平成17年8月23日（火）
13:00〜16:00実施

平成18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10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

１－１　生成多項式として $G(x)=x^4+x+1$ を用い，11 ビットからなる情報ビット系列から符号長 15 の符号語を生成する巡回組織符号を考える．受信された 2 元符号語が (100111000101110)（但し，左のビットほど多項式表記において高次に割り当てられるものとする）であった場合について下記の問いに答えよ．

(i) 受信符号語を多項式表記で表せ．

(ii) 受信符号語に誤りがあるかどうか判定せよ．また，誤りが含まれている場合には，誤りを訂正し，送信されたと判断される 2 元符号語，ならびに，送信されたと判断される情報ビット系列を求めよ．

１－２　PCM (Pulse Code Modulation)によるアナログ情報の 2 元符号化（アナログ信号の 2 元符号化）に関する下記の問いに答えよ．

(i) 標本化ならびに量子化，それぞれの過程の役割と可逆性について説明せよ．

(ii) 標本化周波数を f_S (Hz)，量子化レベル数を L（但し，$L=2^m$ であり m は正の整数とする）としたとき，PCM により符号化されたビット系列のビットレートを f_S および L を用いて表せ．なお，ここでは多重化ならびにフレーム同期などを目的として付加されるビットについて考慮しなくて良い．

(iii) 上記の問い(ii)に対する解答より，PCM により符号化されたビット系列のビットレートを低減するためには，標本化周波数ならびに量子化レベル数をどのように設定すれば良いか説明せよ．また，PCM によるアナログ情報（アナログ信号）の符号化において標本化周波数ならびに量子化レベル数をそのように設定した場合，元のアナログ信号の再現における忠実性においてどのような影響が現れるか説明せよ．

通信方式

2－1　FM 変調において，変調信号を $m(t)$，搬送波の振幅を A，搬送周波数を f_c [Hz]，時刻を t [s] とするとき，FM 変調された信号は，次式で与えられる．

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

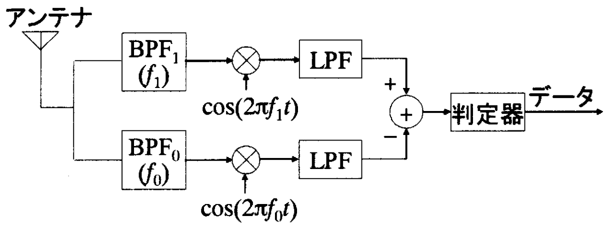
ここで，変調信号が $m(t) = \cos[10^4 \pi t]$ で与えられるとき，以下の問いに答えよ．

(i)　$k_f = 2\pi \times 10^5$ で与えられるとき，最大周波数偏移，最大位相偏移（変調指数）を求めよ．

(ii)　FM 変調された信号の帯域幅がカーソンの法則で規定されるものとして，その値を求めよ．

(iii)　変調信号の周波数を 2 倍にするとき，FM 変調された信号の最大周波数偏移，最大位相偏移，帯域幅を求めよ．

2－2　FSK 同期検波器の構成が右図で与えられるものとする．送信データが「1」の場合の受信信号は $A\cos(2\pi f_1 t)$，送信データが「0」の場合の受信信号は $A\cos(2\pi f_0 t)$ で与えられ，BPF_1 および BPF_0 は，それぞれ周波数 f_1 および f_0 を中心とする必要最小限の周波数成分のみを通過させるフィルタであるものとする．また，BPF_1 の出力に含まれる雑音成分は $n_1(t) = x_1(t)\cos(2\pi f_1 t) - y_1(t)\sin(2\pi f_1 t)$，$BPF_0$ の出力に含まれる雑音成分は $n_0(t) = x_0(t)\cos(2\pi f_0 t) - y_0(t)\sin(2\pi f_0 t)$，$x_1(t)$，$y_1(t)$，$x_0(t)$，$y_0(t)$ は，平均 0，分散 σ^2 のガウス過程に従うランダム変動をするものとするとき，以下の問いに答えよ．



