ (sector)
④転送時間 (transfer time)　⑤ゾーン (zone)　⑥ヘッド (head)
⑦サーチ時間あるいは回転遅延時間 (search time)　⑧ボリューム (volume)
⑨ディレクトリ (directory)　⑩磁性体 (magnetic material)
⑪キャッシュ時間 (cache time)　⑫シリコン (silicon)　⑬トラック (track)
⑭シーク時間 (seek time)　⑮シリンダ (cylinder)

(2-2) 平均シーク時間が 10 [ミリ秒], 回転速度が 4800 [回転/分], セクタ長が 512 [バイト], 1トラックあたりの記録容量が 10240 [バイト/トラック]のディスクから, 1つのセクタを読み出すのに必要な平均アクセス時間を求めよ. 導出過程も示すこと.

(2-3) 様々なシリンダ上のデータへのアクセス要求が複数ある場合に, ヘッドを移動させるシリンダの順序を決定する手法として,

(ア) 最短シーク順 (shortest seek first) : 現在のヘッドの位置に最も近いシリンダへのアクセスを優先させる

(イ) エレベータ順 (エレベータ・アルゴリズム, elevator algorithm) : ヘッドを移動させる方向を決め, その先にアクセス要求がなくなるまでその方向にヘッドを移動させながらアクセスを行い, その後, 逆方向にヘッドを移動させアクセスを続ける

がある. (イ)が(ア)に対してアクセス時間の観点で優れている点を 100 字以内で述べよ.

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

図 1 に示すように電位V_1, V_2 （$V_2 > V_1$）の無限平行導体板(parallel infinite conducting planes) の間に，誘電率(dielectric constant) $\varepsilon_1, \varepsilon_2$，導電率(electric conductivity) σ_1, σ_2，厚さd_1, d_2の 2 種類の物質が挿入されている．以下の各問に答えよ．導出過程も示すこと．ただし，x, y, z 軸は図 1 のように定めている．

(1)　2 種類の物質が導電率 0（$\sigma_1 = \sigma_2 = 0$）の誘電体(dielectric material)である場合を考える．以下の各小問に答えよ．

(1-1)　無限平行導体板間の各部における電束密度(dielectric flux density) $\boldsymbol{D}$ の大きさと方向を求めよ．

(1-2)　電位V_1の導体板から誘電体内部の法線方向(normal direction) にx （$0 \leq x \leq d_1 + d_2$）の距離の点の電位(electric potential) V を求めよ．

(1-3)　無限平行導体板における単位面積当たりの静電容量(electric capacitance)を求めよ．

(2)　2 種類の物質の導電率が共に 0 でない（$\sigma_1 \neq 0$，$\sigma_2 \neq 0$）場合を考える．このとき，無限平行導体板間の各部における電界(electric field) の大きさを求めよ．

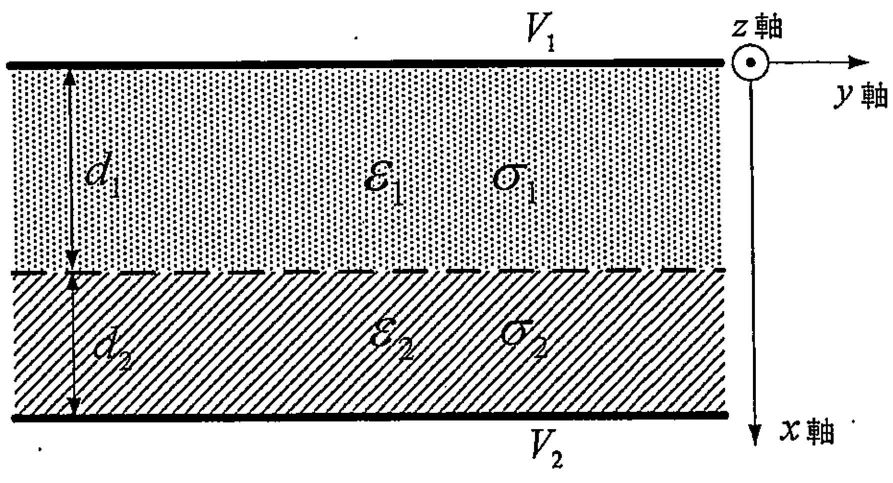


図 1

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

図 1 は，自己インダクタンス（self inductance）L_1，L_2，抵抗（resistance）R_1，R_2 の回路が相互インダクタンス（mutual inductance）M で結合され，一次側に角周波数 ω の電源電圧 $e(t) = E\sin\omega t$ を加えた定常状態にある回路である．図 1 の回路について，フェーザ（phaser）法を用いて以下の各問に答えよ．

