

平成 17 年 8 月 6 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 18 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 19 枚である。
 - 解答用紙は全部で 5 枚である。
 - 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) の解答を
 - 3 枚目 (黄色) の解答用紙には問題 3 (必須問題) の解答を
 - 4 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
 - 5 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、選択問題調査票に、選択した問題の番号 (4～11 から 2 つ) を明記すること。
 - 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。

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

各節点の子の数が2以下で、左の子と右の子を区別して扱う木を2分木という。全ての節点の要素は、1以上の整数値をとる。次の規則に従って、サイズMの配列Aで表現される2分木を考える。

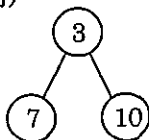
- 各節点には節点番号が付与され、節点番号iを付与された節点を節点iと呼ぶ。根の節点番号を0とする。節点iの左の子には節点番号 $2i+1$ 、右の子には節点番号 $2i+2$ が付与される。
- 配列のインデックスが2分木の節点番号に対応する。つまり、節点iの要素は、配列要素A[i]に格納される。節点iが存在しないとき、配列要素A[i]の値を-1とする。

この2分木に関する以下の各問に答えよ。

- (1) 以下の配列(サイズ7)で表現される2分木を、例にならって図示せよ。なお、この例は、要素が3の節点を根とする節点数3の2分木を表している。

$A[0]=3, A[1]=8, A[2]=4, A[3]=-1, A[4]=5, A[5]=-1, A[6]=13$

(例)



- (2) 次ページのC言語プログラムにおいて、関数insertは、ある規則に従って、要素xを2分木(配列A, 節点数n)に挿入する関数である。関数mainにおいて、36行目を実行し終えた時点の2分木を図示せよ。なお、プログラムにおける各行の行頭の数字およびコロンは、行番号を表している。
- (3) 次ページのC言語プログラム中の関数insertの最悪時間計算量を、2分木の節点数nを用いたオーダー表記で示せ。また、その理由を200字程度で簡潔に説明せよ。ただし、関数insertの引数nは、35行目および36行目のように、 $n=0$ から始まり、関数insertの実行の度に1ずつインクリメントされるものとする。
- (4) 次ページのC言語プログラムにおいて、関数inorderは、2分木の各節点の要素を中順に出力する関数である。空欄(ア)、(イ)を埋めよ。なお、中順に出力するとは、木を左から右へ深さ優先探索でたどる際に、各節点の要素を2回目の訪問時(葉の場合は1回目の訪問時)に出力することを表す。
- (5) 次ページのC言語プログラムにおいて、37行目における`inorder(root, A)`を実行した結果を示せ。

```

1: #include <stdio.h>
2: #define M 100
3: #define root 0
4:
5: void exchange(int n, int *A) {
6:     int i, tmp;
7:     if (n == 0) return;
8:     i = (n - 1) / 2;
9:     if (A[i] > A[n]) {
10:         tmp = A[i]; A[i] = A[n]; A[n] = tmp;
11:         exchange(i, A);
12:     }
13:     return;
14: }
15:
16: int insert(int x, int *A, int n) {
17:     if (n >= M || x <= 0) exit(1);
18:     A[n] = x;
19:     exchange(n, A);
20:     return ++n;
21: }
22:
23: void inorder(int n, int *A) {
24:     if (n >= M) return;
25:     if (A[n] == -1) return;
26:     (左) ;
27:     printf("%d ", A[n]);
28:     (右) ;
29: }
30:
31: int main() {
32:     int A[M];
33:     int i, n;
34:     for (i = 0; i < M; i++) A[i] = -1;
35:     n = 0;
36:     n = insert(3, A, insert(6, A, insert(2, A, insert(10, A, insert(11, A, n)))));
37:     inorder(root, A);
38:     return 0;
39: }

```

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

以下の各問に答えよ。

(1)

2 の補数形式で表されている n bit の 2 つの 2 進数 $A(a_{n-1} a_{n-2} \dots a_0)$ および $B(b_{n-1} b_{n-2} \dots b_0)$ を入力として、加算 ($A+B$) および減算 ($A-B$) の演算を行い、演算結果を $R(r_{n-1} r_{n-2} \dots r_0)$ 、オーバーフローが発生したときのみ OV の値を 1 とするオーバーフロー検出回路付演算器を作成したい。 R も 2 の補数形式とする。具体的には、図 1 のように、 A , B および選択端子 Sel を入力とし、出力を R および OV とする。演算器は選択端子 Sel が 1 のとき加算, 0 のとき減算を実行する。オーバーフローの状態とは、演算の結果、桁あふれが生じ、演算結果が正しくない状態である。また、 a_{n-1} , b_{n-1} , r_{n-1} は最上位ビットを表し、 a_0 , b_0 , r_0 は最下位ビットとする。以下の各小問に答えよ。

(1-1) オーバーフローが発生している状態を、 A と B の符号、演算種別および R の符号の組合せとしてすべて答えよ。

例: (A の符号, B の符号, 演算種別, R の符号) = (正, 負, 加算, 正)

(1-2) OV を表す論理式を、 a_{n-1} , a_{n-2} , \dots , a_0 , b_{n-1} , b_{n-2} , \dots , b_0 , r_{n-1} , r_{n-2} , \dots , r_0 および Sel の最小積和形 (最簡積和形) で表せ。

