

平成 16 年 8 月 24 日 (火)

13:00～16:00 実施

平成 17 年度大学院前期課程入学試験

情報通信工学

入試問題

注意事項

- ・ 問題は 7 問、問題用紙の枚数は 9 枚（表紙を含まず）である。
- ・ 問題 1～問題 3 は必須問題である。すべて解答すること。
- ・ 問題 4～問題 7 は選択問題である。この中から 2 題を選択し、解答せよ。
- ・ 解答用紙は、
 - 問題 1 を 1 枚目（白色）の解答用紙
 - 問題 2 を 2 枚目（赤色）の解答用紙
 - 問題 3 を 3 枚目（青色）の解答用紙
 - 問題 4 を 4 枚目（黄色）の解答用紙
 - 問題 5 を 5 枚目（水色）の解答用紙
 - 問題 6 を 6 枚目（桃色）の解答用紙
 - 問題 7 を 7 枚目（緑色）の解答用紙に記入すること。選択問題については、選択した問題番号に対応する解答用紙を選択すること。解答用紙を間違えると採点されない場合があるので注意すること。
- ・ 試験終了後の解答用紙回収について
 - 1) 試験終了後、指示に従い、選択しなかった問題の解答用紙を 2 つ折りにすること。
 - 2) 選択問題については、すべてについて解答した後、2 つ折りにする段階で、選択問題を確定させても良い。
 - 3) 解答した解答用紙、および 2 つ折りにした解答用紙は、試験監督者の指示に従って提出すること。提出する解答用紙は 5 枚、2 つ折りにする解答用紙は 2 枚である。

情報理論

1. 情報源記号として $\{X_1, X_2, X_3\}$ を有する3元情報源 S を考える. この情報源 S は記憶の無い情報源であり, また, この情報源 S から出現する各情報源記号の生起確率は等しいとする. この情報源 S に関する以下の設問に答えよ. 但し, 解答に際して対数の計算が必要な場合は, $\log_{10}2=0.30$, $\log_{10}3=0.48$ を利用して計算すること.

表1 符号化則

情報源記号	2元符号語
X_1	0
X_2	01
X_3	10

- (i) 情報源 S のエントロピーを求めよ.
- (ii) 情報源 S に対する2元符号化を行う符号化則(各情報源記号と、『0』および『1』により構成される2元符号語の対応関係)として表1に示す符号化則を用いた場合, それによって得られる2元符号は瞬時符号であるか, 非瞬時符号であるか答えよ. また, 非瞬時符号である場合は, 瞬時に復号できない例を示せ.
- (iii) 情報源 S について, ハフマン(Huffman)符号化に基づいて各情報源記号の2元符号化を行う符号化則を求めよ. 但し, 符号化則の導出過程についても図示すること.
- (iv) 上記(iii)で求めた符号化則に従った場合の平均符号長を求めよ.
- (v) 情報源 S から出現する情報源記号を2個ずつまとめて, 新たに1つの情報源記号とする情報源(2次の拡大情報源) S^2 を考える. この2次の拡大情報源 S^2 のエントロピーを求めよ.
- (vi) 2次の拡大情報源 S^2 について, ハフマン符号化に基づいて各情報源記号の2元符号化を行う符号化則を求め, その符号化則に従った場合の平均符号長を求めよ.
- (vii) 情報源記号をできるだけコンパクトに伝送するという観点から, 上記(iii)で求めた符号化則と上記(vi)で求めた符号化則の優劣について比較せよ.

通信方式

2-1 FM変調方式について以下の問いに答えよ。

- (i) FM変調における瞬時周波数の定義を説明せよ。
- (ii) FM変調における瞬時周波数と変調信号の関係を説明せよ。
- (iii) 変調信号を $m(t)$ 、搬送波の振幅を A 、搬送周波数を f_c とするとき、FM変調された信号は、次式で与えられることを説明せよ。

$$e_{FM}(t) = A \cos \left[2\pi f_c t + k_f \int_{-\infty}^t m(\tau) d\tau \right]$$

