情報シケプリ

はじめに

U-Taisakuを見ますと教科書をわかりやすくまとめたもの、過去問に関して丁寧な解説を行ったものを偉大な先人の方々が執筆されていることがわかります。そのような物がある中で私が同じような物を作っても見劣りするだけなので今回は、過去問の傾向を分析して、その出題傾向に即したシケプリを執筆しました。つまり問題を解くためのシケプリです。授業が分からない方でもこれを読むだけで成績良以上、がんばれば優という内容を目指しますが、内容を簡易に、成績を取るため最小限にするために一部至らない（専門的に言えば例外もあるけど……）内容があるかもしれません。
出題範囲の教科書の内容はほぼ網羅しているはずなので、これを読めば教科書を読む必要はありませんが、過去問（2006～2011年共通問題過去問）を解く必要があります。これの解答については、2010年までの分は「only my imformation（マシュー執筆）」を参照してください、2011年の問題の解答は無いので作成しました。

試験傾向分析

教科書を見てその内容の広さに圧倒された人もいるかもしれませんが出題される（出題できる）範囲は限られています。私は独自に出題範囲を11種類に分けました。記号は重要度です。05年以前の問題は傾向が違い、まだまだ6年分しかデータがないために、傾向分析という点では不十分かもしれませんが。

分類	過去問2011	過去問2010	過去問2009	過去問2008	過去問2007	過去問2006
基礎		2-1 2-2				
インターネット・情報システム	1-1(1) 1-1(3) 1-1(4) 1-2	3A	1-2		2	1-1c 3
データモデル		1		1		
プログラムとアルゴリズム	2	3B	2	3B	3B	2
暗号	1-1(2)	2-5			1	1-1d
標本化・量子化			1-1			1-1a 1-1b
情報量と確率	3B			2		
論理回路			3B			
ユーザーインターフェイス	3A-1	2-3		3A(1)	3A(3)	
著作権		2-4	3A	3A(2)	3A(1)	
情報と社会	3A-2				3A(2)	

A/Bは選択
○基礎
情報・コンピューターに関する基礎的なことです。雑学的な内容を網羅する必要があります。基本的に出題はされませんが、後の内容を理解するにあたって重要なのでしっかりと覚えるように。
◎インターネット・情報システム
インターネットの仕組みや、一般的なネットワーク、サーバー・クライアントに関する知識を問われます。本質的な理解はとても難しいので、せめて用語の暗記をお願いします。
○データモデル
ツリー構造について出題されます。アルゴリズムと複合することも。
◎プログラムとアルゴリズム
最重要そして最難関な所。自分で仕組みを考えて記述していくパズルのような問題であるアルゴリズムの問題と、は試験用紙に書かれた擬似プログラムを読んでどのような処理が行われているか推測するプログラム問題がでます。とにかく問題を解きましょう。
◎暗号
毎年出題されている重要単元ですが、覚えることも多くなく、解くのは簡単です。
△標本化・量子化

情報量＋アナログとデジタルの内容になりますが、狙って、CDの情報量の問題が過去に2回出されており、周期的に今年が出されそうなので載せておきます。
○情報と確率
ただの計算問題です。情報量の定義を覚えて問題を少し解けば満点が取れます。
△論理回路
過去に1度だけ出たことがあります。優先度は低いですが狙われるポイントは限られていますので、そこを重点的に取り組んで終わらせましょう。
○ユーザーインターフェイス
△著作権
△情報と社会
初歩的な内容が問われますが、ほとんどは第3問の選択問題の片方として出題されます。優先度は高くはないが、基本的なことは第1問、第2問で出ることがあるので覚えておきましょう。
大問3題構成で、3問目はA、Bに分かれていてAは文系様（笑）のための問題です。とても高度な思考力を要求されるので、素直にBを選びましょう。Bのほうが一見とっかかりにくいですが、慣れれば・しっかり読めばちゃんと解けます。各章の内容は0.概要、1.内容、2.演習、3.コメント で構成されています。演習は過去問から該当する範囲のものを撮ってきていて各章の中に2006～2011年の過去問のすべてが入っています。1.内容は、これらの過去問を解くために必要なことが書いてあるのでおそらく簡単に解けるはずです。

基礎

教科書対応範囲：1.4.2　2進数モデル
2.2.1 ピクトグラム
基礎と言いながらあまり内容は無いけど

2進数

コンピューターの内部では2進数の演算が行われている。普通コンピューターの中の電子回路では2進数の「1」は高い電圧、「0」は低い電圧で表される。10進数では1234を4桁の数と呼ぶが、2進数の場合は1010のような数を4桁の数とは言わずに、寧ろ4ビットの数という。例えば2進数の100010は6ビットの数である。コンピューターが一度に処理できるビット数をそのコンピューターのビット数という、例えばIntelのcore iシリーズは64ビットのCPUである。例えば**8ビットのCPUなら0～256(=2^8-1)の非負の数**を、16ビットのCPUなら0～65535(2^{16}-1)の非負の数を表すことが出来る（非負とは負でない数つまり0と正の数のこと）。2進数の足し算ができるといいかも。
2進数における概念として**ハミング距離**がある。これはある2つの数の距離を表す概念で2つの数のうち値が違う桁の数をいう。たとえば4(0100)と5(0101)は一の位の桁のみが違うのでハミング距離は1、7(0111)と12(1010)は一の位、百の位、千の位が違うのでハミング距離は3である。

コンピューターの仕組み

コンピューターの基本は以下のような部分から成る。

演算装置	中央処理装置：CPU
制御装置	
記憶装置	主記憶装置（メモリ）

現在のコンピューターは記憶装置に格納するプログラムを変えることで様々な動作をすることができる、このようなコンピューターを**プログラム内蔵方式**または**フォンノイマン型コンピューターという。**