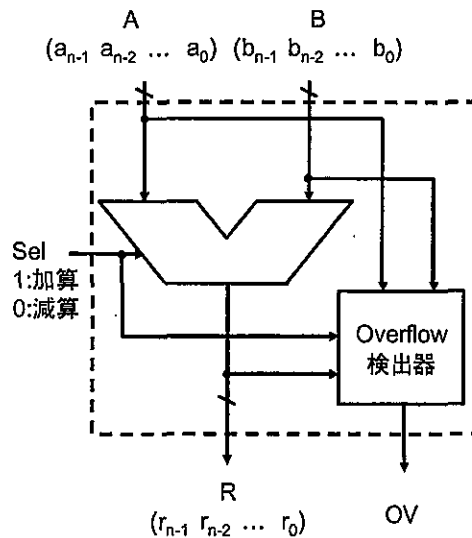


図 1

(2)

図2に示す錠システムを考える。錠は正しいカードキーが読み取り機に挿入されると開く。カードキーには2列6桁の位置に、コードに相当する穴があいている。

読み取り機は同時に2列を走査し、穴の有無に応じて信号を発生して制御回路へ送る。第1列に穴がある場合には X_1 が1となり、穴がない場合には0となる。同様に、第2列に穴がある場合には X_2 が1となり、穴がない場合には0となる。第1桁から順に第6桁まで各桁を読み取るごとに、読み取り機から X_1 , X_2 と同期信号(クロック)が制御回路に与えられるものとする。

制御回路は初期状態より入力信号系列 $(X_1, X_2)=(0, 1), (0, 0), (0, 1), (1, 0)$ が入力される場合に限り、正しい入力コードとして錠を開ける出力信号 Z は1となる。錠は Z が1となるパルスが与えられると一定時間、開錠状態となる。入力信号系列と上記の系列が異なると判明した時点で、制御回路は初期状態に戻る。1桁目、6桁目は必ず両列に穴があるものとし、1桁目、6桁目の読み取り直後に、初期状態となる。初期状態では、 Z は0で錠は閉じている。この制御回路をDフリップフロップを用いて設計する。以下の各小問に答えよ。

(2-1) 状態数4のMealy型順序回路として設計する場合の状態遷移図を示せ。初期状態を S_0 とせよ。また状態数5のMoore型順序回路で設計する場合の状態遷移図も同様に示せ。ただし、Mealy型順序回路は出力が現状態と入力に依存して決まる順序回路、Moore型順序回路は出力が現状態のみに依存して決まる順序回路である。

(2-2) 以降の各小問ではMealy型順序回路として回路設計を進めるものとする。状態遷移表を示せ。

(2-3) 各状態を2ビットの変数(Q_1, Q_0)で符号化する。 Q_1, Q_0 に対応するDフリップフロップの入力 D_1, D_0 を、 Q_1, Q_0, X_1, X_2 の最小積和形(最簡積和形)で表せ。解答の導出過程も示すこと。ただし、初期状態 S_0 は $(Q_1, Q_0)=(0, 0)$ で符号化せよ。

(2-4) 出力 Z を Q_1, Q_0, X_1, X_2 の最小積和形(最簡積和形)で表せ。

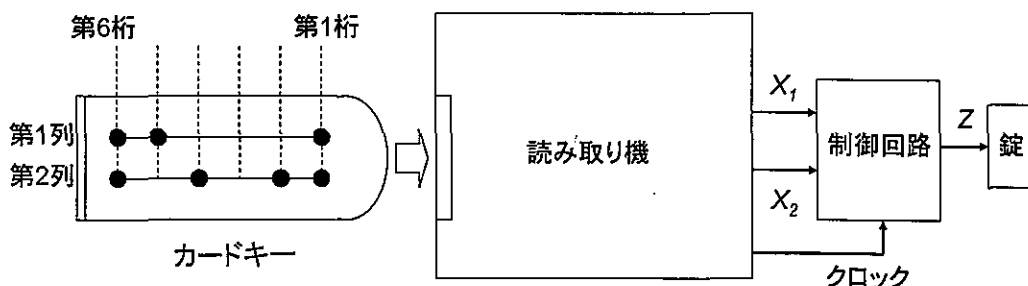


図 2

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

(1) CPU (中央処理装置) は、演算処理部と実行制御部から構成される。図 1 は、CPU の基本構成の一例を示しており、プログラムカウンタ (PC)、命令レジスタ (IR)、メモリアドレスレジスタ (MAR)、メモリデータレジスタ (MDR)、汎用レジスタ ($R1 \sim Rn$)、一時的に演算結果などを記憶するレジスタ (X, Y, Z)、算術論理演算器 (ALU) が、バスを介して信号を転送している。なお、主記憶装置との信号転送は、アドレスバス、データバスを介して行われる。図中の●は、制御ゲートを示しており、制御信号がオンとなった際にデータを送受できる。c は、桁上げ入力である。

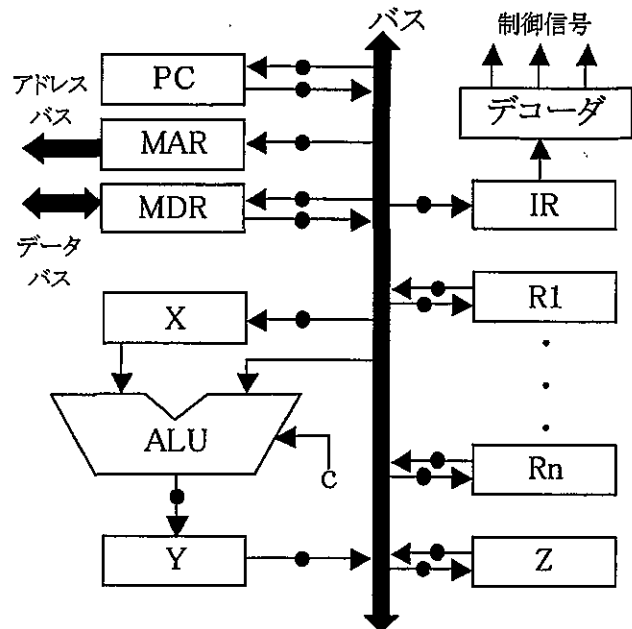


図1 CPUの基本構成

(1-1)