BPF (Band Pass Filter)
LPF (Low Pass Filter)

(i)　図の FSK 同期検波器において，各 BPF の出力，各 LPF の入力，各 LPF の出力，および判定器の入力の数式表現を，送信データが「0」の場合について示せ．ただし図中の LPF は，不要な高周波成分を除去するためのフィルタである．

(ii)　図の FSK 同期検波器において，送信データが「1」の場合の判定器入力を示せ．

(iii)　判定器入力 $r(t)$ の確率密度関数を，送信データが「1」，送信データが「0」の場合に分けて説明せよ．

ネットワーク工学

3．回線速度 C（ビット／秒）の専用全２重回線を用いて，ホストAからホストBへ固定長 L_D（ビット）のパケットを stop-and-wait ARQ プロトコルと呼ばれる以下の手順に従い伝送することを考える．ホストAはパケットを一つ送信した後，ホストBからの ACK を待つ．もし，送信後，タイムアウト時間 T（秒）以前に ACK を受信すれば，直ちに次のパケットの送信を行う．一方，送信後，タイムアウト時間 T までに ACK を受信しなければ，送信は失敗したと見なし，再送する．ホストAは先行するパケットに対する ACK を受信するまで後続のパケットの送信を開始しないことに注意する．他方，ホストBはパケットを正しく受信した場合，直ちにホストAへ固定長 L_A（ビット）の ACK を送信する．また，受信したパケットに誤りがある場合はそれを破棄する（NAK は送信しない）．なお，誤り検査に必要な時間は無視できると仮定する．ホストAとホストBの間の往復伝搬遅延を R（秒）とする．ホストAから送信された任意のパケットに対する ACK をホストAが受信する確率を $1-p$ $(0<p<1)$ として，以下の問いに答えよ．

(i) ホストAがパケットを送信後，タイムアウトが起こる前に，送信したパケットに対する ACK を受信したとする．このとき，ホストAがこのパケットの最初のビットを送信してから ACK の最後のビットを受信するまでの時間 S（秒）を求めよ．

以下の問い (ii)–(v) では $T>S$ と仮定する．

(ii) ホストAが一つのパケットの送信を開始してから最終的に ACK を受け取るまでに必要な時間の平均 H（秒）を S, p ならびに以下で与えられる F を用いて表せ．

$$F = L_D/C + T$$

(iii) ホストAには常に送信待ちパケットがあると仮定する．このとき達成される最大スループット θ を H を用いて表せ．ただし，スループットとは１秒当たりの平均送信成功パケット数を指す．

(iv) ホストBへ送信すべきパケットが到着率 λ の Poisson 過程に従ってホストAに到着すると仮定する．到着したパケットは先着順サービス規律に従ってホストBへ送信される．なお，ホストAには十分に大きなバッファがあり，バッファあふれによるパケット損は起こらないと仮定する．このとき，ホストAで送信待ち状態にあるパケット数が定常状態分布をもつための条件を θ を用いて示せ．

(v) 問い (iv) と同じ状況を考え，問い (iv) で求めた条件が成立していると仮定する．このとき，定常状態において，ホストAのバッファで送信開始を待っているパケット数の平均を S, F, p, λ を用いて表せ．ただし，到着率 λ，利用率 ρ，サービス時間の２次モーメント $b^{(2)}$ をもつ M/G/1 待ち行列モデルの平均待ち時間 W が次式で与えられることを用いて良い．

$$W = \frac{\lambda b^{(2)}}{2(1-\rho)}$$

光・電波工学

4．図に示すように，誘電率，透磁率，および導電率がそれぞれε，μ，およびσの導体が，無限に広い平面を隔てて空気と接している．空気の側から導体に向かって，角周波数ωの平面電磁波を垂直に照射する．このとき，次の問いに答えよ．ただし，空気の誘電率および透磁率をそれぞれε_0およびμ_0とする．