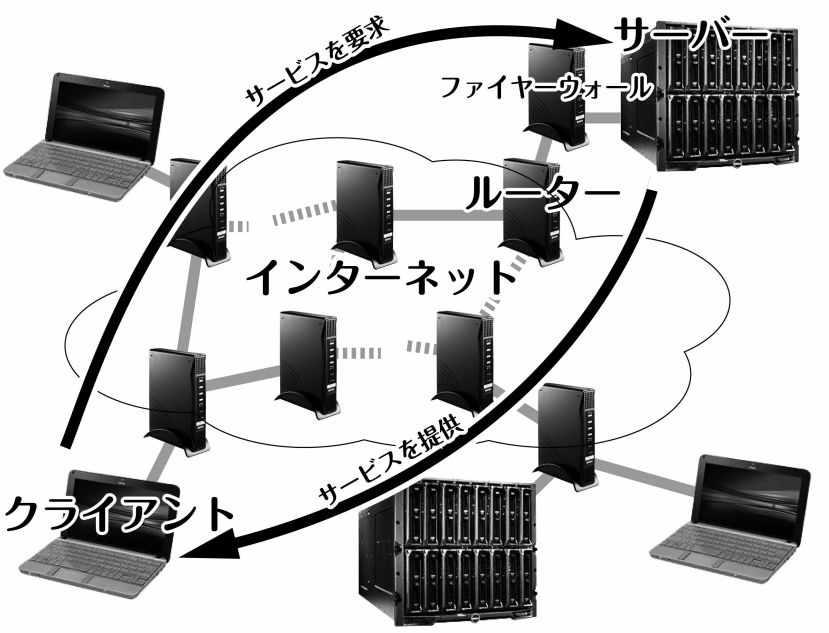
文字コード

コンピューターは数字しか扱えないので、ワープロなどのソフトで文字を扱うためには文字を数字と対応させて操作しなければならない。この対応のことを文字コードという。**日本語の文字コードは文字が多いために複雑でかつ複数の文字コードがありそれらが混在している。**例えば**JIS**や**シフトJIS**、**EUC-JP**などがある。例えばJISの文字コードで描かれた文章をEUC-JPで解読してしまうと、文章が間違ったものになってしまう。この現象を**文字化け**という。

インターネット・情報システム

教科書対応範囲：3.2 情報通信（3.2.1、3.2.2(a)、3.2.2(a)）
3.4 インターネット（3.4.1、3.4.2、3.4.3、3.4.7）
8.2 情報システムの仕組み
これらを本質的に理解することはとても難しいので、雑学的で散逸な内容の暗記が中心となる。イメージがしやすいように色々な例えを出すが、何を行っているのかわからないのなら、用語の暗記だけでも。

インターネットの仕組み（クライアントとサーバー・プロトコル）



インターネットで行われているのは一言でまとめるとメッセージの送受信といえるがその内容は様々でありそれらのメッセージの形式の約束を**プロトコル**という。例えばはがきでは裏面の真ん中に大きく宛名、右側に送り先の住所を書いて、左側に小さく自分の名前、住所を書くというのは郵便のプロトコルと言える。
例えば自宅のパーソナルコンピューター（以下パソコン）で、WWW を使っている時（=Web ページを見ている時）、パソコン（正確に言うとパソコン上の WeB ブラウザ）とインターネット上のサーバーコンピューター（以下 Web サーバー）がメッセージをやり取りしている。例えば wikipedia の情報という名前のページを見たい時は、Web ブラウザは wikipedia の Web サーバーに情報という名前の Web ページを送れというメッセージを送り、サーバーはそれに応えて Web ブラウザに情報のページのファイルを送る。Web ブラウザとは InternetExproler や FireFox、Safari などのアプリケーションプログラムのこと。これらの **Web ブラウザと Web サーバーメッセージのやり取りの形式を規定しているプロトコルを HTTP**（HyperText Transfer Protocol）という。また一般に**要求を出す方（この例ではパソコン）をクライアント、要求に答える方（今回はサーバーコンピューター）をサーバー**という。
この **HTTP に即したソフトウェアなら Web ブラウザでなくても Web サーバーと通信をすることが出来る。**例えば「YouTube Downloader」というツールは Web ブラウザはではないが、You Tube の Web サーバーと通信して、動画データをダウンロードをするソフトである。また Web ブラウザはではないが自動的に文章を収集するプログラムがある。
また HTTP を用いた通信はパソコンと Web サーバーだけに限られず、スマートフォンとパソコンをインターネットを用いてつないで、外出先から自宅のパソコンの中にあるファイルをダウンロードすることも出来る。
HTTP の親戚としてメールの送受信を行うプロトコルの SMTP（Simple Mail Transfer Protocol）がある。
マシン同士の通信では時に送信側のマシンと受信側のマシンは直接つながっている必要はない。両者がインターネットを介して間接的に継がっていれば、次の章で述べる仕組みによって通信を行える。

インターネットの仕組み（4 つの層、実際の通信のプロセス）

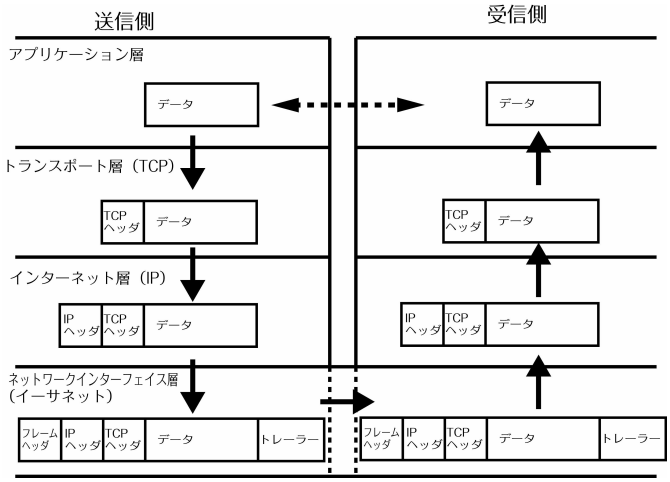
インターネットで Web ページを見ている時の通信は全部で 4 つの層に分けることができます（下の表を丹念に覚える必要は無いと思います。まあ並び替えが出来るぐらいで大丈夫）。読んどくだけでよいです。

層の名前	主なプロトコル	役割
①アプリケーション層	HTTP	アプリケーションプログラム間の通信
②トランスポート層	TCP	1 対 1 の通信
③インターネット層	IP	ネットワーク間の通信
④ネットワークイン	Ethernet	ネットワーク内の通信