CPU が演算命令を実行する手順を説明する以下の文章として、文中の空欄 (a) ~ (j) に当てはまる適当な語句を下記の選択肢から一つ選び、その番号を答えよ。同じものを複数回用いても良い。

命令フェッチ (命令読み出し) では、(a) の指示する主記憶アドレスから命令を取り出し、(b) に格納する。このことは、次のような複数のステップに分けて実行される。まず、(c) の内容を (d) に転送し、次に主記憶に対して読み出し指示を与える。ここで、主記憶からの読み出し動作が完了し、MDR へデータが格納されるまで、CPU は待機する。なお、この間に PC の内容を ALU により 1 だけインクリメントしている。MDR への格納完了を通知する信号を主記憶から受け取った後に、その内容を (e) に転送する。以下の各手順も、同様に複数のステップで実行される。

命令の解釈では、IR に取り出された命令語の命令コード部を解釈し、実行すべき動作を決定する。オペランドフェッチ (オペランド読み出し) では、取り出された命令語の (f) から、オペランドの格納されているレジスタ、あるいは主記憶内の位置を知り、そこからオペランドを取り出す。この際の処理は、(g) で若干異なり、(h) の場合は、命令語のアドレス部で指定された汎用レジスタの内容がオペランドになる。直接アドレス指定の場合は、アドレスそのものが命令語のアドレス部で指定される。(i) の場合は、命令語のアドレス部で指定された汎用レジスタの内容が、実効アドレスとなる。オペランドフェッチが終了すると、命令コード部で指定された演算を実行する。この後、演算結果を指定されたレジスタ、あるいは主記憶に格納する。主記憶に格納するためには、書き込むアドレスを (j) に格納した後、書き込むデータを MDR に格納し、書き込み命令を実行する。引き続き、命令フェッチに戻って、次の命令が実行される。

選択肢

- ① PC ② MDR ③ アドレス部 ④ レジスタ間接指定 ⑤ MAR
 ⑥ 即値指定 ⑦ インデックス指定 ⑧ IR ⑨ アドレス変換 ⑩ レジスタ
 ⑪ アドレス指定モード (アドレス指定形式) ⑫ ALU ⑬ レジスタ指定

(1-2)

命令を実行する手順の各ステップでの制御信号は、制御ゲートの信号のオン、オフや演算器などの動作を指定するビットパターンにとらえることができる。このビットパターンを、各ステップでのハードウェアユニットの動作を規定する命令とみなすことで、制御論理を構築することができる。このようなビットパターンをマイクロ命令と呼び、これを用いて実行を制御する方式を、マイクロプログラム制御方式という。この際に、例えばレジスタへのデータ格納などの操作（マイクロ操作）ごとに各1ビットを割り当てると、各ステップの動作を記述するのに多数のビットが必要となり、しかもその大部分が0である。そこで、マイクロ命令の語長を以下のように短縮することを考える。図1に示すCPUの基本構成の汎用レジスタ数を4とする。例えば、各レジスタからバスへの出力を R_{out} で表すと、 R_{out} としては、次のような8種類のマイクロ操作が含まれ、これらは同時に実行されることはない。

$$R_{out} = \{PC_{out}, MDR_{out}, R1_{out}, R2_{out}, R3_{out}, R4_{out}, Y_{out}, Z_{out}\}$$

このように同時に実行されることのないマイクロ操作をまとめて、区別できるようにすることで、マイクロ命令の語長を短くできる。図1に示したCPUの基本構成に対するマイクロ操作の例を、表1に示す。なお、表中のNONEは「操作なし」を示している。

表1 マイクロ操作

R_{out}	R_{in}	ALU	R/W	Clear X	Y_{in}	c	WMS
NONE	NONE	NONE	NONE	NONE	NONE	c=0	NONE
PC_{out}	PC_{in}	Add	Read	Clear X	Y_{in}	c=1	WMS
MDR_{out}	MAR_{in}	Sub	Write				
$R1_{out}$	MDR_{in}	AND					
$R2_{out}$	IR_{in}	OR					
$R3_{out}$	$R1_{in}$	XOR					
$R4_{out}$	$R2_{in}$						
Y_{out}	$R3_{in}$						
Z_{out}	$R4_{in}$						
	X_{in}						
	Z_{in}						
レジスタから バスへのデー タ出力	バスからレジ スタへのデー タ入力	ALUの演算	主記憶からの 読み出し、主 記憶への書き 込み	レジスタXの 内容をクリア	レジスタYへ のデータ入力	桁上げ入力 の値	主記憶からの 読み出し動作 完了の待機

この時、命令フェッチは、次の3ステップで示される。

	R _{out}	R _{in}	ALU	R/W	Clear X	Y _{in}	c	WMS
ステップ1	PC _{out}	MAR _{in}	Add	Read	Clear X	Y _{in}	c=1	NONE
ステップ2	Y _{out}	PC _{in}	NONE	NONE	NONE	NONE	c=0	WMS
ステップ3	MDR _{out}	IR _{in}	NONE	NONE	NONE	NONE	c=0	NONE

