

平成 23 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 24 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 枚である。
- 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 (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 (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, 「問」の欄には対応する問題番号 (1~7 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 (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。

配点 : (1-1) 10 点, (1-2) 5 点, (1-3) 15 点, (1-4) 14 点, (1-5) 10 点
(2) 10 点, (3-1) 10 点, (3-2) 10 点, (3-3) 16 点

図 1 は ANSI-C 準拠である C 言語のプログラムである。28 行目, 29 行目のようにデータを配列 (array) で表現し, 同一の添字 (index) i ($i=0, 1, \dots$) の $key[i]$ と $label[i]$ の対 (pair) を一個のデータとみなす。このプログラムは, key と $label$ で表されるデータを key の要素 (element) の値 (value) に従って整列 (sort) するプログラムである。以下の各問に答えよ。

(1) プログラムの 16 行目および 18 行目において

ア を $i=a; i<b; i++$

イ を $j=b; j>i; j--$

とする。このプログラムを「プログラム 1」と呼ぶ。以下の各小問に答えよ。

- (1-1) プログラム 1 は, データを key の要素の値に従って昇順 (ascending order), 降順 (descending order) のどちらに整列するか答えよ。
- (1-2) プログラム 1 で実現されている整列アルゴリズム (sorting algorithm) は一般に何と呼ばれているか名称を答えよ。
- (1-3) プログラム 1 を実行したとき, 19 行目の比較判定 (comparison test) の回数の総数を整列要素数 n を用いて答えよ。導出過程も示すこと。
- (1-4) key の要素の値が同じデータ対に対して整列前のデータの並び順の前後関係が整列後も維持されるとき, 安定な (stable) 整列アルゴリズムという。プログラム 1 で実現されている整列アルゴリズムが安定かどうかを理由も含めて説明せよ。
- (1-5) プログラム 1 を 34 行目まで実行したときの $key[0] \sim key[11]$ と $label[0] \sim label[11]$ の値を答えよ。28 行目や 29 行目の表記のようにコンマ区切りで回答してよい。

(2) ア を $i=b; i>a; i--$

イ を $j=a+1; j<=i; j++$

とする。このプログラムを「プログラム 2」と呼ぶ。プログラム 2 は, データを key の要素の値に従って昇順, 降順のどちらに整列するか答えよ。

(3) ア を $i=a; i<b; i+=skip$

イ を $j=b; j>i; j--$

とする。このプログラムを「プログラム 3」と呼ぶ。以下の各小問に答えよ。

- (3-1) プログラム 3 の実行において, 24 行目の printf 関数が初めて呼び出されるとき, 出力結果を答えよ。

(3-2) プログラム 3 は、プログラム 1 と比べて、16 行目の for ループの 2 巡目の実行における 19 行目の比較判定の回数を何回削減できるか答えよ。

(3-3) プログラム 3 において、整列要素が 28 行目、29 行目で与えられたデータではなく、昇順もしくは降順に整列済みの n 個のデータに変更したときを考える。なお、 n 個のデータの key の要素の値は互いに異なるものとする。この変更したプログラムを 34 行目まで実行したとき、昇順および降順のそれぞれの場合について、19 行目の比較判定の回数の総数を整列要素数 n を用いて答えよ。

```

1  #include <stdio.h>
2
3  void swap(int x, int y, int *key, int *label){
4      int temp;
5      temp=key[x];
6      key[x]=key[y];
7      key[y]=temp;
8
9      temp=label[x];
10     label[x]=label[y];
11     label[y]=temp;
12 }
13 void sort(int a, int b, int *key, int *label){
14     int i, j, skip=1;
15
16     for(  ){
17         skip=b-i;
18         for(  ){
19             if(key[j] < key[j-1]){
20                 swap(j, j-1, key, label);
21                 skip=j-i;
22             }
23         }
24         printf("%d\n", skip);
25     }
26 }
27 int main(){
28     int key[]={1,1,2,5,4,5,2,6,3,4,3,6};
29     int label[]={1,2,2,2,1,1,1,2,2,2,2,2};
30     int n=12;
31
32     sort(0, n-1, key, label);
33
34     return 0;
35 }

```

図 1 プログラム

配点：(1-1) 20 点, (1-2) 18 点, (1-3) 12 点, (2-1) 28 点, (2-2-1) 12 点, (2-2-2) 10 点

(1) 計算機における数の表現および算術演算 (arithmetic operations) に関する以下の各小問に答えよ。なお、 $(X)_r$ は、 X が r 進数 (base- r number) で表された数値であることを示す。また、「最上位ビット (most significant bit) を符号ビット (sign bit) とする N ビットの 2 の補数表現 (2's complement representation) で表されたビット列 (bit stream)」を、単に「 N ビット 2 の補数表現」と記し、ビット列においては、左を上位ビット (more significant bits)、右を下位ビット (less significant bits) とする。

