

# TEX の初心者用チュートリアルを作成

吉田 直子

## 目 次

1	研究のテーマ	1
2	誰のために作ったか・どのように役立つのか	1
3	TEX の問題点	2
4	作ろうとしているのも	3
5		4
1	全体の構成 . . . . .	4
6	第 1 章 TEX と L <sup>A</sup> T <sub>E</sub> X	5
1	TEX について . . . . .	5
2	L <sup>A</sup> T <sub>E</sub> X について . . . . .	5
7	TEX の処理方式	6
8	第 2 章 TEX の起動の仕方	8
1	コマンドで行う場合 . . . . .	8
2	ターミナルの起動 . . . . .	8
3	エディタの起動 . . . . .	9
4	L <sup>A</sup> T <sub>E</sub> X の起動 . . . . .	10
9	第 3 章 LaTeX 入力の基本	11
1	LaTeX の入力ファイル . . . . .	11
10	第 4 章 文書の構造	15
1	タイトルと概要 . . . . .	15
2	改行の扱い . . . . .	16
3	注釈 . . . . .	16
4	行頭の字下げの有無 . . . . .	17
5	空白の扱い . . . . .	17

6	書体を変える命令	18
7	文字サイズを変える命令	18
8	環境	18
9	箇条書き	19
10	長さの単位	20
11	空白を出力する命令	20
12	脚注	21
11	第5章 パッケージと自前の命令	21
1	パッケージ	21
2	表組み	22
3	『数学の基本』	23
4	『ページレイアウト』	23
5	ページレイアウトの変更	23
12	研究のまとめ	24

## 1 研究のテーマ

「人の役に立つのもを作る」というのが研究のテーマである。このテーマに沿って、授業で使用していた T<sub>E</sub>X の文系向けの初心者用 T<sub>E</sub>X を製作しようと考えている。

授業で使用していた T<sub>E</sub>X という教材は、自分はなかなか使いこなすことができず、行き詰まってばかりいた。なぜなら、T<sub>E</sub>X という教材が理系向けの教材だからである。T<sub>E</sub>X という教材は、文系向けの教材ではないということで、文系の人々が T<sub>E</sub>X を使うには難しいということが分かったのである。

そこで私は、「人の役に立つのもを作る」という研究のテーマに沿って、文系の人でも利用することができる初心者向けの T<sub>E</sub>X の教材を製作し、文系の人々が T<sub>E</sub>X を利用するときに役立てて欲しいと考えているのである。

## 2 誰のために作ったか・どのように役立つのか

大谷大学人文情報学科の生徒、文系の人、日常生活でパソコン (Windows やワープロ) を使っている人などに向けて、初心者でも使いやすい初心者向けの T<sub>E</sub>X を作成しようと考えている。なぜ T<sub>E</sub>X を作る必要があるのか。

それは、例えば大谷大学人文情報学科の生徒、あるいは文系の人々が論文やレポートを書くときに Word で文章を書くことがほとんど多いの

で、その場合、書式などをはじめからから設定していく必要がある。しかし TEX を使えば、決まった書式で文書が書け、段落も自動的に処理してくれるので仕上がりが大変きれいにできる。

日常生活でパソコンを使っている社会人などは、会社での資料政策など TEX を使えば図や表の挿入などができるので資料の仕上がりがきれいにできる。

TEX は、ワープロソフトとは一味違った洗練さを持った文書を作ることができるので、論文やレポートを書くときに役立つのである。だから、人文情報学科の生徒や文系の人には TEX を使うと良いと考えているのである。

### 3 TEX の問題点

TEX という教材は、もともと理系向けの教材であり文系向けの教材ではないため、理系の人を使うことが多い。TEX は、文系向けの初心者用の TEX の教材がなくインターネットでの情報、参考文献があまりないのが現状である。

文系の人には、日頃からワープロソフトを使い慣れているため、新しいソフトを使うということが苦に感じているので TEX を使ってみようとする人が少ない。

TEX は円記号と英単語で構成された命令を用いて文書を整えていくので、このような命令をその都度打ち込んでいくのは面倒と考えている人もいる。そして、初心者の人には難しすぎて使いにくいいため、あまり利用されていないのである。