- (iv) 問い(iii)で得られた被変調信号を復調する手段について説明せよ。

2-2 2相位相変調された信号を受信機で受信する場合、雑音がない場合の受信波形は、送信データが「1」の場合、 $A \cos(2\pi f_c t)$ 、送信データが「0」の場合、 $A \cos(2\pi f_c t + \pi)$ で表されるものとする。それに対して、受信時に発生する雑音は、

$$n(t) = x(t) \cos 2\pi f_c t - y(t) \sin 2\pi f_c t$$

ただし、 $x(t)$ 、 $y(t)$ は、確率密度関数が

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right), \quad p(y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{y^2}{2\sigma^2}\right)$$

で与えられる、互いに独立なランダム変数であるものとする。この雑音が受信信号に加わった信号を同期検波するとき、以下の問いに答えよ。

- (i) 雑音を含む受信信号 $z(t)$ を、送信データが「1」の場合と送信データが「0」の場合に分けて記せ。
- (ii) 同期検波後の波形 $r(t)$ はどのように表されるかを説明せよ。
- (iii) 送信データが「1」の場合の $r(t)$ の確率密度関数 $p(r|\text{データ}=1)$ と、送信データが「0」の場合の $r(t)$ の確率密度関数 $p(r|\text{データ}=0)$ が数式でどのように表されるかを、説明せよ。
- (iv) 送信データにおける「1」と「0」の生起確率は等しいものとするとき、ビット誤り率を求めよ。ただし、

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt$$

とする。

ネットワーク工学

3. コンピュータネットワークでの通信によく用いられる TCP について、以下の問いに答えよ。

- (i) TCP とは何の略か。
- (ii) TCP はプロトコル階層のどの階層に実装されているか。
- (iii) TCP を(ii)で答えた階層のプロトコルとして用いるアプリケーションを2つ挙げよ。
- (iv) TCP の提供する機能に、フロー制御と輻輳制御がある。それぞれの役割について説明せよ。
- (v) TCP において輻輳の発生を検知する仕組みを説明せよ。
- (vi) (v)で解答した輻輳検出の仕組みをとらなければならない理由を、TCP が実装されるプロトコル階層と関連付けて説明せよ。
- (vii) 以下の文中の空欄を埋めよ。

送信側ホストの TCP が輻輳を検出すると、Congestion Window を減少させる。TCP では、送信ホストから送出されたセグメントに対し、受信ホストがこれを正しく受信すると ① と呼ばれる応答を返信する。基本的には、Congestion Window 分だけセグメントを送出した後に、送信ホストと受信ホスト間の往復遅延である ② の後に ① が返ってくるために、おおまかには ② の間に Congestion Window 分だけのセグメントが送出されることになる。つまり、Congestion Window を減少させることは、送信ホストの送出レートを下げることにつながる。

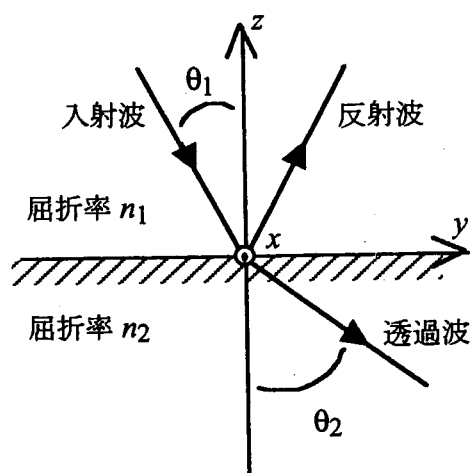
- (viii) Congestion Window を変化させる際の基本原理として、Additive Increase/ Multiplicative Decrease という概念が用いられる。この概念を簡単に説明するとともに、なぜこの考え方が適用されるのかについて論ぜよ。

光・電波工学

4. 屈折率が異なる2種の誘電体媒質が xy 平面を境界面として接している。電界ベクトルが

$$\mathbf{E} = \mathbf{i}_x E_0 \exp[-jk_1(y \sin \theta_1 - z \cos \theta_1)]$$

で与えられる角周波数 ω の平面電磁波が、図に示すように境界面に斜めに入射するとき、以下の問いに答えよ。ただし、入射波の波面は x 軸と平行であり、入射波の伝搬方向と z 軸のなす角を θ_1 とする。また、入射電界の振幅を E_0 (定数)、 $z > 0$ および $z < 0$ の領域を占める誘電体の屈折率をそれぞれ n_1 および n_2 ($n_1 > n_2$) とする。