(1-1) 以下の各設問に答えよ。

(1-1-1) $(11110011)_2$ を 10 進数 (decimal number) で表せ。

(1-1-2) $(9A)_{16}$ を 10 進数で表せ。

(1-1-3) $(1111000010011010)_2$ を 16 進数 (hexadecimal number) で表せ。

(1-1-4) $(24)_{10}$ を 2 進数 (binary number) で表せ。

(1-1-5) $(-24)_{10}$ の 8 ビット 2 の補数表現を示せ。

(1-2) $(91)_{10} - (85)_{10} = (6)_{10}$ の減算 (subtraction) を考える。 $(91)_{10}$ および $(-85)_{10}$ の 8 ビット 2 の補数表現を示し、これらのビット列と符号無し 8 ビット加算器 (unsigned 8-bit adder) を用いて、 $(91)_{10} - (85)_{10} = (6)_{10}$ の減算が行えることを示せ。但し、加算における最上位ビットからの桁上げ (carry) の扱いを明記すること。

(1-3) $(10)_{10} \div (3)_{10} = (3)_{10} \cdots (1)_{10}$ の正数除算を引き戻し法 (回復型除算) (restoring division) で行うことを考える。ここで、 $(3)_{10} \cdots (1)_{10}$ は、商 (quotient) が 3、余り (remainder) が 1 であることを示す。除数 (divisor) 3 を 3 [bit] 左シフトした状態から計算を開始し、加算には符号無し 8 ビット加算器を用いる。計算過程を示した以下の説明の空欄にあてはまる適切な数値または式を、下記の選択肢から選んで記号を答えよ。同じ記号を複数回用いてよい。なお、最下位ビット (least significant bit) を第 0 ビットとする。また、以下の説明では、「符号無し 8 ビット加算器によるビット列の加算」を単に「加算」と記す。

<pre> 00001010 +) 11101000 ----- 11110010 +) 00011000 ----- 00001010 </pre>	<p>$(10)_{10}$ の 8 ビット 2 の補数表現。</p> <p>① $(-3)_{10} \times 2^3$ の 8 ビット 2 の補数表現を加算</p> <p>(a) なので、商の第 (b) ビットは (c)。</p> <p>$(3)_{10} \times 2^3$ の 8 ビット 2 の補数表現を加算。</p>
<pre> +) 11110100 ----- 11111110 +) 00001100 ----- 00001010 </pre>	<p>② ①を 1 [bit] 右シフトしたビット列を加算。</p> <p>(d) なので、商の第 (e) ビットは (f)。</p> <p>$(3)_{10} \times 2^2$ の 8 ビット 2 の補数表現を加算。</p>
<pre> +) 11111010 ----- 00000100 +) 11111101 ----- 00000001 </pre>	<p>③ ②を 1 [bit] 右シフトしたビット列を加算。</p> <p>(g) なので、商の第 (h) ビットは (i)。</p> <p>③を 1 [bit] 右シフトしたビット列を加算。</p> <p>(j) なので、商の第 (k) ビットは (l)。</p>

【選択肢】

(ア) 0, (イ) 1, (ウ) 2, (エ) 3, (オ) 計算結果 < 0, (カ) 計算結果 ≥ 0

(2) 仮想記憶 (仮想メモリ; virtual memory) に関して以下の各小問に答えよ。

- (2-1) 次の説明文を読み、空欄 (a) ~ (d) にあてはまる最も適切な語句を下の (ア) ~ (サ) の選択肢から選んで記号で答えよ。また、空欄 ① ~ ④ には適切な数値を埋めよ。ただし、1 [Kbyte] = 2^{10} [byte], 1 [Mbyte] = 2^{20} [byte], 1 [Gbyte] = 2^{30} [byte] とする。

仮想記憶は、主記憶 (メインメモリ; main memory) と二次記憶 (secondary storage) (補助メモリ (auxiliary memory) ともいう。) を用いて、計算機で実行されるプロセスに対して、その計算機の主記憶のもつ (a) よりも大きな (b) を提供する。代表的な仮想記憶の実現手法としては、主記憶と二次記憶の間のデータ転送を固定長のブロック (block) 単位で行う (c) と、可変長のブロック単位で行う (d) などがある。