ターフェイス層		

WWWサービスで「http://ja.wikipedia.org/」のページ（Wikipediaのトップページ）を見る時の具体的なデータの流れは以下。
① Webブラウザが（パソコン）が「http://ja.wikipedia.org/」に対応するIPアドレスを調べて、「208.80.152.201」というIPアドレスを得る（これについては後述）。
② Webブラウザが TCPポート80からIPアドレス208.80.152.201に対してHTTPのメッセージを送る。これは次のように処理される。

(a:トランスポート層)HTTPのメッセージを大きいのでこれを小さな単位（約1.5KB）のセグメントに分けてTCPヘッダを付ける（TCPヘッダには元に戻せるように分割した中の何番目なのか等の情報が入っている）。
(b:インターネット層)上記の者にIPヘッダをつける。
(c:ネットワークインターフェイス層)これにフレームヘッダとトレーラーを付ける。
③②のメッセージがWebサーバー（wikipediaのサーバー）にたどり着いて、Webサーバーは要求されたページをWebクライアントに向けて送る（上記の①、②の動作を行う）。

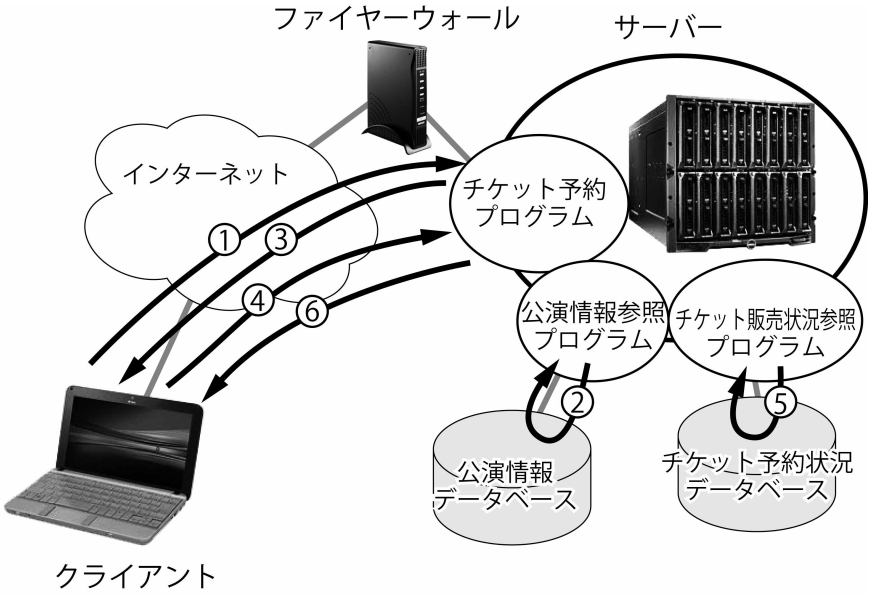


ヘッダとはデータの先頭につけるもの、トレーラーとはデータの末尾に付けられる物のこと。
上のようにデータにヘッダやトレーラーをつけることをデータを**カプセル化**するという。
上のような**分割されたデータにヘッダとトレーラーがついたものをパケット**と言う。インターネット上では実際はこのパケットのやり取りを行なっている。
IPアドレスは32ビット（＝32桁の2進数で表される）の数値で普通は8ビットずつの4つに分けた値を10進数（0～255）で、例えば「192.168.255.1」のように表す。**インターネットにつながっている全てのマシン（コンピューターのこと）は固有のIPアドレスを持つ**。つまり、IPアドレスが208.80.152.201と言うとwikipediaのサーバーの1つに限定できる。
TCPのポート番号は16ビットの数値である。通常**マシン上では複数のアプリケーションプログラムが動いており、そのどれにデータを送るかをこれで決める**。これによりマシン上でWebブラウザとメールソフトが同時に動いていても、混ざらずに正しく通信ができる。
このように層を分けるのは面倒な事に見えるが、これによってWebクライアントとWebサーバーが直接線でつながっているかのように通信することができる。つまり、Web上で通信をするプログラムを書くときにプログラマは面倒な途中の経過を気にすることなく、プログラムの本体に集中できる。

ホスト名とIPアドレス、DNS

ホスト名
URLのはじめの方の部分（http://www.u-tokyo.ac.jp/index/b00_j.html の www.u-tokyo.ac.jp）を**ホスト名**という。**ホスト名はIPアドレスと1対1で対応している**。コンピューターが処理する場合は数字の羅列であるIPアドレスの方が便利だが、人間が使う時は意味のある文字列であるホスト名の方が使いやすい。これは携帯電話の番号と人の名前を対応させているのに似ている。
ホスト名は階層構造を取っており、上の階層から「jp」→「ac」→「u-tokyo」→「www」となっている。jpは日本のサーバーであること（「us」だとアメリカ）を、「ac」は教育機関であること（「co」だと企業）を表します。これは仮想上での住所に当たるが、**だいたい実際の住所、運営主体と関連性がある**。
ホスト名とIPアドレスの対応表を保管しているコンピューターをDNSサーバーと言う（携帯電話の電話帳の機能に当たる）。またこの仕組自体もDNSと言う。

Webサービスの例



Webサービスの例としてインターネット上でのチケット予約サービスの仕組みを解説する。このサービスには複数の要素がある。
・ジャンル別、地域別の購入可能な公演チケットの照会サービス
・チケットのハンバンスケジュールの照会サービス
・クレジットカードによる決済サービス
またこのときに必要なデータとしては
・公演内容（日時、演目）
・公演者（名前、プロフィール）
・公演会場（会場名称、住所）
等がある。
システムに柔軟性を持たせるため、セキュリティー上の都合などから通常チケット予約システムとは別のシステムにこれらの情報を保管する。**このようなシステムをデータベースという**。チケット予約システムの動作の一例をいかに示す。
① Webクライアントが Webサーバーに公演情報の要求をする。
② Webサーバーのチケット予約プログラムか応えて、公演情報参照プログラムが公演情報データベースにアクセスして公演情報を取り寄せる。
③ Webサーバーのチケット予約プログラムは Webクライアントに要求されていた公演情報を提供する。
④ Webクライアントはチケットの購入を Webサーバーに要求する。
⑤ Webサーバーのチケット予約プログラムが応えて、チケット販売状況参照プログラムがチケット予約情報データベースにアクセスして、チケット予約状況を確認する。
⑥ Webサーバーが Webクライアントに返答する。
……
上記の例では、Webクライアントと webサーバーというクライアントサーバーの関係があるが、Webサーバー（公演情報参照プログラム）とデータベース（公演情報データベース）の間の通信を見た場合、webサーバーがクライアント、データベースがサーバーの仕組みとなっている。
またクライアントから来た要求は、サーバーの上で動いている具体的な処理をするプログラム（この例ではではチケット予約プログラム）が受け取り、他のプログラムと強調して処理をしている。このようなプログラムを**アプリケーションプログラム**（適用プログラム）と言う。

