

令和 8 年度大学院[春・秋]入学 3 月試験

情報科学専攻

論 述 試 験

【60 分】

[注意事項]

1. 試験開始の合図があるまで、この問題冊子を開かないでください。
2. この問題冊子には論述試験の問題が記載されています。万一、落丁・乱丁があった場合は、手をあげて申し出てください。
3. 受験票、黒鉛筆、シャープペンシル、消しゴム、鉛筆けずり、時計、眼鏡、ハンカチ、目薬、袋から取り出したティッシュペーパー以外は机上に置かないでください。
4. 監督者の指示に従い、解答用紙に受験番号・氏名を記入してください。
5. 試験終了の合図で解答を終了し、監督者の指示があるまで席を立たないでください。また問題冊子は持ち帰ってください。

以下の問題群（Q 1～Q 6）の中から 2 問を選択し、解答せよ。選択した問題番号は解答用紙の指定箇所に記入すること。Q1～Q5 の問題に対しては、解答用紙の裏面を利用しても構わない。

Q1

5 つの情報源記号 $S = \{A, B, C, D, E\}$ に対して、以下の表のように発生確率を割り当てたとき、以下の(1)～(3)の問いに答えよ。

記号	A	B	C	D	E
発生確率	0.40	0.15	0.15	0.15	0.15

- (1) この表の記号に対する平均情報量を求めよ。答えは小数点以下第 2 位まで（第 3 位を四捨五入）求めること。ただし、 $\log_2 3 \approx 1.585$ 、 $\log_2 5 \approx 2.322$ とする。
- (2) この表の各情報源記号に対し、シャノン・ファノ符号化を行い、各情報源記号に対し割り当てられるビットを答えよ。
- (3) (2)で求めたシャノン・ファノ符号化の結果よりも、平均符号長の短い符号を考え、各情報源記号に対し割り当てられるビットを答えよ。また、その符号が(2)の符号よりも平均符号長が短いこと、および、瞬時符号であることも示すこと。

解答例：

(1) 平均情報量は以下で求まる

$$\begin{aligned}
 H &= -(0.40 \times \log_2 0.40 + 4 \times 0.15 \times \log_2 0.15) \\
 &= -(0.40 \times (1 - \log_2 5) + 0.6 \times (\log_2 3 - 2 - \log_2 5)) \\
 &= -(0.40 \times (1 - 2.322) + 0.6 \times (1.585 - 2 - 2.322)) \\
 &= 2.171 \approx 2.17 \text{ (bit, bit/記号)}
 \end{aligned}$$

(2) シャノン・ファノ符号化は、以下のようなになる

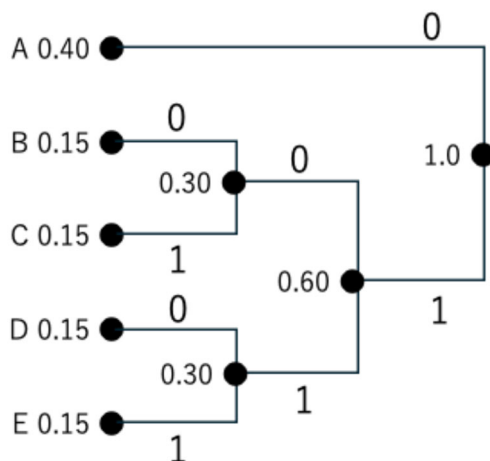
記号	確率	分割 1	分割 2	分割 3
A	0.40	0	0	
B	0.15		1	
C	0.15	1	0	
D	0.15		1	0
E	0.15	1	1	1

結果、A=00, B=01, C=10, D=110, E=111 となる。

ただし、0/1 はビット反転していても良い。また、記号 BCDE の割当はどのような組み

合わせでも良い．2 ビット割当の記号が 3 つ，3 ビット割当の記号が 2 つとなっていること．また，瞬時符号ではない（符号木の節に符号が割り当てられるような符号化）は NG とする．