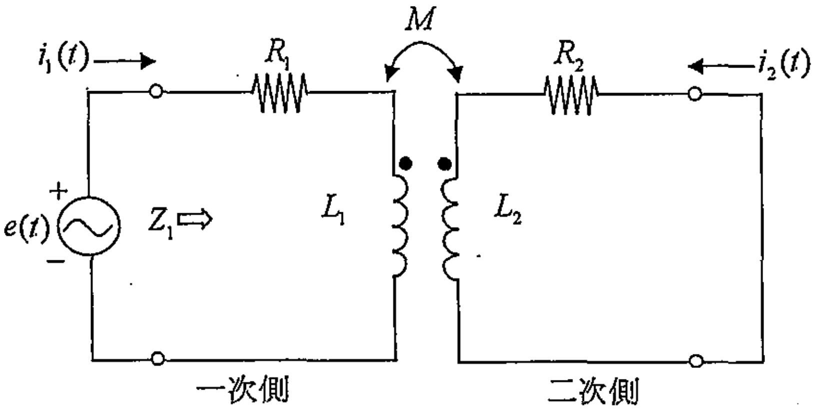


図 1

(1) 一次側に流れる電流 $i_1(t)$ のフェーザ表示を $\dot{I}_1$，二次側に流れる電流 $i_2(t)$ のフェーザ表示を $\dot{I}_2$ とする．$\dot{I}_1$ と $\dot{I}_2$ の関係式を，電源電圧 $e(t)$ のフェーザ表示 $\dot{E}$ を用いずに求めよ．

(2) 一次側から見た合成インピーダンス Z_1 のフェーザ表示について，実部，虚部をそれぞれ求めよ．

(3) 相互インダクタンスを a 倍とした時に，一次側に流れる電流を $\dot{I}_1$ のまま，二次側に流れる電流 $\dot{I}_2$ を $\frac{1}{a}$ 倍に変更したい．二次側の R_2，L_2 はどのように変更すればよいか答えよ．

(4) 図 1 の回路は，図 2 のようにフェーザ法で表示された相互インダクタンスを含まない等価回路で表すことができる．このときの $Z_a(j\omega)$，$Z_b(j\omega)$，$Z_c(j\omega)$ をそれぞれ求めよ．

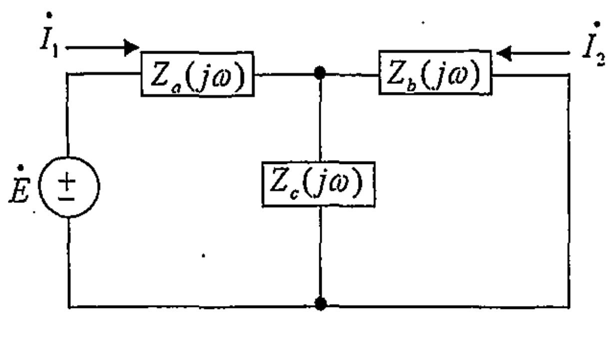


図 2

(配点: (1)50 点, (2)50 点)

以下の各問に答えよ.

(1) X-Y 平面における曲線について考える. この曲線上の点 $\mathrm{P}(x, y)$ における法線 (normal line) へ原点 O から垂線 (perpendicular) をおろし, 法線との交点を $\mathrm{N}(\alpha, \beta)$ とする. また, P から X 軸への垂線の足 (foot of perpendicular) を H とする. 直線 OP と直線 HN とが平行になる場合, この曲線は $2xyy' = y^2 - x^2$ なる微分方程式 (differential equation) を満足することを示せ.
また, この微分方程式を解いて, この曲線の式を求めよ. ここで, $y' = dy/dx$ である.

(2) $f(x) = \cos \mu x \ (-\pi \le x \le \pi)$ をフーリエ級数展開 (Fourier series expansion) せよ.
また, 得られた結果を利用して,

$$\sum_{n=1}^{\infty} \frac{2\mu}{\mu^2 - n^2}$$

の値を求めよ. ただし, μ は整数でないものとする.