同様に、表1のマイクロ操作表記をもとに、以下の加算命令に関して、命令解釈後の各ステップを示せ。

- (i) 加算命令（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 R2 に格納）
- (ii) 加算命令（レジスタ R1 の内容で指定される主記憶アドレスに格納されている内容とレジスタ R2 の内容を加算し、レジスタ R2 に格納）

(2) 以下の各小問に答えよ。

(2-1)

仮想記憶を説明する以下の文章として、文中の空欄 (a) ～ (j) に当てはまる適当な語句を下記の選択肢から一つ選び、その番号を答えよ。

仮想記憶とは、計算機に搭載される主記憶の容量よりも大きな (a) を提供するための技法である。実行中のプロセスが参照するアドレスは、(b) と呼ばれ、これに対して主記憶上のアドレスは、(c) と呼ばれる。プロセスの進行に伴って、2 次記憶に格納されている仮想空間の一部が動的に主記憶上に読み出される。このような仮想記憶を実現するには、通常、動的アドレス変換機構と呼ばれるハードウェアを用いて、(b) を (c) へ変換しなければならない。主記憶と 2 次記憶との情報転送はバイト単位ではなく、ある程度の大きさを持つ (d) 単位で行われる。(d) の大きさが固定の時には、(e) と呼び、その時の仮想記憶実現方式を (f) と呼ぶ。一方、(d) の大きさが可変の時には、(g) と呼び、その時の仮想記憶実現方式を (h) と呼ぶ。

(f) においては、主記憶はページ枠と呼ばれる (d) に分割され、仮想アドレス空間も同じ大きさの (e) と呼ばれる (d) に分割される。(e) が主記憶上に存在しない場合、ページフォルト割込みが発生し、当該ページが 2 次記憶から読み出されて主記憶上のページ枠に置かれる。このことを (i) という。また、主記憶上に空きページがない時には、空きを作るために主記憶上の (e) が 2 次記憶に書き込まれる。このことを (j) と呼ぶ。

選択肢

- ① 実アドレス ② 仮想アドレス ③ ブロック ④ アドレス空間 ⑤ ページ
- ⑥ セグメント ⑦ ページング ⑧ セグメンテーション ⑨ ページイン
- ⑩ ページアウト

(2-2)

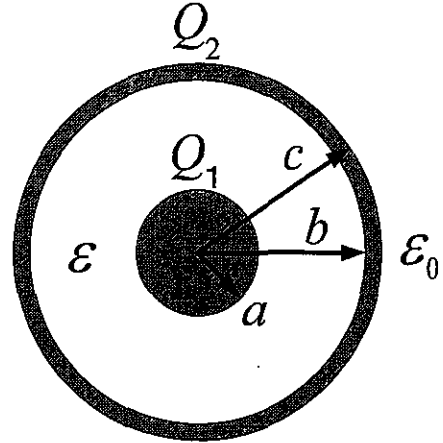
ページングアルゴリズムとして、FIFO (First In First Out), LRU (Least Recently Used) などがあるが、それらの中から適切なアルゴリズムを選択する際、ページフォルト率を基準とするのが一般的である。例えば、仮想アドレス空間の特定のページ参照列に対してアルゴリズムを実行し、ページフォルト数を計算することによってアルゴリズムを評価することができる。

ページ枠の数を 3 とし、以下に示すあるプロセス P のページ参照列に関して、FIFO, LRU を用いた際のページフォルト数と、その導出過程を示せ。ただし、プロセスの実行前には、ページ枠の内容は「空」に初期化されているものとする。

プロセス P のページ参照列 : 0, 1, 2, 3, 0, 2, 4, 3, 1, 0, 3, 4

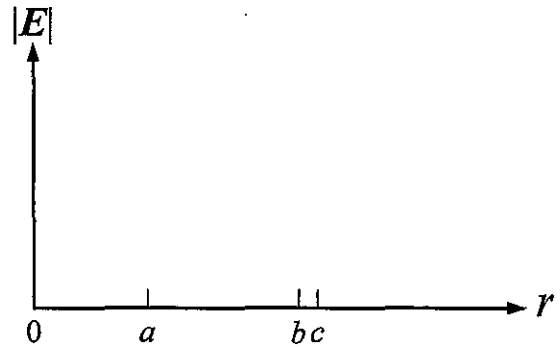
(配点: (1) 25 点, (2) 25 点, (3) 25 点, (4) 25 点)

右の図に示すように、半径 a の内部完全導体球と、内側の半径が b で外側の半径が c の外部完全導体球殻からなる同心球状導体系がある。この導体系の内部の空間は誘電率 ϵ の誘電体によって満たされている。外側の空間は真空（誘電率 ϵ_0 ）である。内部導体球に電荷 $Q_1 (> 0)$ 、外部導体球殻に電荷 $Q_2 (> 0)$ を与えた。以下の各問に答えよ。ただし、座標系として導体球の中心を原点とする球座標系 (r, θ, φ) を用いよ。必要であれば、 r 方向、 θ 方向、および φ 方向の単位ベクトルとして、 i_r 、 i_θ 、 i_φ を用いよ。



- (1) 各部 ($0 \leq r \leq a, a < r < b, b \leq r \leq c, c < r$)