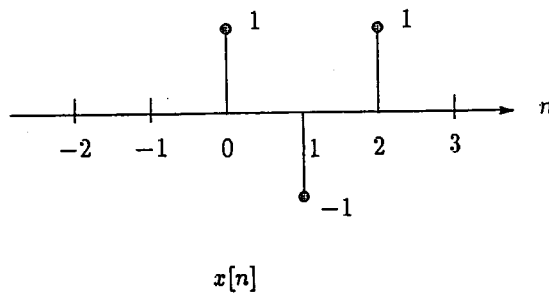


- (i) 入射波の波数の大きさ k_1 を角周波数 ω 、真空中の光速 c 、および誘電体の屈折率 n_1 を用いて表せ。
- (ii) 入射波の磁界ベクトルを求めよ。
- (iii) 入射波が $-z$ 方向に運ぶ単位面積当たりの電力の時間平均値を求めよ。
- (iv) 電界の反射係数および透過係数をそれぞれ r および t とするとき、反射波および透過波の電界ベクトルを示せ。
- (v) 反射係数 r および透過係数 t を求めよ。
- (vi) 入射波が境界面で全反射されるための条件を示せ。

信号処理

5. 線形性と時不変性を有する離散時間信号処理システム L (単に線形時不変システムと呼ぶ) の入力信号, 出力信号を各々 $x[n]$, $y[n]$, (n は整数) とし, L の入出力関係を $y[n] = L[x[n]]$ と表す. 以下の問いに答えよ.

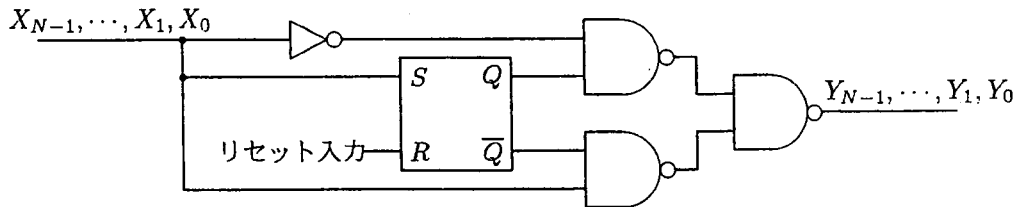
- (i) 信号処理システム L の線形性および時不変性とはどのようなものか. それぞれの性質を数式を用いて説明せよ.
- (ii) インパルス応答 $h[n]$, (n は整数) の定義を数式で表現せよ.
- (iii) 線形時不変システム L の出力信号 $y[n]$ が入力信号 $x[n]$ とインパルス応答 $h[n]$ のたたみ込みで表されることを導出せよ.
- (iv) 線形時不変システム L に対する $x[n]$ を下図のように与え, かつ $h[n] = x[n-1]$ であるとき, $y[n]$ を z 変換により求めよ.



- (v) 線形時不変システム L をいかなる順で縦続接続しても同一出力となることを証明せよ.
- (vi) 線形時不変システム L が無記憶 (ある時刻における出力は同時刻の入力によってのみ決定される) であるときのインパルス応答 $h[n]$ が満たす条件を求めよ.
- (vii) 線形時不変システム L が因果的 (ある時刻における出力はその時刻またはその時刻以前の入力によってのみ決定される) であるときのインパルス応答 $h[n]$ が満たす条件を求めよ.

コンピュータ工学（ハードウェア）

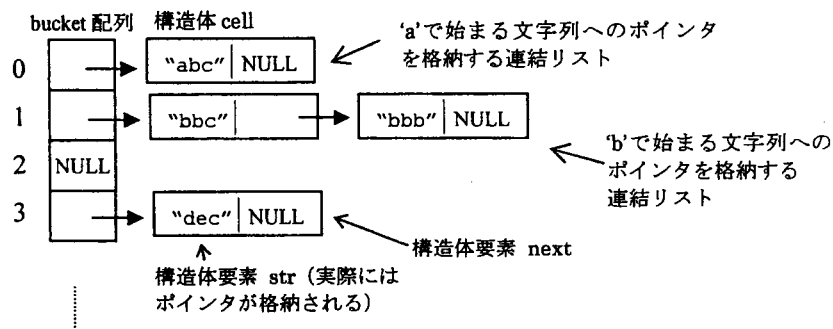
6. 図に示された回路を考える。最初にリセット入力が入り、SR フリップ・フロップの状態 Q が 0 になった後、 N 個の入力 $X_0, X_1, X_2, \dots, X_{N-1}$ が入力され、それに応じて出力 $Y_0, Y_1, Y_2, \dots, Y_{N-1}$ が得られる。この回路の動作について以下の問いに答えよ。



