東工大情報理工学院情報工学系 R1 年度試験解答

文責：天才 (@johrinotensai)

2020 年 2 月 19 日

（講評）

大問 1

数学の分野である．ここで問われている数学は大学 1 年生の春学期で扱うものか, 大学 2 年生の春学期で扱うものである．用語を知っていればすぐに計算できて簡単であり, 知らなければもちろん解けないというレベルである．ゆえに広く浅く勉強しておけば全く問題ないと思われる．微分積分に関しては高校で扱う数 III の主張が強くなり偏微分とかそういう知識が必要ないので簡単である．

1. 以下の問に答えよ.

1) 以下の極限を求めよ.

a) $\displaystyle\lim_{x\to\infty}\{\log_e(2x+3)-\log_e(x)\}$

b) $\displaystyle\lim_{x\to 0}\frac{1-\cos x}{x\sin x}$

c) $\displaystyle\lim_{x\to 0}\frac{e^{3x}-\cos x}{x}$

解説)

面倒なので, 対数の底の表記を省略する.

極限として, 線形関数より指数関数のほうが発散速度が速いことや対数関数より線形関数のほうが発散速度が速いなどがある. 有名な極限としては

$$\lim_{x\to 0}\frac{\sin x}{x}=1 \tag{1-1}$$

がある. また, 関数 $f(x)$, $g(x)$ において求める極限の形が $f(x)/g(x)$ の形であり, $f(x)\to 0$ かつ $g(x)\to 0$ または $f(x)\to\infty$ かつ $g(x)\to\infty$ のときロピタルの定理が使えて

$$\lim_{x\to a}\frac{f(x)}{g(x)}=\lim_{x\to a}\frac{f'(x)}{g'(x)} \tag{1-2}$$

となる. これらの考え方を用いて極限を求めていくこととなる.

a) 対数の差は対数の中身の商の形に変換できるので

$$\begin{aligned}\lim_{x\to\infty}\{\log(2x+3)-\log x\}&=\lim_{x\to\infty}\log\frac{2x+3}{2}\\&=\lim_{x\to\infty}\log\left(2+\frac{3}{x}\right)\\&=\log 2\end{aligned}$$

b) $\displaystyle\lim_{x\to 0}1-\cos x=0$ かつ $\displaystyle\lim_{x\to 0}x\sin x=0$ よりロピタルの定理から

$$\begin{aligned}\lim_{x\to 0}\frac{1-\cos x}{x\sin x}&=\lim_{x\to 0}\frac{\sin x}{\sin x+x}\\&=\lim_{x\to 0}\frac{\cos x}{\cos x+1}\\&=\frac{1}{2}\end{aligned}$$

c) $\displaystyle\lim_{x\to 0}e^{3x}-\cos x=0$ かつ $\displaystyle\lim_{x\to 0}x=0$ よりロピタリの定理から

$$\begin{aligned}\lim_{x\to 0}\frac{e^{3x}-\cos x}{x}&=\lim_{x\to 0}3e^{3x}-\sin x\\&=3\end{aligned}$$

2) 以下の実行列の積の行列式を計算せよ.

$$\begin{pmatrix}1&0&0\\x&2&1\\x^2&3&2\end{pmatrix}\begin{pmatrix}31&23&17\\0&11&11\\0&4&5\end{pmatrix}$$

解説)

行列 A, B に対し行列式の積には以下のような関係がある.

$$|AB|=|A||B| \tag{1-3}$$

また行列式の計算においては余因子展開という考え方が存在する．行列 A に対する (i,j) 成分の余因子を A_{ij} とする．この余因子とは行列 A の i 行と j 列を抜き去ったものの行列式 D_{ij} に対して以下である．

$$A_{ij} = \begin{cases} D_{ij} & (i+j = 2n, n \in \mathbb{N}) \\ -D_{ij} & (i+j = 2n+1, n \in \mathbb{N}) \end{cases}$$