の電界ベクトル E を求めよ。導出過程も示すこと。次に、右の図のように、横軸に r 、縦軸に電界の大きさ $|E|$ をとり、電界の分布を図示せよ。



- (2) 電位の基準（電位 0）を無限遠点 ($r = \infty$) にとり、各部の電位 ϕ を求めよ。導出過程も示すこと。次に、横軸に r 、縦軸に電位をとり、電位の分布を図示せよ。
- (3) 内部導体球および外部導体球殻にそれぞれ与えた電荷 Q_1, Q_2 による電荷密度の分布を求めよ。導出過程も示すこと。次に、横軸に r 、縦軸に電荷密度をとり、電荷密度の分布を図示せよ。
- (4) 外部完全導体球殻を接地し、十分な時間が経過した後の各部の電界ベクトル E 、電位分布 ϕ 、および電荷密度を求めよ。導出過程も示すこと。

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

図1に示すRC回路が，端子対 $a-b$ を持ち，抵抗 R ，容量 C に加わる電圧をそれぞれ $v_R(t)$ ， $v_C(t)$ とするととき，以下の各問に答えよ。

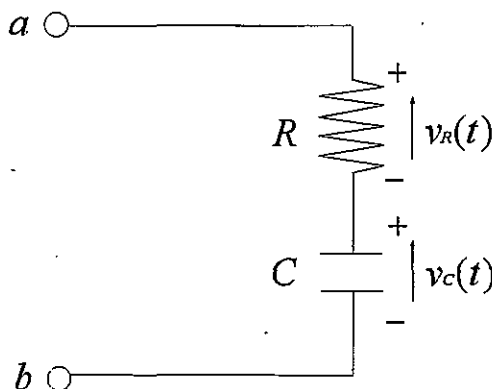


図1

(1) 図1の回路について，以下の各小問に答えよ。

- (1-1) 端子対 $a-b$ 間の角周波数 ω におけるインピーダンス $Z(j\omega)$ を求めよ。次に求めたインピーダンス $Z(j\omega) = R(\omega) + jX(\omega)$ について，角周波数 ω の値を $0 \rightarrow \infty$ まで変化させるときの， $Z(j\omega)$ の軌跡を複素平面上に図示せよ。
- (1-2) 端子対 $a-b$ 間の角周波数 ω におけるアドミタンス $Y(j\omega)$ を求めよ。次に求めたアドミタンス $Y(j\omega) = G(\omega) + jB(\omega)$ について，角周波数 ω の値を $0 \rightarrow \infty$ まで変化させるときの， $Y(j\omega)$ の軌跡を複素平面上に図示せよ。

(2) 図1の端子対 $a-b$ 間に加わっている電圧が $e(t)$ である時，以下の各小問に答えよ。ただし， $t < 0$ の C の蓄積電荷は0のままであったとする。

(2-1) $e(t) = u(t)$ とするととき，電圧 $v_C(t)$ を求めよ。導出過程も示すこと。ただし，

$u(t)$ は以下に示す関数とする。

$$u(t) = \begin{cases} 0 & (t < 0) \\ 1 & (t \geq 0) \end{cases}$$

(2-2) $e(t) = u(t) - u(t-a)$ (ただし $a > 0$)とするととき，電圧 $v_C(t)$ を図示せよ。導出過程も示すこと。ただし， $u(t)$ は(2-1)と同様の関数とする。

(配点: (1) 50 点, (2-1) 25 点, (2-2) 25 点)

以下の各問に答えよ.

(1) $y(t)$ に関する次の微分方程式を, ラプラス (Laplace) 変換を用いて解け.

$$\frac{dy}{dt} + y = t + 2 \int_0^t y(u) du$$

ただし, $y(0) = -1$ とする.

(2) 次の偶関数 $f(x)$

$$f(x) = \begin{cases} 1 - |x|/2 & (|x| \leq 2) \\ 0 & (|x| > 2) \end{cases}$$

に関して以下の各小問に答えよ.

(2-1) $f(x)$ のフーリエ変換 $F(\omega)$ を求めよ.

(2-2) (2-1) で得られたフーリエ変換を利用して,

$$\int_0^\infty \left(\frac{\sin \omega}{\omega} \right)^2 d\omega$$

の値を求めよ.

(配点: (1) 40 点, (2-1) 20 点, (2-2) 20 点, (2-3) 20 点)

(1) 1 個以上の節点をもつ任意の 2 分木が, 次の性質を持つことを帰納法を用いて証明せよ.

節点のうち, 子を 2 つ持つものの数を D , 葉の数を L としたとき, $D = L - 1$.

(2) 巡回路を持たない有向グラフ $G = (V, E)$ (V, E はそれぞれ節点と有向辺の有限の集合) を考える. このグラフのすべての有向辺には左または右のラベルがついている. 各節点 v から出る有向辺のうち, 左のラベルがついているものは高々 1 つであり, 右のラベルがついているものも高々 1 つである. 左のラベルがついている有向辺の行き先の節点を v の左の子という. 右の子も同様に定義される. 子が 0 個の節点を葉とよぶ. また, 1 つの節点の左の子と右の子が一致する場合もある. 例を図 1 に示す. このグラフでは, 左および右のラベルのついた有向辺をそれぞれ点線と実線で表している.