ここで、実アドレス長が 28 [bit], 仮想アドレス長が 32 [bit], ページサイズを 4 [Kbyte] とする (c) を導入した計算機について考える。この計算機では、1 [byte] ごとにメモリアドレス (memory address) が割り付けられているとする。また、仮想アドレスから実アドレスへの変換を行う際に用いられるページテーブル (page table) は 1 段で構成されているとし、ページテーブルの 1 エントリは 2 [byte] とする。なお、ページテーブルによる主記憶容量の減少はないものとする。この計算機で利用できる最大の主記憶容量は ① [Mbyte] となり、このときページ枠 (page frame) は ② 個となる。一方、仮想アドレス空間の大きさは ③ [Gbyte] であり、ページテーブルのサイズは ④ [Mbyte] となる。

選択肢

- (ア) DMA (direct memory access) (イ) 動的再配置 (dynamic relocation) (ウ) キャッシュ (cache)
 (エ) コンパクション (compaction) (オ) 静的再配置 (static relocation) (カ) ページング (paging)
 (キ) ページ枠 (page frame) (ク) 実アドレス空間 (real address space)
 (ケ) フラグメンテーション (fragmentation) (コ) 仮想アドレス空間 (virtual address space)
 (サ) セグメンテーション (segmentation)

- (2-2) ページ枠の数が 4 のページングについて考える。プロセスの処理開始前には、ページ枠は「空」に初期化されているものとし、以下の各設問に答えよ。

- (2-2-1) プロセスが次に示すページ参照列 P で処理を行うものとする。ページ置き換えアルゴリズムとして FIFO (first in first out), LRU (least recently used) を用いるそれぞれの場合において、ページフォルト (page fault) が発生するページ参照に○を記入せよ。

ページ参照列 P : 0, 1, 2, 3, 4, 5, 0, 4, 2, 3, 5, 1, 0

- (2-2-2) プロセスが次に示すページ参照列 Q で処理を行うものとする。このとき、ページ置き換えアルゴリズムとして LRU を用いた場合、設問 (2-2-1) で LRU を用いた場合と比較してページフォルトの回数が減少した。このようなページフォルトの回数の減少が生じる理由について簡潔に説明せよ。

ページ参照列 Q : 0, 1, 2, 3, 3, 1, 2, 0, 0, 4, 5, 4, 5

配点:(1) 24 点, (2-1)12 点, (2-2) 14 点, (3-1) 15 点, (3-2) 10 点, (4-1) 15 点, (4-2) 10 点

論理式 (logic formula) の記号 $\Leftrightarrow, \Rightarrow, \vee, \wedge, \neg$ は, それぞれ, 等価 (equivalence), 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, 真 (true), 偽 (false) を表す true, false の二値からなる集合を B とする.

- (1) 一階述語論理 (first-order logic) における式 (formula) の解釈 (interpretation) I は (D, C, F, P) の 4 項組で与えられる. ここで, D は値集合, C は各定数記号への D の要素の割り当て, F は各 n 引数関数記号への $f: D^n \rightarrow D$ なる関数 f の割り当て, P は各 n 引数述語記号への $p: D^n \rightarrow B$ なる述語 p の割り当てである.

例えば, 一階述語論理式 $\forall x p(f(b, x), a)$ に対して, 解釈 $I_0 = (D_0, C_0, F_0, P_0)$ として次を与え
ると, その評価値は true となる.

- D_0 は非負整数 (non-negative integer) 全体からなる集合とする.
- C_0 は定数記号 a, b にそれぞれ値 $0, 1$ を割り当てる.
- F_0 は 2 引数関数記号 $f(u, w)$ へ非負整数上の加算 $u + w$ を割り当てる.
- P_0 は 2 引数述語記号 $p(u, w)$ へ非負整数上の比較演算 $u > w$ を割り当てる.

問 (1) では各解釈 $I_k = (D_k, C_k, F_k, P_k)$ ($1 \leq k \leq 3$) を定めるにあたり, D_k を k によらず, $\{a, b, c\}$ とする. また, C_k, F_k はそれぞれ任意の割り当てとし, P_k として述語記号 r に以下で定める $r_k: D_k^2 \rightarrow B$ を割り当てる.

$$r_1(x, y) = \text{true} \Leftrightarrow (x, y) \in \{(a, a), (a, b), (a, c)\}$$

$$r_2(x, y) = \text{true} \Leftrightarrow (x, y) \in \{(a, a), (a, b), (b, b), (c, b), (c, c)\}$$

$$r_3(x, y) = \text{true} \Leftrightarrow (x, y) \in \{(a, b), (b, c), (c, a)\}$$