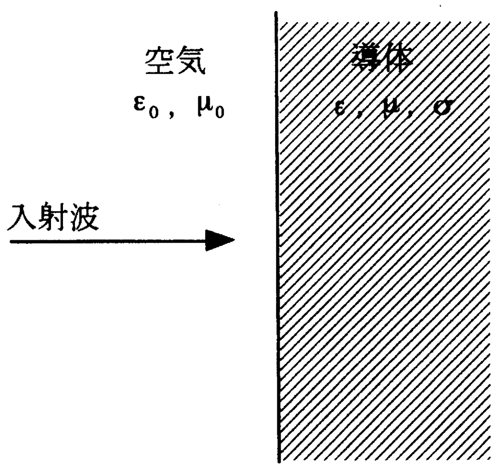
(i) 導体中において，電界ベクトルおよび磁界ベクトルのフェーザ表示$\dot{\mathbf{E}}(x,y,z)$および$\dot{\mathbf{H}}(x,y,z)$が満たす方程式を示せ．

(ii) 導体中の平面電磁波の伝搬定数を求めよ．ただし，導電率σが十分大きく，$\sigma >> \omega\varepsilon$が成り立つものとする（伝搬定数が複素数になることに注意せよ）．

(iii) 入射波が境界面で反射される際の，入射波の位相に対する反射波の位相の変化を求めよ．

(iv) 照射された電磁波電力の時間平均値のうち，導体内で消費される電力の割合を求めよ．

(v) 導体の導電率が$\sigma \to \infty$（無限大）となる極限では，(iii)および(iv)で求めた位相の変化および消費電力の割合がそれぞれどのような値になるかを示せ．



信号処理

5．以下の空欄を埋め，問いに答えよ．

離散時間信号 $x[n], (n$ は整数) に対する離散時間フーリエ変換 (DTFT: Discrete Time Fourier Transform) は，Ω を角周波数を表す実数変数，j を虚数単位とするとき，

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] \exp(-j\Omega n)$$

で定義される．$X(\Omega)$ は $\boxed{(あ)}$ と呼ばれ，周期 $\boxed{(い)}$ の周期関数となる．このDTFTをコンピュータなどを用いて実際に計算するには (a) 問題があることも知られている．

そこで，実際には次に示す (b) 離散フーリエ変換(DFT: Discrete Fourier Transform) を用いる．DFTは長さ N の離散時間信号（データ系列）$x[n], (n = 0, ..., N-1)$ と長さ N の $\boxed{(あ)}$ 列 $X[k], (k = 0, ..., N-1)$ との変換（N 点DFTと呼ぶ）を定義したもので，

$$X[k] = \boxed{(う)} W_N^{nk}, \quad 但し, W_N = \exp(\boxed{(え)})$$

である．N 点DFTを直接計算するには高々，$\boxed{(お)}$ 回の複素乗算と $\boxed{(か)}$ 回の複素加算で十分である．

高速フーリエ変換 (FFT: Fast Fourier Transform) は，DFTを計算する際の乗加算回数を減らすために考案されたアルゴリズムで，ある問題を部分問題に分割し，それらを解き，解を統合することによって元の問題の解を与える $\boxed{(き)}$ 法に基づくものである．具体的には，N 点DFT(ここでは N を2のべき乗とする) を2回の $\boxed{(く)}$ 点DFTに分割し，さらに $\boxed{(く)}$ 点DFTを2回の $\boxed{(け)}$ 点DFTに分割するという操作を繰り返し，最終的に $\boxed{(こ)}$ 点DFTに帰着させるのがFFTの考え方である．

FFTのアルゴリズムとして (c) 時間間引きFFTと周波数間引きFFT が知られており，何れの場合も N 点DFTを高々，$\boxed{(さ)}$ 回の複素乗算と $\boxed{(し)}$ 回の複素加算で実行することができる．また (d) 逆離散フーリエ変換 (IDFT : Inverse DFT) もFFTを用いて高速に計算すること が可能である．