- (i) Y_i を X_i と Q を用いて表現せよ。
- (ii) $Q = 0$ と $Q = 1$ の場合について、それぞれ Y_i を X_i のみで表現せよ。
- (iii) X_i の入力後 $Q = 0$ であったとする。 $X_{i+1} = 0$, $X_{i+1} = 1$ の場合について、それぞれ X_{i+1} の入力後の Q の値を求めよ。
- (iv) X_i の入力後 $Q = 1$ であったとする。 $X_{i+1} = 0$, $X_{i+1} = 1$ の場合について、それぞれ X_{i+1} の入力後の Q の値を求めよ。
- (v) 以上の考察から、 $(X_0, X_1, \dots, X_9) = (0, 0, 0, 0, 1, 1, 0, 1, 0, 0)$ ($N = 10$) と入力に加えられた場合について、その出力 (Y_0, Y_1, \dots, Y_9) を求めよ。
- (vi) 一般に $(X_0, X_1, \dots, X_{N-1})$ のうち、最初の k 個の入力 X_0, X_1, \dots, X_{k-1} が 0 であり、 X_k が 1 であるとする。この時の出力 $(Y_0, Y_1, \dots, Y_{N-1})$ を Y_0, \dots, Y_k と Y_{k+1}, \dots, Y_{N-1} に分けて求めよ。
- (vii) 各入力 X_i ならびに各出力 Y_i に対して 2^i の重みをもたせ、それぞれ N 桁の 2 進数とみなす。ただし、 X_i ($i = 0, 1, \dots, N-1$) のうち少なくとも 1 つは 1 であるとする。このとき入力 $(X_0, X_1, \dots, X_{N-1})$ と出力 $(Y_0, Y_1, \dots, Y_{N-1})$ との和 K を 10 進数で表せ。
- (viii) N 桁の 2 進数 $(X_0, X_1, \dots, X_{N-1})$ を上記の回路の出力 $(Y_0, Y_1, \dots, Y_{N-1})$ に変換する機能を有している計算機が存在するが、このような機能は通常どのような目的のために利用されるか述べよ。

コンピュータ工学 (ソフトウェア)

7-1 プログラム A は文字列を複数の連結リストに格納するデータ構造 (バケットと呼ばれる) を実装したものである。文字列は固定長 K で英小文字 a, b, c, \dots, z から構成され、同じ文字列はないものとし、文字列へのポインタが構造体 `cell` をノードとする連結リストに格納される。構造体 `cell` は、格納する文字列へのポインタ(`str`)と、次の `cell` へのポインタ(`next`)から構成される。文字列は先頭の文字によって、対応する連結リスト p_j ($0 \leq j < 26$, 'a'のとき $j=0$, 'b'のとき $j=1 \dots$ とする) に格納される。連結リスト p_j の先頭ノードへのポインタは、下図の例のように、配列 `bucket` の要素 `bucket[j]` に格納される。つまり、文字列 `str` の先頭の文字が、例えば 'a'であれば `bucket[0]`, 'b'であれば `bucket[1]`に格納されるポインタによって指し示されるノードを先頭とする連結リストに、文字列へのポインタが格納される。プログラム A を読み、以下の問いに答えよ。