(3) ハフマン符号化を行えば良い．ハフマン符号化は，以下のようになる．



結果，A=0，B=100，C=101，D=110，E=111 となる．

ただし，0/1 はビット反転していても良い．また，記号 BCDE の割当はどのような組み合わせでも良い．1 ビット割当の記号が 1 つ，3 ビット割当の記号が 4 つとなっていること．また，瞬時符号ではない（符号木の節に符号が割り当てられるような符号化）は NG とする．

シャノン・ファノ符号化の結果とハフマン符号化の結果について，それぞれ平均符号長を計算する．

シャノン・ファノ符号化の平均符号長

$$0.4 \times 2 + 0.15 \times 2 + 0.15 \times 2 + 0.15 \times 3 + 0.15 \times 3 = 2.3 \text{ (bit, bit/記号)}$$

ハフマン符号化の平均符号長

$$0.4 \times 1 + 0.15 \times 3 + 0.15 \times 3 + 0.15 \times 3 + 0.15 \times 3 = 2.2 \text{ (bit, bit/記号)}$$

以上より，ハフマン符号化の平均符号長が短いことがわかる．なお，瞬時符号であることは，ハフマン符号化の符号木より明らかである．クラフトの不等式が成立することを示しても，必ずしも瞬時符号であるとは限らないため，符号木を示す必要がある．

Q2

グレースケール画像に対して、以下に示すオペレータを用いたフィルタ処理はどのような結果を得るための処理なのか、また、図中のパラメータ α を大きくした場合（あるいは小さくした場合）、どのような効果が得られるのか、理由もあわせて答えよ。ただし、 $\alpha > 0$ とする。

$$\begin{pmatrix} -\alpha & -\alpha & -\alpha \\ -\alpha & 1+8\alpha & -\alpha \\ -\alpha & -\alpha & -\alpha \end{pmatrix}$$

ただし、 $\alpha > 0$

図1 オペレータ

解答例：

画像の輪郭や細部を強調して、くっきりさせる鮮鋭化を行うフィルタ処理です。具体的には、図1のオペレータは、次式のように、元の画像そのままを得るオペレータから、8近傍でのラプラシアンオペレータに係数 α を掛けたものを引いて求めることができます。

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} - \alpha \begin{pmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

この α を大きくすることによって、輪郭付近での、明るさが変わる境界の濃淡差が大きくなるため、画像をよりくっきりとさせた鮮鋭化を行うことができます。ただし、 α を大きくしすぎると輪郭や細部が強調され過ぎて不自然な画像になってしまいます。

Q3

オブジェクト指向プログラミング言語における抽象クラスの関数（メソッド）について論ぜよ。必要に応じて図を作成して解答しても良い。

解答例：

抽象クラスは、インスタンス（オブジェクト）を生成せずにもっぱらサブクラスの実装を前提として作成される。この時、抽象クラスで宣言された関数は、1) 共通の機能を提供する、2) 共通の機能に同一の名称を提供する、という、それぞれ二つの目的を持つ。

1) 共通の機能を提供することで、同一の抽象クラスから継承されたサブクラスでは、機能の実装が不要になり、共通の機能の修正が行われた場合に個別のサブクラスの関数の修正が不要である。

例えば、抽象クラスとして、Reader クラスを作成し、サブクラスとして、ファイルシステムからデータを読み込む FileReader クラスと、ネットワークからデータを読み込む SocketReader クラスを実装してする。

Reader クラスで、「データを最後まで一括で読み込む readAll() 関数」を共通機能として実装する。これにより、サブクラスである FileReader クラス、SocketReader クラスは個別に一括読み込みの処理を書く必要がなくなる。

2) 共通の機能に同一の名称を提供することで、同一の抽象クラスから継承されたサブクラス間では、実装が異なっているとしても、同一の名称が割り当てられるので、プログラマが動作を理解しやすいと期待できる。

例えば、抽象クラスとして、Reader クラスを作成し、サブクラスとして、ファイルシステムからデータを読み込む FileReader クラスと、ネットワークからデータを読み込む SocketReader クラスを実装して、それぞれの実装（ファイルシステムとネットワーク）に依存した read() 関数を作成する。