ファイヤーウォール

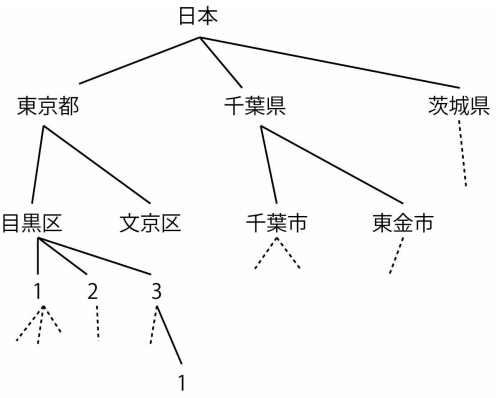
通常の Webブラウザから上記の予約システムの Webサーバーにアクセスすることは許されるが、不正なアクセスによりデータベースに直接アクセスしてデータベースの内容を書き換えられてしまうことがあり。それを防ぐために Webサーバーとインターネットの間に**ファイヤーウォール（防火壁）を備えてこのような不正なアクセスをブロックしている**。

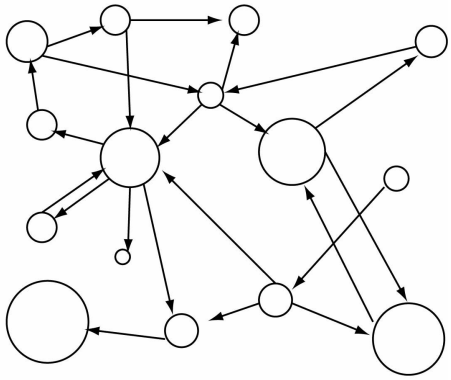
データモデル

教科書対応個所 4.3 代表的なデータモデルと演算（ネットワークモデルと階層モデル）
難しいことがたくさん書いていますが、内容は簡単です。階層モデルとネットワークモデルの違いに注意をして両者を理解しましょう。過去問を見るとどれも、頭の体操のような物で、何を言っているのかがわかれば前提知識なしで解けますが。用語を問う問題が出るかもしれないので、そこを押さえておけば、あとは演習をやるだけです。

階層モデル

ツリーモデル、木構造と同義で、教科書には住所が例に挙がっている。いわゆる末広がり構成。
個々の「日本」、「駒場」などのものを**ノード**、これらをつないでいる線を**有向グラフ**という。本来有向グラフは矢印付きの線分で表すが、階層構造ではでは向きは明白なため、わざわざつけることはしない（付けてもかまわない）。





階層モデル　　ネットワークモデル

性質は以下の2つ
・一意性
例えば『駒場キャンパスの住所」はと聞かれたときに「日本→東京都→目黒区→駒場→3→8→1」と一意的に決められる。逆に言うと1つの場所に複数の住所の表し方は存在しない。
・階層の上下があるということ
「目黒区」の1つ上の階層は「東京都」。1つ下の階層には「青葉台」、「大岡山」……「駒場」、「五本木」……「祐天寺」があるとはっきりと言える。
この階層構造は非常に便利な物だが例えば本の分類をするとき「工学と芸術」なんていう本があったらそれは1つのカテゴリに出来ないのでこの構造では表せない。以下はこの階層構造が適当な物。
・住所
・パソコン内のファイルの位置（ファイルシステム）
・生徒と所属クラス、学年
例えば家族構成は微妙に階層構造ができない物だが（ある人の親は父と母の2人いる）、もし家長制度があり「ある人の家長をたどっての上の世代を探す」という処理をしたいなら階層構造を適用できる。このようにデータそれ自体も行いたい処理によって階層構造が使えるかが変わる。

ネットワークモデル

インターネットのウェブページがその典型です。複数のノードが複数のグラフで複雑に結ばれています（このグラフは有向・無向かは場合によるが簡易化のため有向とする）。
性質は以下
・構造に上下がない
あるノードをさしているノード（階層構造なら上のノードに相当）は複数あり、これをたどると自分にかえってくることもある。
この階層構造は関係複雑なでも表せるという点では非常に便利だが、これをもとに分類調査を行うことは非常に大変。

プログラム・アルゴリズム

教科書対応範囲：第5章、第6章
ほとんどがパズルです。以下のプログラムを読めと書いてあっても問題文中に構文の意味などが解説されていて、ただ「AがBだったらCを行う」を「if A が B then C endif」としているだけです。ひたすら問題を解きましょう。

構文説明

今回は擬似プログラムを使う。プログラムは本当はコンピューター上で実際に実行出来るもののことを言いますが、これには細かい仕様がたくさんあり、アルゴリズムを厳密に書く用途には少々大振り過ぎるので、プログラムっぽいものを使ってアルゴリズムを書く。この時に使う記号は各自の好きなように定めて構わないが、今回は教科書に載っている例を載せる。特に構文については説明が問題中でされるはずだが、ここで覚えていくべきです。
・演算子
+:足す　　-:引く　　*:掛ける　　/:割る　　←:代入

a ← 3 と書くと変数 a に 3 を代入するという事になる。
・条件式
a<b :a が b より小さい　　a<=b :a が b 以下　　a>=b :a が b 以上　　a>b :a が b より大きい

```
条件分岐構文
if 条件
        then 条件が成立した時の処理
        else　条件が成立しない時の処理
endif
（else の 1 行はなくても構いません）
```

```
ループ構文（条件が成立している間、処理を繰り返す）
while 条件 do
        処理
done
```

a=b :a と b が等しい　　a!=b :a と b が等しくない
配列
意味のある数字の列、と言うよりもただ変数を扱いやすくしたものと理解するべき。例えば生徒 1～10 までいた時に、各々のテストの点数を score1、score2、……score10 と 10 個の変数を定義するよりも、$score_i$ として i には 1～10 の値が入るとしたほうがわかりやすいよねということ。$score_i$ ではなく score[i]と表すこともある。