TEX を使う場合、環境が整っていないと TEX を起動することもでき

ないという問題もでてくるのである。

インターネットでの初心者用の T<sub>E</sub>X の情報は、詳しく説明しており、わかりやすい部分もあるが、初心者が使うにはまだまだ説明が型苦しい部分が多い。実際にパソコンで起動してみてもうまく起動せず、次の段階に進めないために行き詰まってしまうということがある。

参考文献として奥村晴彦著者の L<sup>A</sup>T<sub>E</sub>X2e 美文書作成入門を使ってみても、内容がつまりすぎているので初心者が使うにはやはり少し難しいと思われる。

初心者が使うには、L<sup>A</sup>T<sub>E</sub>X2e 美文書作成入門の中から必要最小限のものを絞り込んだものが必要であると考えている。

#### 4 作ろうとしているのも

T<sub>E</sub>X という教材は、理系向けの教材であるので文系向けの T<sub>E</sub>X の教材があまりない。そこで私は、大谷大学人文情報学科の生徒、初心者用、文系の人でも使える初心者向けの T<sub>E</sub>X を作成しようと考えている。

初心者向けの T<sub>E</sub>X を作成するにあたって、参考文献として奥村晴彦著者の L<sup>A</sup>T<sub>E</sub>X2e 美文書作成入門を用いた。

その中から初心者が使うにあたって必要最小限の項目を選び、T<sub>E</sub>X の処理方式、L<sup>A</sup>T<sub>E</sub>X の入力の基本、最小限のルール、L<sup>A</sup>T<sub>E</sub>X のコマンドについての説明を第 1 章から第 5 章にまとめている。

### (1) 全体の構成

第 1 章は、T<sub>E</sub>X と L<sup>A</sup>T<sub>E</sub>X についての説明である。T<sub>E</sub>X(テフ)とは Donald E.Knuth(クヌース) 教授が作成したコンピュータプログラムのことである。

T<sub>E</sub>X はフリーソフトなので、他のフリーソフトと組み合わせて組版結果を画面や紙に出力したり、PDF といった形で出力する。また、Windows、Unix 等などあらゆる種類のコンピュータで使える。ここでは、Windowsds での T<sub>E</sub>X の使い方などを説明している。

L<sup>A</sup>T<sub>E</sub>X(ラテフ) は DEC(現 HP) コンピュータ科学者 (レスリー・ランポート) によって機能強化された T<sub>E</sub>X である。L<sup>A</sup>T<sub>E</sub>X の特徴は、文章の論理的な構造と視覚的なレイアウトとを分けて考えることができる点などを説明している。

第 2 章では、Windows で T<sub>E</sub>X を使うときのターミナル (端末) の起動の、L<sup>A</sup>T<sub>E</sub>X の起動、エディタの起動の説明をしている。また、エディタで起動した場合の入力の注意点などを説明している。

第 3 章では、L<sup>A</sup>T<sub>E</sub>X 入力の基本のなかで、L<sup>A</sup>T<sub>E</sub>X を入力するときに気をつけることや、特殊文字についての説明をしている。そして L<sup>A</sup>T<sub>E</sub>X のコマンドの説明、最低限のルール、文書ファイルを作る際のルールなどを説明している。

また、ユーザーデータを処理したりするプログラム部分をドキュメントクラスといい、その説明も詳しくしている。

第 4 章では、文章の構造について、文章を書くときにどのようにして書いていくのか、手順を説明している。文章の書体や文字を変えたりす

る方法も説明している。

第 5 章では、パッケージ、表組み、ページレイアウト、ページレイアウトの変更の説明をしている。

## 6 第 1 章 T<sub>E</sub>X と L<sub>A</sub>T<sub>E</sub>X

### (1) T<sub>E</sub>X について

T<sub>E</sub>X(テフ)とは、Donald E. Knuth (クヌース) 教授が作成したコンピュータプログラムであり、テキストと数式の組版を目指したものである。L<sub>A</sub>T<sub>E</sub>X はコンピュータによる組版のことをさしている。

