

易しくないパソコンリテラシー(4/4)

演習編

4. プログラミングの演習

4.1 はじめに

4.1.1 表題に「易しくない」を付けた理由

筆者(島田)は、2000年代、或る文科系大学の情報学部で、コンピュータプログラミングの授業と演習の講師を依頼されました。英語では、通称、**Computer Literacy**と言います。英語の Literacy(リテラシー)は、読み・書きの教養があることを言いますので、俗に言えば、「読み・書き・ソロバン」に代える「読み・書き・パソコン」と当てることができます。学科名に採用するには、内容を具体的に構成し難いところがあります。これに、「コンピュータプログラミング」と当てた、と理解して下さい。授業内容の組み立てには多くの課題があります。これを言外に含ませて、「易しくない」を表題の頭に付けました。授業と演習の経験を文書化したものが、この文書の主題です。内容の組み立ては、パソコン端末を備えた特別の教室で、実習してもらうことを主目的にしました。講師側からの一方的な講義方式を最小限に抑え、個に異なる作業内容について、対話を持つ方法を探りました。学生の知識の幅は、何も知らない全くの初心者から、ほどほどのパソコン経験のあるレベルまでの広がりがありました。女子学生・海外からの留学生もいました。このすべての人に、納得してもらえる、実(じつ)のある内容を、どのように組み立てればよいかの模索が必要でした。女子商業高校では、秘書教育を目的として、英文タイプライタの使い方を実習する科目を持つ例がありました。書店に行けば、多種類のパソコンの参考書が売られています。しかし、パソコン本体があって、実際に操作する環境が必須です。「畳の上の水練」という諺があります。理論や方法を知っていても、実場面での練習経験をしなければ身に付かないことを言います。その環境の構成は、対話ができる学校が最も効率の良い場所です。

4.1.2 お料理のレシピの文書化を例題とした

一般教養としてのプログラミング教育の課題に、何を採り上げるかにも悩みが起きます。プログラミングは、歴史的には、高速で数値計算をさせることを目的としました。数学の応用は、専門的に多方面に亘ります。情報処理は、現在では理工系だけでなく、文科系の分野でも広く応用されるような学際的な分野になりました。そこで、疑似的なプログラミングのモデルとして、お料理のレシピを使うことを思いつきました。レシピは、お料理の手順を書いた、正にプログラム文書だからです。日本語だけでなく、英文レシピも大型書店で売られています。したがって、日本語と英語との表現方法の違いを説明できます。レシピは、お料理を作る指示書ですので、具体的に矛盾なく作成しなければなりません。これがプログラミングの本質を備えています。ハードウェアとして鍋・釜・食器などがあり、キッチン設備があります。データは材料です。必ず数量的(デジタル的)に定義しなければなりません。あらかじめ調理済みの食材、例えばゆで卵などは、ユーティリティプログラムで作ります。並列処理もあります。お料理は人数に合わせたバッチ処理です。並列処理もあります。高級なプログラミングを勉強する前に、肩の凝らない例題で初心者プログラミングの実質を理解してもらうために、レシピを HTML 文書に作成して、講師と学生間との通信を大学内の LAN(local area network)環境で構成することにしました。学生間で相互に閲覧ができます。

4.1.3 パソコンを擬人化して対話すると考える

パソコンが利用されている環境は、パソコンを疑似的に人とみなし、公用語として、人の方は英語を、パソコン側では、電気信号を符号化したバイトを使うと考え、相互に翻訳することで対話が成り立つとします。人の方は、国ごとに固有の母語(母国語)がありますが、方言違い、と解釈することにします。パソコン側の言語にも母語を考えます。これは、CPU が機能する符号コードですので、製品ごとに異なる機械語です。これも、方言違いです。公用文字は、アスキーコード(ASCII)とします。母語と公用語間に翻訳が必要です。人とパソコン間では公用語間の翻訳があって、初めてパソコンの利用が成り立ちます。これがユーザインタフェースです。「パソコンにデータを入力する・パソコン側からのデータ出力を得る」二方向の情報交流のセットを言います。この実現にハードウェアが使われます。人とメインフレームとの交流の道具(ツール)は、標準入出力装置と言いました。これをコンソールの愛称で呼び、全体は巨大設備でしたが、現在ではノート型パソコンに納まってしまいました。驚異的な縮小です。

4.1.4 作業環境の構成

一般論から説明を始めます。個人が自宅でパソコンを利用したいとき、全くのゼロからシステムの構成をする時の手順については、第2.3.2項で説明しました。インターネットに接続する使い方が普通になりました。しかし、学校教育の場では、学生が外部のネットワーク環境に自由なアクセスができないように、閉鎖的な LAN(local area network)の環境を構築して、厳重なシステム管理をする必要があります。これは、一般企業においても、業務の安全管理上、重要な課題になってきました。私的な環境では、マイクロソフト社の Windows を搭載したパソコンの利用が主流ですが、学術関係や企業では、別の OS、例えば UNIX なども採用されることがあります。したがって、共同利用の環境に参加する人には、システムの利用について、丁寧なガイダンスの文書を作成しておくと共に、対話の機会が持てる必要があります。この人の知識レベルについては、パソコン利用の経験の有る・無しとは関係しません。教育環境では、毎年新入生に、この種のガイダンスをすることを年次行事に組み込みます。システム側の担当者は、毎年、同じような作業の繰り返しになることから、マンネリ化して、丁寧さが無くなることがあります。一般企業でも、ガイダンスは、本来、管理者側が真剣に対応すべきことなのですが、重要であると認識していないことが多くみられます。そうであると、新しく参入した人は、見様見真似で先輩の作業を覚えることとなります。これは、間接的ですが、いじめ(虐め)になります。新人の定着率が低下する原因になります。多くの人が顔を合わせる場所に参加する、つまり出社し、大部屋で作業をすることは、実は非常に良い環境を育てます。とりわけ、技術の継承では必須と言えます。インターネットを介して、自宅での作業形態を採る孤独化も増えています。これらの環境の変化は、精神的な疾患が社会問題化する原因にもなってきました。

プログラミングの教育についての話に戻ります。OS 環境の違いは、テキストエディタのような基本的なアプリケーションの使い方も、一応の説明が必要になります。この第4章では、マイクロソフト社の提供する[メモ帳(notepad)]を利用することから解説しています。使い方に慣れた頃から、より複雑なワードプロセッサの使い方に進ませます。モニタ上のスタイルの例を下に例示します。図4A は、デスクトップ上に複数のアプリケーションのフォーム(ウインドウ)が表示されています。左上は、マイクロソフト社のワープロ (Word 97-2003) です。識別子は、(*.doc)です。改定版では(*.docx)です。図 4B は、そのメニュー部分を示した図です。メニュー項目が多いため説明用です。文書表示用アプリケーションは、多種類あります。インターネットの利用で多く使われるものは、(*.pdf)と(*.html)です。Word で作業している文書を、「名前を付けて保存」の機能を利用して、別構造の文書ファイルに作成することができます。