以下の各論理式 (a)~(d) と解釈 I_k ($1 \leq k \leq 3$) のすべての組み合わせ 12 通りについてその評価値を答えよ. 答は「(a): true, true, true」のように各論理式の記号を明記した上で解釈 I_1, I_2, I_3 の順に評価値を記すこと.

$$(a) \quad \forall x \exists y r(x, y) \quad (b) \quad \exists y \forall x r(x, y) \quad (c) \quad \forall y \exists x r(x, y) \quad (d) \quad \exists x \forall y r(x, y)$$

- (2) 以下で与えられる論理式 F が恒真 (valid) であることを次の (i), (ii) を行うことによって示したい.
(i) F の否定のスコレム化 (Skolemization), (ii) 導出原理 (resolution principle) による充足不能 (unsatisfiable) であることの確認. 以下の各小問に答えよ.

$$A = p(g(f(a)))$$

$$B = p(f(g(b)))$$

$$C(x) = p(g(x)) \Rightarrow p(x)$$

$$D(x) = p(x) \Rightarrow p(f(x))$$

$$E(x) = p(f(f(x)))$$

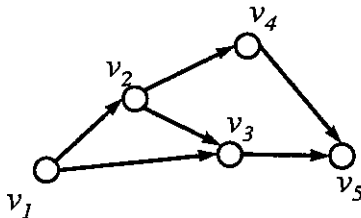
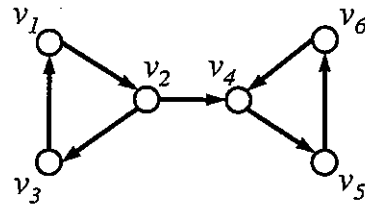
$$F = (A \wedge B \wedge \forall x C(x) \wedge \forall x D(x)) \Rightarrow \exists x E(x)$$

ただし, a, b は定数記号, f, g は関数記号, p は述語記号である.

- (2-1) $\neg F$ のスコレム連言標準形 (Skolem conjunction normal form) F' を答えよ. F' 中では記号 A, B, C, D, E, F を用いないこと. 導出過程は不要.

- (2-2) 導出原理を用いて F' が充足不能であることを示せ. 導出過程も示すこと.

- (3) $G_1 = (V_1, E_1)$ と $G_2 = (V_2, E_2)$ は下に示した有向グラフであり, V_1 と V_2 は頂点の集合, E_1 と E_2 は辺の集合である.

 $G_1=(V_1, E_1)$  $G_2=(V_2, E_2)$

以下の R_1 と R_2 はそれぞれ G_1 と G_2 のグラフの頂点の集合上の 2 項関係 (binary relation) である.

$R_1 = \{(v, v') \in V_1 \times V_1 \mid G_1 \text{ 上で } v \text{ から } v' \text{ への経路が存在するか, または, } v = v' \text{ である}\}$

$R_2 = \{(v, v') \in V_2 \times V_2 \mid G_2 \text{ 上で } v \text{ から } v' \text{ への経路と } v' \text{ から } v \text{ への経路がともに存在するか, または, } v = v' \text{ である}\}$

以下の各小問に答えよ.

- (3-1) R_1 が反射的 (reflexive) かつ反対称的 (antisymmetric) であることを証明せよ.
- (3-2) R_2 は同値関係 (equivalence relation) である. R_2 の同値類 (equivalence class) をすべて示せ.
- (4) n 変数論理関数 $f: B^n \rightarrow B$ に対して, $C(f) = \{x \mid x \in B^n, f(x) = \text{true}\}$ と定める. さらに n 変数論理関数の集合上の関係 \geq を次のように定める. ここで, f と g は n 変数論理関数である.

「 $f \geq g$ となるのは, $C(f) \subseteq C(g)$ であるとき, かつ, そのときに限る」

以下の各小問に答えよ. なお, 論理関数に対する論理積 $f \wedge g$, 論理和 $f \vee g$, 否定 $\neg f$ については, それぞれ, $(f \wedge g)(x) = f(x) \wedge g(x)$, $(f \vee g)(x) = f(x) \vee g(x)$, $(\neg f)(x) = \neg f(x)$ を意味するものとする.

- (4-1) 任意の二つの n 変数論理関数 f と g の論理積 $f \wedge g$ を考えたとき, $f \wedge g \geq f$ であることを, 上記の定義に基づいて証明せよ.
- (4-2) $k (\geq 2)$ 個の n 変数論理関数 f_1, f_2, \dots, f_k に対して, 次の条件を満足する f を f_1, f_2, \dots, f_k および論理演算子を用いて記せ.

