

# 目次

第 1 章	表組み	2
1.1	表組みの基本	2
1.2	列割の一時変更	3
1.3	罫線	3
1.3.1	罫線の引き方	3
1.3.2	部分的に罫線を引く方法	3
1.4	横幅の決まった表	4
1.5	table 環境	4
1.6	表組みの応用	5
第 2 章	数式	7
2.1	数式モード	7
2.2	基本的な数式環境	7
2.2.1	math 環境	7
2.2.2	displaymath 環境	7
2.2.3	equation 環境	7
2.2.4	eqnarray 環境	8
2.3	数式の参照	9
2.4	文字列の扱い	9
2.5	空白	9
2.6	添え字	10
2.7	分数	10
2.8	様々な数式	11
第 3 章	BIB <sub>T</sub> E <sub>X</sub>	12
3.1	BIB <sub>T</sub> E <sub>X</sub> について	12
3.2	BIB <sub>T</sub> E <sub>X</sub> の使用	12
3.3	BIB <sub>T</sub> E <sub>X</sub> マクロの設定	13
3.4	属性の解説	14

# 第1章 表組み

## 1.1 表組みの基本

次のような表があるとします．

Table 1.1 最も基本的な表

書名	数量	金額
進化する人工物	1	1400 円 ( 税別 )
未来革命	1	1700 円 ( 税別 )

上記のように出力するには次のようなソースにします．

Fig. 1.1 最も基本的な表のソース

```
\begin{center}
\begin{tabular}{lcr}
  書名          & 数量 & 金額          \\
  進化する人工物 & 1    & 1400 円 ( 税別 ) \\
  未来革命       & 1    & 1700 円 ( 税別 ) \\
\end{tabular}
\end{center}
```

`center` 環境は表組みと直接関係ありませんが，`center` 環境がなかったら，表は別行立てにならず，本文の中に入り込んでしまいます．これは，表は一つの大きい文字として扱われているためです．

`\begin{tabular}...\end{tabular}`までが表そのものを出力する `tabular` 環境です．では，`tabular` 環境の書式と引数の指定を以下にまとめておきます．

Fig. 1.2 `tabular` 環境の書式

```
\begin{tabular}{引数 (列指定)}
      :
\end{tabular}
```

Table 1.2 `tabular` 環境の引数

指定	解説
l	左寄せ
c	中央
r	右寄せ
	縦の罫線
	縦の二重罫線

引数 ( 列指定 ) は列の数だけ並べます．Table 1.1 では列指定は `lcr` でしたので，1 列目は左寄せ，2 列目は中央，3 列目は右寄せになっています．

列の区切りは `&`，行の区切りは `\\` です．

## 1.2 列割の一時変更

一時的にいくつかの列をまとめて 1 列のように扱う命令は

`\multicolumn{まとめる列数}{引数(列指定)}{中身}`

というように書きます。

例えば、次のような表を考えます。

Table 1.3 1 行目をまとめた表  
請求書

書名	数量	金額
進化する人工物	1	1400 円(税別)
未来革命	1	1700 円(税別)

上記の表は「請求書」が 3 列分まとめて中央揃えで出力されているのがわかります。  
上記の表のソースは次のようになります。

Fig. 1.3 1 行目をまとめた表のソース

```
\begin{tabular}{lcr}
\multicolumn{3}{c}{\textbf{請求書}} \\
\multicolumn{1}{c}{書名} & 数量 & 金額 \\
進化する人工物 & 1 & 1400 円(税別) \\
未来革命 & 1 & 1700 円(税別) \\
\end{tabular}
```

`\textbf{}` は、囲まれた文字を太字する命令です。

`\multicolumn{1}{c}{書名}` は「書名」を中央揃えに直すということです。

## 1.3 罫線

### 1.3.1 罫線の引き方

Table 1.4 罫線を引いた表

請求書		
書名	数量	金額
進化する人工物	1	1400 円(税別)
未来革命	1	1700 円(税別)

Fig. 1.4 罫線を引いた表のソース

```
\begin{tabular}{|l|c|r|} \hline
\multicolumn{3}{c}{\textbf{請求書}} \\ \hline
\multicolumn{1}{|c|}{書名} & 数量 & 金額 \\ \hline
進化する人工物 & 1 & 1400 円(税別) \\
未来革命 & 1 & 1700 円(税別) \\ \hline
\end{tabular}
```

横罫線は { 列指定 } の中の該当個所に半角縦線 | を入れるだけです。また 2 重の縦罫線にしたいときは || と書きます。  
横罫線は \hline です。 \hline \hline と書くと、2 重の横罫線になります。

### 1.3.2 部分的に罫線を引く方法

\hline 命令を使用すると、表の幅の文だけ横の罫線を出力します。しかし、場合によっては、1, 2, 4 列目だけに引きたい場合なども出てくると思います。例えば、Table 1.5 のような表を作りたいとします。

Table 1.5 部分的に罫線を引いた表

出版社	書名	著者	定価
オーム社	進化する人工物	三木光範	1400 円
講談社	団塊世代はなぜインターネットが苦手か		800 円

Fig. 1.5 部分的に罫線を引いた表のソース

```

出版社 & 書名 & 著者 & 定価 \\ \hline \hline
\begin{center}
\begin{tabular}{|l|l|l|l|} \hline
オーム社 & 進化する人工物 & 三木光範 & 1400 円 \\
\cline{1-2}\cline{4-4}
講談社 & 団塊世代はなぜインターネットが苦手か & & 800 円 \\
\end{tabular}
\end{center}