プログラム A

```
#include <stdio.h>
#include <stdlib.h>

#define M 26 /* 文字種の数 */
#define K 3 /* 文字列の長さ (文字数) */
#define CI 0 /* 参照する文字の位置 (先頭が 0) */

struct cell {
    char *str;
    struct cell *next;
};

int main () {
    struct cell *bucket[M];
    int j;

    for (j=0; j<M; j++) (1);
    insert(bucket, "abc", CI); insert(bucket, "bbc", CI);
    insert(bucket, "aaq", CI); insert(bucket, "aaa", CI);
    insert(bucket, "ccd", CI);
    <A>
    delete(bucket, "aaq", CI);
    delete(bucket, "ccd", CI); insert(bucket, "acc", CI);
    <B>
}
```



```

/* 文字列 str を c 文字目に従って bucket に示される連結リストに入れる */
void insert(struct cell *bucket[], char *str, int c) {
    struct cell *newcell;
    int bi;

    bi = index(str, c);
    newcell = (struct cell *)malloc(sizeof(struct cell));
    newcell->str = str;
    (2);
    bucket[bi] = newcell;
    return;
}

/* 文字列 str を bucket から削除する */
void delete(struct cell *bucket[], char *str, int c) {
    int bi;
    struct cell *p,*q;

    bi = index(str, c);
    p = (3);
    q = NULL;
    while (p != NULL) {
        if (strcmp(p->str, str)==0) {
            if (q==NULL) (4);
            else (5);
            free(p);
            return;
        }
        q = p; p = p->next;
    }
}

/* 文字列 str を入れる bucket の番号 h (0<h<M) を返す */
int index(char *str, int c) {
    return (str[c] - 'a'); /* 文字のコードは連続しているものとする */
}

```

(i) <A>, におけるバケットの状態を例のようにそれぞれ図示せよ。ただし, bucket[j] ($0 \leq j < 4$) の範囲だけでよい。

(ii) プログラムA中の空欄 (1) から (5) を埋めよ。

7-2 プログラムA (main 関数を除く) にプログラムBを加えることで, n 個の文字列を辞書順にソートするプログラムを以下のような考えで実現する。文字列へのポインタは配列 $s[j]$ ($0 \leq j < n$) に格納されており, 固定長 K の英子文字で構成されたとする。まず文字列の最後の文字 ($s[j][K-1]$) を参照して文字列をソートし, 次に $s[j][K-2]$ についてソートするというように, 文字列の文字ごとにソートすることで文字列全体をソートする (一般に基数ソートと呼ばれる)。 k 文字目 ($0 \leq k < K$, 先頭文字を 0 文字目とする) に対するソートでは, 7-1 で実装したバケット構造を用いて, 7-1 では先頭の文字を参照して文字列を格納したが, ここでは k 文字目 ($s[j][k]$) を参照して文字列へのポインタをバケット構造に格納する。次に, bucket 配列を参照して連結リスト

をたどりながら文字列を順に取り出し、配列 s に順に文字列へのポインタを格納する
(バケットソートと呼ばれる)。以下の問いに答えよ。

プログラムB

```
#define N 50 /* 最大の要素数 (n<N であるとする) */

int main () {
    char *s[N]; /* ソートする対象の文字列 */

    s[0] = "aac"; s[1] = "abc"; s[2] = "aba";
    radixsort(s, 3);
}

/* n 個の文字列 s[i] (i<0<n) に対して基数ソートを行う */
void radixsort(char *s[], int n) {
    int k;

    for ( (6) ; k>=0; k--) {
        bucketsort(s, n, k);
    }
    return;
}

/* n 個の文字列 s[i] を, c 文字目 でソートする */
void bucketsort(char *s[], int n, int c) {
    struct cell *bucket[M], *p, *q;
    int i, j;

    for (j=0; j<M; j++) (1);
    for (j=0; j<n; j++) insert(bucket, s[j], c); /* プログラムA */
    (7);
    for (i=M-1; i>=0; i--) {
        p = (8);
        while ( (9) ) {
            (10); j=j-1; q = p->next; free(p); p=q;
        }
    }
    return;
}
```

- (i) プログラムB中の空欄 (6) ~ (10) を埋めよ。ただし, (1) はプログラムA中の (1) と同じであるので答える必要はない。
- (ii) k 文字目のソートにおいて, バケット構造を参照して, 配列 s に文字列を格納する際に, $\langle C \rangle$ の初期条件にあるように, 最後の要素 $bucket[M-1]$ から降順に取り出している。 k を用いて, その理由を説明せよ。
- (iii) このソートの時間計算量のオーダーを求めよ。文字列の数を n , 文字列の長さ(文字数)を K , 文字の種類(バケット数)を m として, これらを用いて表したオーダーと, その算出根拠を答えよ。