平成 29 年 4 月入学及び平成 28 年 9 月入学
大学院修士課程・専門職学位課程入学試験

情報理工学院　情報工学系
筆答専門試験科目
想　定　問　題

平成 28 年 1 月

東京工業大学

※　出題される分野、問題数等本想定問題の内容は、実際の試験問題とは異なる場合があります。

※　各系の試験概要については，２月上旬に公表予定です。

※　本入学試験にかかる募集要項は，４月上旬に本学ホームページで公表し，志願票等を含む冊子を５月上旬より配布する予定です。

専門科目　　　　　　　　　　　大修

情報工学系

注　意　事　項

1. 各解答用紙に受験番号を記入せよ．
2. 次の５題のうち，１番～３番の３題は必ず解答し，４番と５番からは１題を選び解答すること．
3. 解答は１題ごとに別々の解答用紙に，問題番号を明記した上で記入せよ．必要であれば，解答用紙の裏面に記入して良いが，解答用紙の表面にその旨を明記すること．
4. １枚の解答用紙に２題以上の解答を記入した場合はそれらの解答を無効とすることがある．
5. １題の解答を２枚以上の解答用紙に記入した場合はその解答を無効とすることがある．
6. 電子式卓上計算機等の使用は認めない．

1. xy 平面上において，原点を中心とする半径 a の円の内部（円周上の点は含まない）を領域 D とし，

$$I = \int\int_D e^{-x^2-y^2} dxdy \tag{1.1}$$

とする．以下の問に答えよ．

1) $x = r\cos\theta, y = r\sin\theta$ と変数変換したとき，領域 D の範囲を r, θ で表せ．

2) 上記の変換に対するヤコビ行列 J

$$J = \frac{\partial(x, y)}{\partial(r, \theta)}$$

および，その行列式を求めよ．

3) 式 (1.1) を計算し，a を用いて表せ．

4) $\lim_{a\to\infty} I$ の値を求めよ．

5) 4) の結果を利用して，

$$\int_{-\infty}^{\infty} e^{-x^2} dx$$

の値を求めよ．

6) ガンマ関数

$$\Gamma(x) = \int_0^\infty e^{-t}t^{x-1}dt$$

について，$\Gamma(\frac{1}{2})$ の値を求めよ．

2. 以下の数理論理学の問題に答えよ.

1) 以下の命題論理の問題に答えよ.

a) α, β, γ を命題変数とする．以下の命題論理式がトートロジー (恒真式) で あるか否かを述べよ.

イ)　$\alpha \vee (\beta \Rightarrow \alpha)$

ロ)　$\alpha \vee (\alpha \Rightarrow \beta)$

ハ)　$\alpha \Rightarrow (\beta \Rightarrow \alpha)$

ニ)　$((\alpha \vee \beta) \Rightarrow \gamma) \Rightarrow (\alpha \Rightarrow \gamma)$

b) ブール関数 $H(\alpha, \beta)$ と $G(\alpha, \beta)$ の真理値表がそれぞれ以下のように与えられている．α, β は命題変数であり，T は真，F は偽を表す.

α	β	$H(\alpha,\beta)$
F	F	T
F	T	T
T	F	T
T	T	F

α	β	$G(\alpha,\beta)$
F	F	T
F	T	T
T	F	F
T	T	T

イ)　$\neg\alpha$ を関数 H のみを用いて表せ.

ロ)　$\alpha \wedge \beta$ を関数 H のみを用いて表せ.

ハ)　$\alpha \vee \beta$ を関数 H のみを用いて表せ.

ニ)　$G(\alpha, \beta)$ を関数 H のみを用いて表せ.

2) 以下の述語論理の問題に答えよ.

a) x, y, z が自然数であるとき，次の各論理式の真偽を理由と共に述べよ.

イ)　$\exists x \forall y ((x < y) \vee (x = y))$

ロ)　$\forall x \forall y \exists z ((x < z) \wedge (y < z))$

ハ)　$\forall x \forall y \exists z ((x < y) \Rightarrow ((z < x) \vee (y < z)))$

ニ)　$\forall x \forall y ((x < y) \Rightarrow \exists z ((x < z) \wedge (z < y)))$

b) x, y, z が実数であるとき，上記イ)，ロ)，ハ)，ニ) の各論理式の真偽を理由と共に述べよ.