(配点: (1-1) 20 点, (1-2) 20 点, (2-1) 30 点, (2-2) 30 点)

(1) 次のように, 5つの集合 X_1, X_2, X_3, X_4, X_5 それぞれの上で定義された関係 R_1, R_2, R_3, R_4, R_5 を考える. ここで, N^* は 2 以上の整数の集合, N は 1 以上の整数の集合である.

$X_1=\{a,b,c,d\}$, $R_1=\{(a,a),(b,b),(c,c),(d,d),(a,b),(a,c),(a,d),(b,d),(c,d)\}$
$X_2=\{1,2,3,4,5\}$, $R_2=\{(1,1),(1,2),(2,2),(2,1),(3,3),(3,5),(4,4),(5,5),(5,3)\}$
$X_3=\{a,b,c\}$, $R_3=\{(a,a),(b,b),(c,c)\}$
$X_4=N^*$, $R_4=\{(x,y)\mid x$ と y は 1 より大きい公約数をもつ $\}$
$X_5=N$, $R_5=\{(x,y)\mid y$ は x の約数 $\}$

(1-1) R_i $(i=1,2,3,4,5)$ のうち, 同値関係 (equivalence relation) であるものをすべて挙げよ. また, 同値関係として挙げた R_i それぞれについて同値類 (equivalence class) を全て記せ.

(1-2) R_i $(i=1,2,3,4,5)$ のうち, 半順序関係 (partial order relation, semiorder relation) であるものをすべて挙げよ. また, 半順序関係として挙げた R_i それぞれについて, X_i の最大元があればそれを示せ. 最大元が存在しない場合はそのように記せ. ただし, $l \neq m$ であるような $(l,m) \in R_i$ について, m の方が l より大きいと定義する.

(2) S をある集合とし, f を S から S への写像とする. S から S への写像 $f^1, f^2, f^3, \cdots$ を, 任意の $x \in S$ について, $f^1(x)=f(x)$, $f^2(x)=f(f(x))$, $f^3(x)=f(f(f(x)))$, $\cdots$ により定義する. ある自然数 k が存在し, 次の条件が成立しているとする.

条件「任意の $x \in S$ について $f^k(x)=x$ である」

(2-1) 「$x=f^i(y)$ となるような自然数 i が存在するとき, かつ, そのときに限り $x \sim y$」と定めることにより, S 上の関係「$\sim$」を定義する. このとき「$\sim$」が同値関係であることを証明せよ.

(2-2) f が全単射 (bijection) であることを証明せよ.

(配点:(1)20 点，(2-1)10 点，
(2-2-1)10 点，(2-2-2)15 点，(2-2-3)30 点，(2-2-4)5 点，(2-2-5)10 点)

複数の地点からなる集合を考える．集合中の２つの地点間のコスト (交通費，時間など) を求めたい．任意の地点から別の任意の地点へ,「直接コスト」と呼ぶコストが定義されるものとする．

次のように述語 (predicate) T を定義する．

$T(x,y,u)$: 「地点 x から地点 y へコスト u で到達可能である」

また，関数記号 f,m,s は，すべて２引数の関数とし，それぞれ以下を表すとする．

$f(x,y)$: 地点 x と地点 y 間の直接コスト;

$m(x,y)$: x と y のうち小さい値;

$s(x,y)$: x と y の和

以下の各問に答えよ．なお，解答には論理記号として $\to, \wedge, \vee, \neg, \forall, \exists$ を用いよ．

(1) 次の各内容を表す閉論理式 (closed formula)A, B をそれぞれ記述せよ．
但し，x,y,z,w,v は変数とする．

A 「地点 x から地点 y へコスト v で到達可能である」ならば「地点 y から地点 x へコスト v で到達可能である」

B 「地点 x から地点 y へコスト w で到達可能であり」かつ「地点 y から地点 z へコスト v で到達可能である」ならば,「地点 x から地点 z へ,『w と v の和のコストと，直接コスト $f(x,z)$ の２つのうち，より小さいコスト』で到達可能である」

(2) 以下，３つの地点 n_1,n_2,n_3 からなる集合上で考える．以下の各小問に答えよ．

(2-1) 次の各内容を表す論理式 C, D をそれぞれ記述せよ．

C 地点 n_1 から地点 n_2 へコスト $f(n_1,n_2)$ で到達可能

D 地点 n_3 から地点 n_2 へコスト $f(n_3,n_2)$ で到達可能

(2-2) 論理式 $(A \wedge B \wedge C \wedge D) \to (\exists z\, T(n_1,n_3,z))$ を E とする．
式 E の否定から導出原理 (resolution principle) を用いて，地点 n_1 から地点 n_3 へのコストを表す式 (expression) を求めたい．次の各小問に答えよ．