Q4

次のプログラムは、配列の要素をあるアルゴリズムに従って並び替えるものである。このプログラムについて、以下の(1)～(3)の問いに答えよ。

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

```
void PrintData (int n, int data []) {  
    for (int i= 0; i< n; i++) {  
        printf("%d ", data[i]);  
    }  
    printf("%n");  
}
```

```
void Swap (int* x, int* y) {  
    int tmp = *x;  
    *x = *y;  
    *y = tmp;  
}
```

```
void Sort(int n, int data []) {  
    PrintData (n, data);
```

```

for (int m = n; m >= 2; m--) {
    int swapped = 0;

    for (int i = 0; i < m - 1; i++) {
        if (data[i] > data[i + 1]) {
            Swap(&data[i], &data[i + 1]);
            PrintData(n, data);
            swapped = 1;
        }
    }

    if (!swapped) break;
}

int main(void) {
    int data[5] = {3, 7, 1, 9, 5};
    Sort(5, data);
    return 0;
}

```

- (1) Swap 関数で行われる処理について「仮引数」、「ポインタ」、「間接参照」という用語を使って説明せよ。
- (2) このプログラムをコンパイル・実行したときに表示される内容を答えよ。
- (3) 配列の要素を並び替えるアルゴリズムについて、このプログラムで実装されているもの以外のアルゴリズムを一つ挙げ、データ数が大きい場合にどちらのアルゴリズムを利用すべきか説明せよ。

解答例：

(1)

仮引数 **x** と **y** は **int** 型へのポインタであり、それぞれ呼び出し元から渡された整数変数のアドレスを受け取る。このようにポインタを仮引数にすることで、関数の外部にある変数の値を間接参照によって変更できる。まず、ポインタ **x** が指すアドレスを間接参照して得た値を一時変数 **tmp** に代入し、元の値を退避する。次に、ポインタ **x** が指す先に、ポインタ **y** が指す先の値を代入する。最後に、ポインタ **y** が指すアドレスに、退避しておいた **tmp** の値を代入する。これにより、2 変数の値が交換される。

(2)

3 7 1 9 5

3 1 7 9 5

3 1 7 5 9

1 3 7 5 9

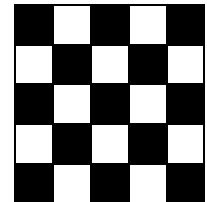
1 3 5 7 9

(3)

このプログラムはバブルソートを実装したものである。バブルソートの平均時間計算量のオーダーは $O(n^2)$ である。一方、他のアルゴリズムとしてクイックソートがある。クイックソートの平均時間計算量のオーダーは $O(n \log n)$ であり、プログラムとして実装した場合に、要素同士の比較や転送などの演算を考慮した手間が平均的に小さいことが知られているため、データ数が増加してもバブルソートほど急激には処理時間が増えない。したがって、データ数が大きい場合には、バブルソートよりもクイックソートの方が高速にソートできることが期待される。

Q5

右に示す1枚の白黒の画像（画像 A とする）は、 5×5 の 25 画素で構成され、黒色を 0、白色を 1 で表現する。各ピクセルは最低 1 ビットで表現できるため、この画像は 25 ビットで表現できる。以下の(1)～(5)の問いに答えよ。

