

東京大学 2010年度「情報」共通問題 解答例

(作成：2011 年度入学 文三 19 組 横山 g130456@mail.ecc....)

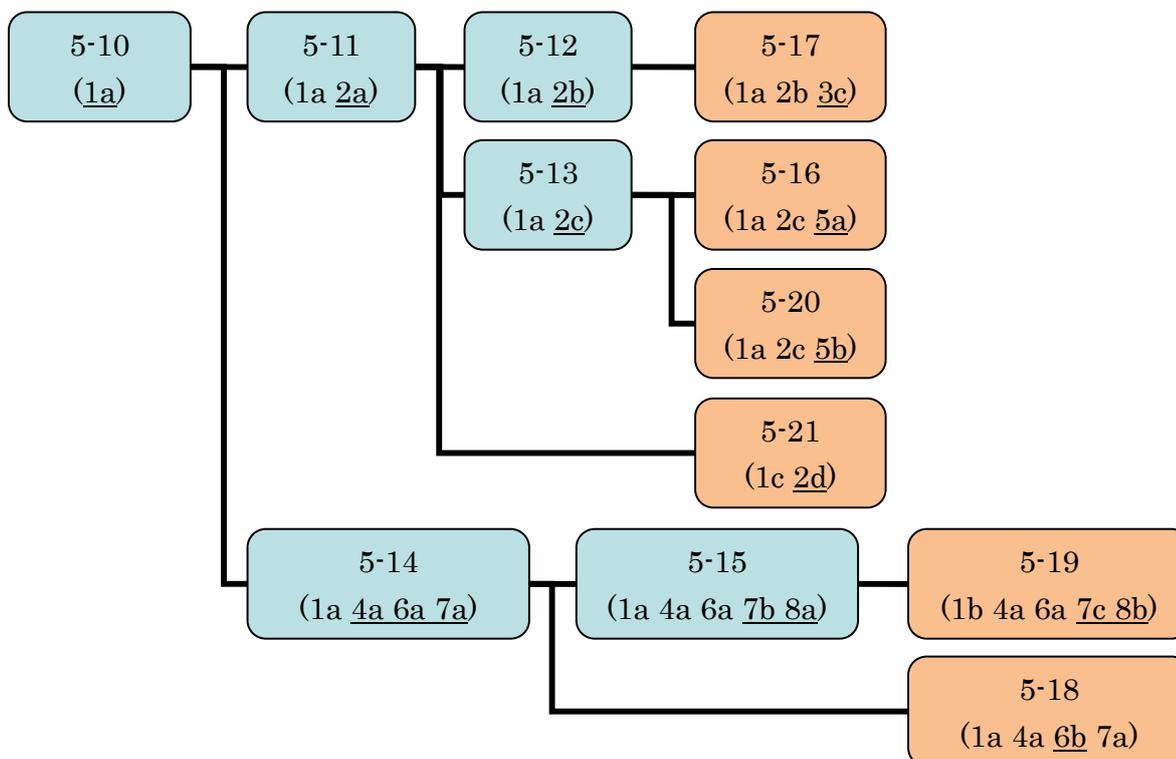
この文書の配布に制限はない。あくまで解答「例」であり、「模範解答」ではないことに注意。この内容を信用・利用したことにより発生した損害については、作者は一切の保障をしない。誤りを発見した方は、ぜひ作者まで連絡していただきたい。

共通問題 1

- 1) 1 章, 2 章, 5 章
- 2) 5-10, 5-11, 5-12, 5-13, 5-14, 5-15
- 3) 7 章
- 4) (1 章) 複数あり、5-19, 5-21 (2 章) 複数あり、5-12, 5-13, 5-21 (3 章) 5-17 (4 章) 5-14 (5 章) 複数あり、5-16, 5-20 (6 章) 5-18 (7 章) 5-19 (8 章) 5-19
- 5) 各章について、書き直した際に元となる版を「親」、書き改めた後の版を「子」して、階層モデルを用いた木構造グラフを描く。ここで、「子」を持たない版、つまりその版を元にして書き改められた版が存在しないものを集めることで、最新のものを集めることができる。

◆ 解説

まず、問題文に記されている操作をグラフにして描いてみよう。ここでは、「書き直し」によって新たな版が出来るたびに、その版に 1a, 1b といった英小文字を振り、版の区別をする。



この図は、「階層モデル」を用いて描かれている。全ての版は、ある一つの「元となる版（親）」の子供であるから、こうして表現することができるのである。

この図を見ることで、2) の問題を解くことができる。図中で茶色に塗られているものは、削除してはいけないファイルである。それ以外のファイルは、それを元に改善したファイルが既に存在しているから、削除しても問題ない。