3. 異なる n 個のものから重複を許さず k 個を選ぶ組合せの数 ${}_nC_k$ の求め方として，再帰的な定義を用いる方法や，階乗による定義 ${}_nC_k = \frac{n!}{k!(n-k)!}$ を用いる方法などがある．図 3.1 のパスカルの三角形の最上段を 0 段目，最左を 0 番目とすると，n 段目の左から k 番目の数は ${}_nC_k$ を表している．${}_nC_k$ を求める C 言語の関数 comb1, comb2, comb3 について，以下の問に答えよ．

```
int comb1(int n, int k) {
  if ((k == 0) || (k == n)) return 1;
  else return comb1([A],[B]) + comb1([C],[D]);
}
int comb2(int n, int k) {
  int i, j;
  if (n >= 2 * k) {
    i = n - k + 1;
    j = k;
  } else {
    i = k + 1;
    j = n - k;
  }
  return pf([E],[F]) / pf([G],[H]);
}
int pf(int m, int n) {
  int i, f;
  f = 1;
  for (i = m; i <= n; i++) {
    f = f * i;
  }
  return f;
}
```

```
int comb3(int n, int k) {
  int i, j;
  int a[MAXARG];
  if (n - k < k) k = n - k;
  if (k == 0) return 1;
  if (k == 1) return n;
  for (i = 0; i < k; i++) {
    a[i] = i + 2;
  }
  for (i = 3; i <= [I]; i++) {
    a[0] = i;
    for (j = 1; j < k; j++) {
      a[j] += a[j - 1];
    }
  }
  return [J];
}
```

1) 関数 comb1 の $\boxed{\mathrm{A}}$, $\boxed{\mathrm{B}}$, $\boxed{\mathrm{C}}$, $\boxed{\mathrm{D}}$を答えよ．

2) ${}_7C_3$ を comb1(7,3) で求める際に，関数 comb1 が呼び出される回数を答えよ．ただし最初の呼び出し comb1(7,3) も 1 回と数えるとする．

3) 関数 comb2 の $\boxed{\mathrm{E}}$, $\boxed{\mathrm{F}}$, $\boxed{\mathrm{G}}$, $\boxed{\mathrm{H}}$を答えよ．

4) 関数 comb1 の関数 comb2 に対する，実行時の長所と短所を述べよ．

5) パスカルの三角形における ${}_2C_1$, ${}_3C_2$, ..., ${}_{k+1}C_k$ を初期値とする配列 a (a[0] $= {}_2C_1$, a[1] $= {}_3C_2$, ...) を用いて ${}_nC_k$ を計算する関数 comb3 の$\boxed{\mathrm{I}}$, $\boxed{\mathrm{J}}$を答えよ．また計算時に配列 a の内容がどう更新されるか，図 3.1 を用いて説明せよ．ただし MAXARG は k より大きいものとする．

```
                1
              1   1
            1   2   1
          1   3   3   1
        1   4   6   4   1
      1   5  10  10   5   1
    1   6  15  20  15   6   1
  1   7  21  35  35  21   7   1
1   8  28  56  70  56  28   8   1
.....................................
```

図 3.1 パスカルの三角形

4. 以下の式で定義された関数 $f(x)$ と関数 $F(\omega)$ の間の変換をフーリエ (Fourier) 変換という．ただし，i は虚数単位である．このとき，以下の設問に答えよ．