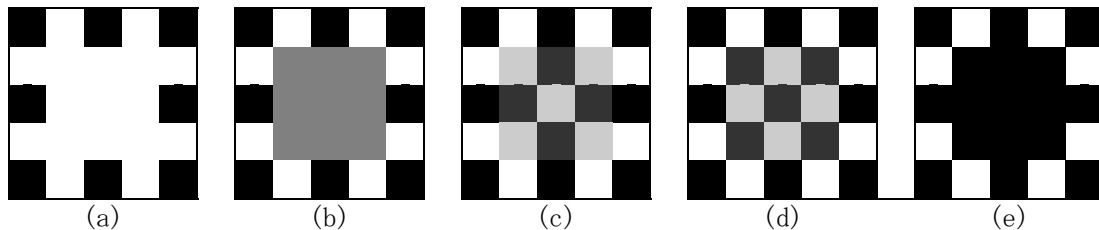


画像 A

- (1) 画像 A を 100,000 ビット毎秒 (bps) の通信路で伝送するとき、1 秒間に何枚の画像を送信できるか答えよ。ただし、プロトコルのヘッダなどによるオーバーヘッドは考慮しないものとする。
- (2) 画像 A と同じ形式で、全画素が黒色の画像を画像 B、全画素が白色の画像を画像 C とする。画像 A, B, C について、画素のデータを一行に並べたもの（画像 A の場合、0101010101…のようになる）をランレングス法で圧縮したとき、圧縮後のサイズが最も大きくなる画像はどれか答えよ。
- (3) 画像 A と同様の 25 画素で、各画素が 0～255 の値を持ち得るグレースケールの画像について考える。ここでは黒色を 0、白色を 255 で表現し、値が大きい方がより明るい（白色に近い）灰色を表わすことにする。この形式で画像 A と同じ黒色（画素の値が 0）と白色（画素の値が 255）を交互に配置した画像（画像 D）を作成すると、そのデータ量は画像 A の何倍になるか答えよ。

(4) 画像 D にある加工をした画像（画像 E）を作成するプログラムを次ページに示す（出題の都合上、型名を伏せている）。このプログラム中にある 空欄 に当てはまるコードを答えよ。この加工では、ある画素の値を、その画素および上下左右に隣り合う計 5 画素の平均値（小数点以下を切捨）に設定するものである。ただし、上下左右の端の画素については、加工を行わない。

(5) 画像 E を、以下の選択肢(a)～(e)から選びなさい。



```
#include <stdio.h>
#define N 5 /* 画像の一辺の画素数 */
#define B 0 /* 黒色を示す値 */
#define W 255 /* 白色を示す値 */

int main(void) {
    unsigned 型名 D[N][N] = {{B, W, B, W, B}, {W, B, W, B, W}, {B, W, B, W, B},
                             {W, B, W, B, W}, {B, W, B, W, B}}, E[N][N];
    for (int y = 0; y < N; y++) {
        for (int x = 0; x < N; x++) {
            if ( 空欄 ) { /* 上下左右の端の画素を示す条件 */
                E[y][x] = D[y][x]; /* 上下左右の端の画素を画像 D と同じ値に設定 */
            } else { /* 平均値を算出 */
                E[y][x] = (D[y][x] + D[y-1][x] + D[y+1][x] + D[y][x-1] + D[y][x+1]) / 5;
            }
        }
    }
    return 0;
}
```

解答例：

- (1) 4000 枚
- (2) 画像 A
- (3) 8 倍
- (4) `x == 0 || x == N-1 || y == 0 || y == N-1`

他に、`x < 1 || x > N-2 || y < 1 || y > N-2` や

`!(0 < x && x < N-1 && 0 < y && y < N-1)` などの別解もある

(5) (c)

Q6

あなたの研究について、以下の(1)～(2)の問いに対し、解答用紙 1 ページ以内に記述せよ。

- (1) 卒業研究の題目を冒頭に記述し、研究背景、解決しようとした問題、提案した解決手段や方法、手法の有効性の証明方法、得られた結果を述べよ。ただし、解答の際は、少なくとも 1 つ以上の図または表を用いること。
- (2) 大学院に入学後の研究計画・展望を、少なくとも 1 つ以上の図または表を作成の上、卒業研究の研究成果との関係が分かるように述べよ。

解答例：

- (1) 卒業研究についての研究背景、解決しようとした問題、提案した解決手段や方法、手法の有効性の証明方法、得られた結果、図または表が記述されていること。また、記述内容が論理的に矛盾なく書かれていること。
- (2) 大学院での研究計画と展望、卒業研究の成果との関係、図または表が記述されていること。また、記述内容が論理的に矛盾なく書かれていること。