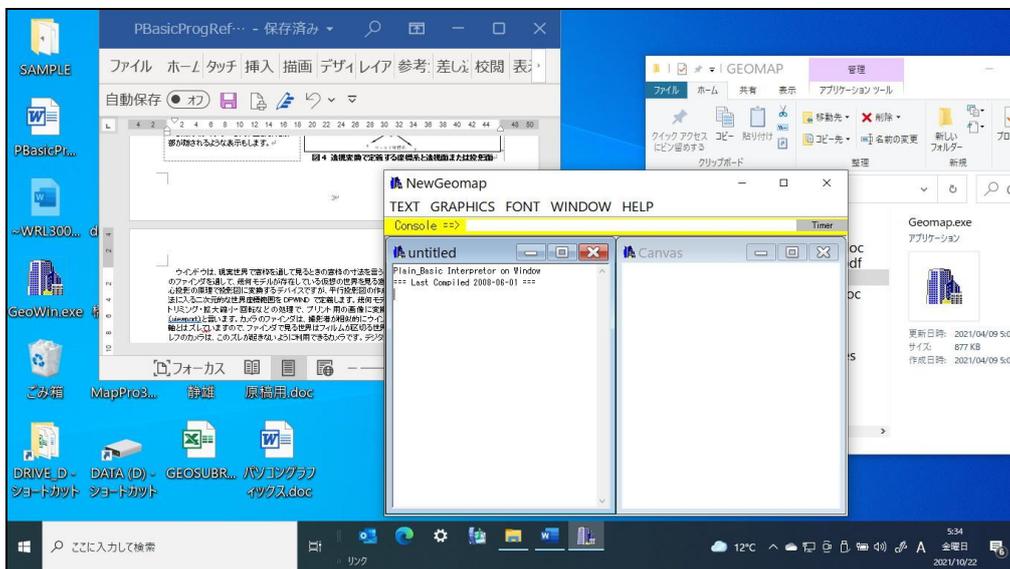


図 4A 複数のアプリケーションの疑似的な並列作業の画面



図 4B メニュー項目が多くなって使い方が難しくなったことの説明用です

4.2 タイピングの技能習得から

4.2.1 機械式の英文タイプライタ

活字を使う活版印刷機は、1439年頃、グーテンベルクによって発明されました。この機械を簡易化した製品が英文タイプライタであって、1907年にレミントン社が発表した形式がほぼ標準になりました。活字の寸法を一種類に抑え、12ポイントの等幅フォント(パイカ)で、1ページ単位で用紙に印刷します。今では実物を教育現場で見ることがありません。パソコンのキーボードの元点ですので、機械式タイプライタの構造について説明しておきましょう。図4.1は、インターネットを検索して探図しました。活字の付いたバーが放射状に並び、文字キーを叩くと、**インクリボン**を介して**プラテン**に挟んだ用紙に印字されます。各部の名称のあらましを図4.2から図4.4までに示します。



図 4.1 標準的な機械式
英文タイプライタ

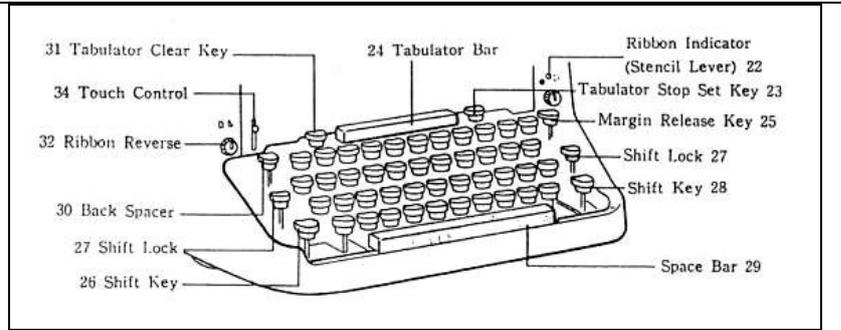


図 4.2 古典的なキー配列

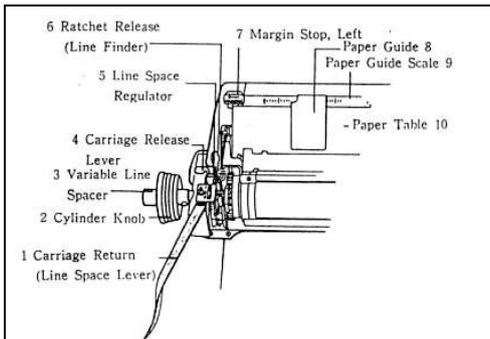


図 4.3 プラテンの左部分
復帰・改行(CR・LF)用レバーは左手用

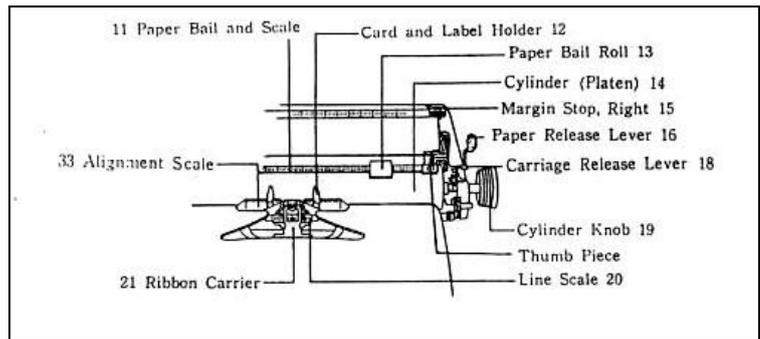


図 4.4 プラテンの右部分
CR・LF レバーが右にある機種が標準です



図 4.5 テンキーの無い日本語用キーボード(インターネットから)

4.2.2 文字キーの呼び方

- (1) キーボードのキーには文字や記号が表示されています。キーの種類を口頭で伝えるためには、文字や記号の正しい呼び方が必要です。
- (2) 電動の英文タイプライタをコンピュータに繋いで印刷をさせるとき、文字以外の制御コードが必要です。標準として使われるものは、改行(CR)、行送り(LF)、空白(SP)、タブ(HT)、ベル(BEL)の5個です。文字データだけを記録するテキストファイルは、英字・数字・記号のコードと、5個の制御コードを使います。
- (3) 電話線を介して通信を行うとき、以前は6ビットのコード系を使っていました。この場合、英字の大文字と小文字にコードの区別ができませんので、この打ち分けタイプライタの shift 状態を切り換える SI と SO のコードを使いました。8ビットのコード系を使うようになって、英字の大文字と小文字とを別コードで表せるようになりましたので、この shift 変更コードを使う必要がなくなりました。
- (4) 電報などの送受信に使うテレタイプライタでは、文字データを送る前後に種々の信号の遣り取りをします。その目的に使うためのキーがコンピュータ用のキーボードに引き継がれています。代表的なものが、ESC, DEL です。この他のコードは別文字やラベルを当ててありますので、下に示す文字は直接にはキーボードに表示されていません。
- (5) 電動の英文タイプライタをコンピュータに繋いで印刷をさせるとき、文字以外の制御コードが必要です。標準として使われるものは、改行(CR)、行送り(LF)、空白(SP)、タブ(HT)、ベル(BEL)の5個です。文字データだけを記録するテキストファイルは、英字・数字・記号のコードと、5個の制御コードを使います。
- (6) 電話線を介して通信を行うとき、以前は6ビットのコード系を使っていました。この場合、英字の大文字と小文字にコードの区別ができませんので、この打ち分けタイプライタの shift 状態を切り換える SI と SO のコードを使いました。8ビットのコード系を使うようになって、英字の大文字と小文字とを別コードで表せるようになりましたので、この shift 変更コードを使う必要がなくなりました。
- (7) 電報などの送受信に使うテレタイプライタでは、文字データを送る前後に種々の信号の遣り取りをします。その目的に使うためのキーがコンピュータ用のキーボードに引き継がれています。代表的なものが、ESC, DEL です。この他のコードは別文字やラベルを当ててありますので、下に示す文字は直接にはキーボードに表示されていません。