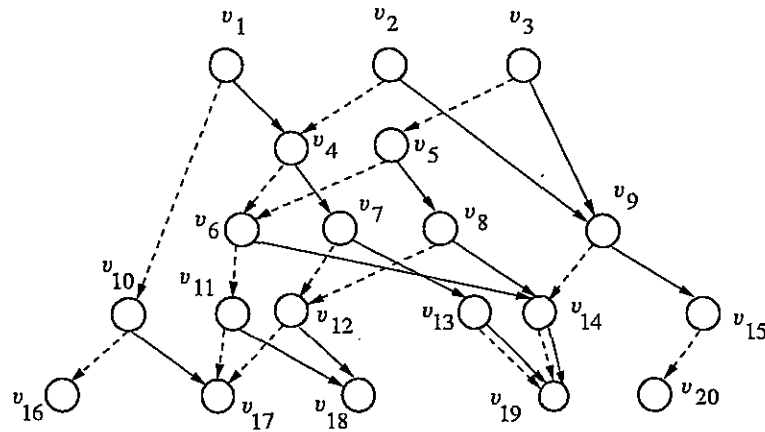


図 1: G の例

このとき, 節点間に以下のように定義される関係 $R \subseteq V \times V$ を導入する. この R は与えられたグラフに対して一意に定まり, かつ, 同値関係であることを示すことができる.

- すべての $v \in V$ に対して, $(v, v) \in R$.
- $v, u \in V$ ($v \neq u$) の一方または両方が葉の場合, $(v, u) \notin R$.
- $v, u \in V$ ($v \neq u$) がともに葉でない場合, $(v, u) \in R$ であるのは, 次の 2 つの条件 (i)(ii) を満足するとき, かつその時に限る.
 - (i) v, u の左の子は両方とも存在しないか, あるいは, それぞれの左の子 v^l, u^l が存在して, $(v^l, u^l) \in R$ である.
 - (ii) v, u の右の子は両方とも存在しないか, あるいは, それぞれの右の子 v^r, u^r が存在して, $(v^r, u^r) \in R$ である.

以下の各小問に答えよ.

(2-1) 図 1 で示されたグラフについて, $(v_{11}, v_{12}) \in R$ が成り立つことを簡潔に示せ.

(2-2) 図 1 で示されたグラフについて, 2 個以上の節点を含む同値類をすべて示せ.

(2-3) n を 1 以上の整数とする. 節点数が n であり, 同値類の個数も n であって, 連結であるようなグラフ G の例を示せ.

(配点: (1-1) 10 点, (1-2) 6 点, (1-3) 8 点, (1-4) 8 点, (1-5) 8 点, (1-6) 10 点,
(2-1) 18 点, (2-2) 27 点, (2-3) 5 点)

(1) Java の インスタンスやクラスに関する述語 S, P, A を次のように定義する.

「インスタンス x がインスタンス y と同一クラスである」ことを表す述語を $S(x, y)$ とする.

「インスタンス x のクラスがインスタンス y のクラスの直接の親クラスである」ことを述語 $P(x, y)$ で表す.

「インスタンス x のクラスがインスタンス y のクラスの先祖クラスである」ことを述語 $A(x, y)$ で表す.

これらの述語の定義を用いて以下の各小問に答えよ. なお, $\wedge, \vee, \neg, \rightarrow, \forall, \exists$ は, それぞれ, 論理積, 論理和, 論理否定, 含意, 全称作用素, 存在作用素を表す.

(1-1) 閉論理式 B を $\forall x \forall y \forall z ((A(x, y) \wedge P(y, z)) \rightarrow A(x, z))$ とおく. B の内容を上記 S, P, A の説明のような文章で記述せよ.

(1-2) 「インスタンス x がインスタンス y と同一クラスであれば y も x と同一クラスである」ことを表す閉論理式 C を作成せよ.

(1-3) 次の「」の内容を表す閉論理式 D を記述せよ. 「以下が成り立てばインスタンス x はインスタンス z と同一クラスである. 『インスタンス x がインスタンス y と同一クラスであり, かつ, インスタンス y がインスタンス z と同一クラスである.』」

(1-4) 次の「」の内容を表す閉論理式 E を記述せよ. 「以下が成り立てばインスタンス z のクラスがインスタンス x のクラスの直接の親クラスである. 『インスタンス x がインスタンス y と同一クラスであり, かつ, インスタンス z のクラスがインスタンス y のクラスの直接の親クラスである.』」

(1-5) 次の「」の内容を表す閉論理式 F を記述せよ. 「以下が成り立てばインスタンス x のクラスがインスタンス z のクラスの先祖クラスである. 『インスタンス x のクラスがインスタンス y のクラスの直接の親クラスであり, かつ, インスタンス y のクラスがインスタンス z のクラスの直接の親クラスである.』」

(1-6) b を定数記号とする. 「あるインスタンス x が存在して, x のクラスはインスタンス b のクラスの先祖クラスである」ことを表す閉論理式 G を記述せよ.

(2) (1-6) で用いた定数記号 b の他に, 定数記号 a, c, d, e を導入し, (1) で定義した C, D, E, F, G を用いて閉論理式 H を次のように定義する.

$$H = ((C \wedge D \wedge E \wedge F \wedge S(b, a) \wedge S(c, a) \wedge P(d, c) \wedge P(e, d)) \rightarrow G)$$

$\neg H$ が充足不能であることを導出原理を用いて示すことにより, H が恒真であることを示したい. 以下の各小問に答えよ.

(2-1) $\neg H$ の冠頭標準形かつ和積形 (述語 A, S, P とそれらの否定からなる和項の積) である論理式を示せ. 導出過程は明記しなくてもよい.

