

# 情報 2011 年度共通試験問題 解答例・解説

2012 年度入学理科一類 10 組情報シケ対 編

2012 年 7 月 24 日

この解答例と解説の正確さは保証しかねます。また、共通問題 3 の A-2 の答えは載せていません。

## 共通問題 1

1-1

- (1) 誤り 必要がある。( 必要はない。)

説明 送信者がインターネット上で送ったメールは、一つ一つが小さなサイズのパケットに分割され、いくつかのルータを通じて複数のネットワークを越えて送信者へと届く。(76 字)

- (2) 誤り 送信者と受信者がお互いに鍵を公開し、公開された鍵を用いて暗号化と復号を行う。

誤り(別解) 公開鍵暗号方式( 共通鍵暗号方式)

説明 公開鍵暗号方式は、受信者が予め公開鍵と秘密鍵の組を作っておき、公開鍵のみを一般に公開した上で、送信者はその公開鍵を用いて平文を暗号化し、それを受信者が秘密鍵で復号する、という暗号化方式である。(96 字)

- (3) 誤り 従って  $\sim a \times 10$  秒より長くかかることはない結論づけられる。

説明 自宅のマシンとサーバとは、ルータによりつながった複数のネットワークを介して繋がっており、通信がなされる際はこの複数のネットワークを越えて複数個のパケットがやり取りされる。この通信の経路はサーバの所在により違うので、通信に要する時間もサーバの所在により異なる。(130 字)

- (4) 誤り 完全に仮想的な名前であり、接続者の国、地域、所属組織などとの対応関係は一切ない。

説明 ドメイン名を地域や組織に対応させることで、ホスト名中のドメイン名から地域や組織を読み取り、その地域や組織に IP アドレスを問い合わせる、といった IP アドレスとホスト名との対応の分散的な管理ができる。実際にはドメイン名に階層的な構造を持たせている。(120 字)

解説 これらはいずれも教科書・解説スライド 3 章「情報の伝達と通信」からの出題である。(1) 及び (2) については 2006 年度の問題 1 とほぼ同じであるので、この部分の解説に関してはシケプリ”only my information”等を参照されたい。前者は教科書(第 9 刷)の p.56 前後、後者は p.49-51 の内容である。(3) はどう説明すればいいかわかりづらい問題だが、ダウンロードの速度が時と場合により異なるのは経験上分かることであろう。速度が異なる原因として、他にはサーバの混雑による処理の遅延等が考えられる。(4) のドメインの分散管理については 2007 年度の問題 2(3) にも問題が出ている。解答例には登場させなかったが、DNS というワードは重要なので意味を知っておこう。(教科書 p.64)

1-2

- (ア) (b) [HTTP]
- (イ) (c) [プログラム]
- (ウ) (d) [文書を収集]
- (エ) (b) [プログラム]
- (オ) (c) [データベース]

解説 この問題も第3章「情報の伝達と通信」からの出題である。(ア)のHTTPはプロトコルの一種である。プロトコルとは通信の際の決め事のことである。(教科書 p.46) (イ)については(d)が微妙なところだが、電子メールソフトとすると限定しすぎであろうし、メールを送信する時のプロトコルはSMTPである。また、(ウ)についても(a)が微妙なところだが、「ウェブブラウザ」の目的を考えると「文書を収集」が最も適切であろう。(エ)と(オ)については、正答以外の選択肢はいずれもかなりの外れであるので、消去法でも正答にたどり着くだろう。チケット予約システムについての問題は2009年度の問題1-2にもある。

## 共通問題 2

- (1) (一例：繰り返し処理を用いたもの)

```
i    1
j    2
while j ≤ 4 do
  if seatsi < seatsj
    then i    j
  endif
  j    j + 1
done
```

- (2)

c	votes <sub>1</sub>	votes <sub>2</sub>	votes <sub>3</sub>	votes <sub>4</sub>
7	51000	81000	30000	20000
4	34000	27000	30000	20000
0	20400	20250	15000	20000

- (3)

seats <sub>1</sub>	seats <sub>2</sub>	seats <sub>3</sub>	seats <sub>4</sub>
4	3	1	0

解説 この問題は主に教科書・解説スライド 5 章「計算の方法」からの出題である。問題文中のプログラムの  
ような部分は「擬似コード」と呼ばれるもので、教科書 p.99-103 や解説スライドの「八十八夜」や「平  
方根の計算」等と同じようなものが載っているだろう。このコード中の

```
X    Y
```