(i) 空欄（あ）〜（し）に当てはまるものを示せ．ただし，記号が同じ空欄には同じ字句が入るので注意すること．

(ii) 下線部 (a) について，どのような問題があるのか具体的に述べよ．

(iii) 下線部 (b) について，離散時間フーリエ変換DTFTと離散フーリエ変換DFTの関係，すなわち $X(\Omega)$ と $X[k]$ の関係について説明せよ．

(iv) 下線部 (c) について，時間間引きFFTあるいは周波数間引きFFTを用いて，4点DFTを計算するFFTの信号流れ図（バタフライ演算を基本計算単位とする）を図示せよ．また，データ系列 $(x[0], x[1], x[2], x[3]) = (0, 8, 2, 4)$ の4点DFTを求めよ．

(v) 下線部 (d) について，FFTによるIDFTのアルゴリズムを記述せよ．また，問い (iv) における4点DFTの計算結果にそのアルゴリズムを適用し，元のデータ系列に戻ることを示せ．

コンピュータ工学（ハードウェア）

6－1 整数部n桁，小数部m桁の2進数Aを固定小数点表現で

$$A = a_n 2^n + \sum_{i=-m}^{n-1} a_i 2^i \quad (0 \le a_i \le 1)$$

と表す．最上位桁$a_n 2^n$（$n+1$桁目）を符号部と呼び，正数のときa_n=0，負数のときa_n=1 とし，残りを数値部$|A|$と呼ぶ．以下の問いに答えよ．

(i) 整数部n桁，小数部m桁で表される負数Aの2の補数表現$\overline{\overline{A}}$の数値部 $|\overline{\overline{A}}|$ に関して，$|\overline{\overline{A}}| = 2^n - |A|$が成り立つことを示せ．

(ii) 10進数の −18.242 の固定小数点2進数表現（整数部5桁，小数部3桁）の (a) 符号－絶対値表現，(b) 1の補数表現，(c) 2の補数表現をそれぞれ示せ．なお，2進数Aの1の補数表現を $(\overline{A})_{c1}$，2の補数表現を$(\overline{\overline{A}})_{c2}$と表すものとする．

(iii) 1の補数で表現された整数の負数A, Bを加算する際には，加算結果において，最上位ビットの桁上がり2^{n+1}を無視して，1を減算すればよい．小数部がなく，またオーバーフローは起こらないとして，このことが成り立つことを示せ．

(iv) 2進数の除算に関する以下の文章中の空欄に当てはまるものを示せ．なお，同じ記号の欄には同じ内容が入る．

2進数の除算も10進数の場合と同様に被除数から除数を引くという操作を各桁について行えばよい．いま，2進数の正の整数A, Bについて，$A \div B$ を実行する場合を考え，計算途中で除数Bを減算した結果をRとする．この場合，減算後の次の操作はRに関する条件により次のように変わる．ただし、2は左に1ビットシフトする操作を表す．また，空欄ア、ウにはRに関する条件式が入る．

(a) $2R-B$ （$\boxed{\text{ア}}$）
(b) $2(\boxed{\text{イ}})-B$ （$\boxed{\text{ウ}}$）

また，上記 (b) の操作を単純化して計算の高速化を図る手法があり，その場合，上記 (b) の操作は以下のようになる．

(b') $\boxed{\text{エ}}$ （$\boxed{\text{ウ}}$）

たとえば，$(55)_{10} \div (3)_{10}$をした場合，(b)を用いた場合の減算回数は$\boxed{\text{オ}}$回であり，(b')を用いた場合の減算回数は$\boxed{\text{カ}}$回である．

6－2 以下の演算を，命令フェッチ(IF)，命令デコード(D)，オペランドフェッチ(OF)，命令の実行(EX)，結果の格納(S)の 5 ステージ命令パイプライン制御により実行する．なお，命令語，データ，アドレスの長さはいずれも 1 語であり，主記憶装置へのデータバスは 1 本であるとする．以下の問いに答えよ．