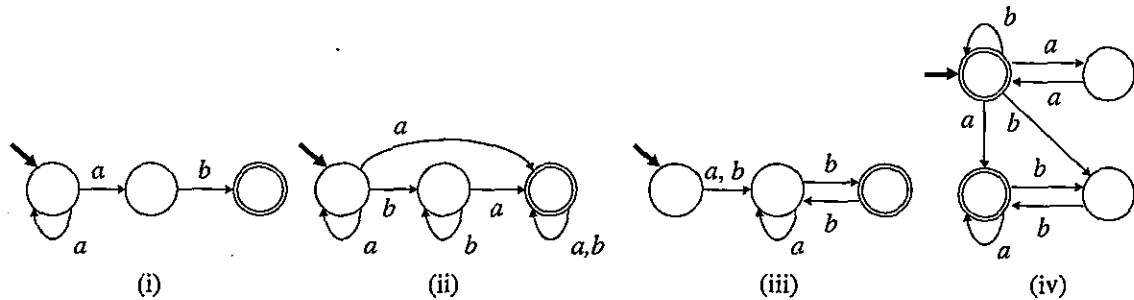
(2-2) 導出原理を用いて空節を導出することにより, $\neg H$ の充足不能性を示せ.

(2-3) (2-2) の空節導出における b の先祖クラスを答えよ.

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

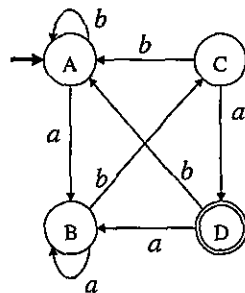
(1) 有限オートマトン (finite automaton) に関する以下の各小問に答えよ。ただし、入力記号の集合を $\{a, b\}$ とする。

(1-1) 下の (i), (ii), (iii), (iv) は、4つの有限オートマトンの状態遷移図 (state transition diagram) を表している。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、2重丸で表されている状態が受理状態である。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認識する言語 (language) を表している正規表現 (regular expression) を、1~8の中から1つ選べ。



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

(1-2) 下の状態遷移図で表される決定性有限オートマトンを M とする。



(1-2-1) M が受理する長さ 4 以下の語 (word) をすべて書け。

(1-2-2) M と等価な決定性有限オートマトンで、状態数が 3 のものは存在しない理由を簡潔に示せ。

(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 は非終端記号の有限集合, T は終端記号の有限集合, P は生成規則の有限集合, S は始記号である。

(2-1) 文脈自由文法 $G_1 : (\{S, A\}, \{d, +, *\}, P_1, S)$ が生成する長さ 5 の文 (sentence) を全て示せ。ただし, P_1 は以下の通りとする。

$$\{S \rightarrow d, S \rightarrow SSA, A \rightarrow +, A \rightarrow *\}$$

(2-2) 文脈自由文法 $G_2 : (\{S, A\}, \{0, 1, 2, 3, +, *\}, P_2, S)$ について考える。ただし, P_2 は以下の通りとする。

$$\{S \rightarrow 0, S \rightarrow 1, S \rightarrow 2, S \rightarrow 3, S \rightarrow SSA, A \rightarrow +, A \rightarrow *\}$$

下の文は, G_2 が生成する文の 1 つである。

$$1032 * + +$$

この文について, 対応する導出木 (derivation tree) を示せ。

(2-3) 下の再帰アルゴリズム $\text{calc}(t)$ は, 文脈自由文法 G_2 が生成する文に対する導出木を t とした場合, 出力として整数値を返す。アルゴリズム中の演算子 $+$, $*$ は, 整数集合上の加算, 乗算を表す。また, $\text{left}(t)$, $\text{center}(t)$, $\text{right}(t)$ は, 木 t の根が 3 個の子を持つ場合に, 左の子を根とした部分木, 中央の子を根とした部分木, 右の子を根とした部分木を表す。

```

module calc( $t$ )
  if  $t$  が葉を 1 つしか持たない
  then
    return  $t$  の葉の記号が表す値
  else
    if  $\text{right}(t)$  の葉の記号が  $+$  である
    then
      return ( $\text{calc}(\text{left}(t)) + \text{calc}(\text{center}(t))$ )
    else
      return ( $\text{calc}(\text{left}(t)) * \text{calc}(\text{center}(t))$ )

```

(2-3-1) t を (2-2) で求めた導出木としたときに, $\text{calc}(t)$ が返す値を答えよ。

(2-3-2) G_2 が生成する文で, 以下の 2 条件を同時に満たすものを 1 つ示せ。

- 長さが 15 以下である。
- 導出木を t とした場合, $\text{calc}(t)$ が返す値が 100 になる。

(配点: (1)20 点, (2-1)12 点, (2-2)12 点, (3-1)12 点, (3-2)12 点, (4-1)16 点, (4-2)16 点)

元 (n, k) 線形符号 C が, ハミング重み (以下, 単に重みという) 奇数の符号語を含むとする. このとき, C の偶部分符号 C_e は,

$$C_e = \{\bar{v} \mid \bar{v} \in C, \text{かつ} \bar{v} \text{の重みは偶数}\}$$

と定義される. また, C の拡大符号 C_{ex} は, 次式で定義される.

$$C_{\text{ex}} = \{\text{ex}(\bar{v}) \mid \bar{v} \in C\}$$

ここで, 2元ベクトル $\bar{v} = (v_1, v_2, \dots, v_n)$ に対して, $\text{ex}(\bar{v})$ は,

$$\text{ex}(\bar{v}) = (v_1, v_2, \dots, v_n, v_1 + v_2 + \dots + v_n)$$

と定義される成分数 $(n+1)$ の 2元ベクトルである.

最小重み d が奇数の 2元 (n, k) 線形符号 C に対して, 次の各問に答えよ.

1) 符号 C の生成行列として次の条件を満たす行列 G が存在することを, C の任意の生成行列から行基本操作により条件を満たす行列が得られることを示すことにより, 証明せよ.