組版とは、印刷関係で活字を組んで字を刻んだ印刷用の板を作ることを意味している。L<sub>A</sub>T<sub>E</sub>X はフリーソフトなので、他のフリーソフトと組み合わせて組版結果を画面や紙に出力したり、PostScript や PDF といった形式で出力したりする。

また、Windows, Unix 等々、あらゆる種類のコンピュータで使うことができる。

### (2) L<sub>A</sub>T<sub>E</sub>X について

L<sub>A</sub>T<sub>E</sub>X(ラテフ)はDEC(現HP)のコンピュータ科学者(レスリー・ランポート)によって機能強化されたT<sub>E</sub>Xである。

L<sub>A</sub>T<sub>E</sub>Xの特徴は、文章の論理的な構造と視覚的なレイアウトとを分けて考えることができる点である。

例えば「題名」という節の見出しがあれば文書ファイルには

```
\section{題名}
```

のように書いておく。この `\section...` という命令が、紙面上のデザインになる。レイアウトに対応するといったことは、様式・判型ごとに別ファイル (クラスファイル、スタイルファイル) に登録されているので、クラスファイルのデザインは自由に変更できる。

クラスファイルの設定を変更したり、新たな機能を付け加えたりするには、スタイルファイルを読み込む必要がある。

L<sub>A</sub>T<sub>E</sub>X は章・節・図・表・などの番号や、目次・索引・引用文献の処理も自動的に行ってくれる。また、参照個所には番号ページを自動挿入できるようになっている。

文書の中に、`\` に続く英語らしき列がある。これが、T<sub>E</sub>X で使われる「命令」(正確には「コントロール・シーケンス」)なのである。これらの「命令」を解釈してくれるソフトが T<sub>E</sub>X である。

T<sub>E</sub>X は、`\` に続く命令を使って、ワープロソフトとは一味違った洗練さを持った文書を作っていくことができるのである。

## 7 T<sub>E</sub>X の処理方式

T<sub>E</sub>X では、原稿は自分の使いなれたソフトで書いて、テキストファイルとして保存しておきます。これをあとで T<sub>E</sub>X でコンパイルし、印刷用のイメージファイル (dv:ファイル) を作ります。

例えば pL<sub>A</sub>T<sub>E</sub>X 2<sub>ε</sub> では、

```
\documentclass{j s a r t i c l e}
```

```
\begin{document}
```

これはサンプルの文章です。

テキストファイルの中では、

どこで改行してもかまいません。

印刷結果の改行の位置は勝手に決めてくれます。

段落の切れ目には空の行を入れておきます。

```
\end{document}
```

のようなテキストファイルを入力すると、

<p>これはサンプルの文章です。テキストファイル中では、どこで改行してもかまいません。印刷結果の改行の位置は勝手に決めてくれます。</p> <p>段落の切れ目には空の行を入れておきます。</p>
---

のように出力される。

テキストファイルを使うことの利点は、テキストファイルはコンピュータの機種に依存がなく、どんなコンピュータや携帯電話でも、テキストファイルなら安全にやりとりできることである。

出力についても、いろいろな種類のコンピュータの画面やプリンタ、写植機があり、今後どんな装置が現れるのかもわからないので、TeX 本体は最終的な印刷の直前までの仕事を行い、その状態を中間ファイルに書き出しておく。この中間ファイルのことを dvi ファイルという。

Dvi は device independent (装置に依存しない) という英語の略である。この dvi ファイルを読み込んでパソコンの画面、各種プリンタ、写植機などに出力するための専用ソフト (dvi ドライバ、dvi ウェア) が、出力装置ごとに別に用意されているのである。

特に画面出力用の dvi ドライバのことをプレビューアともいいう。

## 8 第2章 TEX の起動の仕方

### (1) コマンドで行う場合

windows のコマンドプロンプト (MS-DOS プロンプト)、Mac OS X や Linux 等のターミナルでコマンドを打ち込んで使う場合についての説明である。