表 4.1 ラテン文字・記号の呼び方

文字・記号	英語の名称	日本語での名称	解 説
!	exclamation mark	感嘆符	俗称で「びっくりマーク」
“	quotation mark	引用符	活字では左右で違う字
#	number sign, sharp	番号記号、シャープ記号	
\$	dollar sign	ドル記号	
%	percent sign	パーセント記号	
&	ampersand	アンパサンド	俗称でアンドマーク
‘	apostrophe	アポストロフ	single quote とも言う
(left parenthesis	左小かっこ	左丸括弧
)	right parenthesis	右小かっこ	右丸括弧
*	asterisk	アステリスク	星印
+	plus sign	正符号	プラス記号
,	comma	コンマ	日本語の読点(、)相当
-	hyphen, minus	ハイフン、マイナス	
.	full stop, period	ピリオド	日本語の句点(.)相当
/	solidus, slush, diagonal	斜線	スラッシュ
0 - 9	digit	数字	アラビア数字
:	colon	コロン	
;	semi colon	セミコロン	
<	less than sign	不等号(より小)	左側が右より小さい
=	equals sign	等号	イコール
>	greater than sign	不等号(より大)	左側が右より大きい
?	question mark	疑問符	俗称で「はてな記号」
@	commercial at	単価記号	アット記号
A - Z	Latin capital letter, UC	英大文字、上段文字	UC=upper case
[left square bracket	左大かっこ	左角括弧
¥	yen sign, back slash(\)	円記号	(備考参照)
]	right square bracket	右大かっこ	右角括弧
^	circumflex accent	アクセント シルコンフレックス	フランス語で使う
`	grave accent	アクセント グラフ	フランス語で使う
~	tilde	オーバライン	
_	low line, underscore	アンダーライン	下線、アンダースコア
a - z	Latin small letter, LC	英小文字、下段文字	LC=lower case
{	left curly bracket opening brace	左中かっこ	左波括弧
	vertical line, bar	縦線	縦棒
}	right curly bracket closing brace	右中かっこ	右波括弧

備考: 記号または文字の表記方法や呼び方は、参考にする出典によって異なります。ここでの定義は、コンピュータに関連した場合です。印刷関係では、記号文字の呼び方が違うことがあります。ASCII コードの back slash(\)は、日本語の字形では¥記号がでます。

表 4.2 通信用の制御コードの呼び方

文字	英語での名称	16進数コード	解 説
NULL	null	00	空白。スペースと紛らわしい用語です
SOH	start of heading	01	ヘッディング開始
STX	start of text	02	テキスト開始
ETX	end of text	03	テキスト終結
EOT	end of transmission	04	伝送終了
ENQ	enquiry	05	問合せ
ACK	acknowledge	06	肯定応答。了解の意義に良く使う
BEL	bell	07	ベルを鳴らすコード
BS	backspace	08	後退。前の文字を消すときに使う
HT	horizontal tab	09	水平タブ。タブキー
LF	line feed	0A	改行。CR と一緒になっている
VT	vertical tabulation	0B	垂直タブ
FF	form feed	0C	書式送り。プリンタの改ページコード
CR	carriage return	0D	復帰。テキストの改行コードに使う。
SO	shift out	0E	シフトアウト。下段文字への切り換え
SI	shift in	0F	シフトイン。上段文字への切り換え
DLE	data link escape	10	伝送制御拡張
DC1~DC4	device control	11 - 14	装置制御 1~4
NAK	negative acknowledge	15	否定応答。了解できないと知らせる
SYN	synchronous idle	16	同期信号
ETB	end of transmission block	17	伝送ブロック終結
CAN	cancel	18	取消
EM	end of medium	19	媒体終端
SUB	substitute character	1A	置換キャラクタ
ESC	escape	1B	拡張。取り消しの目的によく使う
FS	file separator	1C	ファイル分離キャラクタ
GS	group separator	1D	グループ分離キャラクタ
RS	record separator	1E	レコード分離キャラクタ
US	unit separator	1F	ユニット分離キャラクタ
SP	space	20	間隔。スペースバーで打つコード
DEL	delete	7F	抹消。後ろの文字を消すときなど

表 4.3 キーボードショートカット(Windows 用)

実行する機能		キー操作
システム	スタートメニューを開く	Ctrl+Esc、旗マークキー
	システムの強制終了	Ctrl+Alt+Delete
ウインドウ	アクティブウインドウを閉じる	Alt+F4
	アクティブウインドウのコントロールメニューを開く	Alt+スペース
	アプリケーションの切り替え	Alt+Tab
表項目選択	アイコンの移動	カーソルキー(→←↑↓)で
	次の項目を選択	Tab
	複数の項目を選択	Shift+カーソルキー
編集	すべてを選択	Ctrl+A
	切り取り	Ctrl+X、Shift+Delete
	コピー	Ctrl+C、Ctrl+Insert
	貼り付け	Ctrl+V、Shift+Insert
	元に戻す(Undo)	Ctrl+Z、Alt+Backspace
	モニタ画面をクリップボードにコピー	Fn+Insert
	カーソル位置より前の文字削除	Backspace
	カーソル位置より後の文字削除	Delete
	HELP の呼び出し	F1
	OK	Enter
	キャンセル	Esc

備考:メニューバーのメニュー項目に英字が記してあるものは、Alt+英字で選択できます

4.3 タイピング練習のヒント

4.3.1 指使いの練習から始める

- (1) タイピングの基本は、機械式のタイプライタの使い方が原点になっています。最も標準的なタイプライタのキー配列は図 4.4 のようです。文字キーはレバー構造になっています。前後 4 段に並べるとき、レバーが重ならないようにするため、キー並びが斜めになります。どのキーにどの英字を割り当てるかの約束の一つを QWERTY 配列と言います。
- (2) まず、指をホームポジションに置いて、正しくまた楽な姿勢を取るようにします。ホームポジションとは、**ASDF**、**JKL**のキー位置を言います。親指を除く左右4本の指を、このキーの上に軽く載せた状態が基本姿勢です。親指はスペースバーを打つ位置に載せます。普通、**F**と**J**のキー(または**G**と**H**)表面には小さな突起が付いていますので、人さし指でそれを確かめることでブラインド・タッチ・タイピングを助けます。