演習問題

次の各々のプログラムの結果（プログラムの最後にある return x における x の値）を求めよ
1-1

```
a←5
b←6
y←0
if a>b
        then y←a
        else y←b
endif
return y
```

ただの if 構文の練習問題 a>b ならば y に a を代入、そうでないならば（a<=b ならば）b を代入する。つまり a、b のうちで大きい方を y に代入する。よって答えは 6

1-2

```
a←10
y←0
when a>0 do
        y←y+a
        a←a-1
done
return y
```

while のループの問題です。a は始めは 10 でそれから 1 まで減って、a が 1 の時に「a←a-1」が実行されることで、a が 0 になり、「a>0」を満たさなくなるのでそこでループから抜けます。つまり y には 10、9、8～1 の和となり答えは 55

1-3

```
x は 5 個の配列とする（つまり x1～x5 までがある）

x1←69
x2←87
x3←30
x4←90
x5←77
i←1
y←0
while i<=5 do
        if(xi>y)
                then y←xi
        endif
done
return y
```

配列の問題です。i が 1 から 5 まで変化していき、その各々で y と x_i を比較して、大きい方を y に入れます（y>=$score_i$ の時は何もしませんが、つまりこれは y に大きい方である y を代入するのと同等の行為になる）。よって y は $score_i$ のうちの 5 つ全てと比較して大きいまたは等しい score のうちのどれかの値となるので、y には score のうちの最大値が入ります。よって 90

暗号

教科書対応箇所 3.2.3(a)通信の秘密と相手の認証（共通/公開鍵暗号）
共通鍵暗号方式・公開鍵暗号方式の違いに注意して両方を理解すれば問題ありません。計算や頭をつかうような問題はなく、文章題で正誤や選択式の問題となります。ほぼ確実に出題されますので確実に点を稼ぎましょう。

暗号とは

メッセージ・データを送信者から受信者に送る時に、送った内容を途中で第三者に盗み見られてもわからなくする為に行うのが暗号化である。
①まず送信者は**暗号鍵**を用いて**平文（伝えたいもとの文章）**を**暗号文**を作る。
②暗号文を送信者から受信者に送る。この暗号文は第三者が見ても読むことはできない。
③受信者は暗号文を**復号鍵**を用いて平文に直す。
暗号鍵、復号鍵をどのようにするかによってによって暗号の種類が以下の 2 つに別れます。

共通鍵暗号

一番簡単なものとして「文字を指定の数後ろにずらす」という暗号を考えます。この場合何文字ずらすかが鍵になります。今回は「ノルマンデイ」という文を送り、1 文字ずらすとします。以下の暗号通信をするためには、「ずらす文字数は 1」という事を二者が共有しなければなりません。この**共有しれなければいけない物を共通鍵**と言います。**暗号鍵、復号鍵は共有鍵から生成します**（共有鍵→暗号鍵「後ろに 1 文字ずらす」、共有鍵→復号鍵「前に 1 文字ずらす」）。
①「ノルマンデイ」を 1 文字ずらして「ハレミアドウ」とします（暗号化）
②送信者から受信者に「ハレミアドウ」という暗号文を送ります。第三者はこの文を見ても何かはわかりません。またこれとは別に共通鍵をも受信者に送ります。
③受信者は 1 文字前にずらして「ハレミアドウ」→「ノルマンデイ」と平文に直します。
送信者はこの共有鍵を受信者に伝えなければなりませんがどうやって伝えて共有するかが問題になります。②で使うような**第三者に盗み見されるかもしれない経路よりも信頼性の高い経路で共有鍵を伝える必要があります**。共通鍵が第三者に盗み見られると、暗号文を解読されてしまいます。

公開鍵方式

簡単に説明できる具体的な例がないので概念のみの説明になりますが……そういうものだと思って暗記してください。
以下の説明では、**公開鍵＝暗号鍵**、**秘密鍵＝復号鍵**になります。
①受信者は秘密鍵を決めそれから公開鍵を生成する。
②受信者は送信者に公開鍵を送る。この暗号鍵は第三者が盗み見ても構わない
③まず送信者は公開鍵を用いて平文（伝えたいもとの文章）から暗号文を作る。
④暗号文を送信者から受信者に送る。この暗号文は第三者が見ても読むことはできない。
⑤受信者は暗号文を秘密鍵を用いて平文に直す。
秘密鍵から公開鍵を生成できますが、公開鍵から秘密鍵を生成できないことがこの方式の特徴です（これには素因数分解の困難性を使います。つまり2つの素数をかけて大きな値を得ることは簡単だが、大きな値を2つの素数の積に素因数分解をするのは難しいということ）。つまり**第三者は暗号鍵と暗号文の両方を知っていても平文を作ることはできません**。共通鍵方式であるように鍵を別な安全な経路で送る必要がありません。

標本化･量子化

教科書対応範囲 2.3 アナログとデジタル
過去に2回出題されたことがありますがどちらもCDに関するほとんど同じような問題で基本的な内容です。周期的に今年は出題される可能性が高いのでしっかりと得点できるようにしましょう。中身は計算問題がちょっと入る程度で簡単ですが、覚えるべき用語が少々あります。特に1.2以降の内容ではアナログレコードからデジタルなCDに音楽データを変換することを前提として説明がされています。
連続的に表されるアナログな現実世界の何かしらの物を、デジタルデータとして記録するためには**標本化**と**量子化**が必要である。