(2-2-1) 式 $\neg E$ に対応するスコーレム連言標準形 (Skolem conjunctive normal form) を求めよ．

(2-2-2) (2-2-1) で求めたスコーレム連言標準形を表現する節 (clause) 集合を挙げよ．

(2-2-3) (2-2-2) で求めた節集合に対して，導出原理を用いてコストを表す式を求めよ．導出の過程も示すこと．

(2-2-4) (2-2-3) で求めたコストの式の内容を文章で説明せよ．

(2-2-5) 関数 $s(x,y), m(x,y), f(x,y)$ の解釈を以下に与える．

$$s(x,y) = x + y;$$
$$m(x,y) = \begin{cases} x & : \ x \leq y \\ y & : \ x > y \end{cases}$$
$$f(n_1,n_2) = 10, \quad f(n_1,n_3) = 25, \quad f(n_2,n_3) = 10,$$
$$f(n_1,n_1) = 0, \quad f(n_2,n_2) = 0, \quad f(n_3,n_3) = 0, \quad f(x,y) = f(y,x)$$

このとき (2-2-3) で求めたコストの式に対して具体的な値を求めよ．

(配点: (1-1) 20 点, (1-2) 30 点, (1-3) 10 点, (2-1) 10 点, (2-2) 15 点, (2-3) 5 点, (2-4) 10 点)

(1) オートマトン (automaton) に関する以下の各小問に答えよ．なお，オートマトン M が受理する言語，およびその補集合を，それぞれ $L(M)$ と $L^c(M)$ で表す．

(1-1) 決定性有限オートマトン (deterministic finite automaton) M_a を右の表で定める．例えば状態 q_0 では，入力が 0 の場合の次状態は q_2 であり，入力が 1 の場合の次状態は q_1 である．初期状態 (initial state) は q_0，受理状態 (accepting states) は q_4 と q_5 とし，入力記号 (input symbols) は 0 と 1 である．

	0	1
q_0	q_2	q_1
q_1	q_5	q_4
q_2	q_0	q_3
q_3	q_5	q_4
q_4	q_4	q_5
q_5	q_4	q_5

$L(M_a) = L(M_b)$ を満たす最小状態数 (minimum number of states) の決定性有限オートマトン M_b を求め，状態遷移図 (state transition diagram) の形で示せ．

(1-2) 任意に与えられる有限オートマトン (finite automaton) M_1 と M_2 より，新たな有限オートマトンを構成する 3 つの手続き $\mathcal{U}, \mathcal{P}$ および $\mathcal{S}$ が存在するとする．構成される有限オートマトンはそれぞれ以下の性質を持つ．

$M' = \mathcal{U}(M_1)$ に対して $L(M') = L^c(M_1)$

$M' = \mathcal{P}(M_1, M_2)$ に対して $L(M') = L(M_1) \cap L(M_2)$

$M' = \mathcal{S}(M_1, M_2)$ に対して $L(M') = L(M_1) \cup L(M_2)$

2 つの有限オートマトン M_x と M_y を，それぞれ以下の条件を満たすように構成したい．

- $L(M_x) = \phi \iff L(M_1) = L(M_2)$
- $L(M_y) = \phi \iff \forall z \in L(M_1) : z \notin L(M_2)$

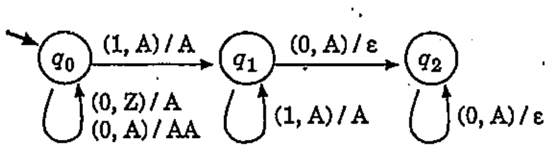
下記の選択肢は $M_1, M_2, \mathcal{U}, \mathcal{P}, \mathcal{S}$ を組み合わせて構成した有限オートマトンである．M_x と M_y に対し，条件に合致するものを選択肢からそれぞれ 1 つ選び，その記号を書け．

選択肢: (a) $\mathcal{P}(\mathcal{U}(M_1), M_2)$　(b) $\mathcal{P}(M_1, \mathcal{U}(M_2))$　(c) $\mathcal{P}(M_1, M_2)$　(d) $\mathcal{S}(M_1, \mathcal{U}(M_2))$