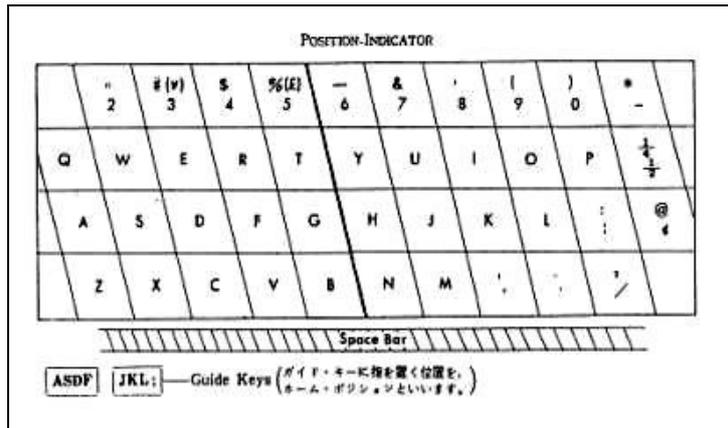


図 4.6 指使いの範囲は縦の配列単位です。ただし、人さし指は2列を打ちます。

- (3) 指使いは、打ちたい文字を打つ指だけを縦に移動させて打ち、すぐにホームポジションに戻すことが基本です。人さし指は、2列分の守備範囲を持ちます。打つ操作は、一字ごとに押すのではなく、リズムカルに叩くように打ちます。コンピュータのキーボードでは、ある程度長い時間押しているとリポート機能が働いて、同じ文字を連続して入力させることができます。人さし指に比べて、他の指を独立に動かし難いので、タイピングの時以外にも動かす練習をすることを薦めます。
- (4) 大文字を打つときは、打ちたい文字キーに使う指と左右反対側の、空いた方の小指で shift キーを押しながら打ちます。キーボードショートカットを使うときは、ctrl キーを同じように小指で押しながら文字キーを打ちます。最初の中は、練習は、英小文字だけを使ってブラインド・タッチの練習に集中するのが良いでしょう。
- (5) 指使いの練習は、2文字の語(in,on,up to,be,do,...)、3文字の語(the,for,ing,...)を何回も繰り返します。長い単語も、このような短い語を繋ぐからです。練習は、下の例のように、同じ文字を3行続けて間違いないく打てることを単位とします。練習時間は、一回一時間程度にして休憩を取ります。毎日1~2回が適当です。

```
ing ing
ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing
ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing ing
```

- (6) 指使いは、人さし指から始めて、小指が使いこなせるように練習を進めます。この練習に沿うように選んだ文字を下に並べました。上の(5)のように練習します。

```
ff jj fj dd kk dk ss ll sl aa ;;
a; as ad ass add sad dad; fad jad lass fall salad;
hh hj ha has hd salad ash hash dash hall shall
```

gg gf gas gd gad gag gaff gall
gash glass gags; lad has; all galls half;

ee ed el eld eg egg elf edge desk ease eagle
ii ik ke kid il ill in his idle idea like siege
rr rf ra rag rd red rid ride rise reel eider
uu uj ju jug us use ugh used urge lush sugar
dag keg due his her sir far jug hug laga rush

fell; slide slid; hear heard; shear sheared
tt tf tg tag te tea at hat task take late tease
yy yi yh hay yt yet yes year yard stay yield
ww ws wa was wy way why walk well what wheel
oo ol od old oil on off work good flow order

qq qa qu qua quo quit queer quest quarter
pp p; pa pad pe peg put poor puff pulp pearl
the key set for you who jug too
feel good your desk with type just quip law
sharp skill great quite pearl judge forty weight

vv fg vi via vs vis vie very view have value
mm mj ma map me met may mere mild most moral
cc cd ca cad cu cue cat cede cock dock color
nn nj no not ne new net in ing nice name noon nerve
have from time need come cave each give can

which thank cover heavy money human month minute
makeup course vacant custom jangle scrawl quaver
bb bf bh bag bu but bad back bear club thumb
xx xs sx six xa tax fox next text taxi excel
zz za ze zed zo zoo zip zone zeal size zebra

, , , k k, one, two three, four, five, six, seven
.. .l f.o.b., c.o.d., c.i.f., a.m., p.m., ex.
buy bob beg tab rub wax nob mix zee vex quiz
did and for key due row end man men big she ham
pay eye cut own fix rub aid fur sit fir jam sir box qua pen

hand kept half rush risk then work they make them
sigh form sign dock with firm paid name city both busy when wish down
world right forms title eight spent usual civic visit spend
was are get ear few egg eat act bar see red fat saw
ill oil pop joy ink lip hop mop pin kin ohm nun him hum kin date lump

data jump gate only were save read fast case test
 card ever race ease area dear upon fact best rate free gage
 draft union aware imply react union cases nylon acted pupil
 state plump stage onion deeds brass grade wares cases jolly
 reward effect starts estate faster defect wasted states regret

- (7) タイピング速度を計算するときは、1分間のワード数で言います。ワード数は、平均して5ストロークとして数えます。ストロークとは、文字・区切り記号・スペースを含みます。タイピングの能力は、速度が早く打てることと、ミスタイピング件数の少ないことで測ります。商工会議所の検定は、10 分間の作業で下のよう分類しています。

級	速度 (word/minute)	速度 (stroke/minute)	最大誤件数	同誤まり率
A	50	250	11	4.4x10 ⁻³
B	40	240	11	5.5x10 ⁻³
C	30	180	16	11.6x10 ⁻³
D	25	150	16	12.7x10 ⁻³

- (8) 機械式のタイプライタのキーは、叩くように打ちます。打つ強さが一定になるように練習して、印刷文字に濃淡のムラが出ないような技能が必要です。ただし、コンマやピリオドは、僅かに弱めに打たないと、用紙に穴を開けてしまいます。電動機械式のタイプライタは、濃淡の調整ができる設計でした。
- (9) パソコン用のキーボードは、電気的な接点を持つキーに変わります。指先でキーを打つときに感じる感触は、機械式のタイプライタよりも柔らかくなりますが、確実なストローク感が得られるように、人間工学的に研究されて製品化されています。
- (10) テンキー部分のない、横幅の狭いキーボードの方が普通です。テンキーの利用もできるように、英数字キーボードの(7 8 9 U I O J K L, M)をテンキーとして切り替えて利用するように設計された製品もあります。ここで、テンキーの数字並びですが、プッシュボタン式の電話機の数字並びが左上を1として、右下に(1 2 3),(4 5 6),(7 8 9)の順に並ぶのとは上下が逆になっています。テレビのチャンネル切り替え用は、(10 11 12)が追加されています。