行列 G の条件) G の第 j 行 ($1 \leq j \leq k$) の行ベクトルを \bar{g}_j と書くとき, 行ベクトルは以下の (a), (b) を満たす.

(a) \bar{g}_1 の重みは奇数.

(b) \bar{g}_j ($2 \leq j \leq k$) それぞれの重みは偶数.

2) C の偶部分符号 C_e について, 以下の小問に答えよ. 結果だけを書けばよい.

-1) C_e の生成行列を, 問 (1) の行列 G の行ベクトルを用いて表せ.

-2) C_e の符号化率を n, k を用いて表せ.

3) C の拡大符号 C_{ex} について, 以下の小問に答えよ. 結果だけを書けばよい.

-1) C_{ex} の符号化率を n, k を用いて表せ.

-2) C_{ex} の最小重みを, C の最小重み d を用いて表せ.

4) 符号の性能評価に関して以下の小問に答えよ.

-1) 通信路の効率的利用の重要な尺度として符号化率がある. 符号化率という観点で比べたとき, 符号 C_{ex} と C_e では, どちらが優れているかを説明せよ.

-2) 同一の最小重みをもつ符号の誤り訂正能力を比較するとき, 一般に, 最小重みの符号語数が少ない符号の方がよいことが知られている. まず, C の重み w の符号語数を A_w とするとき, C_{ex} の重み $2t$ は $n/2$ 以下の正整数) の符号語数が $A_{2t-1} + A_{2t}$ であることを示せ. 次に, C_e の重み $2t$ の符号語数 A_w ($0 \leq w \leq n$) を用いて表せ. そして, C_{ex} と C_e の最小重みが同一の場合に, 最小重みの符号語数という観点で比べたとき, 符号 C_{ex} と C_e では, どちらが優れているかを説明せよ.

(配点: (1-1)10点, (1-2)20点, (2)16点, (3-1)20点, (3-2-1)10点, (3-2-2)10点, (3-2-3)14点)

(1) 以下の図は、情報通信ネットワークにおけるプロトコルの階層化モデルとして、OSI 参照モデルを示したものである。以下の各小問に答えよ。

OSI 参照モデル

(A) アプリケーション層
(B) プレゼンテーション層
(C) セッション層
(D) トランスポート層
(E) ネットワーク層
(F) データリンク層
(G) 物理層

(1-1) 以下の a ～ e に示す、情報通信ネットワークにおける機能が、OSI 参照モデルのどの層の機能に相当するかを(A)～(G)から選べ。

- a. イーサネット上で多重アクセス制御を行う機能
- b. 仮想端末、ファイル転送などの機能
- c. ルータやホスト間を接続する回線上でフレームを誤りなく送受信する機能
- d. ホスト間に回線やノードが存在する場合に、どの経路を通して送受信を行うかを決定する機能
- e. 物理的な通信回線を用いてビット列を送受信する機能

(1-2) 情報通信ネットワークにおける機能を OSI 参照モデルのように階層化することの利点と欠点をそれぞれ2点ずつ挙げよ。

(2) OSI 参照モデルのネットワーク層における情報交換原理には、大別して(ア)パケット交換原理と(イ)回線交換原理がある。以下の a ～ h の文章がそれぞれ(ア)(イ)のどちらをより適切に説明しているかを答えよ。

- a. 情報を受信した際に到着順序が入れ替わっていることがある
- b. 遅延が小さく、かつ、その変動がない通信が実現できる

- c. 輻輳時に通信要求が拒否されることがある
- d. エンドホスト間の通信品質がネットワークの輻輳状態によって変動する
- e. 送信する情報がバースト(間欠)的に発生する場合に適している
- f. いったん通信が開始されると、終了まで通信品質を維持することができる
- g. ストアアンドフォワード方式によって情報が運ばれる
- h. 送信する情報が大きい場合には、一定の大きさに分割して送信する

(3) イーサネットに関する以下の各小問に答えよ。

(3-1) イーサネットで用いられている CSMA/CD 方式は、純 ALOHA 方式を拡張したものである。以下の a および b に示す拡張機能のそれぞれについて、その内容および利点を簡潔に述べよ。

- a. 搬送波検知 (Carrier Sense)
- b. 衝突検出 (Collision Detection)

(3-2) イーサネットの規格においては、回線速度、ステーション(コンピュータ)間最大距離、システム最大距離、最小/最大フレーム長などが規定されている。

今、2 台のステーション A および B が同軸ケーブルで接続されていて、A-B 間のケーブル長は L [km]、回線速度を C [Mbps] とする。なお、ケーブル内を信号が伝播する速度を 2.0×10^8 [km/sec] とし、他にはステーションは接続されていないものとする。以下の(3-2-1)～(3-2-3)に答えよ。

(3-2-1) A が B に向かってフレームを送信する時、フレームの先頭が B に到着した時に A が送信し終わっているビット数を上記記号を用いて表せ。ただし、フレームの先頭が B に到着した時点では、A はフレーム全体を送信し終えていないものとする。

(3-2-2) A および B が互いにフレーム送信を行う場合において、送信タイミングによっては、フレームの衝突が発生したにもかかわらず、それが検出されないままフレーム送信が終了することがある。そうならないための、最小フレーム長 F_{min} [bits] が満たすべき条件を示せ。

(3-2-3) 最小フレーム長を固定したままイーサネットの回線速度を大きくすると、イーサネットの規格にどのような制約が生じるかを、(3-2-2)の結果を使って説明せよ。