平成22年7月31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23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（問題1～3），選択問題8題（問題4～11），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20枚である．
- 解答用紙は全部で8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（緑色）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（黄色）の解答用紙には問題3（必須問題）の(1)の解答を
  5枚目（黄色）の解答用紙には問題3（必須問題）の(2)の解答を
  6枚目（黄色）の解答用紙には問題3（必須問題）の(2)の解答の続きを
  7枚目（白色）の解答用紙には問題4～11（選択問題）から選択した1題の解答を
  8枚目（白色）の解答用紙には問題4～11（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．
- 解答用紙は8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，「問」の欄には対応する問題番号（1～11から一つ）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（4～11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

配点：(1) 30 点，(2-1) 32 点，(2-2) 15 点，(2-3) 23 点

二分ヒープ（binary heap）とは，以下に挙げるヒープ特性（heap property）および形状特性（shape property）を満たす二分木（binary tree）である．

- ヒープ特性：各節点（node）の値はその親（parent）の値以下であること．
- 形状特性：すべての葉（leaf）の深さ（depth）の差が高々1 であり，深い方の葉は左詰めに並んでいること．ここで，ある節点の深さとは，根（root）からその節点までの経路（path）の長さを指す．

以降では，節点番号（node number）i を持つ節点を節点 i と呼ぶ．また，根の節点番号を 1 とする．節点 i の左の子（child）は節点 2i であり，右の子は節点 2i+1 である．図 1 に二分ヒープの例を示す．節点の肩にある数は節点番号を表し，節点内部の数は値を表す．以下の各問に答えよ．

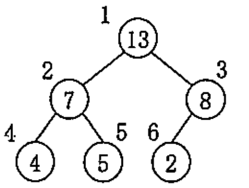


図 1: 二分ヒープの例

(1) 以下の空欄（ア）～（オ）を適切な数式で埋めよ．なお，$\lfloor r \rfloor$は r を超えない最大の整数を表し，$\lceil r \rceil$は r 以上の最小の整数を表す．

木（tree）において，深さの最大値を木の高さ（height）と呼ぶ．n(＞0)個の節点を持つ二分ヒープの高さは $\boxed{\text{（ア）}}$ である．節点 i(2 ≤ i ≤ n)の親は節点$\lfloor i/2 \rfloor$であり，深さ d には高々 $\boxed{\text{（イ）}}$ 個の節点がある．すべての値が異なる場合，最大値は節点 $\boxed{\text{（ウ）}}$ にあり，最小値は節点 $\boxed{\text{（エ）}}$ ～ $\boxed{\text{（オ）}}$ のいずれかにある．

(2) 配列（array）を用いて，n 個の整数（integer）を持つ二分ヒープを構築することを考える．図 2 の ANSI-C 準拠である C 言語プログラムは，配列 data を並び替え，ヒープ特性を持つようにする．並び替え後，data[i]は節点 i の値を持つ．ただし，先頭の値 data[0]は配列を初期化するためのダミー（dummy）であり，29 行目以外では参照されない．また，図 2 では関数 readData(int *h)の定義は省略されている．関数 readData(int *h)は，1 個の整数をポインタ h が指す番地に読み込み，整数を読み込めた場合に 0 を，読み込めなかった場合に 1 を返すものとする．図 3 は，図 1 の二分ヒープを配列に格納した例である．以下の各小問に答えよ．

(2-1) このプログラムの実行が 35 行目に到達したとき，n および data の値が以下の通りであるとする．
n=8, data[1]=4, data[2]=5, data[3]=6, data[4]=3, data[5]=8, data[6]=7, data[7]=1, data[8]=2
この場合，36 行目の処理を終えたときの data[1]～data[8]の値を示せ．

(2-2) 関数 swap の呼び出し回数を T(n)とする．n を固定したもとで，T(n)が最大となるとき，節点の値がどのような移動をするか，木の用語を用いて簡潔に答えよ．

(2-3) T(n)を n に関するオーダー表記（order notation）で示せ．導出過程を示すこと．$\sum_{k=0}^{\infty} \frac{k}{2^k} = 2$ を用いてよい．

```
#include <stdio.h>
#define MAX 1000

void swap(int *x, int *y) { int t=*x; *x=*y; *y=t; }

void heapify(int *h, int i, int n) {
        int left, right, x;

        left=2*i; right=2*i+1;
        if (left<=n && h[left]>h[i]) x=left;
        else x=i;
        if (right<=n && h[right]>h[x]) x=right;
        if (x!=i) {
                swap(&h[x], &h[i]);
                heapify(h, x, n);
        }
}

void recursiveBuild(int *h, int n) {
        int i;

        for (i=n/2; i>0; i--) heapify(h, i, n);
}

int main() {
        int data[MAX+1];
        int i, n;

        n=0; data[0]=-1;          /* data[0]はダミー */
        while (!readData(&i)) {
                n++;
                if (n>MAX) { printf("Out of range.¥n"); return 1; }
                data[n]=i;
        }

        recursiveBuild(data, n);

        return 0;
}
```

図 2: 二分ヒープを構築するプログラム

1	2	3	4	5	6
13	7	8	4	5	2

図 3: 図 1 の二分ヒープを配列に格納した例

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

(1)
次の 2 つの論理式(logical expression)について，以下の各小問に答えよ．

$$f = x_1 \cdot x_2 \cdot \overline{x_0 \cdot x_3} + \overline{x_0} \cdot (x_1 \oplus x_2) \cdot \overline{x_3} + \overline{x_0} \cdot x_1 \cdot \overline{x_2} + \overline{x_1} \cdot x_2 \cdot \overline{x_3}$$
$$g = (x_0 + \overline{x_1}) \oplus (\overline{x_2} + x_3)$$

(1-1) g の論理式を変形することで最小積和形(最簡積和形，minimal sum-of-products expression)を求めよ．途中の導出過程を示すこと．

(1-2) f の出力を与える論理回路(logic circuit)を，NOR ゲートのみを用いて構成し図示せよ．ただし，入力としては，x_0, x_1, x_2, x_3 に加えてそれらの否定(not)も利用できるものとする．ここで，各ゲートへの入力本数(fan-in)に制限はないが，用いるゲート数を最小とし，その中でそれらのゲートへの総入力本数を最小とすること（最小であることの証明は不要である）．

(1-3) g の出力を与える論理回路を NAND ゲートと f のみを用いて構成し図示せよ．ただし，入力としては，x_0, x_1, x_2, x_3, f に加えてそれらの否定も利用できるものとする．ここで，各ゲートへの入力本数に制限はないが，f に追加して用いるゲート数を最小とし，その中でそれらのゲートへの総入力本数を最小とすること（最小であることの証明は不要である）．