(e) $\mathcal{P}(\mathcal{S}(M_1, \mathcal{U}(M_2)), \mathcal{S}(\mathcal{U}(M_1), M_2))$　(f) $\mathcal{S}(\mathcal{P}(M_1, \mathcal{U}(M_2)), \mathcal{P}(\mathcal{U}(M_1), M_2))$

(g) $\mathcal{S}(\mathcal{P}(M_1, M_2), \mathcal{P}(\mathcal{U}(M_1), \mathcal{U}(M_2)))$

(1-3) 空スタック受理 (acceptance by empty stack) を行なう決定性プッシュダウンオートマトン (deterministic pushdown automaton) M_p を状態遷移図の形で右に示す．始点を持たない太い矢印が指し示している状態が初期状態である．例えば状態 q_0 から q_0 自身への矢印に添えられたラベル (0, Z) / A は，入力テープ上の文字が 0 でスタックトップの記号が Z のとき，スタックトップの記号をポップするとともに入力テープのヘッドを次に進め，次の状態を q_0 にし，スタックに A をプッシュする動作を表している．合致する動作が無い場合は M_p は動作を停止する．プッシュする記号が ε の場合は何もプッシュしないことを表す．なお Z はスタックの初期 (底) 記号 (initial symbol) であり，実行に先立って Z がひとつスタックに入れられる．入力を全て読み終えてスタックが空になるとき，空スタック受理という．



M_p が空スタック受理する入力の内，長さが 5 以下のものを全て示せ．

(2)　文脈自由文法 (context-free grammar) に関する以下の各小問に答えよ．以下では文脈自由文法を 4-組 (N,T,P,S) で表す．ここで N は非終端記号 (non-terminal symbol) の有限集合, T は終端記号 (terminal symbol) の有限集合，P は生成規則 (generating rule, production rule) の有限集合，S は始記号 (start symbol) である．ε は空語 (empty word) を表すものとする．

文脈自由文法 $G_1 = (N_1, T_1, P_1, S_1)$ を考える．N_1, T_1, P_1, S_1 は次の通りとする．

$$
\begin{aligned}
&N_1 = \{\, A, B, C, D, E, F \,\},\\
&T_1 = \{\, \texttt{<p>}, \texttt{</p>}, \texttt{
}, \texttt{
}, \texttt{
}, \texttt{
}, \texttt{}, \texttt{}, \texttt{text} \,\},\\
&P_1 = \{\, A \to \varepsilon \mid BA,\quad B \to C \mid D \mid E,\quad C \to \texttt{<p>}\ A \mid \texttt{<p>}\ A\ \texttt{</p>},\quad D \to \texttt{
}\ F\ \texttt{
},\\
&\qquad E \to \varepsilon \mid \texttt{text} \mid E\ \texttt{}\ E\ \texttt{}\ E,\quad F \to \varepsilon \mid \texttt{
}\ A\ F \mid \texttt{
}\ A\ \texttt{
}\ F \,\},\\
&S_1 = A
\end{aligned}
$$

(2-1)　G_1 によって生成される文 (sentence) のうち，すべての終端記号が現れるもので，最も長さが短いものを 1 つ示せ．文の長さとは，文を構成する終端記号の数とする．

(2-2)　以下の文は G_1 によって生成される．

$$\texttt{<p>}\ \texttt{<p>}$$

始記号 A からこの文への導出の一例を，以下に示す．

$$A \to BA \to CA \to C \to \texttt{<p>}\ A \to \texttt{<p>}\ BA \to \texttt{<p>}\ CA \to \texttt{<p>}\ C \to \texttt{<p>}\ \texttt{<p>}\ A \to \texttt{<p>}\ \texttt{<p>}$$

上の例と同様の形式で，始記号 A からこの文への最左導出 (leftmost derivation) と最右導出 (rightmost derivation) を，それぞれ 1 つずつ示せ．

(2-3)　文脈自由文法 $G_2 = (N_2, T_2, P_2, S_2)$ を考える．N_2, T_2, P_2, S_2 は次の通りとする．

$$
\begin{aligned}
&N_2 = N_1,\ T_2 = T_1,\\
&P_2 = \{\, A \to \varepsilon \mid BA,\quad B \to C \mid D \mid E,\quad C \to \texttt{<p>}\ A\ \texttt{</p>},\quad D \to \texttt{
}\ F\ \texttt{
},\\
&\qquad E \to \varepsilon \mid \texttt{text} \mid E\ \texttt{}\ E \mid E\ \texttt{}\ E,\quad F \to \varepsilon \mid \texttt{
}\ A\ \texttt{
}\ F \,\},\\
&S_2 = A
\end{aligned}
$$