1. (B) = (A) + R1　　アドレス A の主記憶の内容とレジスタ R1 の内容を加算し，主記憶のアドレス B に格納する．
2. (D) = (C) – R2　　アドレス C の主記憶の内容からレジスタ R2 の内容を減算し，主記憶のアドレス D に格納する．

(i) これらの演算を以下の機械命令順で実行する場合について，表 1 のような表を解答用紙に作成し，各クロックサイクルにおいて実行されるステージの名前を，表 1 中の命令 1 の例のように，上記の記号を用いて記入し，表 1 を完成させよ．なお，表は必要に応じて大きくし，またクロックサイクルとの関係が明確に分かるように記述すること．

1. (A) → R3　　主記憶アドレス A の内容(A)をレジスタ R3 に格納する
2. Add R3, R1, R4　　レジスタ R3 とレジスタ R1 の内容を加算し，結果をレジスタ R4 に格納する．
3. [R4] → B　　レジスタ R4 の内容[R4]を主記憶アドレス B に格納する．
4. (C) → R5　　主記憶アドレス C の内容(C)をレジスタ R5 に格納する．
5. Sub R5, R2, R6　　レジスタ R5 の内容からレジスタ R2 の内容を減算し，結果をレジスタ R6 に格納する．
6. [R6] → D　　レジスタ R6 の内容[R6]を主記憶アドレス D に格納する．

表 1

		クロックサイクル								
		1	2	3	4	5	6	7	....	
1	(A) → R3	IF	D	OF	EX	S				
2	Add R3, R1, R4									
3	[R4] → B									
4	(C) → R5									
5	Sub R5, R2, R6									
6	[R6] → D									

(ii) 一般に，機械命令の実行順序を工夫することで，パイプラインの乱れ（ストール）を減らし，実行効率を向上させることが可能である．必要なクロックサイクルが最小になるように，問い(i)における命令の実行順序(1～6)を変更して，表 2 を完成させよ．ただし，表 2 にあるように，1 番目に (A) → R3，3 番目に Add R3, R1, R4 を実行する命令実行順序だけを考えよ．

表 2

		クロックサイクル								
		1	2	3	4	5	6	7	....	
1	(A) → R3	IF	D	OF	EX	S				
2										
3	Add R3, R1, R4									
4										
5										
6										

コンピュータ工学（ソフトウェア）

7－1．図7－1のように，親の保持するデータ値が子の保持するデータ値以下となる2分木に対して，図7－2に示す配列による実現をヒープと呼ぶ．プログラムAは，図7－2の配列heapを，データ値の大きさに応じてソートするプログラムである．以下の問いに答えよ．

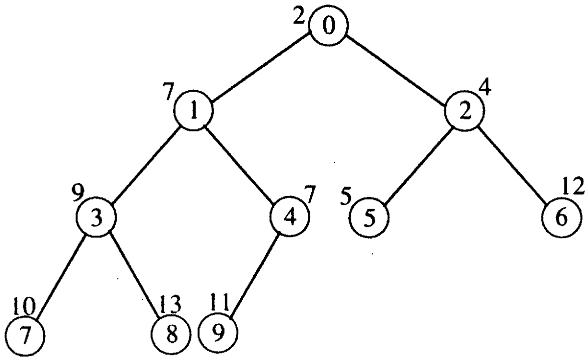


図7－1　　ヒープの条件を満たす2分木
（円内は節点番号，肩の数字はデータ値）

2	7	4	9	7	5	12	10	13	11

heap[0] ････････････････････････ heap[9]