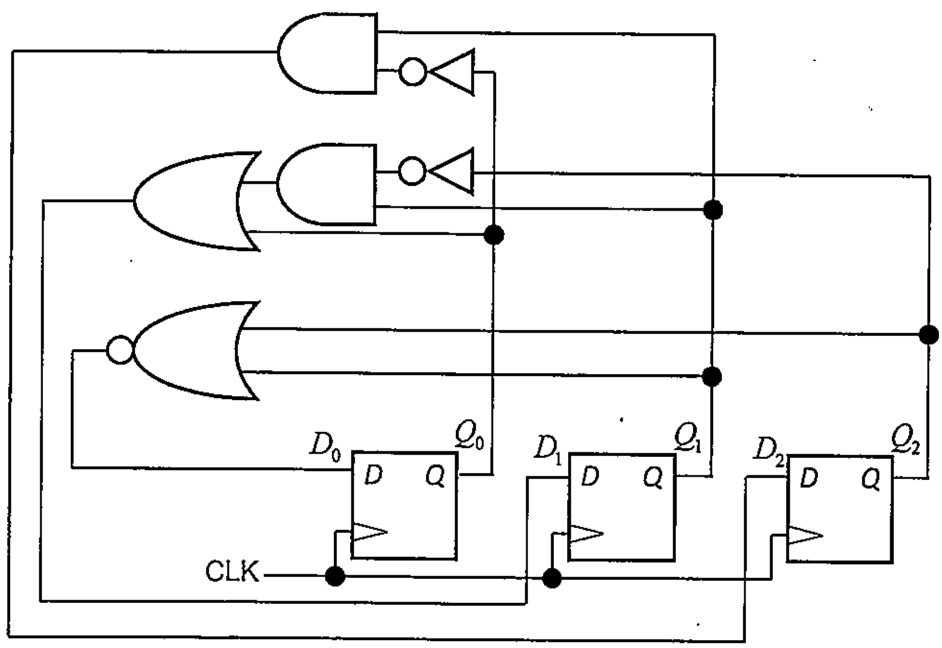
(2)

3 個のエッジトリガ型 D フリップフロップ(edge-triggered D flip-flop)を用いたカウンタ(counter)について，次の各小問に答えよ．

(2-1) 図の回路において，出力(Q_2, Q_1, Q_0)の遷移をカウンタ 1 周期分求めよ．ただし，フリップフロップの初期状態(initial state)は，$(Q_2, Q_1, Q_0)=(0,0,0)$とする．

(2-2) 3 個の D フリップフロップの入力をI_2, I_1, I_0，出力をO_2, O_1, O_0とする．(O_2, O_1, O_0)が(0,0,0), (0,0,1), (0,1,1), (1,1,1), (1,1,0), (1,0,0), (0,0,0), (0,0,1)・・・と遷移するカウンタを設計する．(O_2, O_1, O_0)の初期状態は(0,0,0)とする． 状態遷移表(state transition table), カルノー図(Karnaugh map)を作成して，I_2, I_1, I_0をそれぞれO_2, O_1, O_0の論理式で表し，本カウンタを実現する回路を図示せよ．ただし，論理式は最小積和形で表すこと．用いることができるゲートは，NOT，2 入力 AND，2 入力 OR である．

(2-3) (2-2)の回路にゲートを追加して(2-1)と同じ出力遷移になるような回路を設計したい．Q_2, Q_1, Q_0をO_2, O_1, O_0の論理式で表せ．また，NOT，2 入力 AND，2 入力 OR を用いてよいとき，回路の実現のために追加されるゲートの最小数を求めよ．



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) 10 点，(1-2-1) 5 点，(1-2-2) 5 点，(1-2-3) 5 点，(1-3) 25 点
(2-1) 12 点，(2-2-1) 12 点，(2-2-2) 12 点，(2-2-3) 14 点

(1) 機械語の 1 つの命令(instruction)を，IF(instruction fetch)，ID(instruction decode)，OF(operand fetch)，EX(execution)，MW(memory write)という 5 つのステージを用いて実行する同期型プロセッサを考える．以下の各小問に答えよ．

(1-1)上記のプロセッサにおいて，下記の文中の空欄を埋める適切な用語を選択肢①～⑩から選べ．

プロセッサ高速化技術の一つとしてのパイプライン処理(pipeline processing)は，複数の命令を少しずつずらして同時並行的に実行する方式である．パイプラインを構成するステージでは，以下の処理が行われる．IF ステージでは，(a)のアドレスを用いて主記憶から命令語が読み出される．ID ステージでは，命令語の(b)部を解読し，命令操作が決定される．OF ステージでは，命令語の(c)部によって指定される(d)が取り出される．次に，EX ステージでは，命令操作が実行される．MW ステージでは，実行結果の主記憶ないしはレジスタへの(e)が行われる．

選択肢
①スタックポインタ(stack pointer)　②オペランド(operand)　③即値指定(immediate mode)
④プログラムカウンタ(program counter)　⑤取り出し(load)　⑥格納(store)
⑦アドレス(address)　⑧アドレス変換(address conversion)　⑨分岐命令(branch operation)
⑩命令コード(operation code)

(1-2)上記のプロセッサにおいて，各ステージを独立に処理する機能ブロックの遅延時間(delay time)のみを考え，その他の遅延時間はないものとし，パイプライン処理をクロックサイクル(clock cycle) τ [ナノ秒]([ns])で実行する時に，以下の各問に答えよ．

(1-2-1)プロセッサを設計する際に τ を決定する上での制約について，「遅延時間」という言葉を用いて簡潔に述べよ．

(1-2-2)1 つの命令の実行時間(execution time) [ナノ秒]を τ を用いて示せ．

(1-2-3)パイプライン処理の乱れがない時の m 個の命令 (m は 1 以上の整数) の実行時間[ナノ秒]を τ を用いて示せ．

(1-3)上記のプロセッサにおいて，ロード命令 lw，加算命令 add，減算命令 sub，ストア命令 sw とレジスタ R1〜R5 を用いて，以下の命令 1〜命令 6 を順にパイプライン処理で実行する．なお，命令 6 の実行終了時において，主記憶及びレジスタの内容は，逐次処理(sequential processing)で実行した場合と同じになるものとする．