$$F(\omega) \equiv \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx$$

1) 図 4.1 に示す関数 $f_1(x)$ のフーリエ変換 $F_1(\omega)$ を求めて，ω の関数として図示せよ．ただし a は正数とする．

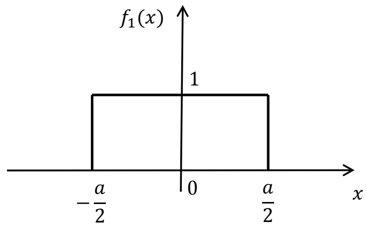

図 4.1

2) 図 4.2 に示す関数 $f_2(x)$ のフーリエ変換 $F_2(\omega)$ を求めて，ω の関数として図示せよ．ただし a は正数とする．

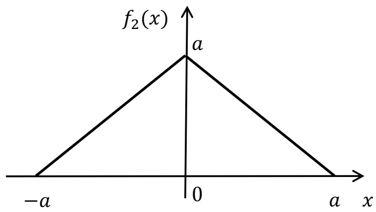

図 4.2

3) 二つの関数 $f(x)$ と $g(x)$ のたたみこみ積分 $h(x)$ は次式で定義される．

$$h(x) \equiv \int_{-\infty}^{\infty} f(x-y) g(y) dy$$

このとき，それぞれのフーリエ変換に対して次の関係が成り立つことを示し，係数 k を求めよ．

$$H(\omega) = kF(\omega)G(\omega)$$

4) たたみこみ積分の考え方を用いて，1) の関数 $f_1(x)$ と 2) の関数 $f_2(x)$ の関係を考察し，簡潔に述べよ．

5.

1) 以下の論理代数（ブール代数）の等式を証明せよ．ただし $x \vee y$ は x と y の論理和（OR），xy は x と y の論理積（AND），$\overline{x}$ は x の否定（NOT）を表す．

 a) $(x_1 \vee \overline{x_2})(x_2 \vee \overline{x_3})(x_3 \vee \overline{x_1}) = x_1 x_2 x_3 \vee \overline{x_1}\,\overline{x_2}\,\overline{x_3}$

 b) $(x_1 \vee \overline{x_2})(x_2 \vee \overline{x_3}) \cdots (x_{n-1} \vee \overline{x_n})(x_n \vee \overline{x_1}) = x_1 x_2 \cdots x_n \vee \overline{x_1}\,\overline{x_2} \cdots \overline{x_n} \qquad (n \geq 2)$

2) 4 進同期式カウンタを実現する順序回路の設計を考える．入力を x，出力を y の各 1 ビット，現在の状態を $q_1\,q_0$，次の状態を $q_1'\,q_0'$ の各 2 ビットとする．初期状態は $q_1\,q_0 = 00$ とする．$x = 1$ が入力されるたびに，$q_1\,q_0$ の値は $00 \to 01 \to 10 \to 11 \to 00 \to \cdots$（以後，同じ繰り返し）と変化する．$x = 0$ の時，$q_1\,q_0$ の値は変化しない．$x = 1$ かつ $q_1\,q_0 = 11$ の時のみ $y = 1$ となり，それ以外では $y = 0$ とする．

表 5.1 : カルノー図

x \ $q_1\,q_0$	00	01	11	10
0				
1				

 a) q_1', q_0', y それぞれについて，x, q_1, q_0 によるカルノー図を示せ．カルノー図の作成には表 5.1 の形式を参考にせよ．

 b) a) の解を用いて，x, q_1, q_0 に関する NOT-AND-OR 形式（「積項の和」形式，積和標準形）で q_1', q_0', y それぞれを表せ．ただし，項数が最小となるように簡単化すること．

 c) b) の解を用いて，順序回路図を示せ．2 入力の NAND ゲート，3 入力の NAND ゲート，および立ち上がり動作のエッジトリガ型 D フリップフロップのみを用いること．クロックへの入力は順序回路図では省略して良い．

3) メモリキャッシュについて，空欄 $\boxed{\text{A}}$ 〜 $\boxed{\text{P}}$ に最も適切な語句を，図 5.3 の選択肢群から選んで，その番号を解答せよ．同じ選択肢を何度選んでも良い．

 a) メモリキャッシュは CPU とメモリの間に配置する記憶装置であり，メモリよりも小容量で高速に動作する．一般的にメモリアクセスには参照の $\boxed{\text{A}}$ があるため，メモリキャッシュにメモリ中のデータの一部を一時的に保存することで，全体の平均アクセス時間を向上させることができる．アクセスされたメモリがすぐに再度アクセスされる可能性が高いことを $\boxed{\text{B}}\boxed{\text{A}}$ といい，アクセスされたメモリと近い番地のメモリが，同時あるいはすぐにアクセスされる可能性が高いことを $\boxed{\text{C}}\boxed{\text{A}}$ という．

 b) メモリキャッシュの書き込み操作には大きく 2 つの方式がある．$\boxed{\text{D}}$ 方式は書き込み操作をメモリキャッシュとメモリの両方に同時に行うため，メモリキャッシュとメモリの内容が常に一貫しているが，書き込み速度は向上しない．一方，$\boxed{\text{E}}$ 方式はメモリキャッシュのみに書き込みを行い，メモリへの書き込みはフラッシュ時などに遅延して実行されるため，書き込み速度が向上する．