G_1 で生成されるが，G_2 では生成されない文を，1 つ示せ．

(2-4)　文脈自由文法 $G_3 = (N_3, T_3, P_3, S_3)$ を考える．N_3, T_3, P_3, S_3 は次の通りとする．

$$N_3 = N_1,\ T_3 = T_1,\ P_3 = P_1 \cup P_2,\ S_3 = A$$

G_3 で生成されるが，G_1 では生成されない文を，1 つ示せ．

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

ビット誤り率 p $(0 < p < 1/2)$ の 2 元対称通信路 (binary symmetric channel) における通信を考える. 2 元ベクトル $\bar{v}$ を送信したときに 2 元ベクトル $\bar{u}$ が受信される確率を $P(\bar{v}, \bar{u})$ と書くことにする.

(1) 確率 $P(\bar{v}, \bar{u})$ に関して, 以下の小問に答えよ.

(1-1) $\bar{v} = (1,1,1,1,1,1,1)$, $\bar{u} = (1,0,1,0,1,1,1)$ のとき, 確率 $P(\bar{v}, \bar{u})$ を p を用いて表せ.

(1-2) $\bar{v}$, $\bar{u}$ の成分数は共に n であるとし, $\bar{v}$ と $\bar{u}$ のハミング距離 (Hamming distance) を D で表すとき, 確率 $P(\bar{v}, \bar{u})$ を p, n, D を用いて表せ.

(2) 最小距離 (minimum distance) が d, 符号長 (code length) が n の 2 元線形符号 (binary linear code) を上述の通信路で用いる場合を考える. 復号法として, 最大尤度基準 (maximum likelihood criterion) による復号法 (または最尤復号法), 限界距離復号法 (bounded distance decoding) などが知られている. 以下の小問に答えよ.

(2-1) 最大尤度基準による復号法では, 受信語 $\bar{u}$ に対して, $P(\bar{v}, \bar{u})$ を最大にする符号語 $\bar{v}$ に復号する. 符号語 $\bar{v}$ は受信語 $\bar{u}$ とのハミング距離が最小となる符号語であることを示せ.

(2-2) 限界距離復号法の復号過程を説明する以下の文の空欄 (あ)~(う) を埋めよ.

限界距離 t は, $\boxed{\text{(あ)}}$ を満たす非負整数である. 復号の過程は以下の 2 つの場合に分けられる.
(場合 1) 受信語から $\boxed{\text{(い)}}$ に符号語 $\bar{v}$ が存在する場合, $\bar{v}$ を復号結果とする.
(場合 2) そのような $\bar{v}$ が存在しない場合は, $\boxed{\text{(う)}}$ する.

(2-3) 限界距離復号法の (場合 1) において, 符号語 $\bar{v}$ に復号されるような受信語の数は, どの符号語 $\bar{v}$ についても, $\displaystyle\sum_{i=0}^{t} \binom{n}{i}$ であることを示せ. ここで, $\binom{n}{i}$ は二項係数であり ${}_n\mathrm{C}_i$ とも表記される.

(2-4) 2 元ハミング符号に対して, 限界距離 1 の復号法を用いるとき, 小問 (2-2) における限界距離復号法の (場合 2) は生じないことを示せ. 以下の事実は使ってよい.

2 元ハミング符号の符号長 n, 情報記号数 k は, $2^k(1+n) = 2^n$ を満たす.