デジタルとアナログ

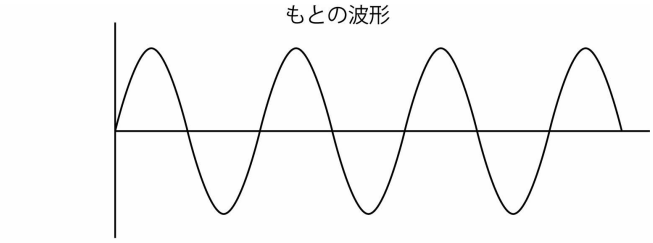
	アナログ	デジタル	
何らかの状態を	連続的な値として	ある一定間隔の尺度に近似した離散値として	表す。
例えば	24.3284……℃の	24.3℃	のような値となり、
精度については	無限の精度を必要とする。	精度は有限で良い。	
複製では	劣化が避けられない。	正確に出来る。	
媒体としては	レコード、オーディオテープ、ネガフィルム、ビデオテープ	MD、CD、メモリーカード、DVD	がある。

アナログ媒体からデジタル媒体にデータを変換する時は元々のアナログデータの持っていた連続的な性質が少なからず切り捨てられることに注意。**満足な結果にするためには、標本化、量子化の間隔を十分な物にする必要がある。**

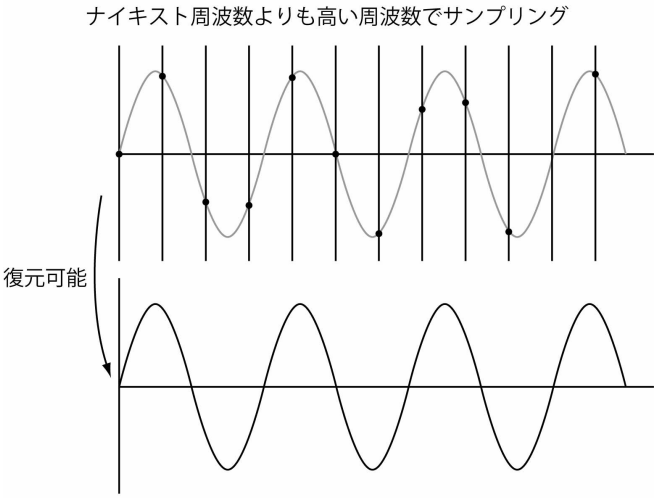
標本化

アナログデータをデジタルデータに変換する過程で、元のデータをある一定の時間間隔で抽出する必要がある。例えば、ある地点の気温を1分間隔で記録するのか、それとも1時間、1ヶ月間隔ですればいいのかという問題だ。1秒間に何回データを抽出するかを**サンプリング周波数**（単位はHz）という、例えば1秒間に100回抽出（サンプリング）する時のサンプリング周波数は100Hzとなる。元のデータの半分の周波数を**ナイキスト周波数**という。もとのデータの周波数成分の内ナイキスト周波数以下の周波数成分は完全に再生できることが証明されていて、これを**標本化定理**という。**音楽CDのサンプリング周波数は44.1kHz（=44100Hz）であるのでその半分の22.05kHzまでの周波数成分までを正しく再生できる。**これより高い周波数が元のデータに入っていた場合はデータが変わってしまう（正しく復元できない）。この現象を**エイリアシング**という。

量子化



標本化した時の個々の値はもとはアナログデータから取ってきた値なので無限小数で表される。これを一定の尺度の1つに近似することを量子化という。
音楽CDでは16ビットで量子化している。つまり、**音の振幅を65536（=2^16）段階で表している**ということである。ただ**この1段階が人間には認知できないようなものならば問題ない**とされる。
フルカラー画面では赤（R）、青（B）、緑（G）の成分が8ビットで表されている（256段階で表されている）。つまりその3つを合わせた色は256*256*256=16,777,216段階で表される。これの1段階も人間には認知できない程度なので問題は無い。

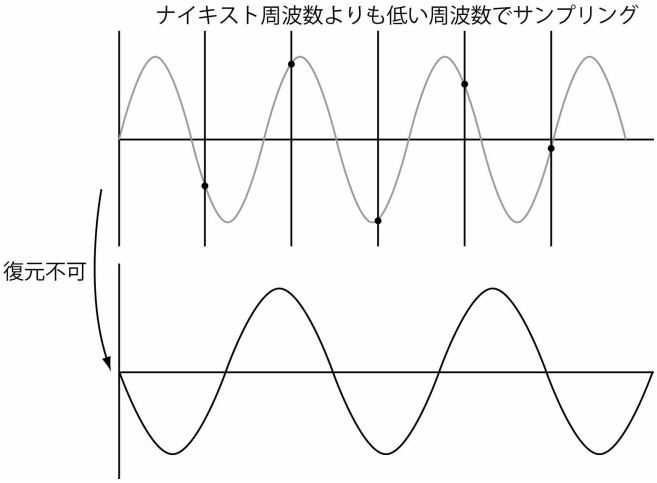


情報と確率

教科書対応範囲 3.1.1 3.1.2 情報の伝達と情報量（情報の伝達、情報の大きさ）
ただの計算問題ですが、「情報量とは何か」が分からないと計算のしようがないのでそこの定義を覚えましょう。ただの確率の問題です。大問3の選択問題として出される可能性が高いです。確実に得点できるように。

定義：情報量

ある情報が与えられたとき、ある物の $\log_2$（事前の可能性/事後の可能性）をそのものの情報量として、単位はbitとする。



情報量（場合の数の場合）

よくわからないので例を見ましょう。
犯人の候補が、ヤス、山川、沢木、平田の4人が候補の時に「犯人はヤス」という情報があると犯人の候補はヤス1人になるのでこの時「犯人はヤス」という情報の情報量は $\log_2(4/1)=2$ となります。
また「犯人はヤスではない」という情報は、これにより犯人の候補が3人になるので情報量は $\log_2(4/3)=0.415$…となります。
また犯人の候補がヤス、山川の2人の時に「犯人はヤス」という情報の情報量は $\log_2(2/1)=1$ となります。
また4人が候補の時にに「東京スカイツリーの高さは634Mである」という情報が入ってきても犯人を絞ることはできないので、その情報量は $\log_2(4/4)=0$ となります。
このように同じ文章でも背景によって情報量は変わりますし、「AはBである」とその否定の「AはBではない」の情報量は大抵の場合は違います。