図7－2　　図7－1に示す2分木のヒープによる実現

プログラムA

```
#include <stdio.h>
#define N 10
void deletemin(int heap[ ], int *n)
{
      int p,q,min,tmp;
      if(*n < 0){ printf("Empty ¥n"); }
      else{
              min=heap[0]; heap[0]=heap[*n]; *n = *n-1; p=0;
              while(p*2+1 <= *n){
                  q=p*2+1;
                  if(p*2+2 <= *n && [    (1)    ] > heap[p*2+2]){ q= [    (2)    ] ; }
                  if([    (3)    ] ) break;
                  else{ tmp=heap[p]; heap[p]=heap[q]; heap[q]=tmp; }
                  p=q;
               }
               heap[*n+1]=min;
       }
}
int main()
{
      int k,n,heap[N]={2,7,4,9,7,5,12,10,13,11};
      n=N-1;
      for(k=0;k<N-1;k++){
          deletemin(heap, &n);
                  < α >
      }
}
```

(i) 図7－1において，親の節点番号を i とすると，その左の子と右の子の節点番号は，i を用いてどのように表されるか述べよ．但し，i=0,1,2,3 とする．

(ii) プログラムＡの空欄(1)，(2)，(3)を埋めよ．

(iii) プログラムＡの main 関数における＜ α ＞の位置において，k=2 のときの heap[0]，heap[1] ‥‥ heap[9]の値を示せ．

(iv) プログラムＡの計算量のオーダは，データ数 N を用いて $O(N\log N)$ となることを示せ．

7－2．プログラムＢは，図7－3のグラフにおいて，始点となる節点０から他のすべての節点までの最短経路を求めるダイクストラ法のプログラムである．このプログラムは，節点０から最短距離の未探索節点を探索済み節点集合へ逐次加える．以下の問いに答えよ．

(i) プログラムＢの空欄(4)，(5)，(6)を埋めよ．但し，(5)は２カ所あるが，同一の内容が入るので注意すること．

(ii) プログラムＢの下から３行目の printf 文に関して，k=2 の時の出力結果を示せ．

(iii) プログラムＢを変更し，図７―４の有向グラフに対する最短経路を求めるプログラムを答えよ．但し，解答にはプログラム全体ではなく，すべての変更箇所について変更前と変更後を対比させる形で記せ．

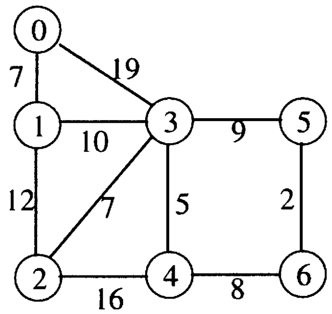


図７－３　対象とする無向グラフ
(円内は節点番号，枝に付された数字は節点間の距離)

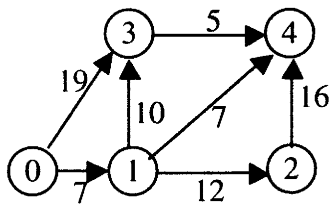


図７－４　対象とする有向グラフ

プログラムB

```
#include <stdio.h>
#define BN 100000
#define N 7
#define START 0
void sub(int *next, int *min, int  (4) , int distance[], int visited[], int prev[])
{   int i,j;
    i=*next; visited[i]=1; *min=BN;
    for(j=0;j<N;j++){
       if(visited[j]) continue;
       if(a[i][j] < BN &&     (5)      < distance[j]){
            distance[j] =     (5)     ;
            prev[j] = i;
       }
       if(distance[j]<(*min)){ *min=distance[j];     (6)     ;}
    }
}
int main()
{  int i,k,next,min,distance[N],visited[N],prev[N];
   int a[N][N] = { 0, 7,BN,19,BN,BN,BN,
                   7, 0,12,10,BN,BN,BN,
                  BN,12, 0, 7,16,BN,BN,
                  19,10, 7, 0, 5, 9,BN,
                  BN,BN,16, 5, 0,BN, 8,
                  BN,BN,BN, 9,BN, 0, 2,
                  BN,BN,BN,BN, 8, 2, 0};
   for(i=0;i<N;i++){ visited[i]=0; distance[i]=BN; }
   distance[START]=0; next=START;

   for(k=0;k<N-1;k++){
       sub(&next, &min, a, distance, visited, prev);
       printf("k=%d  %d %d %d¥n",k,next,min,prev[next]);
   }
}
```