4.4 文書作成のタイピング練習

4.4.1 実文書のタイピング

ここでの演習に使う文章原稿は、四種類です。新聞・雑誌の記事から選んだ文章と、擬似的プログラム文としてのレシピと、それぞれ英文と日本語文です。この演習は、プログラミング講義と平行して学生に課題として出しますので、年度によって見本原稿が変わります。教育目的がありますので、原稿の選定は下に示すような点に注意しました。

- ・ まず、5W1H を満たしていることです。新聞記事では Who を満たす署名記事を選びますので、かなり注意して探さないと良い見本原稿が見つかりません。また、表題・抄録・段落などは、演習のときに再構成を指導します。
- ・ 英文原稿は段組みの幅が狭いので、ハイフネーションが多く使われています。ハイフンを省いて単語を繋いだ原稿にタイプし直すことを指導します。日本語の文章の禁則処理は、テキストエディタで原稿を作成するときに解説しています。
- ・ レシピは擬似的なプログラム文として解説しています。マニュアル文章は情緒的に書いても、プログラム文は事務的に構成することを指導します。

英文について、原稿処理の演習見本は、この報文の後半に例示しました。

4.4.2 テキストエディタの使い方から

文書作成の最初は草稿の準備です。草稿と言う言葉は、編集作業の前段階の下書きを意味します。コンピュータを使って体裁のよい文書を作る場合、草稿の作成に相当する作業は、テキストファイルを作成することです。Windows で標準に利用されているのは「メモ帳」またはノートパッド(実行形式では Notepad.exe)は、システムの標準ソフトとして含まれています。デスクトップの画面で、「スタート→プログラム→アクセサリ」の順にたどって開くことができます(図1)。メモ帳で扱うテキストファイルの標準拡張子は(.txt)です。原稿作成時の注意として、改行は段落の最後で使うようにします。これは、編集メニューの「右端で折り返す」にチェックを入れて作業することです。チェックを外すと原稿がどうなるかを試してみます。

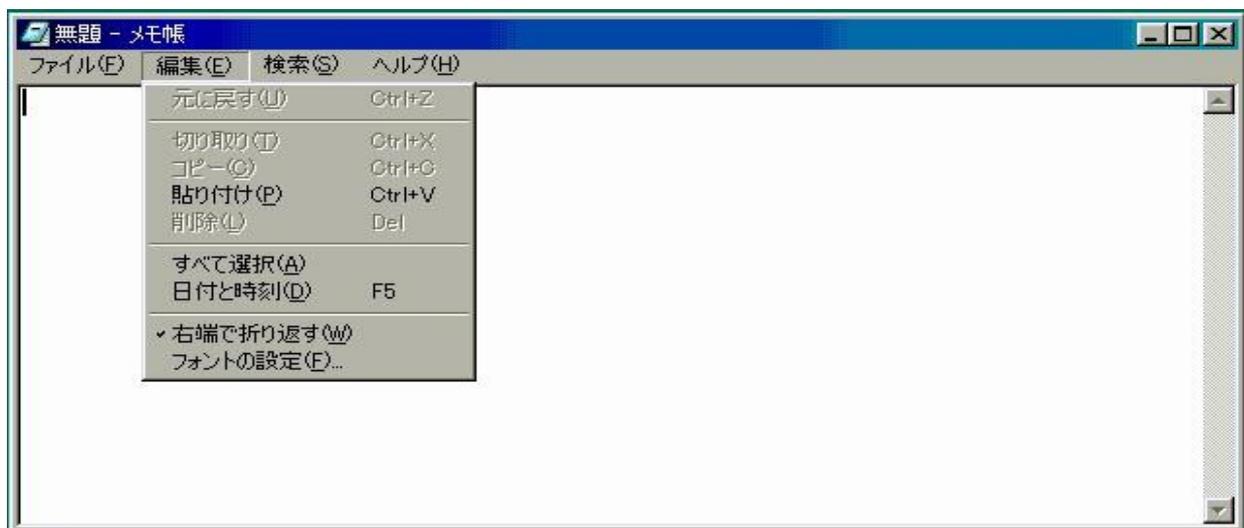


図 4.7 プログラム「メモ帳(NotePad)」の作業画面

4.4.3 ワードパッドプログラム

前項のメモ帳は、非常に単純な編集機能を備えたプログラムです。8ビットのミニプロセッサを使っていた簡易なワープロは、草稿の大きさとして 32KB 以内の制限がありましたので、ワードパッド(実行形式は Wordpad.exe)は、この文字数制限がありませんし、メモ帳よりも機能が拡張されていました。現在は MS_WORD が標準となりましたが、歴史的な経緯がありますので、作業画面を紹介しておきます。「書式」メニューが増えたことに注目して下さい。なお、ツールバーは、作業を便利にするため、メニュー項目の一部の機能をアイコンの図柄で「表示」したものです。ワードパッドでは表示の切り換えができますが、ノートパッドには、最初からツールバーはありません。



図 4.8 プログラム「ワードパッド」の作業画面

4.4.4 ワードプロセッサの使い方

下の図はマイクロソフト社の WORD です。ワードパッドよりもメニュー項目がさらに増えています。機能が増えた分だけ使い方に慣れるのは大変ですが、表示を印刷レイアウトにして体裁を工夫します。これを文章のお化粧と説明しています。



図 4.9 Microsoft Word の作業画面、ツールバーは一例です。

4.4.5 ワードプロセッサを使う作業の実際

プログラムの名前は Winword.exe ですが、普通の起動は Word のアイコンをクリックします。作業画面のツールバーなどのレイアウトは、利用者ごとにカスタマイズしますので、図3は一つの参考画面です。図1と較べてメニュー項目が増えた分は書式や体裁の設定に関するものです。

- ファイルメニューの「開く」から、以前にテキストエディタを使って作成した単純な文書原稿(プレーンテキスト)を読み出します。ファイルの拡張子が(.txt)になっているファイルです。なお、作業の終了のときには、Word の書式で保存しますが、そのときの拡張子は(.doc)に変えます。以前に作業したファイルを読み出すときは、ファイルのアイコンが Word のマークになっているので、ファイル名をクリックするだけで Word が起動し、以前のファイルを表示してくれます。
- 文章原稿の追加や削除の方法は、テキストエディタの使い方と同じです。幾つかのファイルに分けて作成した原稿を追加するときは、メニューバーの「挿入」でファイルを選びます。図なども割り付けることができますが、差し当たっては欲張らないで、テキストだけにします。
- 書式や体裁を整える作業は、専門用語では「版組み」や「割り付け」と言います。文字並びをマウスで選択

して反転させておいて、ツールバーの項目を選んで実行させます。図3に示してあるアイコンマークを片端から試して、それらがどのように文字並びに反映するかを見れば、文書の書式や体裁をどのように設計するかを体験できます。何かの処理をしたとき、その直後に処理を取り消す方法も覚えておきます。これは編集メニューで、「元に戻す」を使います。

- 処理の個々には、専門用語がからみます。ツールバーのアイコンにマウスのカーソルを合わせると、説明が表示されるようになっています。これが編集用の専門用語です。説明は「ヘルプ」メニューで読むことができます。
- 編集・校正作業が済んだら、ファイルに保存します。このファイルが Word で処理されましたので、Word の拡張子(.doc)を付けて保存します。