次に、5) の解答で述べたように、各章の版について、版の親子関係を木構造によって表現してみよう。ここで、その版が作られた時のファイル名を併せて記す。

1章 a(5-10)	$\begin{cases} b(5-19) \\ c(5-21) \end{cases}$	5章 a(5-16)	
		b(5-20)	
2章 a(5-11)	$\begin{cases} b(5-12) \\ c(5-13) \\ d(5-21) \end{cases}$	6章 a(5-14) — b(5-18)	
3章 a(5-17)		7章 a(5-14) — b(5-15) — c(5-19)	
4章 a(5-14)		8章 a(5-15) — b(5-19)	

上の図で、例えば1章については、5-10 で作成された a 版を元である「親」として、b 版と c 版の 2 つの「子」が平行して作成されていることが分かる。問題文中における「最新」の版とは、上の図において、「子」のない版のことを指す。

であるから、1) についてはこの図を見ることで解くことができる。また、3) については、「書き直したものがさらに書き直されている」、つまりある版の「孫」(=「子の子」) が作成されている章だから、これは7章しかない。

4) についても、上の図に、それぞれの章について「最新」の版が作成されたときのファイル名が書かれているから、それを列挙することで解答することができる。もちろん別解として、2) で考えた「削除してはいけないファイル」だけを用いることもできる。その場合の解答例は以下のとおり。

4 別) (1 章) 複数あり、5-19, 5-21 (2 章) 複数あり、5-16, 5-17, 5-21 (3 章) 5-17 (4 章) 5-18
 (5 章) 複数あり、5-16, 5-20 (6 章) 5-18 (7 章) 5-19 (8 章) 5-19

なお、この問題中で何度も言及した「階層モデル」については、教科書 p84 以降に説明があるほか、過去問にも何度も出題されているため、そちらの解説も参照のこと。重要なのは、ひとつの「親」に複数の子がぶら下がっていると、ある「子」の「親」となるものは常に 1 つ以下であるということである。教科書 p85 の図 4.20 を例にとるならば、「東京都」は子として「文京区」や「目黒区」を持つが、「目黒区」といえば、その親は「東京都」以外にはありえない、ということである。

共通問題 2

(ア) JIS (イ) 文字化け (ウ) 65535 (エ) GUI (オ) CUI (カ) 復号

- (1) 誤りあり：標本化 → 標準化
- (2) 誤りなし
- (3) 誤りあり：マウス → キーボード
- (4) 誤りあり：プログラム言語 → アプリケーションプログラム
- (5) 誤りあり：符号化 → 暗号化

◆ 解説

- (1) 教科書 p19 参照。JIS とは、日本工業規格 (Japanese Industrial Standards) の略であり、工業における日本の国家標準である。JIS コードはその一環として、日本語文字をデータ化する文字コードとして定められているものである。あるデータがどの文字コードで符号化 (情報をデジタルデータとして表現すること) されているかが分からないと、誤った文字コードとして読み取ってしまうことがあり、この際に「文字化け」が発生する。
なお「標本化」とは、連続的なアナログデータ (音声など) をデジタルデータに変換する際に行う処理である (教科書 p24 参照)。
- (2) 教科書 p21 参照。なお時代が下るにつれ、コンピュータのビット数は増大している。16 ビットということは、0 と 1 が 16 個並んでいるということだから、2 の 16 乗で 65536 種類の数値を表現することができる。問題文をよく読むと「0 から」と書いてあるから、上限値は 65535 である。
- (3) 教科書 p219 参照。GUI とは Graphical User Interface の略で、現在広く使われている Windows や Mac OS でも採用されている。マウスとキーボードを併用して操作するものである。対して CUI とは Character User Interface の略であり、Character つまり文字を出入力することでコンピュータを操作する仕組みである。文字の入力に使用するのは、もちろんマウスではなくキーボードである。
- (4) 教科書 p240 参照。著作権は、数学的な操作であるアルゴリズムや、それを記述する方式の規約であるプログラム言語については保護しない。いっぽう、アルゴリズムを組み合わせ、プログラム言語を使用することで構築されたソースプログラムには創作性が認められ、著作権が発生する。オブジェクトプログラムとは、人間の書いたソースプログラムを、機械が理解しやすい形式に翻訳 (変換) したものである。OS やアプリケーションプログラムは、オブジェクトプログラムによって構成されている。なお、解答例では教科書の例示に沿って「アプリケーションプログラム」と訂正したが、その他著作権で保護されるものであれば正答となると思われる。
- (5) 教科書 p48 参照。伝達したいデータの、誰でも読める状態にあるもののことを「平文」といい、それを意図した相手以外に読まれたくない際にする処理を「暗号化」、受信した相手が正当な手段によって再び読めるようにすることを「復号 (化)」という。意図しない第三者がそのデータを読もうとすることを、暗号の「解読」という。なお、音声情報でなくとも「盗聴」という言葉は用いられる。