という部分は、「X という変数に Y という値を代入する」ということを、

```
while (条件) do
```

```
  (処理)
```

```
done
```

という部分は、「(条件) が成立している間 (処理) を繰り返し続ける」ということを表している。

(1) については、解答例は次のような手順で「votes 中の最大値の添え字」を求めようとしている。

(添え字とは、seats や votes の右下についた小さな数字や文字のことである。)

- 暫定的に  $i$  を 1 としておく。
- $votes_i$  と  $votes_2$  を比べ、 $votes_2$  のほうが大きければ、 $i$  に 2 を代入する。
- $votes_i$  と  $votes_3$  を比べ、 $votes_3$  のほうが大きければ、 $i$  に 3 を代入する。
- $votes_i$  と  $votes_4$  を比べ、 $votes_4$  のほうが大きければ、 $i$  に 4 を代入する。

このように順番に全て比較すれば最大値の添え字が求まるのは明らかだろう。これを素直に書くと、

```
i    1
```

```
if seatsi < seats2
```

```
  then i    2
```

```
endif
```

```
if seatsi < seats3
```

```
  then i    3
```

```
endif
```

```
if seatsi < seats4
```

```
  then i    4
```

```
endif
```

となるだろうが、これだと途中があまりにも冗長なので、解答例では繰り返し処理を使用した。但し、  
もちろん上のように答えても正解であるとは思う。

(2),(3) は、問題文中に書かれた手続きを順に追っていくことで解くことができる。まず  $c$  について考  
えずに、それ以外の while ~ done の中身を考えて、

- votes を調べて得票数が最大の政党を見つける。
- その政党の議席数 (seats) を 1 増やす。
- その政党の得票数 (votes) に  $\frac{(\text{議席数})}{(\text{議席数}) + 1}$  を掛ける。

という手順を順番にしようとしていることが分かる。また、 $c$  は while 内の処理を 1 回するごとに 1  
減り、 $c \leq 0$  となると繰り返し処理が終わるので、結局繰り返しは 8 回であることが分かる。

以上を踏まえて seats や votes の値を順に追っていくと、 の処理を終えた回数ごとのそれぞれの値  
の表は次のようになる。(ただし、繰り返し回数 0 回とは、初期状態であることを意味する。)

回数	c	votes <sub>1</sub>	votes <sub>2</sub>	votes <sub>3</sub>	votes <sub>4</sub>	seats <sub>1</sub>	seats <sub>2</sub>	seats <sub>3</sub>	seats <sub>4</sub>
0	8	102000	81000	30000	20000	0	0	0	0
1	7	<b>51000</b>	81000	30000	20000	<b>1</b>	0	0	0
2	6	51000	<b>40500</b>	30000	20000	1	<b>1</b>	0	0
3	5	<b>34000</b>	40500	30000	20000	<b>2</b>	1	0	0
4	4	34000	<b>27000</b>	30000	20000	2	<b>2</b>	0	0
5	3	<b>25500</b>	27000	30000	20000	<b>3</b>	2	0	0
6	2	25500	27000	<b>15000</b>	20000	3	2	<b>1</b>	0
7	1	25500	<b>20250</b>	15000	20000	3	<b>3</b>	1	0
8	0	<b>20400</b>	20250	15000	20000	<b>4</b>	3	1	0

この表から (2) 及び (3) は答えが出てくる。

## 共通問題 3

### 問題 A

A-1

- (ア) ユーザネーム
- (イ) コマンド
- (ウ) GUI (グラフィカルユーザインタフェース)
- (エ) アイコン
- (オ) ポインティング・デバイス
- (カ) タッチスクリーン
- (キ) 直接操作 (direct manipulation)
- (ク) (c) [できる限り多くの操作に対して、マウスとキーボードの両方のインタフェースを用意すること]

解説 この問題は第9章「ユーザインタフェース」からの出題である。この問題は主に前半 1/4 が CUI について、残りが GUI についてである。GUI や CUI についての問題は頻出であるのでよく覚えておきたい。GUI については授業スライドや教科書 p.218-221 が詳しいので、そちらを参照されたい。

解答について、(エ) は文章中 2ヶ所に出てきていて、上の空欄のみであるなら「ボタン」という答えも正解になるが、下の空欄で「～の頭文字をとって WIMP システムと呼ぶこともある」とあり、WIMP とは Window, Icon, Menu, Pointing device(又は Pointer) の頭文字を取ったものであるので、(エ) には「アイコン」が入る。また、(カ) は「タッチパネル」でもおそらく正解であるが、教科書やスライドの表記である「タッチスクリーン」を優先した。

A-2

(ここは自分で頑張ってください...)