この余因子から行列式は

$$|A| = \sum_{i=1}^{n} a_{ij} A_{ij} = \sum_{j=1}^{n} a_{ij} A_{ij} \tag{1-4}$$

問題の行列式の値は

$$\begin{aligned} \det \left| \begin{pmatrix} 1 & 0 & 0 \\ x & 2 & 1 \\ x^2 & 3 & 2 \end{pmatrix} \begin{pmatrix} 31 & 23 & 17 \\ 0 & 11 & 11 \\ 0 & 4 & 5 \end{pmatrix} \right| &= \begin{vmatrix} 1 & 0 & 0 \\ x & 2 & 1 \\ x^2 & 3 & 2 \end{vmatrix} \begin{vmatrix} 31 & 23 & 17 \\ 0 & 11 & 11 \\ 0 & 4 & 5 \end{vmatrix} \\ &= \begin{vmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 3 & 2 \end{vmatrix} \cdot 31 \begin{vmatrix} 11 & 11 \\ 4 & 5 \end{vmatrix} \\ &= 1 \cdot 31 \cdot 11 \\ &= 341 \end{aligned}$$

3) 以下の確率密度関数 $f_X(x)$ に従う連続型確率変数 X の分散 $V(X)$, 累積分布関数 $F_X(x)$ を各々求めよ．ただし, $\lim_{x \to +0} x^n \log_e(x) = 0$ (n は 1 以上の整数) とする．

$$f_X(x) = \begin{cases} -4x \log_e(x) & 0 < x \leq 1 \\ 0 & x \leq 0 \text{ または } x > 1 \end{cases}$$

解説)

面倒なので, 対数の底の表記を省略する．

連続型確率変数 X の平均 $E(X) = \mu$ と分散 $V(X)$ の求め方は次のとおりである．

$$E(X) = \mu = \int_{-\infty}^{\infty} x f(x)\ dx \tag{1-5}$$

$$V(X) = \int_{-\infty}^{\infty} (x-\mu)^2 f(x)\ dx \tag{1-6}$$

まず平均については

$$\begin{aligned} E(X) &= \int_{-\infty}^{\infty} x f_X(x)\ dx \\ &= \int_0^1 -4x^2 \log x\ dx \\ &= -4 \left\{ \left[\frac{x^3}{3} \log x \right]_0^1 - \int_0^1 \frac{x^2}{3}\ dx \right\} \\ &= 4 \left[\frac{x^3}{9} \right]_0^1 \\ &= \frac{4}{9} \end{aligned}$$

次に分散については

$$
\begin{aligned}
V(X) &= \int_{-\infty}^{\infty} (x-\mu)^2 f_X(x)\ dx \\
&= \int_0^1 \left(x - \frac{4}{9}\right)^2 \cdot (-4x\log x)\ dx \\
&= -4\int_0^1 \left(x^2 - \frac{8}{9}x + \frac{16}{81}\right) \cdot (x\log x)\ dx \\
&= -4\int_0^1 x^3 \log x\ dx + \frac{32}{9}\int_0^1 x^2 \log x\ dx - \frac{64}{81}\int_0^1 x\log x\ dx \\
&= 4\int_0^1 \frac{x^3}{4}\ dx - \frac{32}{9}\int_0^1 \frac{x^2}{3}\ dx + \frac{64}{81}\int_0^1 \frac{x^2}{2}\ dx \\
&= \left[\frac{x^4}{4} - \frac{32}{81}x^3 - \frac{16}{81}x^2\right]_0^1 \\
&= \frac{1}{4} - \frac{32}{81} + \frac{16}{81} \\
&= \frac{17}{324}
\end{aligned}
$$

累積分布関数 $F_X(x)$ については, 以下の定義式である.

$$
F(x) = \int_{-\infty}^{x} f(x)\ dx \tag{1-7}
$$

これから求める $F_X(x)$ については, $x<0$ について 0, $x \geq 1$ については 1 である. $0 \leq x < 1$ のときは

$$
\begin{aligned}
F_X(x) &= \int_0^x (-4x\log x)\ dx \\
&= -4\left(\left[\frac{x^2}{2}\log x\right]_0^x - \int_0^x \frac{x}{2}\ dx\right) \\
&= -4(x^2\log x - \frac{x^2}{4}) \\
&= -4x^2\log x + x^2
\end{aligned}
$$

よって, 答えは

$$
F_X(x) = \begin{cases} 0 & (x<0) \\ -4x^2\log x + x^2 & (0 < x \leq 1) \\ 1 & (x > 1) \end{cases}
$$

確認として, $x=0$, $x=1$ のときの値を代入して連続になっているかをみればよい.