### (2) ターミナルの起動

ターミナル (端末) は、コマンド (命令) を打ち込むための画面である。

- ・ windows95 ~ Me では「MS-DO プロンプト」といわれている。通常は [スタート] [プログラム] [アクセサリ] の中にある。
- ・ windowsNT, 2000, XP 以降では「コマンドプロンプト」といわれている。通常は [スタート] [プログラム] [アクセサリ] または [すべてのプログラム] [アクセサリ] の中にある。

ターミナルを起動したなら、そのターミナルの現在位置 (カレントディレクトリ) に注意しておかなければならない。

通常は「ホームディレクトリ」という自分用の作業フォルダがカレントディレクトリになっているが、Windows は自分用の作業フォルダを作ることから始めなければならないことが多いと考えている。

例えば、コマンドプロンプトで、

```
C: \WINDOWS>
```

のようになった場合は、「カレントディレクトリは C:¥WINDOWS です」という意味を表している。ここは Windows 用の大事なものがたく

さん入っているフォルダなので、他の場所に適当なディレクトリを作成 (mkdir=make directory) し、そこに移動 (cd=change directory) する必要がある。

### (3) エディタの起動

T<sub>E</sub>X の文書ファイルを作るには、テキストファイルを作成するための「テキストエディタ (エディタ)」というソフトを使うことになっている。T<sub>E</sub>X の文書ファイル名は、たとえば ex1.tex のように、拡張子 (ファイル名の末尾につに付いている「.(ピリオド) + 英数字 3 文字」で表す記号のようなもので、ファイルの種類を示すのに使われている。

以下では ex1.tex という文書ファイルを作成し、それを L<sup>A</sup>T<sub>E</sub>X で処理してみることにする。

Windows の場合、ターミナルから

```
c: work>notepad ex1.tex
```

のように打ち込むと「メモ帳」が立ち上がるので、「ファイル ex1.tex が見つかりません。新しく作成しますか?」と聞いてきたら「はい」と答える。

エディタを起動したら、最初は実験ですから日本語は使わず、エディタの中にすべて半角文字で次のように打ち込んでみる。これは Hello, TeX! という文を出力するための入力である。

```
\documentclass{article}
\begin{document}
Hello, \TeX!
\end{document}
```

入力にあたって次のことに注意する必要がある。

- すべて直接入力（半角文字）で打ち込んでいかなければならない。  
(例) `d o c u m e n t` は `document` にする。
- 大文字と小文字を間違わないように、必ず上に書いた通りに打ち込んでいかなければならない。  
(例) `LaTeX` を `laTeX` にしてはいけない。
- 各行の最後で Enter キー（または Return）を押して改行していか  
なければならぬ。  
(例)
- スペースキーを何度も押して次の行に移ってはいけない。
- 半角の円記号 ¥ がたくさんあるが、キーボードに ¥ のないコン  
ピュータでは ¥ の代わりに `¥textbackslash`(バックスラッシュ) を  
打ち込む必要がある。

打ち込み終わったら、起動時にファイル名を指定していない場合は、`ex1.tex` というファイル名で保存しなければならない。保存するフォルダ（ディレクトリ）はかならずさきほどのターミナルの現在位置と同じにしておかなければならないのである。

#### (4) L<sub>A</sub>T<sub>E</sub>X の起動

ターミナルで

```
platex ex1.tex
```

と入力し Enter キーを押す。

```
Output written on ex1.dvi(1 page,380bytes).
```

と画面に表示されると L<sub>A</sub>T<sub>E</sub>X での処理は OK. である。ただし、380bytes というサイズは違う値かもしれない。

ここで行ったことは、文書ファイル `ex1.tex` を、`dvi` ファイル `ex1.dvi` に変換する作業である。この T<sub>E</sub>X の処理にあてはめて `dvi` ファイルに変換する作業を コンパイル するといわれているのである。