「全ての i について $f_i \geq f$ であり, かつ, 『全ての i について $f_i \geq g$ である』 任意の g について $f \geq g$ である」

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

(1) 文脈自由文法 (context-free grammar) について以下の各小問に答えよ。

(1-1) チョムスキー標準形 (Chomsky normal form) の定義を示せ。

(1-2) 文脈自由文法 G_1 を以下の通り定める。

$G_1(N_1, T_1, P_1, S_1)$

- 非終端記号 (non-terminal symbol) の集合 $N_1 = \{ S, A, B \}$
- 終端記号 (terminal symbol) の集合 $T_1 = \{ a, b \}$
- 生成規則 (generating rule) の集合 $P_1 = \{ S \rightarrow aB, S \rightarrow Ab, A \rightarrow aAB, A \rightarrow a, B \rightarrow Sb, B \rightarrow aSA, B \rightarrow b, B \rightarrow \varepsilon \}$
- 始記号 (start symbol) $S_1 = S$

文法 G_1 と同じ言語 (language) を生成するチョムスキー標準形に従う文法 G_2 は以下のものである。空欄 (ア)～(オ) に当てはまるものを選択肢 (a)～(i) の中から選べ。

$G_2(N_2, T_2, P_2, S_2)$

- 非終端記号の集合 $N_2 = \{ S, A, B, X_{AB}, X_{SA}, X_a, X_b \}$
- 終端記号の集合 $T_2 = \{ a, b \}$
- 生成規則の集合 $P_2 = \{ S \rightarrow X_aB, S \rightarrow AX_b, S \rightarrow \boxed{\text{ア}}, A \rightarrow X_aX_{AB}, A \rightarrow \boxed{\text{イ}}, A \rightarrow a, B \rightarrow \boxed{\text{ウ}}, B \rightarrow X_aX_{SA}, B \rightarrow b, X_{AB} \rightarrow \boxed{\text{エ}}, X_{SA} \rightarrow \boxed{\text{オ}}, X_a \rightarrow a, X_b \rightarrow b \}$
- 始記号 $S_2 = S$

選択肢:

- (a) a (b) b (c) SA (d) SB (e) X_aA (f) SX_a (g) SX_b (h) AB (i) BA

(2) 文脈自由文法 G_3 を以下の通り定める。以下の各小問に答えよ。

$G_3(N_3, T_3, P_3, S_3)$

- 非終端記号の集合 $N_3 = \{ S, B, C \}$
- 終端記号の集合 $T_3 = \{ a, b, c \}$
- 生成規則の集合 $P_3 = \{ S \rightarrow BC, S \rightarrow a, B \rightarrow CS, B \rightarrow b, C \rightarrow SB, C \rightarrow c \}$
- 始記号 $S_3 = S$

(2-1) 文法 G_3 が生成する文 (sentence) で長さ 3 以下のもの全てを導出木 (derivation tree) と共に示せ。

(2-2) 文法 G_3 が生成する文 $abbabab$ に対する導出木を示せ。

(3) オートマトン (automaton) に関する以下の各小問に答えよ。以下では、決定性有限オートマトン (deterministic finite automaton) を DFA, 非決定性有限オートマトン (nondeterministic finite automaton) を NFA, プッシュダウンオートマトン (pushdown automaton) を PDA と略記する。オートマトン M によって認識される言語 (language recognized by M) を $L(M)$ と表記する。また、すべてのオートマトンについて、入力記号 (input symbol) の集合は $\Sigma = \{0, 1\}$ とする。

(3-1) $L(M_1)$ が次の正規表現 (regular expression) によって表される DFA M_1 について考える。

$$(0 + 1)^* 10$$

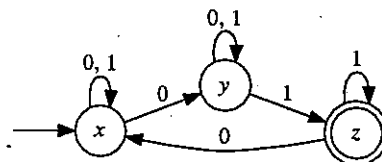
以下は M_1 の状態遷移表 (state transition table) である。

	0	1
$\rightarrow a$	(ア)	b
b	(イ)	b
㊟	a	(ウ)

注. \rightarrow は初期状態, ㊟ は受理状態を示す。

上の状態遷移表の空白部分 (ア)~(ウ) を埋めよ。ただし, M_1 の初期状態 (initial state) は a , 受理状態 (accepting state) は c である。

(3-2) 下の状態遷移図は NFA M_2 を表している。

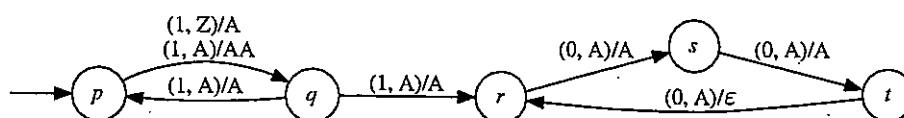


ここで, $L(M_2) = L(M_3)$ である DFA M_3 を考える。以下は M_3 の状態遷移表である。

	0	1
$\rightarrow d$	(エ)	d
e	(オ)	(カ)
㊟	(キ)	(ク)