4) ある製造ラインで生産された製品は 1/1000 の確率で不良品である. 不良品を 99/100 の確率で正しく不良品と判定し, かつ, 不良品でないものを 4/5 の確率で正しく不良品ではないと判定する検査手法がある. この製造ラインにおいて, この手法が不良品と判定した製品が, 不良品である確率を求めよ.

この問題はベイズの定理の典型的問題である. しかし, ベイズの定理を知らなくても条件付き確率から直ちに答えを得られる. 不良品であり, 不良品と判定される確率は

$$
\frac{999}{1000} \times \frac{99}{100}
$$

である. 一方で不良品でないものを不良品と判定する確率は

$$
\frac{1}{1000} \times \frac{1}{5}
$$

これより不良品と判定した製品が, 不良品である確率は

$$
\begin{aligned}
\frac{\frac{1}{1000}\times\frac{99}{100}}{\frac{1}{1000}\times\frac{99}{100}+\frac{999}{1000}\times\frac{1}{5}} &= \frac{\frac{1}{1000}\times\frac{99}{100}}{\frac{1}{1000}\times\frac{99}{100}+\frac{999}{1000}\times\frac{20}{100}} \\
&= \frac{99}{99+999\times 20} \\
&= \frac{11}{11+2220} \\
&= \frac{11}{2231}
\end{aligned}
$$

5) あるカジノで, 4 個のポケット $\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}$ に区切られたルーレットがある. カジノの説明ではそれぞれのポケットにボールが入る確率は同じであるとされている. そのルーレットを 5 回試行したところ, ボールはポケット A に 4 回入った. カジノの説明とは異なる「このルーレットはボールがポケット A により入りやすい」という仮説を, 有意水準 5% で検定せよ. 解答には帰無仮説 H_0, 対立仮説 H_1 を明記すること.

解説)
有意水準とはその仮説の正しさがどのくらいのものかの基準となるものである. つまりある仮説のもとに出した確率が有意水準の値より下回るとその仮説は間違っているとして棄却される. ここで帰無仮説とは正しさを示すための仮説となる. 対立仮説とはその帰無仮説の対となる仮説で帰無仮説が棄却されたときに対立仮説が立証されることとなる. 今回の帰無仮説は「このルーレットはボールがポケット A により入りやすい」としたいがこれを定量的に説明することが困難である. (どの程度入りやすいか明記されている場合は帰無仮説となり得るが... これが出題者のヒントである). 予想だとポケット A に入りやすいとなるが問題の性質上これを示すのは困難なので予想に反する結果になると予測できるというのも出題者のヒントであろう. 本題に戻って帰無仮説, 対立仮説は以下のように立てる.

H_0 :「このルーレットのそれぞれのポケットにボールが入る確率は同じである」
H_1 :「このルーレットのそれぞれのポケットにボールが入る確率は同じではない」

帰無仮説を元にボールがポケット A に 4 回入る確率を求める. すると以下のように求められる.

$$
\begin{aligned}
{}_5\mathrm{C}_4 \cdot \left(\frac{1}{4}\right)^4 \cdot \left(\frac{3}{4}\right) &= 5 \cdot \frac{1}{4^4} \cdot \frac{3}{4} \\
&= \frac{15}{256} \\
&> \frac{15}{300} = 0.05
\end{aligned}
$$

となり, 有意水準 5% を満たすので帰無仮説は棄却されない. ゆえに「このルーレットはボールがポケット A により入りやすい」という仮説は棄却される.