## 9 第 3 章 LaTeX 入力の基本

### (1) LaTeX の入力ファイル

LaTeX の入力ファイルは、ASCII 形式のプレーンテキストファイルなので、入力ファイルの作成にはどんなテキストエディタを使用してもかまわない。

入力ファイルには、文書の内容に加えてさらにテキストをどのように組版するのかを LaTeX に伝えるコマンドも含んでいるのである。

#### (i) スペース

- ・ スペースやタブといった “ホワイトスペース” は、どちらも同じように “スペース” として扱われるのである。
- ・ “ホワイトスペース” は、2 つ以上が連続していても「1 つの “スペース”」として扱われるのである。
- ・ 行頭のホワイトスペースは通常無視され、行末の改行は “スペース” として扱われるのである。
- ・ 入力ファイル中の空行は、段落の区切りを表している。空行は、いくつ連続していても一つの空行と見なして処理が行われているのである。

#### (ii) 特殊文字

L<sup>A</sup>T<sub>E</sub>X では、以下に示す半角記号は T<sub>E</sub>X 特別な意味を持ち、入力した通りには表示されない。その文字などを表示したいときは、前にバッ

クシュラッシュ「\」をつける必要がある。バックシュラッシュの文字自体の表記は、「\textbackslash」とする。

`$ & % # _ { } \ ~ ^ < > |`

### (iii) L<sup>A</sup>T<sub>E</sub>X コマンド

- ・ コマンドは、バックシュラッシュ「\」から始まる。
- ・ コマンドは大文字小文字の区別があり、アルファベットだけからなっている。
- ・ 各コマンドは、スペース、数字、アルファベット以外の文字で区切られている。コマンドのあとのホワイトスペースは無視されている。

スペースを出力したい場合は、コマンド名の後に{ }とスペース、もしくはスペースコマンド(\ )を置く必要がある。{ }は、L<sup>A</sup>T<sub>E</sub>X コマンド名のあとの全てのスペース を無視するのを防いでくれる。

### (iv) 最低限のルール

T<sub>E</sub>X では半角の¥は特別な意味を持つので半角文字で入力しなければならぬ。Windows なら半角/全角キーで直接入力の状態に切り替えて入力するのが安全である。

「¥」は和文(全角)文字でも出力できる。半角の¥がキーボードにないコンピュータでは\ (バックスラッシュ)を入力する必要がある。

各行末では、必ず Enter を打たなければならない。英語の部分は半角文字で入力するが、句読点や括弧は全角文字を使わなければならない。

文章ファイルを作る際にはいくつかのルールがある。

- ・ 文書ファイル名の最後に.tex を付ける。日本語のファイル名や空白は、うまく使えないことがあるので、英語のアルファベット・数字・

アンダーバー ( \_ ) に限るのが安全である。

- ・ 文書の最初に半角で次のようなおまじない書いておく。

```
\documentclass{jsarticle}
```

このオマジナイは、文章を jsarticle という名前の書式 (ドキュメントクラス) で組版することを意味している。

ドキュメントクラスとは、ソフトによっては「スタイル」または「テンプレート」と呼ばれるものである。jsarticle は比較的新しいドキュメントクラスの一つである。これがインストールされない場合は、

```
\documentclass{jarticle}
```

のように jarticle を指定しなければならない。用紙が A4 版でないときは

```
\documentclass[a5paper]{jsarticle}
```

のように a5paper, b4paper, b5paper のいずれかを [] に囲んで入れておく。

- ・ 次に、「これから文書が始まるよ」という意味の次のオマジナイを書いておく。

```
\begin{document}
```

\item 文書の最後には、次のオマジナイを書いておく。

```
\end{document}
```

段落の区切りには空の行を挿入する。つまり Enter キーを 2 回続けて打つことになる。段落の頭にスペースを入れる必要は

ない。

いわゆる半角カナは使わず、全角カナ使わなければならない。

、 、 、 、 、 、 (株) など、いわゆる機種依存文字 (JIS 第 1・第 2 水準外の文字) は、そのままでは文字抜けや文字化けする可能性がある。ローマ数字 、 、 等は、半角のを並べて 、 、 のように書くのが正しい方法である。(株) は全角 3 文字で (株) と書く。