命令 1:	lw R1, (A)	主記憶アドレス A の内容をレジスタ R1 に格納する．
命令 2:	lw R2, (B)	主記憶アドレス B の内容をレジスタ R2 に格納する．
命令 3:	lw R3, (C)	主記憶アドレス C の内容をレジスタ R3 に格納する．
命令 4:	add R4, R1, R2	レジスタ R1 の内容とレジスタ R2 の内容を加算し，レジスタ R4 に格納する．
命令 5:	sub R5, R3, R4	レジスタ R3 の内容とレジスタ R4 の内容を減算し，レジスタ R5 に格納する．
命令 6:	sw R5, (D)	レジスタ R5 の内容を主記憶アドレス D に格納する．

このパイプライン処理を実行する際に，構造ハザード(structural hazard)とデータハザード(data dependent hazard)と呼ばれるパイプライン処理の乱れが発生する．構造ハザードは，複数の命令のステージがプロセッサ内部の資源(resource)を取り合う場合に生じ，データハザードは，ある命令のステージがそれ以前に開始された命令の演算結果を待つ必要がある場合に生じる．

命令 1〜命令 6 までを最短時間で行うパイプライン処理について，下表の命令 1 の表記例に従い，実行されるステージ名を解答用紙に示せ．パイプライン処理の乱れが発生する場合には，ステージ名の代わりに，構造ハザードには「(s)」，データハザードには「(d)」と記入せよ．なお，上記のプロセッサは主記憶アクセスの資源に制約があるために，主記憶アクセスは同時に一つのステージからしか許されないものとする．その他の資源についてはパイプライン処理のために十分な数があるものとする．

		クロックサイクル																			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
命令	1	IF	ID	OF	EX	MW															
	2																				
	3																				
	4																				
	5																				
	6																				

(2)ハードディスク(hard disk)上に構築されるファイルシステム(file system)に関して，以下の各小問に答えよ．

(2-1)上記のファイルシステムに関して，下記の文中の空欄を埋める適切な用語を選択肢①～⑥から選べ．

ファイル(file)は，データやプログラムを格納するための名前付けられた論理的な単位であり，オペレーティングシステム(OS)はファイルシステムを用いてファイルを保存・管理する．OS は，ブロック(block)と呼ばれるデータの最小単位でファイルシステム上のデータ入出力を行う．

ディスク(disk)上のデータ保持の最小単位は(a)と呼ばれ，ブロックは連続する 1 つ以上の(a)によって構成される．ディスク上には円盤の同心円を表す(b)があり，1 つの(b)上に連続する複数の(a)が配置されている．一般的にディスクは複数枚並べて構成される場合が多く，その各表面を(c)が移動してデータの読み書きを行う．この(c)群を移動することなく読めるディスク面数の(b)の組を(d)と呼ぶ．ディスク上では「(d)番号，(c)番号，(a)番号」によって，データ格納位置を特定することができる．

選択肢

①トラック(track)	②バイトオーダ(byte order)	③シリンダ(cylinder)
④ヘッド(head)	⑤セクタ(sector)	⑥シーケンス(sequence)

(2-2)1 ブロックが 1 セクタで構成され，ブロックサイズ(block size)が，

A．8k[バイト]([byte])

B．40k[バイト]([byte])

で構築された 2 種のファイルシステム A，B を考える．これらは各々，容量 160M[バイト]の 1 台のハードディスク装置上に構築されている．以下の各問に答えよ．ただし，以降では 1k[バイト] = 1000[バイト]，1M[バイト] = 1000000[バイト]として計算し，小数第 2 位を四捨五入して解答せよ．

(2-2-1) ファイルシステム A，B に対し，

（a）2k[バイト]のファイル

（b）500k[バイト]のファイル

でファイルシステム全領域を満たした場合，保存できる最大ファイル数とデータサイズ（ファイルサイズ(file size)×最大ファイル数），ハードディスク装置の容量に対する使用効率（%）をそれぞれ求めよ．

(2-2-2) ファイルシステムA，Bに対し，10M[バイト]のファイルを読み出す場合の平均アクセス時間[ミリ秒] (average disk access time [ms])をそれぞれ求めよ．ただし，このファイルを構成する各ブロックはディスク上に連続して配置されないものとし，平均シーク時間(average seek time)は5.0[ミリ秒]，ディスク回転速度(rotation speed)は6000[回転/分]([rpm])，1トラック当たりの記憶容量(track density)は80k[バイト/トラック]([byte/track])，平均回転遅延時間(average rotational delay)はディスクの1/2回転に要する時間とする．

(2-2-3) (2-2-1)および(2-2-2)の結果をふまえ，ブロックサイズがハードディスク装置の使用効率とファイルの平均転送時間に及ぼす影響について，下記の観点から考察し簡潔に述べよ．

・ファイルサイズとブロックサイズの関係
・ブロックサイズとシーク(seek)回数

4 【選択問題】電磁理論

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

下図に示すように，真空 (vacuum) 中における磁界 (magnetic field) および磁界中での振り子 (pendulum) の運動を考える．直交座標系 (orthogonal coordinates) (x, y, z) において，点 $(x, y, z) = (0, 0, \ell)$ で固定された長さ ℓ の絶縁体 (insulator) の糸があり，その先に質量 (mass) m の導体球 (conducting sphere) が吊るされている．導体球は電荷 (electric charge) q に帯電している．$(y, z) = (a, 0)$ の位置には，x 軸と平行に，無限に長い直線状導体 (infinitely long straight conductor) があり，x 軸負の向きに電流 (electric current) I が流れている．系には，z 軸負の向きに，加速度 (acceleration) g の重力 (gravity) がかかっている．

振り子の振れ角 (swing angle) θ は十分に小さく ($\theta \ll \pi/2$)，$\sin\theta \simeq \theta, \cos\theta \simeq 1$ と近似可能であるとする．また，振り子の振れ幅 (swing width) に対して直線状導体は十分に離れており ($\ell\theta \ll a$)，振り子の振れ幅の範囲内では，電流 I によって生ずる磁界は一様 (uniform) と見なせるものとする．