4.4.6 テキストエディタとの違い

ワードプロセッサは、印刷装置に何をを使うかに合わせて、書式制御ができるようにした文字編集用のソフトウェアです。ここでの演習は、前に作成したテキストファイルをワードプロセッサに読み込んで、書式(format)を整えることです。この演習は、マイクロソフト社の Microsoft Word を使います。このソフトウェアは、Windows の OS とは別個に Office シリーズの一部として購入させられます。書式というのは、用紙寸法を考えて、印刷したときの文字並びに調整を加えることです。昔の英文タイプライタは、活字の種類が一定でしたので、体裁を整える技法の種類は多くありませんでした。標準的な方法は、一行の表示で、**インデント・左揃え・中央揃え・右揃え・均等割付**などです。これは、文字数と文字幅を計算し、スペースを挿入して調整します。例外的に活字を部分的に差し替えて印刷できる場合には、**イタリック体(italic)、太字体(bold)**が利用できました。**アンダーライン**は、同じ文字位置で重ね打ちをしますが、モニタ上では前の位置の文字が消えてしまいますので、特別なソフトウェア上の工夫がされています。より専門的な印刷をするとなると、印刷屋さんが使う専門用語の理解が必要になりました。例えば、活字の寸法を表わす**ポイント**、**字形(フォント)の種類**、**段落**、**罫線(ケイ)**、**全角・半角・倍角**などです。日本語の印刷では、**禁則処理**があります。ワードプロセッサのメニュー項目やツールバーを使いこなすには、これらの用語の理解が必要です。これらは使いながら慣れて覚えます。

4.5 Word 文書から HTML 文書への変換

4.5.1 変換の演習

インターネットのホームページを作成することを目的としたソフトウェアは、ホームページビルダーなどがあります。この中身は、HTML 形式のファイルを作成するツールです。しかし、ここでは、Word の文書から HTML 文書に変換する方法です。この演習での手続きと注意事項とを箇条書きで示します；

- Word の画面で、文書を表示しておきます。
- メニューから「ファイル→名前を付けて保存→ファイルの種類→Web ページ(*.htm; .html)」の順に選択して HTML ファイルに変換します。ファイル名の拡張子は(*.html)のように4文字にしておくとい用性が上がります。保存場所は元の Word 原稿(*.doc)と同じフォルダが良いでしょう。
- 変換が済むと、Word の画面のスタイルが変わります。これは、いま作成した HTML ファイルを Word が改めて読み込んで表示するからです。
- ここで Word を終了します。そうして、保存用のフォルダを見て下さい。そこには、新しく(.html)の拡張子が付いたファイルが見つかるはずですが、もし、元の Word 文書に文字データ以外にグラフィックスなどのオブジェクトが含まれていると、同名で(.files)の拡張子の付いた一つのデータフォルダが作られます。
- 新しく作成された(.html)ファイルをダブルクリックすると、Internet Explorer または Netscape Navigator が起動して、このファイルをオフラインで表示してくれます。

下の図は、ワードプロセッサを使って、ほぼ同じ表示形式で作成した原稿を元に HTML 文書を作成した見本です。ワードプロセッサから HTML 文書に変換すると、このソースコードは非常に複雑になっています。下に示した画面は、必要最小限の機能を持たせた HTML ファイル作成に使うテンプレートです。

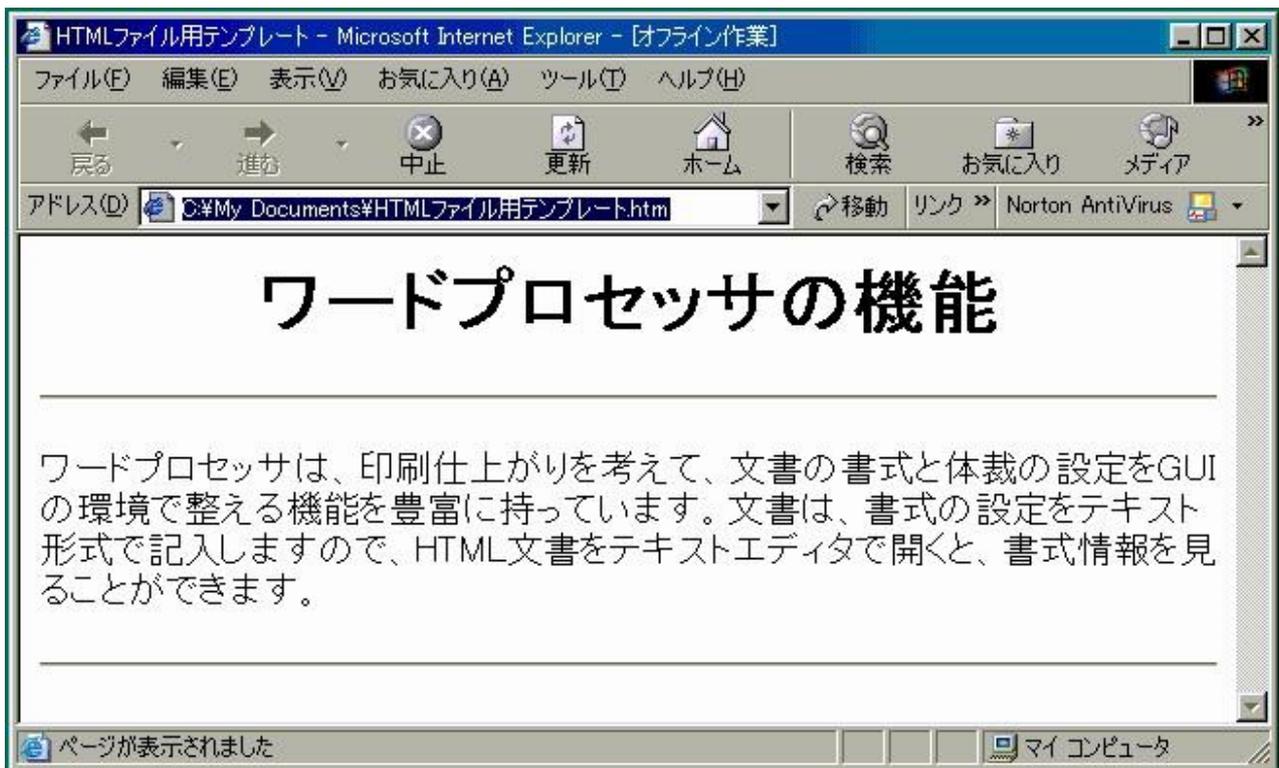


図4.10 HTMLの形式で作成したテンプレート文書の表示例

4.5.2 HTML のテキスト形式を覗く

Internet Explorer のメニューバーで「表示→ソース」をクリックすると、ノートパッドが起動してソースコードのテキストが表示されます。もしファイル寸法が大きければ、ワードパッドが起動します。元の書式を再現させるために、HTML タグを使った表現が並んでいますが、これは Word が自動作成しましたので非常にゴチャゴチャしたリストになっていることを眺めて下さい。書式制御のタグが並んでいますが、かなり複雑です。下に示したものは、前項の HTML 画面のソースファイルです。必要最小限のタグだけを残して簡単にしましたものです。表示に使うタグは、付録 C:基本的な HTML タグにあるものだけですので、解説は簡単にできます。