#### (v) ドキュメントクラス

$\LaTeX$  の文書ファイルの最初の `\documentclass{...}` のような行は、ドキュメント (文書) のクラス (種類) を指定するものである。大学生が書くレポートや論文では、`jsarticle` を選んでおけば間違いない。

¥ `documentclass[a5paper]jsarticle` のような角括弧 [ ] の中は、クラスオプションまたは単にオプションといい、必要に応じて次のような指定をする。

「デフォルト」と書いたものは、標準でそのようになるので、そのオプションを指定する必要はない。

10pt 本文の欧文文字サイズを 10 ポイントにする (デフォルト)。

12pt 本文の欧文文字サイズを 12 ポイントにする。論文を書くときに適している。

次は用紙サイズのオプションである。無指定では A 4 判になる。他に、

a5paper A5 判 (148mm × 210mm)

b4paper B4 判 (257mm × 364mm)

b5paper B5 判 (182mm × 257mm)

などがある。

複数のオプションを指定するときには、

- `¥ documentclass[b5paper,12pt]jsarticle`  
のように、半角のコンマ (,) で区切る必要がある。

## 10 第4章 文書の構造

文書は、タイトル、著者名、章の見出し、節 (セクション) の見出し、段落、のような構造を持っている。L<sup>A</sup>T<sub>E</sub>X で文書ファイル中に書き込むのは、このような文書の構造である。

ワープロソフトでは、文書の構造とレイアウト (文字サイズや書体) とが明確に分かれていない。

文書の構造を記述するために、次のような命令が使われている。

`\section{何々}`      セクション (節) の見出し

`\subsection{何々}`      サブセクション (小節) の見出し

`\begin{quotation}`      引用の始め

`\end{quotation}`      引用の終わり

### (1) タイトルと概要

タイトルを出力するためには次の四つの命令を使う。

`\title{何々}` 文書名の指定

`\author{何々}` 著者名の指定

`\date{何々}` 日付の指定

`\maketitle` 文書名・著者名・日付の出力

`\title`, `\author`, `\date` は`\maketitle`の前ならどこにあってもかまわない。そして、この三つの順序もどれが先でもかまわない。実際にタイトルを出力するのは`\maketitle`である。

タイトル、著者名、日付が長いときは、`¥¥` で区切るとそこで改行する。

## (2) 改行の扱い

TEX 段落を区切るためには、単なる改行ではなく、何もなく (空行) を段落の間に挿入する必要がある。

## (3) 注釈

注釈とは、文章の中で出てきた注意すべき語句を説明するために付けるもので、本文とは関係のない情報を示すために使われるのである。

欧文では改行は空白になるが、場合によっては改行を単に無視してほしいときもある。このようなときは、最後の文字の直後に `%` を書くことよいのである。

入力 `Abcdefghi jklmno%`

pqrstuvwxyz

出力 Abcdefghi jklmnopqrstuvwxyz

この % はその行のそれ以降を T<sub>E</sub>X に無視させる特殊な命令である。この文字からは改行文字 Enter を含めてすべて無視されるので、コメント (注釈) を書くのに便利である。

#### (4) 行頭の字下げの有無

段落の開始には字下げをするものであるが、何らかの理由により字下げを抑制したいときがある。字下げの有無をする場合、¥indent と ¥noindent の命令が使える。

#### (5) 空白の扱い

空白には全角空白と半角空白がある。画面上では半角空白 2 個と全角空白 1 個の区別がつきにくいだが、T<sub>E</sub>X にとってはこれらの意味はまったく違うものである。

- ・ 全角空白 単にその個数分の全角空白が出力されるだけである。
- ・ 半角空白 欧文の単語間のスペースを出力するが、ページの右端をそろえるためにかなり伸び縮みし、半角空白を何個並べても 1 個の空白しか出力できない。

途中で改行されると困るときは `~` (波印、チルダ) を使う。`~` は半角空白と同じ幅の空白を出力する命令であるが、`~` で空けた空白では改行が起らない。チルダを使うときは、`¥` とで挟む必要がある。