11 (選択問題：ネットワーク)

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

(1) トークンパッシング方式 (token passing) は，ホストをリング状に接続し，リングに沿ってトークン (token) と呼ばれる特殊なビット列を巡回させることによって媒体アクセス制御 (medium access control) を行うものである．

トークンパッシング方式の基本動作は次のとおりである．まず，フレーム (frame) を送信したいホストは，トークンの受信を待つ．受信したトークンが，フレーム送信可能であることを表すフリートークン (free token) の場合には，これをビジートークン (busy token) に変えて，ビジートークンに続けてフレームを次のホストに送信する．ビジートークンを受信したホストは，続くフレームの宛先が自ホストであればフレームをコピーして受け取った後，また，宛先が異なるホストである場合にはそのまま，ビジートークンとフレームを次のホストに送信する．フレームの送信元ホストは，自らが送信したフレームのビジートークンを受信すると，ビジートークンと，続くフレームをリングから取り除いた後，フリートークンを生成して次のホストに送信する．

この，トークンパッシング方式に関する以下の各小問に答えよ．なお，トークンの大きさ，およびホストでの処理遅延は無視できるものとする．また，伝搬遅延 (propagation delay) を 1 km あたり 5 マイクロ秒とする．また，M (メガ) を 1.0×10^6，K (キロ) を 1.0×10^3 とする．

(1-1) 回線速度が 4 Mbps でリング長が 100 km のネットワークにおいて，あるホストが 1 Kbit のフレームの送信を開始した．このホストによって新たにフリートークンが送信されるまでの時間を答えよ．

(1-2) 同じ一台のホストのみが 1 Kbit のフレームを連続して送信する際の，回線速度が 4 Mbps でリング長が 100 km のネットワーク，および回線速度が 100 Mbps でリング長が 200 km のネットワークのそれぞれにおける平均スループット (throughput) を答えよ．なお，ホストはフレームごとにフリートークンを生成，送信し，再度フリートークンを受信してから次のフレームを送信するものとする．

(1-3) トークンパッシング方式の利点と欠点を 1 つずつ，イーサネットで使用されている媒体アクセス制御方式である CSMA/CD (Carrier Sense Multiple Access with Collision Detection) 方式と比較して，それぞれ 100 文字程度で述べよ．

(2)　インターネットプロトコル (IP：Internet Protocol) に関する以下の各小問に答えよ．

(2-1) IP のバージョン 4 では，32 ビットの IP アドレスによってホストを特定する．クラス B のアドレスクラスでは，最上位 2 ビットを "10" とし，続く 14 ビットをネットワーク識別子とする．残り 16 ビットのうち 8 ビットをサブネット識別子としたときの，1 つのサブネットでホストに割り当て可能なアドレスの最大数を答えよ．

(2-2) 以下の文中の空欄 (a)～(f) に当てはまる最も適切な用語を選択肢よりそれぞれ選べ．なお，異なる記号には異なる選択肢が当てはまる．

あるホストから送信された IP パケットは，ルーティングプロトコルによって定められた経路を通って宛先ホストへと届けられる．インターネットのルーティングプロトコルは，自律システム (autonomous system) 内で使用される $\boxed{\text{(a)}}$ と，自律システム間で使用される $\boxed{\text{(b)}}$ に大別される．$\boxed{\text{(a)}}$ の代表例である RIP (Routing Information Protocol) は $\boxed{\text{(c)}}$ 型ルーティング，OSPF (Open Shortest Path First) は $\boxed{\text{(d)}}$ 型ルーティングである．$\boxed{\text{(c)}}$ 型ルーティングでは，ノードは，ネットワークのアドレスごとに，そのネットワークに属するホスト宛のパケットを渡す隣接ノード，およびそのネットワークまでのコスト情報などをテーブルとして管理しており，この情報を隣接ノード間で交換することによって，テーブルの内容を更新する．一方，$\boxed{\text{(d)}}$ 型ルーティングでは，ネットワーク内の全てのノードがネットワーク全体の状態を把握しており，$\boxed{\text{(e)}}$ を用いて効率よく最短経路を計算する．OSPF は $\boxed{\text{(f)}}$ なネットワークに適している．

選択肢：

(1)　ディスタンスベクトル (distance vector)
(2)　AODV (Ad hoc On-Demand Distance Vector)
(3)　BGP (Border Gateway Protocol)
(4)　EGP (Exterior Gateway Protocol)
(5)　静的ルーティング (static routing)
(6)　マルチキャスト (multicast)
(7)　AIMD (Additive Increase and Multiplicative Decrease)
(8)　IGP (Interior Gateway Protocol)
(9)　小規模
(10)　大規模
(11)　ブロードキャスト (broadcast)
(12)　Dijkstra アルゴリズム
(13)　Bellman-Ford アルゴリズム
(14)　リーキーバケットアルゴリズム (leaky bucket algorithm)
(15)　トークンバケットアルゴリズム (token bucket algorithm)
(16)　リンクステート (link state)