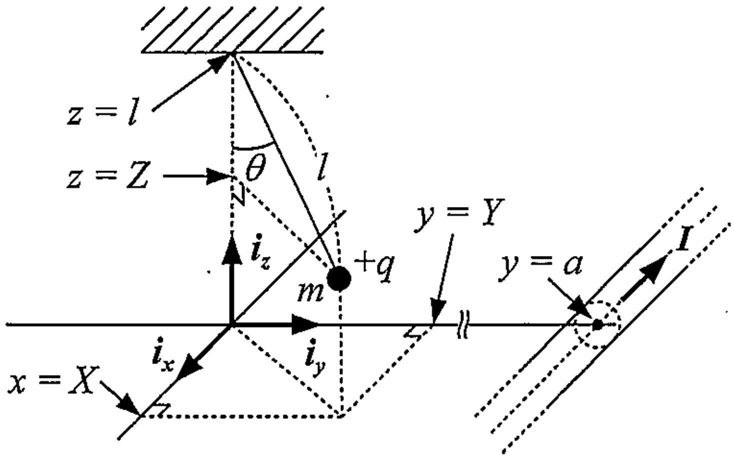
この時，以下の問に答えよ．いずれの問も，導出過程を示すこと．なお，各軸方向の単位ベクトル (unit vector) を $\boldsymbol{i}_x, \boldsymbol{i}_y, \boldsymbol{i}_z$，真空の透磁率 (magnetic permeability) を μ_0，時刻を t とし，糸の質量は無視できるものとせよ．また，導体球の大きさは無視できるものとし，その座標を $(x, y, z) = (X, Y, Z)$，速度ベクトル (velocity vector) を $(\frac{dX}{dt}, \frac{dY}{dt}, \frac{dZ}{dt})$ と表せ．最終的な解答には，θ および独自に設定した変数を用いてはならない (導出過程では用いても良い)．

(1) 電流 I によって生ずる磁界について，原点 (origin) における磁束密度 (magnetic flux density) ベクトルを求めよ．

(2) 運動している導体球が磁界によって受ける力を求めよ．

(3) x 軸方向ならびに y 軸方向における導体球の運動方程式 (equation of motion) を求めよ．

(4) 導体球にある初期位置 (initial position)，初速度 (initial velocity) を与えたところ，導体球の x-y 平面における軌跡 (locus) が原点を中心とする円 (circle) となった．この時，右回り，左回りそれぞれについて，回転角周波数 (rotation angular frequency) を求めよ．なお，回転方向は z 軸負の方向を向いて定義するものとする．



配点: (1) 25 点, (2)25 点, (3) 30 点, (4) 20 点

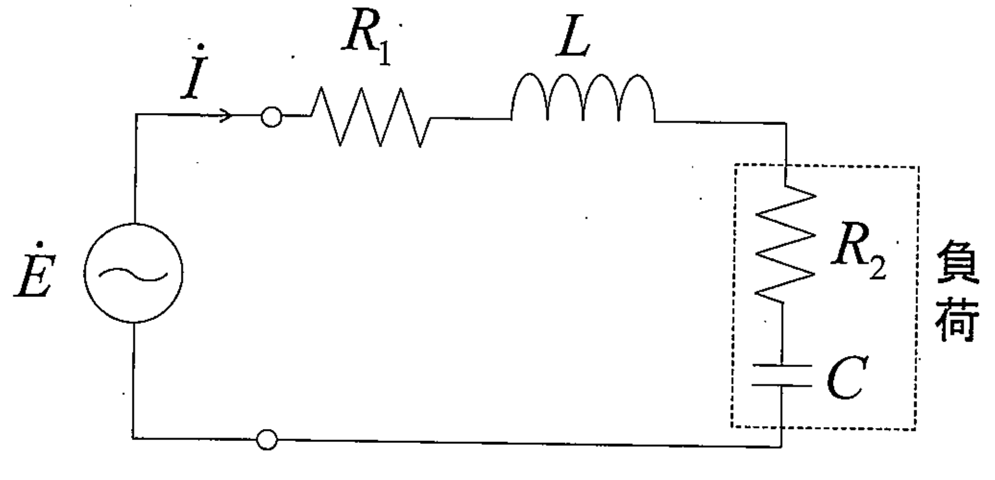
下図の回路(circuit)を考える．角周波数(angular frequency) ω，電圧 $\dot{E}$ の交流電圧源(AC voltage source)から，抵抗(resistance) R_1，インダクタンス(inductance) L の配線(wire)を経て，抵抗 R_2 と容量(capacitance) C よりなる負荷(load)に電力(power)を供給しており，回路は定常状態(steady state)にあるものとする．以下の各問に答えよ．ただし，ωL は X_1，$1/(\omega C)$ は X_2 と表記し，計算過程も記すこと．

(1) 交流電圧源からみた回路の複素合成インピーダンス(complex resultant impedance) $\dot{Z}$ を求めよ．

(2) 回路を流れる複素電流(complex current) $\dot{I}$ ならびにその実効値(effective value) $|\dot{I}|$ を求めよ．

(3) R_2 で消費される電力を最大とするには R_2 がどのような値をとればよいか答えよ．

(4) R_2 に加えて C も自由に変更できるとき，R_2 で消費される最大電力を答えよ．このとき R_1, L からなる複素インピーダンス $\dot{Z}_1$ と R_2, C からなる複素インピーダンス $\dot{Z}_2$ はどのような関係にあるか答えよ．



6 【選択問題】数学解析

配点: (1)50 点, (2-1)30 点, (2-2)20 点

以下の各問に答えよ.

(1) 下記の式を留数定理 (residue theorem) を用いて示せ.

$$\int_{-\infty}^{+\infty} \frac{\cos 2x}{1+2x+2x^2} dx = \pi e^{-1} \cos 1$$

(2) 以下の各小問に答えよ.

(2-1)

$f(x) = x(2\pi - x)\ (0 \leq x \leq 2\pi)$

について余弦級数 (cosine series) に展開 (expansion) せよ.

(2-2)

上の結果と $f(x)$ が偶関数のときの Parseval の等式

$$\frac{1}{\pi}\int_{-\pi}^{\pi} \{f(x)\}^2 dx = \frac{a_0^2}{2} + \sum_{n=1}^{\infty} a_n^2$$

(a_i : フーリエ余弦級数における i 次の係数 $(i = 0, 1, 2, \cdots, n)$)

を用いて,

$$\sum_{n=1}^{\infty} \frac{1}{n^4} = \frac{\pi^4}{90}$$

となることを示せ.