情報の加法性

「犯人は山川ではない」という情報の情報量は上記の通り $\log_2(4/3)=0.415$…で、その後つまり犯人の候補がヤス、沢木、平田の3人になった時の「残りの3人のうち犯人はヤス」という情報の情報量は $\log_2(3/1)=1.584$……となります。
このとき「犯人は山川ではない」、「残りの3人のうち犯人はヤス」の2つの情報量を加えると。
$0.415…+1.584 = \log_2(3/1)+\log_2(4/3)=\log\_2(4/1)=2$
となり犯人の候補が4人の時の「犯人はヤス」という情報と同じ情報量になります。この法則を**情報の加法性**と言います。

情報量と確率

実は上記の議論は前提として4人全員がそれぞれ犯人である確率が等しく1/4でないといけません。では特定の人が特に疑われていた時の情報量は確率を使って以下のように表します。
ある事象が起きる確率をPとするとPが起きるという情報の情報量は$\log_2(1/P)$

ヤス:1/10、山川3/10、沢木3/10、平田3/10の確率で疑われていたとする。「犯人はヤス」の情報量は$\log_2(1/(1/10))=\log_2(10)=3.321…$、「犯人は山川」の情報量は$\log_2(1/(3/10))=\log_2(10/3)=1.736…$となる。
以外なものつまり確率が低いものが起こるという情報の方が情報量は大きい。

平均情報量と平均符号長

何をやっているかわからないだろうが、とりあえず覚えてください。
「ABCD」という文章があった時これをコンピューターで扱うには文字と数を対応させなければなりません(コンピューターは数字しか扱えない)。以下この章で出てくる数字は全て2進数です。単純にAと00、Bと01、Cと10、Dと11を対応させます(この作業を**符号化**と言います)。この時「ABCD」は「00011011」となります。この文章中ではAが出てくる確率は1/4で、その情報量は$\log2(1/(1/4))=2$です。B、C、Dも同様に情報量は2です。ここで情報量の平均を**平均情報量**という。これは

$$\sum(\text{事象の起こる確率})\times(\text{事象の情報量})$$

よってで表せます。今回の平均情報量は1/4*2+1/4*2+1/4*2+1/4*2=2となります。また符号長の長さの平均を**平均符号長**という、これはは　情報の大きさ(bit)÷メッセージの数　であらわせる。
今回の情報は「00011011」なので8bit、メッセージはABCDの4文字なので4で、平均符号長は8/4=2になる。

では次に「ABCDAAAB」という文章を送ることを考える。情報は「0001101100000110」となる。

文字	出現頻度	情報量
A	4月8日	1
B	2月8日	2
C	1月8日	3
D	1月8日	3

よって平均情報量は3/8*1+2/8*2+1/8*3+1/8*3=1.75
平均符号長は変わらず2です。平均情報量<平均符号長になってしまう。**平均情報量よりも平均符号長のほうが大きくなるのは符号化の仕方が非効率的だということをあらわす。つまりもっと良い符号化の仕方によって符号長を短くできる(圧縮できる)ということをあらわす。**

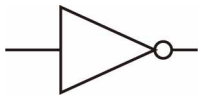
論理回路

情報教科書対応箇所 7.2(P157～164)
以下の3つの演算（+1）からなる計算体型を感じれたら勝ちです。私たちは基本的に足す・引く・掛ける・割るの四則演算を基本にしていますが、コンピューターではNOT、AND、OR（XOR）の演算を基本としてそれから上記の四則演算やコンピューターの中の制御をしています。がそんなことを考えずに以下をさっさと覚えて演習をしましょう。

以下A,B,C…の文字は0か1のどちらかを表します。ちなみに1を「真」、0を「偽」の状態といいます。以下の4つを覚えましょう。ブール代数のところの「+」「・」は普通の計算の記号とは意味が異なるので注意。真理値表とは九九の表のようなものです。例えば「ANDは入力Aが1、入力Bが0の時だけ、出力Yは0になる」ということを言っています。ちなみに真理値表の横にある図形はMIL式といい、この演算を表すブロックのことを「ゲート」と呼びます。

NOT（反転）
A の 0、1 をひっくり返したものを Y に代入
真理値表

A	Y
0	1
1	0

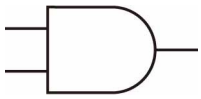


ブール代数　$Y = \overline{A}$

AND（論理積）

A、B 両方が 1 なら Y に 1 を代入
真理値表

A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

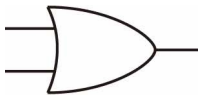


ブール代数　$Y = A \cdot B$

OR（論理和）

A、B の内一つでも 1 があれば Y に 1 を代入
真理値表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

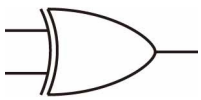


ブール代数　$Y = A + B$

XOR（排他的論理的）

A、B の内に 1 であるものが奇数個なら Y に 1 を代入
真理値表

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0



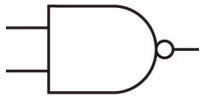
ブール代数　$Y = A \ xor \ B$

以上が大事、これとは別に NAND、NOR、EQ の 3 つの演算がありますが、一部は名前から推測されるように、これらは全て AND、OR、XOR の出力の 0、1 をひっくり返したものになります（NAND=Not AND、NOR=Not OR）。

NAND

A、B の内一つでも 0 があれば Y に 1 を代入
真理値表

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

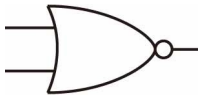


ブール代数　$Y = \overline{A \cdot B}$

NOR

A、B 両方が 0 なら Y に 1 を代入
真理値表

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

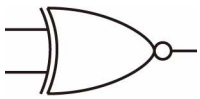


ブール代数　$Y = \overline{A \square B}$

EQ

A、B の内に 1 であるものが偶数個なら Y に 1 を代入
真理値表

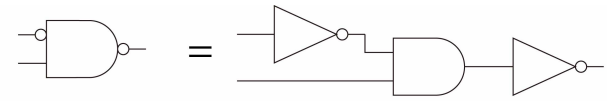
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	1