上の状態遷移表の空白部分 (エ)~(ク) を埋めよ。ただし, M_3 の初期状態は d , 受理状態は f である。

(3-3) 以下の状態遷移図で表される PDA M_4 について, $w \in L(M_4)$ かつ $|w| \leq 10$ であるような語 (word) w をすべて示せ。ただし, M_4 は空スタック受理 (acceptance by empty stack) を行うものとし, スタック記号 (stack symbol) の集合は $\{A, Z\}$, スタックの底記号 (initial stack symbol) は Z である。



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

- (1) 以下の文章を読み、その下の各小問に答えよ。導出の過程は記述しなくてよい。なお、 $P(E)$ は事象 E の生起確率を表す。

情報ネットワークにおいて、できるだけ冗長さなく情報を伝えること（すなわち通信効率を上げること）と、できるだけ誤りなく情報を伝えることとは、一般にトレードオフの関係にある。以下で述べる通信路容量（channel capacity）は、情報が誤って伝わる確率を任意に小さくできる通信効率の上限として知られる重要な概念である。

C を記憶のない通信路（memoryless channel）とし、送信記号（または入力記号）集合（input alphabet）および受信記号（または出力記号）集合（output alphabet）をともに $\{a, b\}$ とする。送信記号を表す確率変数を X 、受信記号を表す確率変数を Y とおく。 X と Y の相互情報量（mutual information） $I(X; Y)$ とは、受信側が Y を受信することによって得られる X の平均の情報量であり、 $I(X; Y) = \boxed{\text{(あ)}}$ と定義される。通信路 C の通信路容量は、 $P(X = a)$ に関する $I(X; Y)$ の最大値 $\max_{0 \leq P(X=a) \leq 1} I(X; Y)$ として定義される。

今、送信記号が a のときの受信記号が b である条件付き確率 $P(Y = b | X = a)$ と、送信記号が b のときの受信記号が a である条件付き確率 $P(Y = a | X = b)$ とが、どちらも ε （ただし $0 \leq \varepsilon < 1/2$ ）に等しいとする。さらに、 X は記憶のない情報源（memoryless source）であるとし、 $P(X = a) = r$ とおく。このとき X の情報エントロピー（entropy） $H(X)$ は r を用いて $\boxed{\text{(い)}}$ と表され、 Y の情報エントロピー $H(Y)$ は r と ε を用いて $\boxed{\text{(う)}}$ と表される。また、この通信路 C の通信路容量を実現する r の値は $\boxed{\text{(え)}}$ である。

- (1-1) 空欄（あ）を、 X と Y を用いた式で埋めよ。なお、必要に応じて、情報エントロピー $H(X)$ 、 $H(Y)$ や条件付きエントロピー（conditional entropy） $H(X | Y)$ 、 $H(Y | X)$ を解答に用いてもよい。
- (1-2) 空欄（い）を適切な式で埋めよ。対数の底は 2 とせよ。なお、次式で定義される関数 F を解答に用いてもよい。

$$F(x) = -x \log_2 x - (1 - x) \log_2 (1 - x)$$

- (1-3) 空欄（う）を適切な式で埋めよ。対数の底は 2 とせよ。なお、上で定義された関数 F を解答に用いてもよい。
- (1-4) 空欄（え）を適切な値で埋めよ。

(2) 情報ネットワークにおける階層化モデル (layered model) に関する以下の各小問に答えよ。

- (2-1) 階層化モデルに関する以下の説明文の空欄 (a) ～ (e) に当てはまる最も適切な語句を選択肢から選び、その番号を答えよ。なお、異なる空欄には異なる語句が当てはまる。

階層化モデルにおいては、情報ネットワークの機能は、上下関係のある複数の階層に分割され、各階層の機能は (a) として上位の階層に提供される。代表的な階層モデルである OSI 参照モデル (OSI reference model) では、7つの階層が規定されている。異なる階層や同じ階層の間でやりとりされる情報のかたまりをメッセージ (message) と呼ぶとき、ホスト (host) やノード (node) 間の通信において、同じ階層でやりとりするメッセージの内容や形式、やりとりの順序などを定めるものを (b) と呼ぶ。また、上下の階層間でのメッセージのやりとりを定めるものを (c) と呼ぶ。OSI 参照モデルでは、同じ階層でホストやノードがやりとりするメッセージの単位を (d) と呼び、トランスポート (transport) 層ではセグメント (segment)、ネットワーク (network) 層ではデータグラム (datagram) と称される。 (d) はヘッダ (header) 部とデータ (data) 部からなり、ある階層の (d) は、 (e) 層から受け取った (d) にその階層のヘッダを付加したものである。