7 【選択問題】離散数学

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

(1) 以下では，N は 1 以上の整数の集合，また，$\mathcal{P}(X)$ は集合 X のべき集合 (power set) をそれぞれ表す．$\lfloor x \rfloor$ は，x を越えない最大の整数．順序対 (x,y) と (u,v) について，「"$(x,y)=(u,v)$" $\Leftrightarrow$ "$x=u$ かつ $y=v$"」と定義する．

このとき，次のように 4 つの集合 X_1, X_2, X_3, X_4 と，それぞれの上で定義された 2 項関係 (binary relation) R_1, R_2, R_3, R_4 を考える．

$$
\begin{aligned}
X_1 &= \mathcal{P}(N), & R_1 &= \{(x,y) \mid x \subseteq y\} \\
X_2 &= N, & R_2 &= \{(x,y) \mid \lfloor x\,/\,5 \rfloor = \lfloor y\,/\,5 \rfloor\} \\
X_3 &= N \times N, & R_3 &= \{((x,y),(z,w)) \mid x \text{ が } z \text{ の倍数，かつ } w \text{ が } y \text{ の倍数}\} \\
X_4 &= N \times N, & R_4 &= \{((x,y),(z,w)) \mid x \text{ と } z \text{ の積が，} y \text{ と } w \text{ の積に等しい}\}
\end{aligned}
$$

以下の各小問に答えよ．

(1-1) R_1, R_2, R_3, R_4 のうち，対称的 (symmetric) でない 2 項関係はどれか．すべて挙げよ．また，それぞれについて対称的でないことを証明せよ (反例を示せ)．

(1-2) R_1, R_2, R_3, R_4 のうち，反対称的 (antisymmetric) である 2 項関係はどれか．すべて挙げよ．また，それぞれについて反対称的であることを証明せよ．

(2) 閉路 (cycle) を持たない有向グラフ (directed graph) $G=(V,E)$ を考える．V は G の頂点集合であり，$E \subseteq V \times V$ は G の有向辺の集合である．次のように定義される 2 項関係 $R_G^i \subseteq V \times V$ $(i=1,2,3,\ldots)$ を考える．

- $R_G^1 = \{(v,v) | v \in V\} \cup E$
- $i = 1,2,3,\ldots$ について，$R_G^{i+1} = \{(u,w) |$ ある頂点 $v \in V$ が存在して $(u,v) \in R_G^i$ かつ $(v,w) \in R_G^i\}$

以下の各小問に答えよ．

(2-1) 閉路を持たない任意の有向グラフ G について，まず，任意の頂点 $v \in V$ について，$(v,v) \in R_G^i$ $(i=1,2,3,\ldots)$ であることを証明せよ．さらに，$R_G^i \subseteq R_G^{i+1}$ $(i=1,2,3,\ldots)$ であることを証明せよ．

(2-2) G として，特に高さ $h \geq 1$ の有向木を考えたとき，$R_G^{i+1} = R_G^i$ となる i の最小値を h を用いて示せ．

(2-3) 閉路を持たないある有向グラフ G について，R_G^1, R_G^2, R_G^3, $\cdots$ を順に構成していったとき，ある整数 $k \geq 2$ に対して，初めて $R_G^{k+1} = R_G^k$ となったとする．このとき，R_G^{k-1} は推移的 (transitive) でないことを証明せよ．

8 【選択問題】情報論理学

配点:(1-1)~(1-4) 各 3 点 (1-5) 6 点, (1-6) 4 点 (1-7-1)~(1-7-3) 各 4 点, (1-8-1) 4 点, (1-8-2) 20 点, (2-1)~(2-6) 各 7 点

論理式 (logic formula) の記号として以下を用いる．$\rightarrow, \vee, \wedge, \neg$ は，それぞれ，含意 (implication)，選言 (論理和)(disjunction, or)，連言 (論理積)(conjunction, and)，否定 (negation, not) の各論理演算子とする．また必要に応じて $\bigwedge_{1\leq i\leq n} F_i$ (あるいは $\bigvee_{1\leq i\leq n} F_i$) の表記を用いる．これは論理式 F_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．なお，$\bigwedge_{1\leq i<j\leq n} F_{ij}$ は $\bigwedge_{1\leq i\leq n} (\bigwedge_{i+1\leq j\leq n} F_{ij})$ を意味する．ここで $\bigwedge_{i+1\leq j\leq n} F_{ij}$ において，$n < i+1$ の場合は真を意味する．

(1) 縦 n マス横 n マス合計 $n\times n$ マス (n は正整数) で構成されるボードの各マスに数字 $1\sim n^2$ を埋めていくパズルを考える．このパズルのゴールは数字 $1\sim n^2$ を重複も欠落もないように埋め，かつ，数字 $1\sim n^2$ をこの順にたどったとき，一筆書きのように，各数字について，その数字のマスとその次の数字のマスが縦，あるいは，横で隣接していることである．なお，数字 1 と数字 n^2 のマスは隣接していなくても良い．

図 1 は $n=3$ のときのゴールの一つである．以降，このパズルを解くための制約式を和積形 (CNF; Conjunctive Normal Form) の命題論理式 (propositional formula) で与えることを考える．なお和積形とは，一つ以上の和項の論理積で表される論理式である．また，和項とは一つ以上のリテラル (literal) の論理和で表される論理式である．ここでリテラルとは，命題変数 (propositional variable) または命題変数の否定を意味する．

9	6	5
8	7	4
1	2	3

図 1: $n=3$ のゴール例

i $(1\leq i\leq n)$ 行　j $(1\leq j\leq n)$ 列のマスを $c(i,j)$ で表記する (行は上から順に $1,\ldots,n$ 行目，列は左から順に $1,\ldots,n$ 列目である)．

マス $c(i,j)$ に数字 k $(1\leq k\leq n^2)$ が入っているとき，かつ，そのときのみ真となる命題変数 x_{ijk} を導入する (i,j,k は正整数)．

以下の各小問に答えよ．

(1-1) 図 1 において命題変数 $x_{117}, x_{221}, x_{135}$ の真偽をそれぞれ答えよ．

(1-2) 命題変数 x_{ijk} の総数は一般にいくつか．n を用いて表せ．

(1-3) 一般に CNF 式 $A(i,j)$ を「マス $c(i,j)$ には，1 以上 n^2 以下の数字が少なくとも一つ入る」制約を表す論理式とする．次式の空欄を埋めよ．

$$A(i,j) = \boxed{}$$

(1-4) 「どのマスについても，1 以上 n^2 以下の数字が少なくとも一つ入る」制約を表す CNF 式 A と等価 (equivalent) な論理式を $A(i,j)$ を用いて示せ．

(1-5) $\neg p\vee\neg q$ は,「p,q がともに真になるということはない」ことを表す論理式である．これを用いて「どのマスについても数字が何か一つ埋まれば，他の数字が（さらに）入ることはない」ことを意味する CNF 式 B を作りたい．次式の空欄を埋めよ．

$$B = \bigwedge_{1\leq i\leq n}\ \bigwedge_{1\leq j\leq n}\ \bigwedge_{1\leq k<h\leq n^2} (\boxed{})$$