2. 以下の問いに答えよ．

1) 命題論理について考える．命題 φ を $(\neg p_0 \to \neg p_1) \wedge (\neg p_1 \to p_0)$ とする．ただし, p_0, p_1 は命題記号である．

a) 以下の表が φ の真理値表になるように（ア）〜（エ）にあてはまる値を答えよ．ただし, 1 は真を表し, 0 は偽を表す．

p_0	p_1	φ
0	0	（ア）
0	1	（イ）
1	0	（ウ）
1	1	（エ）

b) 命題 φ が恒真（トートロジー）であるか否か答えよ．

c) 命題 φ が充足可能であるか否か答えよ．

解説)

問題だけをみると命題 φ は恒真ではなく, 充足可能であるという推測がされる．真理値表を埋めてしまえばすぐに恒真かどうか充足可能かどうかも導き出されるのだが... 論理式変形で次のようなものがある

$$A \to B \Longleftrightarrow \neg A \vee B \tag{2-1}$$

この問題では $\to$ を用いているが多くの参考書や資料では $\Longrightarrow$ で用いられる．一般的には「ならば」という意味で用いられる．これから命題 φ の変形を行うと

$$(\neg p_0 \to \neg p_1) \wedge (\neg p_1 \to p_0) \Longleftrightarrow (p_0 \vee \neg p_1) \wedge (p_1 \vee p_0)$$

この論理式から $p_0 = 1$ なら何でも良いということになる．また $p_0 = 0$ のときどれも満たされない．

a) 論理式は $(p_0 \vee \neg p_1) \wedge (p_1 \vee p_0)$ であることから以下のようになる．

p_0	p_1	φ
0	0	0
0	1	0
1	0	1
1	1	1

b) どの入力に対しても $\varphi = 1$ となるときが恒真であるので, 命題 φ は恒真 (トートロジー) ではない．

c) 充足可能とは $\varphi = 1$ となる入力が存在することであり, a) の答えの通り存在するので充足可能である．

2) 命題論理の自然演繹について考える．p_1, p_2, p_3 は命題記号であり, $\wedge I$ は連言の導入規則, $\wedge E_L$ は連言の除去規則 (左), $\wedge E_R$ は連言の除去規則 ((右), $\to E$ は含意の除去規則である．

a) $(p_1 \wedge p_2) \wedge p_3$ を仮定とし, $p_1 \wedge (p_2 \wedge p_3)$ を結論とする以下の自然演繹の導出 (証明図) を完成させたい．（ア）〜（オ）にあてはまる命題を答えよ．

$$\dfrac{\dfrac{\dfrac{(p_1 \wedge p_2) \wedge p_3}{（ア）}\wedge E_L}{p_1}\wedge E_L \qquad \dfrac{\dfrac{\dfrac{(p_1 \wedge p_2) \wedge p_3}{（イ）}\wedge E_L}{（ウ）}\wedge E_R \qquad \dfrac{(p_1 \wedge p_2) \wedge p_3}{（エ）}\wedge E_R}{（オ）}\wedge I}{p_1 \wedge (p_2 \wedge p_3)}\wedge I$$

b) $p_1 \wedge p_2$ と $p_1 \to (p_2 \to p_3)$ を仮定とし, p_3 を結論とする以下の自然演繹の導出 (証明図) を完成させたい. (ア)〜(エ) にあてはまる命題を答えよ．

$$\dfrac{\dfrac{（ア）}{p_2}\wedge E_R \qquad \dfrac{\dfrac{（イ）}{p_1}\wedge E_L \qquad （ウ）}{（エ）}\to E}{p_3}\to E$$

c) 命題の集合 $\{\neg p_0 \to \neg p_1, \neg p_1 \to p_0\}$ が矛盾するか無矛盾であるか答え, その理由を述べよ．

解説)
この問題は証明木を書かせる問題である．この連言や規則を知らなくても推定で問題を解くことは可能である．今回の問題で連言規則をまとめる．
$\wedge I$ の連言の導入規則は

$$A,\ B \models A \wedge B \tag{2-2}$$

これは具体例で書けば,「今日は日曜日である」というのと「今日は私が休みだ」という二つのから「今日は日曜日であると同時に私は仕事が休みだ」ということになる推論を考える．これから命題変数 $P,\ Q$ について P を今日は日曜日である, Q を私は仕事が休みであるとおくと, 推論は $P \wedge Q$ となる．
論理式について次が成り立つ．

$$A \wedge B \Longrightarrow A \tag{2-3}$$
$$A \wedge B \Longrightarrow B \tag{2-4}$$