（次ページにつづく）

（前ページのつづき）

c) メモリ中のデータをメモリキャッシュのどこに格納することを許すか，すなわちメモリとメモリキャッシュの対応づけの違いで，大きく 3 つの方式がある． $\boxed{\text{F}}$ 方式はあるメモリのデータの格納をメモリキャッシュの一箇所のみに許す（図 5.2 の ①）． $\boxed{\text{G}}$ 方式はメモリキャッシュ内で事前に限定された一部の複数箇所への格納を許し，その複数箇所から 1 つを選んで格納する（図 5.2 の ②）． $\boxed{\text{H}}$ 方式は全ての場所への格納を許し，1 つを選んで格納する（図 5.2 の ③）．この 3 つの方式のうち，一般的に最もキャッシュミスしやすいのは $\boxed{\text{I}}$ 方式である．またキャッシュミスしない場合のアクセス時間が最も短いのは $\boxed{\text{J}}$ 方式である．

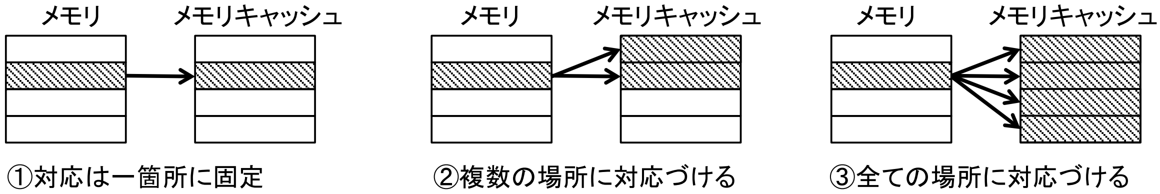

図 5.2：メモリとメモリキャッシュの対応関係

d) キャッシュ置換アルゴリズムとは，メモリ中のデータを新たにメモリキャッシュに格納するために，メモリキャッシュ中から捨てるデータを選択するアルゴリズムのことであり，様々な方式がある． $\boxed{\text{K}}$ は最近のある一定期間内のアクセス頻度が最少のデータを捨てる． $\boxed{\text{L}}$ はメモリキャッシュにデータを格納した時刻が最も古いデータを捨てる． $\boxed{\text{M}}$ は最後のアクセス時刻が最も古いデータを捨てる．

e) メモリ中のデータをメモリキャッシュに転送する方式には大きく 2 つの方式がある． $\boxed{\text{N}}$ はプログラムがデータにアクセスしたときにそのデータを転送する．一方， $\boxed{\text{O}}$ はプログラムがデータにアクセスする前にそのデータを転送する．この 2 つの方式のうち， $\boxed{\text{B}}\boxed{\text{A}}$ が無く，次にアクセスするデータを予測しやすい処理の場合，例えば巨大な配列の各要素を先頭から逐次的に一回ずつアクセスする場合などは， $\boxed{\text{P}}$ の方が性能の向上を期待できる．

(1) インダイレクト・マップ， (2) 完全性， (3) 局所性， (4) 空間的， (5) 健全性，
(6) 構造的， (7) 時間的， (8) スワッピング， (9) 脆弱性， (10) セグメンテーション，
(11) 絶対的， (12) セット・アソシアティブ， (13) 相対的， (14) ダイレクト・マップ，
(15) デマンドフェッチ， (16) 透過性， (17) バッファリング， (18) ビジーウェイト，
(19) 物理的， (20) プリフェッチ， (21) フル・アソシアティブ， (22) ページング，
(23) ポーリング， (24) ライトスルー， (25) ライトバック， (26) ラウンドロビン，
(27) ランダム， (28) 論理的， (29) 割り込み， (30) FIFO， (31) LFU， (32) LRU

図 5.3：選択肢群