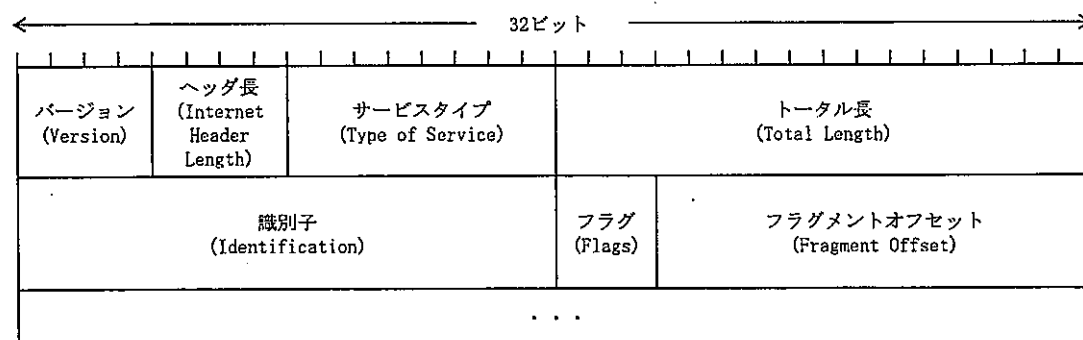
選択肢：

1. 物理 (physical), 2. 回線 (link), 3. PDU (Protocol Data Unit), 4. パケット交換 (packet switching), 5. サービス (service), 6. 下位 (lower), 7. パケット (packet), 8. 回線交換 (circuit switching), 9. インタフェース (interface), 10. アドレス (address), 11. プロトコル (protocol), 12. 上位 (upper), 13. スロット (slot), 14. プレゼンテーション (presentation)

- (2-2) 以下の図は、IP データグラムのヘッダ部のフォーマットの一部を表したものである。図において、それぞれの長方形の領域はフィールドを、また、上部の区切りはビットを表している。例えばバージョンフィールドには 4 [ビット (bit)] が割り当てられている。ヘッダ部の長さが 20 [バイト (byte)] のある IP データグラムについて、以下の (2-2-1), (2-2-2) に答えよ。導出の過程も記述すること。なお、ヘッダ長フィールドでは、ヘッダ部の長さが 32 ビット単位 (32 bit word) で表される。

(2-2-1) ヘッダ長フィールドに入るビット列を答えよ。

(2-2-2) データ部の最大長は何バイトになるか答えよ。



- (2-3) ある通信環境において、ネットワーク層に対して下位層から提供される通信速度が 10 [Mbps] であった。トランスポート層が上位層に提供できる通信速度の最大値を小数第一位まで答えよ (小数第二位を四捨五入)。導出の過程も記述すること。ただし、上位層が下位層から提供される通信速度は、上位層が単位時間あたりに下位層に送出できるヘッダ部、データ部をあわせたメッセージの量とし、単位を Mbps とする。また、トランスポート層で付加されるヘッダ部の長さを 20 [バイト]、ネットワーク層で付加されるヘッダ部の長さを 20 [バイト] とする。また、この時のデータグラムの最大長を 1500 [バイト] とする。なお、トランスポート層、ネットワーク層のそれぞれにおいて、セグメント、またはデータグラムはその下位層へ隙間なく連続して送出するものとし、また、上下の階層間でのメッセージのやりとりによる通信速度のロスはないものとする。

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

エッジトリガ型 D フリップフロップ(edge-triggered D flip-flop)を用いた順序回路(sequential circuit)について、次の各問に答えよ。

3 個の D フリップフロップのそれぞれの出力(output)を O_2, O_1, O_0 とする。(O_2, O_1, O_0)が(0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (0,0,0), (0,0,1), ... と遷移するカウンタ(counter)を設計する。 O_2, O_1, O_0 に対応する D フリップフロップの入力(input)を、それぞれ D_2, D_1, D_0 とする。また、(O_2, O_1, O_0)の初期値(initial value)を(0,0,0)とする。

(1) D_2, D_1, D_0 の論理式(Boolean expression)を次の手順で求め、カウンタを設計する。

(1-1) カウンタの状態遷移表(state transition table)を作成せよ。

(1-2) D_2, D_1, D_0 のカルノー図(Karnaugh map)をそれぞれ作成せよ。

(1-3) D_2, D_1, D_0 を O_2, O_1, O_0 の論理式でそれぞれ表せ。ただし、論理式は最簡積和形(最小積和形, minimal sum-of-products expression)で表すこと。