## 問題 B

$$(1a) \quad -\sum_{i=1}^8 \frac{1}{8} \log_2 \frac{1}{8} = 3$$

$$(1b) \quad \frac{1}{8}(2+3+4+3+2+3+4+5) = \frac{13}{4} = 3.25$$

(1c)

面	1	2	3	4	5	6	7	8
符号	BAB	BBAB	BBBAB	BAABAB	AABAB	AABBAB	AABBBAB	AABBBBAB

$$\text{平均符号長} : \frac{1}{8}(3+4+5+6+5+6+7+8) = \frac{11}{2} = 5.5$$

(1d) (一例)

面	1	2	3	4	5	6	7	8
符号	AAA	AAB	ABA	ABB	BAA	BAB	BBA	BBB

この符号化の平均符号長は 3 であり，平均情報量と一致する．

$$(2a) \quad -\sum_{i=1}^7 \frac{1}{2^i} \log_2 \frac{1}{2^i} - \frac{1}{2^7} \log_2 \frac{1}{2^7} = \frac{127}{64} = 1.984375$$

(2b) (1a) で求めた平均情報量は，(2a) で求めた平均情報量よりも大きい．これは，(2) でのサイコロは各々の目が出る確率が偏っているのに対し，(1) のサイコロは各々の目が等しい確率で出るので，(1) のほうがどの目が出るのかがより予想しづらいためであると考えられる．一般にサイコロの目が何個あっても，どの面が上になるかについての平均情報量は，各面が出る確率がすべて等しいときに最大となる．

(2c) (一例)

面	1	2	3	4	5	6	7	8
符号	A	BA	BBA	BBBA	BBBBBA	BBBBBBA	BBBBBBBA	BBBBBBBB

解説 この問題は第 3 章「情報の伝達と通信」のうち，「情報量」に関する出題である．情報量は頻出であるが，定義の意味が比較的分かりにくいのでしっかりと頭に入れておこう．教科書でいうと p.37-45 のあたりであるし，授業スライドでも多くの分量が割かれている．

(1a) と (2a) は平均情報量の定義  $-\sum_{i=1}^n p_i \log_2 p_i$  に従って計算するだけの問題である．また，(1b) は問題文通りに計算するだけであるし，(1c) も問題文通りに文字を置き換えた後同じように計算するだけである．

(2b) については，教科書 p.43-44 あたりを参照されたい．もちろん書き方は様々あるだろう．

(1d) と (2c) では「平均符号長をなるべく短くする」という指示があるが，いったいどの程度短くすればいいのだろうか．これの 1 つの目安となるのが，次の定理である．

各情報を 2 種類の文字のみで符号化した時，その平均符号長は必ず平均情報量以上であり．その平均符号長の下限は，各情報をその情報量 (ビット単位) と同じ文字数で符号化することで実現できる．

この定理は情報源符号化定理と呼ばれる定理の核をなすものである．ただし，この符号化は (1b) の問題文中にあるような「複数の符号を並べるとその復元の仕方が何通りもある」ことがないような符号化のことである．これを「一意復元可能な符号化」という．基本的に「符号化」といえば「一意復元可能である」と思って良い．

一意復元可能であるためには、「ある符号が他の符号の接頭辞になっている」、即ち、「ある符号全体が他の符号の頭数文字と一致する」ことがなければよい。また、この問題の場合、全ての情報の情報量(ビット単位)は整数となるため、定理よりその情報量と同じ文字数を用いて符号化することで、平均符号長を平均情報量を同じにすることができる。この2点に留意して符号をつくると、解答例のようになる。実際、(1d)と(2c)どちらの解答例の符号化もその平均符号長と平均情報量が一致しているので、これより平均符号長が短い符号を作ることはできない。

より一般に情報量が整数にならない場合では、確率が高い情報には短い符号を、低い情報には長い符号をつけるようにすれば平均符号長は短くなる。