この式 (2-3) を連言の除去規則 (左), 式 (2-4) を連言の除去規則 (右) である．最後に含意の除去規則について

$$A \wedge (A \Longrightarrow B) \models B \tag{2-5}$$

これは A であり A ならば B ということが成り立つならば B であることを言っている．

a) 右にどの除去規則を使っているかが書いてあるのでそれに従って記入していくだけである．

$$\dfrac{\dfrac{\dfrac{(p_1 \wedge p_2) \wedge p_3}{p_1 \wedge p_2}\wedge E_L}{p_1}\wedge E_L \qquad \dfrac{\dfrac{\dfrac{(p_1 \wedge p_2) \wedge p_3}{p_1 \wedge p_2}\wedge E_L}{p_2}\wedge E_R \qquad \dfrac{(p_1 \wedge p_2) \wedge p_3}{p_3}\wedge E_R}{p_2 \wedge p_3}\wedge I}{p_1 \wedge (p_2 \wedge p_3)}\wedge I$$

b) これに関しても誘導に従うだけである．a) に比べると推測しながらの要素は強くなるものの至って簡単である．

$$\dfrac{\dfrac{p_1 \wedge p_2}{p_2}\wedge E_R \qquad \dfrac{\dfrac{p_1 \wedge p_2}{p_1}\wedge E_L \qquad p_1 \to (p_2 \to p_3)}{p_2 \to p_3}\to E}{p_3}\to E$$

c) この命題は無矛盾である．これは 1) の命題 φ へ連言の導入規則で求めたものである．ゆえに

$$\dfrac{\neg p_0 \to \neg p_1 \qquad \neg p_1 \to p_0}{(\neg p_0 \to \neg p_1) \wedge (\neg p_1 \to p_0)}\wedge I$$

また 1) の a) より充足可能であるので, 無矛盾である．

3) 一階述語論について考える．

a) 以下の論理式のモデルは存在するか．存在するならば, モデルのユニバースの濃度の最小値を答えよ．存在しないならば, その理由を述べよ．

$$\exists x \exists y \exists z\ \neg(x = y) \wedge \neg(y = z) \wedge \neg(z = x)$$

b) 以下の論理式のモデルは存在するか．存在するならば, そのモデルを 1 つ挙げよ．存在しなければ, その理由を述べよ．

$$\exists x \forall y (x = y)$$

解説)

a) 存在する．
例えば, $x = 0, y = 1, z = 2$ と取れば良い．このモデルのユニバースについて異なる 3 数が存在すればいいので, 最小値は 3.

b) 存在しない.

ある x に対して全ての y が $x = y$ を満たさないといけないので, y が $y = x + 1$ となるときは論理式は成立しないため存在しない.

4) 一階述語論理の自然演繹について考える. 以下が成り立つか否か答え, その理由を述べよ.
ただし, P はアリティ 2 の述語記号とする.

解説)
例えば, P を x と y は等しいという述語にすればこの一階述語論理の自然演繹は成り立たない.

3. 1) n 個の要素からなる整列の配列 a を昇順に整列させる処理を, 二つの異なるアルゴリズムに基づき C 言語で実装した. プログラム 3.1, 3.2 のキャプション $\boxed{\text{A}}$, $\boxed{\text{B}}$ を埋めるのに最も適切なアルゴリズムの名称を選択肢から選び, その記号を答えよ.

【選択肢】

ア. 選択ソート　イ. バブルソート　ウ. クイックソート　エ. 挿入ソート

プログラム 3.1: $\boxed{\text{A}}$

```
void sort_a(int a[], int n)
{
  int i,j,tmp;

  for(i = 0; i< n-1; ++i) {
    for(j = n-1; i < j; --j) {
      if(a[j] < a[j-1]) {
         tmp = a[j];
         a[j] = a[j-1];
         a[j-1] = tmp;
      }
    }
  }
}
```

プログラム 3.2: $\boxed{\text{B}}$

```
void sort_b(int a[], int n)
{
    int i, j, tmp;

    for(j = 1; j < n; ++j) {
        tmp = a[j];
        i = j - 1;
        while (0 <= i && tmp < a[i]) {
            a[i+1] = a[i];
            --i;
        }
        a[i+1] = tmp;
    }
}
```

解説)

プログラム 3.1 に関しては順々にひっくり返しているのでバブルソートである (イ).

プログラム 3.2 に関しては while 文の中身がある引数以降を 1 ずつずらしてその引数に新しい値を入れている (挿入) ので, 挿入ソートである (エ).