(1-6) 「数字 1 はどこかのマスにある」ことを意味する CNF 式 C を示せ．

(次ページに続く)

(1-7) 「各数字 k $(1 \leq k < n^2)$ について，その数字がどこかのマスに入っているならば，そのマスに隣接するマスのいずれかに数字 $k+1$ が入っている」ことを意味する CNF 式 D を作りたい．

まず,「マスの隣接集合」を次のように定義する．

定義：マス $c(i,j)$ の隣接集合 $N(i,j)$ は $\{c(i-1,j), c(i+1,j), c(i,j-1), c(i,j+1)\}$ である．ただし，ボードの範囲外のマスは $N(i,j)$ から除外する．

したがって角のマスや，辺上のマスの隣接集合のサイズはそれぞれ 2，3 になる．

次いで，論理式 $NP(i,j,k)$ を以下のように定義する $(1 \leq i \leq n,\ 1 \leq j \leq n$ かつ $1 \leq k \leq n^2)$.

$$NP(i,j,k) = \bigvee_{c(p,q) \in N(i,j)} x_{pqk}$$

これはマス $c(i,j)$ に隣接する各マス $c(p,q)$ に対して命題変数 x_{pqk} の論理和をとって得られた式である．

以下の各問に答えよ．

(1-7-1) $n=4$ のときの論理式 $NP(2,3,5)$ を命題変数 x_{ijk} と論理演算子を用いて書き下せ．

(1-7-2) 「マス $c(i,j)$ にある数字 k $(1 \leq k < n^2)$ が入っているならば，$c(i,j)$ に隣接するマスのいずれかに数字 $k+1$ が入っている」ことを意味する CNF 式 D_{ij}^k を作りたい．D_{ij}^k と等価な論理式を論理式 $NP(i,j,k)$ と命題変数 x_{ijk} を用いてを示せ．

(1-7-3) CNF 式 D と等価な論理式を D_{ij}^k を用いて示せ．

(1-8) この小問では $n=2$ に固定する．

(命題 P)「図 2 の状態では残りのマスにどのように数字を埋めてもゴールではないこと」

を示したい．

1	3

図 2: $n=2$ での途中状態

(方針) 命題 P を示すには一般に $A \wedge B \wedge C \wedge D \wedge x_{111} \wedge x_{123}$ の CNF 式，あるいは，その CNF 式からうまく選んだ複数の和項の論理積が $\boxed{\qquad\qquad}$ であることを示せば十分である．

以下の各問に答えよ．

(1-8-1) 上記の (方針) に書かれている文の空欄を埋める正しい語を次の三つの中から選べ．

充足不能 (unsatisfiable)　　充足可能 (satisfiable)　　恒真 (valid)

(1-8-2) この方針にそって，命題 P が成り立つことを具体的に示せ．

(2) 以下の一階述語論理式 (first-order predicate logic formula) が正しい場合は○を，正しくない場合は× を答えよ．理由は不要．なお，$\forall, \exists$ はそれぞれ全称作用素 (universal quantifier)，存在作用素 (existential quantifier) である．また，p, q, r は述語記号 (predicate symbol)，x, y は変数記号 (variable symbol) とする．なお，$\equiv$ は左右の論理式が等価であることを意味する．

(2-1) $\forall\, x\ (p(x) \wedge q(x)) \equiv (\forall\, x\ p(x) \wedge \forall\, y\ q(y))$

(2-2) $\forall\, x\ (p(x) \vee q(x)) \equiv (\forall\, x\ p(x) \vee \forall\, y\ q(y))$

(2-3) $\exists\, x\ (p(x) \vee q(x)) \equiv (\exists\, x\ p(x) \vee \exists\, y\ q(y))$

(2-4) $\exists\, x\ (p(x) \wedge q(x)) \equiv (\exists\, x\ p(x) \wedge \exists\, y\ q(y))$

(2-5) $\exists\, x\ \forall\, y\ r(x,y) \equiv \forall\, y\ \exists\, x\ r(x,y)$

(2-6) $\forall\, x\ \forall\, y\ r(x,y) \equiv \forall\, y\ \forall\, x\ r(x,y)$

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

(1) 決定性有限オートマトン (deterministic finite automaton) M_1 を右の状態遷移表 (state transition table) の通り定める．ただし M_1 の初期状態 (initial state) を p，受理状態 (accepting state) を q と v とする．M_1 の入力記号 (input symbol) は 0 と 1 である．記法 $L(M)$ により，オートマトン M が受理する言語を表す．

$L(M_1') = L(M_1)$ 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_1' を求め，状態遷移図 (state transition diagram) の形で示せ．ただし，初期状態と受理状態が分かるように描くこと．

M_1

	0	1
p	w	x
q	q	v
r	q	u
s	x	u
t	w	r
u	q	r
v	v	q
w	t	r
x	s	u

(2) 決定性有限オートマトン M_2 の状態遷移表を右の通り定める．M_2 の入力記号は 0 と 1 である．

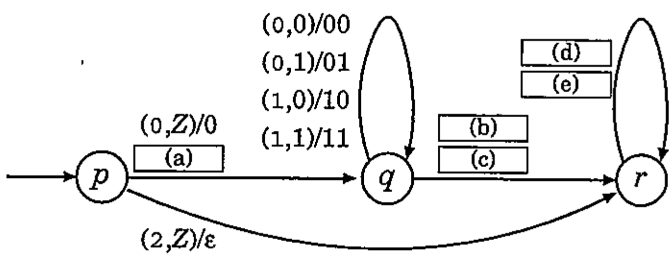
任意の状態から M_2 の動作を開始しても，最終的に状態 r に遷移する入力記号の列で，長さ最小のものを一つ示せ．

M_2

	0	1
p	q	r
q	p	q
r	r	s
s	r	p

(3) 空スタック受理 (acceptance by empty stack) の決定性プッシュダウンオートマトン (deterministic pushdown automaton) M_3 を，$L(M_3) = \{w2w^{\mathrm{R}} : w \in \{0,1\}^*\}$ となるように構成したい．ただし記法 $L(M_3)$ は M_3 が受理する言語を表し，w^{R} は記号列 w の前後を反転 (reverse) した記号列を表す (例: $001101^{\mathrm{R}} = 101100$)．$M_3$ の入力記号 (input symbol) は 0, 1, 2 である．下の図は M_3 の状態遷移図である．