基本的な HTML タグ

HTML タグ	説明
<A>	別のファイルへのリンク(HREF 属性)、または名前(NAME 属性)を指定する
	~までの文字を太字(bold)で表示する
<BASEFONT>	標準寸法のフォントを指定する
<BLOCKQUOTE>	<BLOCKQUOTE>~</BLOCKQUOTE>までの文字が引用文であることを示す
<BODY>	<BODY>~</BODY>までの文字がブラウザで表示される本文であることを示す
 	改行を指定する
<CENTER>	<CENTER>~</CENTER>までの文字を中央揃えする
<COMMENT>	<COMMENT>~</COMMENT>または<!-- ~ -->の間の文字はコメント
<DIV>	<DIV>~</DIV>までを段落として扱い、左、中央、右位置に揃える
	~までの文字のフォント属性を指定する
<HEAD>	<HEAD>~</HEAD>までがヘッダー一部であることを示す
<HR>	水平方向の罫線を引く
<HTML>	<HTML>~</HTML>までが HTML 文書であることを示す
<I>	<I>~</I>までの文字を斜体 (italic)で表示する
	指定した画像を表示する
	~の中で箇条書きの見だし
<META>	汎用的に使われている HTML ファイルに対する制御を指定する
<P>	<P>~</P>までの文字を段落として扱う
<PRE>	<PRE>~</PRE>までの文字でスペースや改行をそのまま表示する
	~までの範囲の文字列に別の属性を与えるときに用いる
<SUB>	_~までの文字を下付き文字にする
<SUP>	[~]までの文字を上付き文字にする
<TABLE>	<TABLE>~</TABLE>までが表(テーブル)の項目であることを示す
<TD>	テーブル内で使用し<TD>~</TD>までがテーブルの列要素であることを示す
<TH>	テーブル内で使用し<TH>~</TH>までがテーブルのヘッダー列であることを示す
<TITLE>	<TITLE>~</TITLE>までを HTML 文書のタイトルとする
<TR>	テーブル内で使用し<TR>~</TR>までがテーブルの行であることを示す
<U>	<U>~</U>までの文字にアンダーラインを表示する
	~一般的な箇条書きリストに作成する
特殊文字	HTML 文書内では、特殊文字を表示するときに下のよう書き換える
<	<
>	>
&	&
スペース	

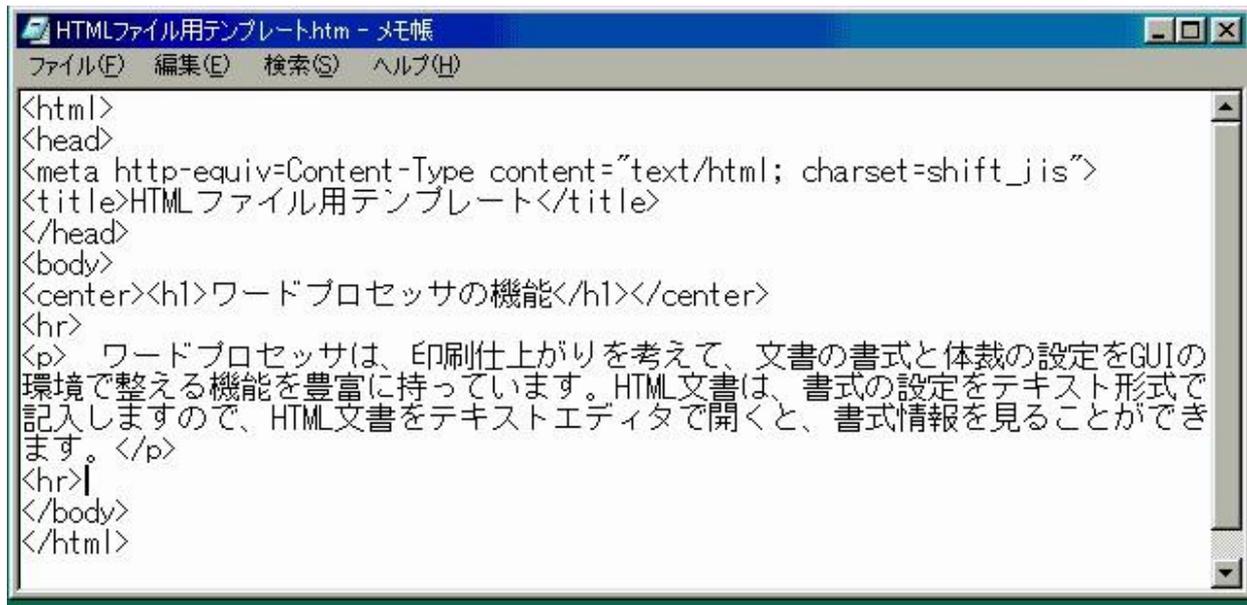


図 4.11 HTML 文書をテキストエディタで開いてタグの記入を見る

4.5.3 例題 HTML 文書の解説

Word 文書から HTML 文書に変換すると、必ずしも Word 文書の書式通りにはなりません。この相違を理解することが、HTML 文書を作成するときの勘所になります。下にその説明をまとめます。

- ・ タグは< >で囲んだ文字です。タグはすべて半角文字です。英字は、大文字と小文字の区別をしません、小文字を使う方向に進んでいます。
- ・ HTML のソースは、見易さのためにスペースを適当に利用します。しかし表示画面では、半角のスペース、タブ、改行は無視されます。したがって、Word 原稿で段落の始めをインデントしても、無視されます。また、段組みも無視されます。
- ・ 文章作成では、まとまりのある文章単位を段落に区切ることを意識的に行う必要があります。段落の始めには、自動的に一行の空白行が入ります。
- ・ 日本語の HTML 文書では、三行目の<META>で、文字コードを宣言しておきます。これが無いと、文字化けが起こることがあります。そのときは、Internet Explorer のメニューから「表示→エンコード」で適切な文字コードを探することができます。
- ・ 日本語の文書では、半角の英数字の混用も起こります。日本語文書でも英語文書でも、フォントの種類を変える表現も使われます。Word 文書から HTML 文書に変換すると、フォントの指定が非常に多く見られます。例題では、フォント指定をすべて外して、システムのデフォルトを採用してあります。そうしておく、メニューから「表示→文字のサイズ」を選択すると表示される文字寸法を変更できます。