2) ヒープソートに関して, 次の問いの答えよ. ただし, n は正の整数とする.

a) 二分ヒープにおいて, 根以外の任意の節点の値が満たすべき条件を説明せよ (max ヒープ条件と min ヒープ条件のどちらでもよい).

b) 二分ヒープが n 個の要素を格納している時, その木の高さを n の式で表わせ. なお, 木の高さは, 根と葉を結ぶ路の長さの最大値として定義される.

c) ヒープソートで n 個の要素を整列させる時, その平均時間計算量と最悪時間計算量の漸近的評価として, 最も適切なものを以下の選択肢からそれぞれ答えよ.

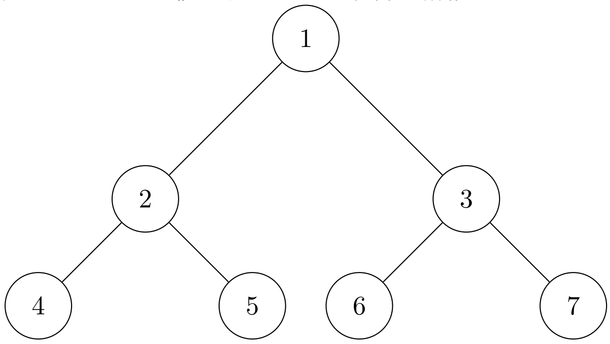
【選択肢】

ア. $\mathcal{O}(1)$　イ. $\mathcal{O}(\log n)$　ウ. $\mathcal{O}(n)$

エ. $\mathcal{O}(n \log n)$　オ. $\mathcal{O}(n^2)$

解説)

まずプログラムが組まれて入力されるときに値は次のような順番で格納されていく.



a) この後ソートする方法として, 根のほうが節点よりも小さいまたは大きいである.

つまり max ヒープ条件においては, 節点はその子の節点よりも大きい. 一方で min ヒープ条件においては節点はその子の節点よりも小さいという条件がある.

b) $n=0$ のときは何もないので高さは 0. $n=1$ のとき路がないので高さは 0. $n=2$ のときは路が 1 つできるので高さは 1. 2^n-1 のとき木の高さは $n-1$ であることがわかる. 2^n のときは高さは n となり, $2^{n+1}-1$ まで高さは n であることが分かる. これをまとめると以下のように高さを表せる.

$$\begin{cases} k-1 & (n=2^k-t \quad (0<t<2^k)) \\ k & (n=2^k) \end{cases}$$

c) ヒープソート (heap sort) とはリストの並べ替えを二分ヒープ木を用いて行うソートアルゴリズムである. 次のようなアルゴリズムである.

1. 未整列のリストから要素を取り出し, 順にヒープに追加する. 全ての要素を追加するまで繰り返し.
2. 最大値または最小値を取り出し, 整列済みリストに追加する. すべての要素を取り出すまで繰り返し.

この最悪計算時間も平均計算時間も $\mathcal{O}(n\log n)$ である.

3) $A=\langle A[0],A[1],...,A[n-1]\rangle$ を n 個の相違なる整数の配列とする. n 未満の非負整数 i,j に対し, $i<j$ かつ $A[i]>A[j]$ のとき, 対 $i,j)$ を A の反転と呼び, A の反転の数を A の反転数と呼ぶ. 例えば, 配列 $\langle 5,7,4,6\rangle$ の反転は $(0,2),(1,2),(1,3)$ であり, 反転数は 3 である. 次の問いに答えよ.

a) 配列 $\langle 1,0,4,3,2\rangle$ の反転数を求めよ.

b) 集合 $\{1,2,...,n\}$ の要素をすべて並べた配列 (要素数は n 個) の中で, 反転数が最大となるものを示せ. また, その反転数を n の式で表わせ.

c) 集合 $\{1,2,...,n\}$ の要素をすべて並べた任意の配列 B(要素数は n 個) をバブルソートで昇順に整列させる. このとき, 「バブルソートにおける $\boxed{X}$ と配列 B の反転数は等しい」という関係が成り立つ. $\boxed{X}$ を埋めるのに適切な語句と, その関係が成り立つ理由を簡潔に答えよ.