#### (6) 書体を変える命令

- ・ 和文書体では、ゴシック体を使おう。

シック体に変える命令は、`¥textgt...` である。

- ・ 欧文書体では、イタリック (Italic) とタイプライター (Typewriter) が使かれている。

イタリック (Italic) の場合は、`¥textit{Italic}` である。これは、強調や書名のときに使かれているのである。

タイプライター (Typewriter) の場合は、`¥texttt{Typewriter}` である。これは、コンピューターの入力例のときに使かれているのである。

#### (7) 文字サイズを変える命令

よく使われるのは `¥small`、`¥normalsize`、`¥large` である。`¥normalsize` は標準の大きさなので特に指定する必要はない。これらの命令は、`{ ¥small 小さな文字 }` のように、命令の後に区切りの半角空白をいれ、適用範囲を `{ }` で囲まなければならない。

#### (8) 環境

`¥begin{何々} ¥end{何々}` のような対になった命令を環境 (environment) という。例えば、`¥begin{ruote} ¥end{quote}` なら quote 環境になる。

環境の内側は一種の別天地で、いろいろな設定が環境の外側と異な

る。例えば `quote` 環境なら左マージン (左余白) が周囲より広くなるので、環境の中で書体などを変えても、環境の中には影響が及ばないのである。

`quote` 以外によく使う環境は次の三つである。

- `flushleft` を左寄せの環境という。
- `flushright` を右寄せの環境という。
- `center` をセンタリング (中央揃え) の環境という。

これらの環境の途中で改行するには `¥ ¥` を使かう必要がある。

## (9) 箇条書き

環境の例として、いろいろな箇条書きの方法を説明する。

- `itemize` 環境

頭に `•` などの記号を付けた箇条書きである。

入力	出力
<code>¥ LaTeX</code> には	<code>LaTeX</code> には
<code>¥ beginitemize</code>	
<code>¥ item</code> 記号付き箇条書き	• 記号付き箇条書き
<code>¥ item</code> 番号付き箇条書き	• 番号付き箇条書き
<code>¥ item</code> 見出し付き箇条書き	• 見出し付き箇条書き
<code>¥ enditemize</code>	
の機能がある	の機能がある。

- `enumerate` 環境

頭に番号を付けた箇条書きです。

入力	出力
<code>¥ LaTeX</code> には	<code>LaTeX</code> には

¥beginenumerate

¥item 記号付き箇条書き

1. 記号付き箇条書き

¥item 番号付き箇条書き

2. 番号付き箇条書き

¥item 見出し付き箇条書き

3. 見出し付き箇条書き

¥endenumerate

の機能がある

の機能がある。

- description 環境

左寄せ太字で見出しを付けた箇条書きで。

#### (10) 長さの単位

T<sub>E</sub>X で使う長さの単位には、

- cm センチメートル (1cm=10mm)
- zw 和文フォントのボディの幅 (全角一文字) がある。

#### (11) 空白を出力する命令

以下で長さとした部分には 1zw 12.3mm など T<sub>E</sub>X で使える単位を付けた数を書き込む。左右にスペースを入れるには次のどちらかの命令を使わなければならない。

- ¥hspace 長さ 行頭・行末では出力されない。
- ¥hspace\*長さ 行頭・行末でも出力される。
- ¥hspace\*1zw は全角空白を一つ入れるのと同じことである。

段落間などにスペースを入れるには次のどちらかの命令を使わなければならない。

`\vspace{長さ}` ページ頭・ページ末では出力されない。段落間に余分のスペースを入れるときによく使われる。

`\vspace*{長さ}` ページ頭・ページ末でも出力される。図を貼り込むスペースを空けるときなどに使われる。

例えば0.5行分のアキを入れるのには `\vspace{0.5\baselinestrip}` としておく。

## (12) 脚注

ページの下部に出力する脚注は `\footnote{...}` という命令を使う。

## 11 第5章 パッケージと自前の命令

### (1) パッケージ

T<sub>E</sub>X にはマクロという自前の命令を作る機能があるのである。L<sup>A</sup>T<sub>E</sub>X は、このマクロ機能を使って T<sub>E</sub>X を拡張したものである。そして、この機能を使うことによって L<sup>A</sup>T<sub>E</sub>X をさらに拡張することができるのである。