(2) (1)で設計したカウンタを論理回路(logic circuit) で実現する。図 1 の点線内部 CL2, CL1, CL0 を組合せ回路(combinational circuit)で実現せよ。ただし、論理ゲート(logic gate)は、NOT ゲート, NAND ゲートのみが使用できるとする。

(3) (2)で完成させたカウンタにおいて、各フリップフロップの出力から各フリップフロップの入力への経路(path)の中で、遅延時間(delay)が最も長くなる経路をすべて示せ。また、その遅延時間を求めよ。ただし、NOT ゲートの遅延時間を 1, n 入力 NAND ゲートの遅延時間は n とせよ。

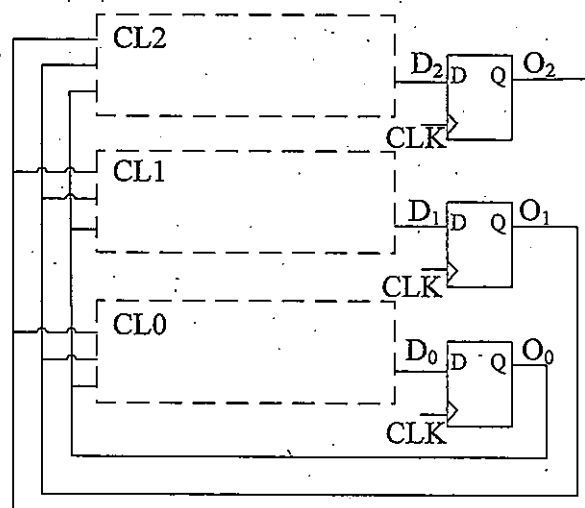


図 1: カウンタ (CLK はクロック信号)

- (4) 論理ゲートによる遅延時間は、厳密にはゲートを構成するトランジスタ(transistor)と負荷容量(load capacitance)により決定される。2入力 NAND ゲートの遅延時間を求めるために、図 2 のように負荷容量 C_L のコンデンサ(capacitor)を駆動するモデルを考える。電源電圧は V_{DD} とする。また、すべてのトランジスタのドレイン(drain)・ソース(source)間の抵抗値(resistance)はオンするときには瞬時に R_{TR} となり、オフするときには瞬時に ∞ になると仮定する。NAND ゲートの入力 A, B が、時刻 $t=0$ でともに 0 から V_{DD} に瞬時に変化するとき、出力電圧 V_O の過渡応答(transient response)を表す式を求めよ。また、出力電圧が $0.5 V_{DD}$ に達するまでの時間を求めよ。ただし、入力が変化する直前の出力電圧 V_O は V_{DD} であったと仮定せよ。

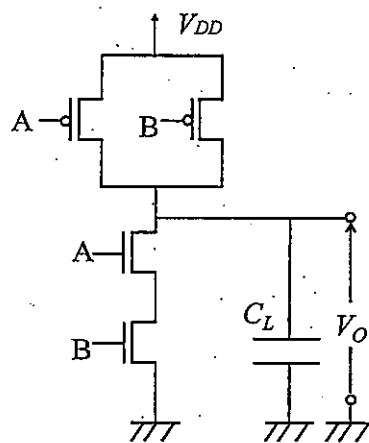


図 2: 容量負荷をモデル化した 2 入力 NAND ゲート

配点(1)30点, (2)40点, (3)30点

以下の各問に答えよ。ただし、導出過程も記述すること。

(1) $\int_0^{\infty} \frac{\sin x}{x} dx = \frac{\pi}{2}$ であることを用いて $\int_0^{\infty} \frac{\sin ax}{x} dx$ を求めよ。

ただし、 a は任意の実数(real number)である。

(2) $f(t) = \frac{\sin t}{t}$ をフーリエ変換(Fourier transform)し、角周波数(angular frequency) ω に対して
図示(plot)せよ。

(3) $f_1(t) = \cos \omega_0 t$ で与えられる信号波形(signal wave)を

窓関数(window function) $f_2(t) = \begin{cases} 1 & -\frac{L}{2} \leq t \leq \frac{L}{2} \\ 0 & |t| > \frac{L}{2} \end{cases}$

で観測(observation)したときのパワースペクトル(power spectrum) $F(\omega)$ を畳み込みの定理(convolution theorem)を用いて求めよ。ただし、 ω_0 は正の実数であり、 $L \geq (\pi / \omega_0)$ とする。