a) この配列で元の位置に存在するものは 3 だけである. つまり反転する際は 3 を選択しないように反転すればよい. まず (0,1) を選択すると $\langle 0,1,4,3,2\rangle$ となる. 最後のに (2,4) を選択すると $\langle 0,1,2,3,4\rangle$ となるので反転数は 2 である.

b) 1 回反転すれば少なくとも 1 個は整列された番号に移動できる. つまり最大 n 回反転すれば整列されることが分かる. ここで a) の例であると 1 という数を整列すると 0 という数も自動的に整列されていた. これを 1 個しか整列されないように組むには以下のようにすればよい.

$$\{n,0,1,2,3,4,...,n-3,n-2,n-1\}$$

このように置くことで, どのように反転させても必ず 1 個しか整列されない. つまり第 0 番目には n を第 $i(i\neq 0)$ 番目には $i-1$ を格納すればよい.

c) バブルソートにおける交換回数と配列 B の反転数が等しくなる.

これの理由として, バブルソートは第 n 番目から順に 0 番目までを決めていく. このとき第 i 番目をバブルソートで整列する際には $i+1$ 番目から n 番目までは整列されている. 第 i 番目が正しい位置に存在するならば交換する必要がなくなるのでカウントされないが, 異なっている場合は順々に交換することが i 番目を決めるのでカウント数と反転数が等しくなる.

4) プログラム 3.3 は, 整数の配列を昇順に整列させる処理を, マージソートアルゴリズムに基づき, C 言語で実装したものである. 次の問いに答えよ.

a) 空欄【 A 】,【 B 】,【 C 】,【 D 】,【 E 】を埋めるのに適切なコードを, 次の選択肢から選び, プログラムを完成させよ. ただし, 整列させたい配列を `a`, 整列時の作業領域に用いる配列を `w` とし, `mergesort` 数を以下の 2 行のコードで呼び出すこととする.

```
int a[5] = {3, 5, 1, 4, 2}, w[5];
mergesort(a, 0, 4, w);
```

【選択肢】

ア.`a[k] = w[i]`	イ.`a[k] = w[i++]`	ウ.`a[k] = w[++i]`
エ.`a[k] = w[j]`	オ.`a[k] = w[j++]`	カ.`a[k] = w[++j]`
キ.`a[k] = w[k]`	ク.`a[k] = w[k++]`	ケ.`a[k] = w[++k]`
コ.`w[k] = a[i]`	サ.`w[k] = a[i++]`	シ.`w[k] = a[++i]`
ス.`w[k] = a[j]`	セ.`w[k] = a[j++]`	ソ.`w[k] = a[++j]`
タ.`w[k] = a[k]`	チ.`w[k] = a[k++]`	ツ.`w[k] = a[++k]`

5) a) に示したコードで mergesort 関数を呼び出したとき, 標準出力の 1 行目に (0,4) が書き出される. 3 行目, 5 行目, 7 行目に書き出される内容をそれぞれ答えよ.

6) `mergesort` 関数が引数で指定された範囲の配列の要素を整列させるまでにプログラム 3.3 の 18 行目を実行した回数の総数 (要素の比較回数) を求めたい. そこで, 17 行目を (コメントアウトを外して)`++c;` に変更したが, これだけでは不十分である. 以下の 2 行のコードで比較回数を変数 `count` に格納するには, プログラム 3.3 をどのように変更すればよいか答えよ.

```
int a[5] = {3, 5, 1, 4, 2}, w[5];
int count = mergesort(a, 0, 4, w);
```

ただし, 帯域変数や静的変数を使ってはいけない. また, 17 行目の変更に加えて行うプログラムの変更は 3 回までの行の差し替えに限定する. 行の差し替えとは, ある行のコードを 60 文字以内 (空白文字は数えない) の別のコードに置き換えることを指す. 解答の際は, 以下の【答案の書き方 (例)】のように, 変更する行の番号と差し替え後のコードを記すこと. 17 行目のコメントは外してあることとし, それ以外に必要な変更を記述すること.

【答案の書き方 (例)】

· 6 行目 : `printf(‘‘Hello\n’’);`

· 30 行目 : `c = end - begin + 1;`