自前の命令は文書ファイルに直接書き込むこともできるが、パッケージ化して別ファイルに保存しておくこともできるのである。

パッケージとは、L<sup>A</sup>T<sub>E</sub>X の機能を簡単に拡張するためのしくみである。L<sup>A</sup>T<sub>E</sub>X の命令のことを一般にコマンドと言われているが、このような自前の命令のことを特にマクロ (macro) ということがある。

マクロ (Macro) は macroinstruction を縮めてできたコンピュータ用語で、一般に「複数の命令に展開されるような一つの命令」という意味をもっているのである。

## (2) 表組み

### 表組みの基本

まずは、表の基本である罫線のない表を書いてみることにする。

商品	金額 (円)	数
スイカ	200	10
すもも	600	20

このように出力するには以下のように入力する。

```
documentclass{j s a r t i c l e}
```

```
\begin{document}
```

```
\begin{center}
```

```
\begin{tabular}{l r r}
```

```
商品 & 金額 (円) & 数 \\
```

```
スイカ & 200 & 10 \\j
```

```
すもも & 600 & 20
```

```
\end{tabular}
```

```
\end{center}
```

```
\end{document}
```

¥begin\{center\}¥end\{center\}は表を左右中央に置く命令であるので、表組みとの直接の関係はない。

その下の¥begin\{tabular\}¥end\{tabular\}が表そのものを出力する命令なのである

(3) 『数学の基本』

数学に無縁な人のために

LaTeX2では、¥印で始まる命令を使って、いろいろな文字・記号を出力することができる。

このような記号は、その記号を出力する命令（ なら ¥forall ）を \$（ドル記号）でサンドウィッチして出力する。  
\$¥forall\$（ドルで囲んだ部分が「数式」です）

(4) 『ページレイアウト』

ページレイアウトとは、メニューやイメージの配置など、ブラウザ内でのページの表示状態を表すこと指している。

(5) ページレイアウトの変更

ノンブルや柱のスタイルを変えるには ノンブルとは、各ページに振るページ番号のことである。各ページの上部または下部に出力する。

柱とは、各ページの上か下に出力する章や節の名前などのことで、ページの上部に出力している。ページスタイルはどのドキュメントクラスにも用意されている。

- ・ plain 何も指定しなければこれに指定される。ノンブルがページ下部に出力されるだけで、柱は出力されない。
- ・ empty ノンブルも柱も出力されない。
- ・ headings ノンブルと柱がページ上部に出力される。例えば article ドキュメントクラスであれば、柱には節 (section) の名前が出力される。

## 12 研究のまとめ

「人の役に立つのもを作る」という研究のテーマに沿って、理系向けの TeX という教材を、文系向けの初心者用 TeX にするというのが課題である。

自分も TeX という教材を利用する上では初心者であるので、自分自身も使いこなせていない状態で、文系向けの初心者 TeX の教材を作成するのは、かなり難しいものがあったのである。

TeX はもともと理系向けの教材であるため、文系の人々が TeX を利用するには当然無理がある。インターネットで TeX のホームページを色々探してみても、一部分でしか TeX の内容を細かく説明していないのである。

現在での新しい TeX の本として、奥村晴彦著者の L<sup>A</sup>T<sub>E</sub>X2e 美文書作成入門があるが、初心者がこの本を使いかいこなすには、初心者の私でも大変苦労したのである。

参考文献として用いた奥村晴彦著者の L<sup>A</sup>T<sub>E</sub>X2e 美文書作成入門の中から、初心者が利用する時には必要であるだろうと思う項目を選び、簡単にまとめている。

自分自身が理解できていないと、それ以上先には進めることが出来ず、何度も立ち止まってばかりいた。

自己評価 できたこととできなかつこと、それはなぜか

——TEX の初心者用チュートリアルの作成——

注

文献表

奥村晴彦

2004 『美文書作成入門』技術評論社