ブール代数　$Y = \overline{A \ xor \ B}$

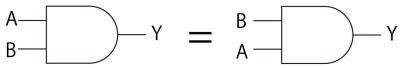
注）真理値表の A、B は入力、Y は出力を表す。

注意点

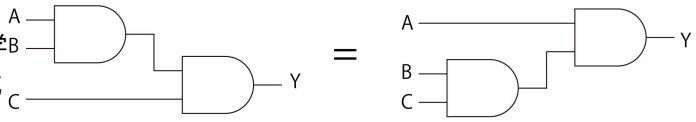


右の図を上から順に

・回路図上の○はNOTを表します。



・A+B=B+A、A・B=B・A、のように演算子の左右を入れ替えても結果は同じです。これは上記のNOT以外の6つの演算全てで成り立ちます（交換則）。

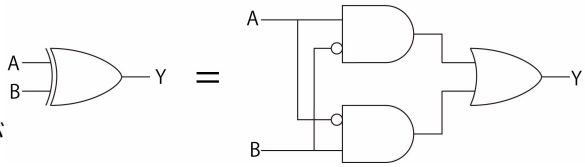


・(A+B)+C=A+(B+C)、(A・B)・C=A・(B・C)と通常の数学と同じように処理できます。これはAND、OR、XORで成り立ちます（結合則）。

・ORの2本の入力、1本の出力の全てにNOTをつけるとANDになります。同様にANDの入出力にNOTをつけるとORになります（ド・モルガン律）。

・XORはAND、OR、NOTから作れます。

以下右の図にはありませんが。

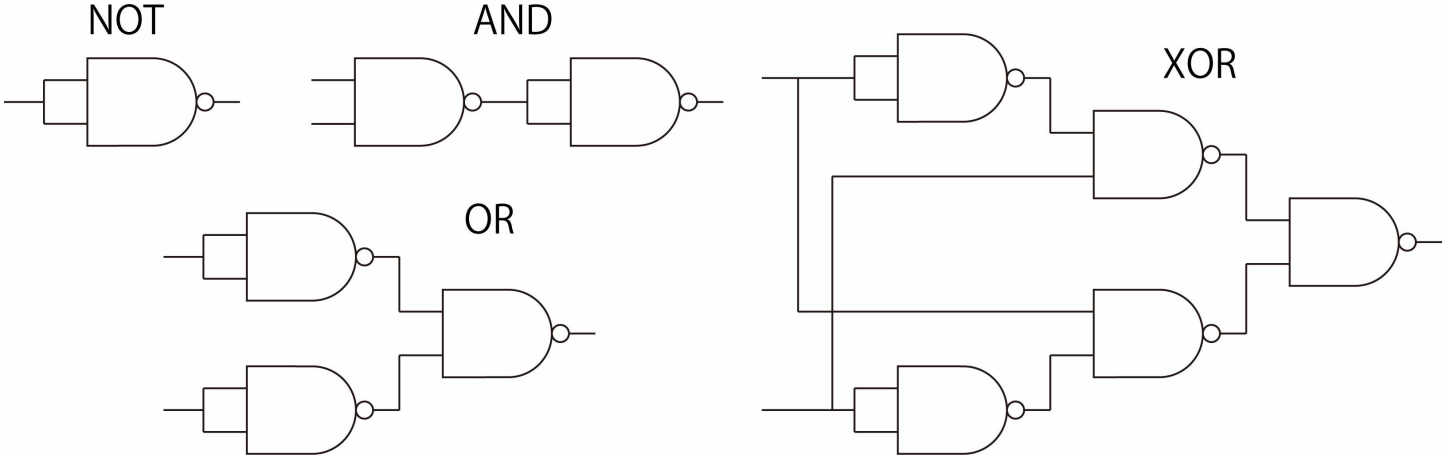


・~A or ~Bと~(AorB)は違うものです。

・否定の否定は無くなります（2重否定）。~(~A)=A

・A+(B・C)=(A・B)+(A・C)、A・(B+C)=(A・B)+(A・C)が成り立ちます（分配法則）。

実はNANDだけまたはNORだけで他の7つの演算ができます。以下にNANDでの表し方を表記します（実用上NANDの方をよく使うので、NORで書けという設問を出したらその人の品性を疑われるぐらいなのでNORでの書き方は覚える必要がない、あってもド・モルガン律でいける）。



コメント：論理回路はハマるまでが大変なので、演習を繰り返して真理値表を書いて覚えるのがいいのでは。

7.2.1の加算標準形・乗算標準形は出ないと思う。「試験で～～を加算標準形で書け」ともし出たらそのようなことを気にせずに適当に書けば1/2の確率で当たります。

どうしてもわからないなら飛ばしましょう。出る確率は低いです。

7.2.4組み合わせ回路の例はちょっとむずかしいので出ないはず。

ユーザーインターフェイス

教科書対応範囲：9.3.2 技術的側面からみたインタフェースデザイン
おそらく 3A で出されると思うので勉強しなくても構わないと思いますが一応、CUI と GUI の違いがわかればそれ以外のことを覚える必要はんく常識で解けます。用語を少々覚えましょう。

GUI と **CUI** はコンピューターの画面上の表示の仕方の違いと理解して構いません。GUI（Graphical User Interface）は普通の表示でウィンドウやアイコンなどのグラフィカルな要素を用いて表示してマウスを使って操作する一方
CUI（character user interface）は **Mac のコマンド**や **Windows のコマンドプロンプト**などのシステムで基本的に文字のみで表示して、キーボードを用いて操作する。詳しくは以下の表の通り。

GUI	CUI
絵や画像を用いて視覚的に情報を表示	キーボードから命令を文字で入力、文字列で結果を出力
直感的でわかりやすい	慣れれば作業は迅速

GUI では主にウィンドウ、アイコン、メニュー、マウスなどのポインティングデバイスが使われて、この 4 つの頭文字を取って **WIMP システム**とも呼ばれる。

著作権・情報と社会

教科書対応範囲：第 10 章 情報技術と社会
ほとんどは常識で解けるものですし、基本的に 3A でしか出しません。気になる人は教科書を読んでください。

あとがき

結構頑張って作ったんだけどな。常識的な量に抑えれて満足。とりあえず過去問をときましょう。
2012_s1_28 eri_A