共通問題 3

問題 A

1)

a) 機器やアプリケーションを、クライアントとサーバに分離するネットワーク方式のこと。クライアントは、利用者の操作に基づいて、ネットワークを介してサーバに対し情報や処理を要求する。サーバは集中的に管理され、随時クライアントからの要求を受け付け、それを処理して返答する。

b) Web サイト。ブラウザアプリケーションがクライアントとなり、利用者の要求した URL に基づき、Web サーバへその URL に対応するファイルの送信を要求し、受信したファイルを利用者が閲覧可能なように表示する。Web サーバは、クライアントから要求されたファイルをサーバ内から探し出し、その内容を返答する。このクライアントとサーバのやり取りには、プロトコルとして HTTP が使用される。

スーパーコンピュータ。高度な計算機能を有するスーパーコンピュータは高価であるため、集中的に管理され、サーバとして動作し、クライアントから計算処理の要求を受け付け、その処理結果を返答する。クライアントは、処理したい数式や統計データなどを利用者に入力させ、その情報をサーバに送信する。

2) 同時にアクセスがあったとしても、ひとつの席に複数の予約を入れてしまう重複予約が発生することがあってはならず、排他処理を確実に行う必要がある。

発売時刻直後などで多くの予約が殺到した場合でも利用者の利便性が損なわれないよう、出来る限り迅速に予約処理が行えるように、高性能なハードウェアを導入することなどにより、負荷に耐える余裕のあるシステムを構築する必要がある。

代金の決済処理をするにあたり、クレジットカードの番号といった重大な個人情報を扱うため、これがバグや盗聴によって漏えいしないよう、システムの入念なチェックや、SSL 技術を用いた通信経路の暗号化を行う必要がある。

◆ 解説

教科書 p46、p193 参照。クライアント/サーバ型のシステムは、ネットワークを利用するあらゆる場面で採用されているため、自分が普段利用しているシステムを思い浮かべることで解答することができる。なお、一般にサーバは多くの利用者からのアクセスを処理するために高価な機器が使用されることが多い。クライアント/サーバ型を採用することで、機器のコストやメンテナンスを一元化できるという利点がある。

なお、クライアント/サーバ型に対する概念としてピアツーピア型 (P2P) がある。通信するコンピュータが互いに対等な関係であり、弱点となりうる特定の中央サーバをもたないため、障害に強いというメリットがある。一方で、中央がないことにより匿名性が高く、違法なファイル交換にも利用されているという実態がある。

問題 B

1) アルゴリズム 1: $\frac{n(n-1)}{2} / O(n^2)$

アルゴリズム 2: $n-1 / O(n)$

2)

アルゴリズム 3

a) $n-1 / O(n)$

b) いちど金貨を天秤ばかりにかける度に、かけた金貨のうち軽いほうの 1 枚が最重金貨の候補から外れることになる。n 枚の金貨から 1 枚の最重金貨を選び出す際には、残りの n-1 枚を候補から外れさせる、つまり敗退させる必要があり、これには n-1 回はかりを使用する必要がある。

アルゴリズム 4

a) $\frac{n(n-1)}{2} / O(n^2)$

b) ※印の操作 1 回目を行うことで、最も軽い金貨が一番右に移動する。このとき、n-1 回の比較を行っている。※印の操作 2 回目には、一番右は既に分かっているから、n-2 回の比較を行うことで、2 番目に軽い金貨を右から 2 番目まで移動させることになる。同様に、※印の操作 k 回目には、n-k 回の比較を行う。k=n-1 までこの操作を繰り返すことで、金貨は重い順に並ぶ。このときの天秤の使用回数の合計は、 $1+2+\dots+(n-1)$ となる。

◆ 解説

計算量のオーダーとは、計算数を大まかに見積もる指標であり、ここでは多項式である天秤ばかりの使用回数のうち、最高次の次数のみに着目し、低次や係数を無視して表記する（教科書 p134）。

1) については、容易に算出することができるだろう。

アルゴリズム 3 は、問題文を整理すれば（勝ち残り式）トーナメントであることがわかる。「重さの比較」は 2 枚の金貨に対してしか行っていない（天秤の定義より、その他の測り方はできない）。

アルゴリズム 4 は、バブルソートと呼ばれるアルゴリズムである（ソートとは、データを順番に並び変えること）。

講評

どの問題も過去に出題されたものの類題であり、過去問を一通り解いておけば必要な知識はほとんど身に付く。特に階層モデルは頻出であるが、そのものについて問題文で触れているのではなく、出題を整理する段階で用いる必要があったため、字面だけの知識では対応しにくかっただろう。

選択問題は、A が同じことを繰り返し述べるに留まりがちになる出題である一方で、B は落ち着いて考えれば平易な数学であり、文理を問わず選択することが可能だったと思われる。