図の空欄 (a)〜(e) それぞれに対し，適切な動作を記号の形で埋めよ．



この状態遷移図の見方とオートマトンの動作を以下に説明する．状態 p が初期状態である．例えば状態 q から q 自身への矢印に添えられた記号 (0,1)/01 は，入力テープ上の文字が 0 でスタックトップの記号が 1 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 にし，スタックに記号列 01 を (逆順の 1, 0 の順で) プッシュする動作を表している．Z はスタックの初期 (底) 記号 (initial symbol) であり，実行開始時には Z のみがスタックに入っている．実行は状態遷移図に従って進む．もし合致する動作が無い場合は動作を停止して，入力を棄却 (reject) する．もしプッシュする記号が ε の場合はスタックに何もプッシュしない．入力を全て読み終えたとき，スタックが空であることが空スタック受理の必要十分条件である．

(次ページへ続く)

(4) 文脈自由文法 (context-free grammar) に関して以下の各小問に答えよ．なお文脈自由文法 G_1 と G_2 を以下の通り定める．

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

- 非終端記号 (non-terminal symbol) の集合 $V_1 = \{ E, T, F \}$
- 終端記号 (terminal symbol) の集合 $T_1 = \{ \mathrm{a}, +, \times, (,) \}$
- 生成規則 (generating rule) の集合 $P_1 = \{ E \to E{+}T,\ \ E \to T,\ \ T \to T{\times}F,\ \ T \to F,\ \ F \to (E),\ \ F \to \mathrm{a} \}$
- 開始記号 (start symbol) $S_1 = E$

$G_2(V_2, T_2, P_2, S_2)$

- 非終端記号の集合 $V_2 = \{ E \}$
- 終端記号の集合 $T_2 = \{ \mathrm{a}, +, \times, (,) \}$
- 生成規則の集合 $P_2 = \{ E \to E{+}E,\ \ E \to E{\times}E,\ \ E \to (E),\ \ E \to \mathrm{a} \}$
- 開始記号 $S_2 = E$

(4-1) 文脈自由文法 G_1 により生成される語で長さ 3 のものを全て示せ．

(4-2) 文脈自由文法 G_1 により生成される次の語 (a), (b) それぞれに対し，導出木 (derivation tree) を示せ．

(a) a + a × a　　　(b) (a + a) × a

(4-3) 文脈自由文法 G_2 が曖昧 (ambiguous) であることを，語 a + a × a を用いて示せ．

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

S は記憶のない情報源（memoryless source）であるとし，その情報源記号集合（source alphabet）を $A = \{a_1, \ldots, a_s\}$ $(s \geq 2)$ とする．各 a_i $(1 \leq i \leq s)$ の生起確率（occurrence probability）を p_i とおく（ただし $p_i > 0$ であるとする）．底を 2 とした情報源 S のエントロピー（entropy）を $H_2(S)$ で表す．通信路記号集合（channel alphabet）として $B_2 = \{0, 1\}$, $B_3 = \{0, 1, 2\}$, $B_4 = \{0, 1, 2, 3\}$ の 3 種類を考え，S をこれらの集合上の系列に情報源符号化（source coding）することを考える．

以下の各問に答えよ．

(1) $s = 8$ とし，各 p_i $(1 \leq i \leq s)$ の値が次のとおりであるとする．

$$p_1 = 1/2, \quad p_2 = 1/8, \quad p_3 = 1/8, \quad p_4 = 1/8,$$

$$p_5 = 1/32, \quad p_6 = 1/32, \quad p_7 = 1/32, \quad p_8 = 1/32$$

以下の各小問に答えよ．

(1-1) 情報源 S のエントロピー $H_2(S)$ を求めよ．

(1-2) S の B_2 上への最適符号化（optimal coding）の復号木（あるいは符号木，code tree）を一つ与えよ．

(1-3) 以下の関数 g は，S の B_4 上への最適符号化を表している．空欄（ア）～（エ）を適切に埋めよ．答えが複数存在する場合，一組だけを記述すればよい．

$$g(a_1) = 0, \quad g(a_2) = 1, \quad g(a_3) = \boxed{\text{（ア）}}, \quad g(a_4) = \boxed{\text{（イ）}},$$

$$g(a_5) = 30, \quad g(a_6) = 31, \quad g(a_7) = \boxed{\text{（ウ）}}, \quad g(a_8) = \boxed{\text{（エ）}}$$

(1-4) S の 2 次拡大（quadratic extension）情報源を $S \times S$ と書く．以下の関数 $N(x)$ $(x = 1, 2, 3, 4, 5)$ は，$S \times S$ の B_4 上への最適符号化において，長さが x に等しい符号語（codeword）の個数を表している．空欄（オ）～（ク）を適切に埋めよ．

$$N(1) = 1, \quad N(2) = \boxed{\text{（オ）}}, \quad N(3) = \boxed{\text{（カ）}}, \quad N(4) = \boxed{\text{（キ）}}, \quad N(5) = \boxed{\text{（ク）}}$$

(2) $s = 5$ とする．$\bar{l}_3$, $\bar{l}_4$ をそれぞれ，S の B_3, B_4 上への最適符号化の平均符号語長（average codeword length）とする．以下の各小問に答えよ．

(2-1) B_3 上への最適符号化が平均符号語長の下界と等しいことを表す次の条件：

$$H_2(S) = (\log_2 3) \cdot \bar{l}_3$$

を満たす $p_1, \ldots, p_5$ を一組与えよ．与えた $p_1, \ldots, p_5$ がこの条件を満たしていることの説明もすること．

(2-2) $p_1 > p_2 > p_3 > p_4 > p_5$ かつ $p_4 + p_5 = 1/100$ とする．さらに，B_3 上への最適符号化と B_4 上への最適符号化の，記号数と平均符号語長から定まる符号化効率が等しいことを表す次の条件：

$$(\log_2 3) \cdot \bar{l}_3 = (\log_2 4) \cdot \bar{l}_4$$

が成り立っているとする．このときの p_3 を求めよ．どのように求めたかも記述すること．$\log_2 3$ はそのまま用いてよい．

11 【選択問題】ネットワーク

配点：(1) 21 点，(2-1) 5 点，(2-2) 30 点，(2-3) 10 点，(2-4) 17 点，(2-5) 17 点

(1) 以下の説明文の空欄 (a) ～ (g) に当てはまる最も適切な語句を選択肢から一つ選び，その番号を答えよ．なお，異なる空欄には異なる選択肢が当てはまる．

コンピュータネットワークは三つの要素 ((a) ， (b) ， (c)) によって構成される． (a) は，コンピュータネットワークにおいて通信の対象となる情報を発生し，また，情報を受け取るためのコンピュータである． (a) は，通信処理の一部も負担する． (b) は，コンピュータネットワーク内部において情報を中継する通信装置である．コンピュータネットワークの機能は階層化されており，その機能や中継の階層におうじて (b) にはさまざまな呼び名がある．例えば，ネットワーク同士を接続して情報を中継するゲートウェイ，ネットワーク層において中継する (d) ，データリンク層において中継する (e) ，物理層において中継する (f) などである． (c) は， (b) 間もしくは (a) と (b) の間をつないで情報を伝達する． (c) の能力は 1 秒間に何ビットの情報を送ることができるかによって測られ， (g) と呼ばれる． (g) の単位は bps (bit per second) である．