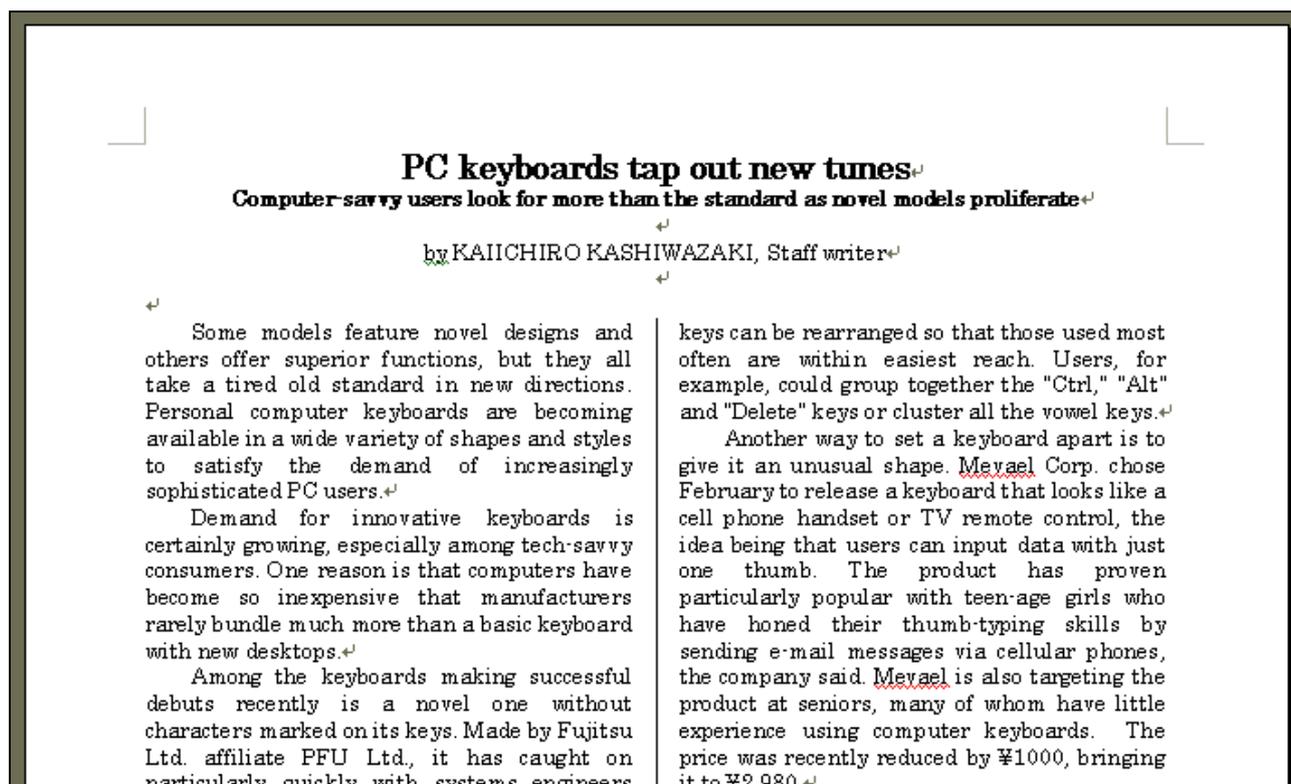


図 4.12 Word を使って2段組にレイアウトした原稿、HTML にすると段組み表示になりません。

MAKING RICE

Serves 4 35 minutes

2 cups short- or medium-grain rice
2²/₅ cups water

1. Place rice grains in the detachable container of an automatic rice cooker or in a large pot. To wash rice, add twice the amount of water to the pot, rinse with your hand until the water becomes clouded, and push on the rice grains with the heel of your hand in a kneading motion (Figure 1). Drain and change the water, and repeat 3 or 4 times, until the water is clear.
2. Drain rice grains well, add the 2²/₅ cups water, and let rice soak for at least 15 minutes before cooking.
3. If using an automatic rice cooker, turn it on to cook the rice. If using a pot, cover the pot tightly, and bring to a boil over high heat. When the pot starts to steam and the lid begins to shake, turn heat down to low. (If the pot seems about to boil over, move the lid off slightly; Figure 2.)
4. Cook another 10 minutes, until all the water has been absorbed, then turn off heat. (If you moved lid off earlier, replace it, so that the rice can steam properly.)
5. Let the rice steam, off the heat, for 10 minutes.

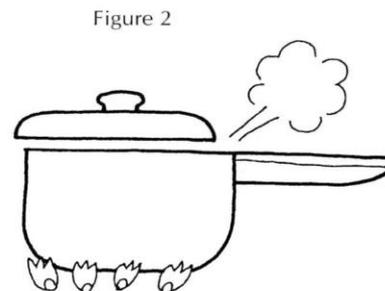
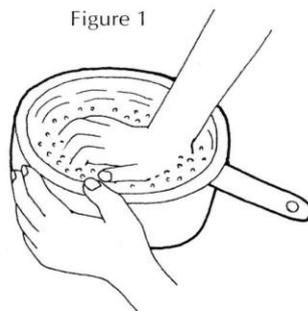


図 4.13 ご飯の炊き方の英文レシピです

出典: 小林カツ代、「The Quick and Easy Japanese Cookbook」、講談社インターナショナル、
ISBN5-7700-2504-1, Tokyo, 2000, p89

下のリストは、図 4.13 のレシピの文章を擬似的な Basic 風プログラム文に書き直したものです。プログラミングの文章は、このような構成になることを理解してもらうために作りました。

```

Program Making_Rice
'----- Comments and instructions
' Basic Technique for Japanese Cokking
' Serve 4
' Cooking time 35 minutes
' by Katsuyo Kobayashi
' programmed by S. SHIMADA
'----- Hardware Environments
  Data Cooker1="Automatic_rice_cooker.Detachable_container"
  Data Cooker2="Large_Pot"
  Dim The_heater As Heat
  Dim Timer As Minute
'----- Data statements as materials
  Dim ContainerNo As Integer      ' 1 or 2
  Dim Cooker As String
  Dim Rice_grains As Cup '--- short_ or medium_grain rice
  Dim Water As Cup
  Dim Timer as Minute
'----- step 1
  Input ContainerNo '--- user selects Cooker
  Select case ContainerNo
    Case 1: Cooker = Cooker1
    Case 2: Cooker = Cooker2
  End Select
  Rice_grains = 2
  Water = 2.4 '--- ratio of rice to water 1 to 1.2
  Place Rice_grains in Cooker
  For i=1 to 4
    Do
      Add Water
      Rinse with your hand
      Loop Until Water becomes clouded
      Push Rice_grains with the heel of your hand
    Do
      Drain Water
      Change Water
      Loop Until the water is clear
  Next
'----- step 2
  Drain Rice_grains
  Add water
  Let Rice_grain Soak 15 meniutes

```

```
'----- step 3
  Select Case ContainerNo
    Case 1:
      Turn on Cooker1
    Case 2:
      Cover Cooker2 'tightly
      Bring_to a boil
      The_heater = high
      If Cooker starts to steam and lid begin to shake Then
        The_heater = down
      EndIf
    End Select
'----- step 4
  Do While Timer < 10
    Water has been absorbed
  Loop
  If ContainerNo = 2 Then The_heater = off
'----- step5
  Do Timer Until < 10
    Let Rice_grains steam
  Loop
End
```