選択肢:
1. アーキテクチャ (architecture)，2. アドレス (address)，3. インタフェース (interface)，4. 回線 (link)，
5. グッドプット (goodput)，6. ジッタ (jitter)，7. ストレージ (storage)，8. ソケット (socket)，
9. 帯域幅 (bandwidth)，10. データグラム (datagram)，11. ノード (node)，12. パケット (packet)，
13. ファイアウォール (firewall)，14. ブリッジ (bridge)，15. プロキシ (proxy)，16. プロトコル (protocol)，
17. ホスト (host)，18. リピータ (repeater)，19. ルータ (router)

(2) 図 1 は，ネットワーク中のあるノード s (始点ノード) から他のすべてのノードへの最短ルート (回線コストの和が最小となるルート) を求めるアルゴリズムである．以下の各小問に答えよ．

(2-1) 図 1 のアルゴリズムの名称を答えよ．

(2-2) 図 2 に示すネットワークモデルに対して，図 1 のアルゴリズムを実行した時の，各ステップにおける N，$d(B)$ ～$d(F)$，$p(B)$～$p(F)$ のすべての値を記入し，表 1 を完成させよ．ただし始点ノードを A とする．

(2-3) 図 2 に示すネットワークモデルにおいて，ノード A からノード B，C，D，E，F への最短ルートおよびそのコストをそれぞれ答えよ．

(2-4) コンピュータネットワークにおいて，図 1 のアルゴリズムを用いているルーティングプロトコルの名称を答えよ．また，そのプロトコルにおける各ノードの動作，特に，他ノードとの情報交換方法および図 1 のアルゴリズム実行方法，それぞれを簡潔に説明せよ．

(2-5) コンピュータネットワークにおけるルーティングには，静的ルーティング (static routing) と動的ルーティング (dynamic routing) の二種類が存在する．図 1 のアルゴリズムは動的ルーティングに用いられる．コンピュータネットワークにおける静的ルーティングと動的ルーティングの違いを簡潔に説明せよ．また，静的ルーティングと比較した時の，動的ルーティングの利点・欠点をそれぞれ二つ以上述べよ．

表 1: アルゴリズムの実行ステップ

ステップ	N	$d(B),p(B)$	$d(C),p(C)$	$d(D),p(D)$	$d(E),p(E)$	$d(F),p(F)$
初期状態	{ A }	3, A	5, A	2, A	∞,ϕ	∞,ϕ
1						
2						
3						
4						
5						

記号の定義

$c(i,j)$　ノード i から j に接続する回線のコスト．ノード i から j に回線で直接接続されていない場合，$c(i,j)=\infty$ とする．$c(i,j)=c(j,i)>0$ とする．

$d(v)$　アルゴリズムの実行途中で，それまでに発見された，始点ノードからノード v までの最短ルートのコスト．

$p(v)$　アルゴリズムの実行途中で，それまでに発見された，始点ノードからノード v までの最短ルートにおけるノード v の隣接ノード．

$\mathbf{N}$　アルゴリズムの実行途中で，それまでに最短ルートが発見されたノードの集合．

初期化

- $\mathbf{N} \leftarrow \{s\}$
- すべてのノード v に対して，v がノード s に隣接しているならば，$d(v) \leftarrow c(s,v)$，$p(v) \leftarrow s$，それ以外の場合，$d(v) \leftarrow \infty$，$p(v) \leftarrow \phi$ (ϕ は値が不定であることを意味する)．

ループ (すべてのノードが集合 $\mathbf{N}$ に含まれるまで以下の操作を繰り返す)

- $\mathbf{N}$ に含まれていないノードのうち，ノード s からの最短ルートのコストが最小のノードを w とし，$\mathbf{N}$ に w を加える．w は $p(w)$ が集合 $\mathbf{N}$ に含まれるノード，すなわち「集合 $\mathbf{N}$ に隣接する」ノードから見つかる．
- $\mathbf{N}$ に含まれないすべてのノード v に対して，$d(v)$，$p(v)$ を更新する．すなわち，ノード v までのルートのコストが，新たに $\mathbf{N}$ に追加されたノード w を経由した方が小さい場合は，ノード w 経由とする．式で表した場合，以下のようになる．

$$d(v) \leftarrow \min\left(d(v), d(w)+c(w,v)\right)$$

$$p(v) \leftarrow \begin{cases} w & \text{if} \quad d(w)+c(w,v)<d(v) \\ p(v) & \text{otherwise} \end{cases}$$

図 1: アルゴリズム

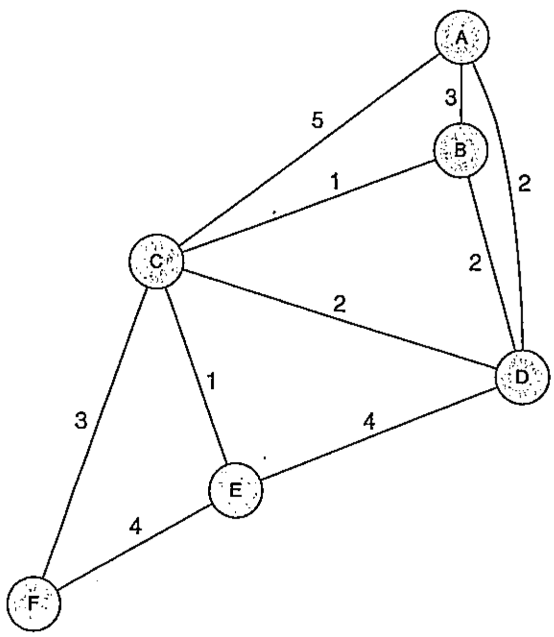


図 2: ネットワークモデル (数字は回線のコスト)