```

このような場合には `\hline` 命令の代わりに `\cline` 命令を使います。

`\cline` 命令は `\cline{数字 - 数字}` のように罫線を引きはじめたい列から引き終わりたい列までの数字を書きます。

## 1.4 横幅の決まった表

全体の横幅の決まった表は `tabular` 環境の代わりに `tabular*` 環境を使います。次のように使います。

Fig. 1.6 横幅を決める表の書式

```

\begin{tabular*}{幅}{@{\extracolsep{fill}}列指定}
:
\end{tabular}

```

幅を合わせるために列間に均等に空きが入ります。Table . 1.6 , Fig . 1.7 は幅を 150mm にしたときの表と、そのソースです。

列指定 `p{10zw}` は「全角 10 文字幅」という意味で、入りきらなければ、次行送りになります。

## 1.5 table 環境

これまで、`tabular` 環境の話ばかりしてきましたが、`tabular` 環境はあくまで、表を作るだけです。例えば、ページの残りが少ないときには表は次のページに送られてしまい空きすぎのページができます。

表全体の領域を確保する `table` 環境で表を囲んでおけば、空きすぎのページができないように最適な位置に表が移動します。

また、`table` 環境で囲んでおけば、`figure` 環境のように

```
\caption{説明}
```

で表の説明をつけたり、

```
\label{参照名}
```

Table 1.6 横幅を 150mm にした表

請求書		
書名	数量	金額
進化する人工物	1	1400 円 (税別)
未来革命	1	1700 円 (税別)

Fig. 1.7 横幅を 150mm にしたときの表のソース

```
\begin{tabular*}{150mm}{@{\extracolsep{\fill}}|p{10zw}|r|r|}\hline
\multicolumn{3}{|c|}{\textbf{請求書}} \\\hline
書名 & 数量 & 金額 \\\hline
進化する人工物 & 1 & 1400 円 (税別)\\
未来革命 & 1 & 1700 円 (税別)\\ \hline
\end{tabular*}
```

で、参照名をつけたりできます。一般的に表の説明は表の上に来るように書きます。  
また、参照名をつけておけば、

```
\pageref{参照名}ページの表\ref{参照名}は ...
```

と、書くと

```
12 ページの表 3 は ...
```

というようにページ数と表番号を出力することができます。

## 1.6 表組みの応用

最後に凝った表に仕上げるために `hhline` パッケージについて触れておきます。  
まずは、`hhline` パッケージを

```
\usepackage{hhline}
```

で、読み込んでください。`hhline` 命令は、

```
\hhline{引数}
```

と、いう書式で使います。引数は `tabular` 環境の引数として指定した列要素 (`c` などの配置指定や、`|` などの垂直罫線指定) に対応するように縦罫線や横罫線の種類を指定します。

```
児玉 憲造
```

と出力するには、

```
\begin{center}
\begin{tabular}{|c||c||}
\hhline{|t::t::t|}
児玉 & 憲造 \\\hline
\hhline{|b::b::b|}
\end{tabular}
\end{center}
```

と、言うふうに記述します。Table 1.7 に `\hhline` 命令の引数を示しておきます。

Table 1.7 hhline 命令の引数

指定	解説
=	列の幅に相当する横の二重罫線
-	列の幅に相当する横の罫線
~	横の罫線を引かない
	横の罫線を貫くたての罫線
:	横の罫線を貫かないたての罫線
#	横の二重罫線と交わる縦の二重罫線
t	二重罫線の上のみ
b	二重罫線の下のみ
*	繰り返し (*{2}{- }という指定は- -と同じ)

下に凝った表の例を示してみました．参考にしていただければ幸いです．

Table 1.8 凝った表の例

知的システムデザイン研究室で紹介された書籍			
書名	著者	出版社	定価
進化する人工物	三木光範	オーム社	1400 円 ( 税別 )
未来革命	浅井隆	第 2 海援隊	1700 円 ( 税別 )

Fig. 1.8 凝った表の例のソース

```

\begin{table}[h]
\begin{center}
\begin{tabular}{|l|l|l|l|r|}
\hhline{|t:*{4}{=}:t|}
\multicolumn{4}{|c|}{\textbf{知的システムデザイン研究室で紹介された書籍}}\\
\hhline{|:t=:t=:t=:t=:|}
書名 & 著者 & 出版社 & 定価 \\
\hhline{|:t=:t=:t=:t=:|}
進化する人工物 & 三木光範 & オーム社 & 1400 円 ( 税別 ) \\
未来革命 & 浅井隆 & 第 2 海援隊 & 1700 円 ( 税別 ) \\
\hhline{|b:b=b:b=b=b=:b|}
\end{tabular}
\end{center}
\end{table}

```

## 第2章 数式

### 2.1 数式モード

L<sup>A</sup>T<sub>E</sub>X は通常「段落モード」か「L Rモード」のどちらかで動作しています。前者は単語や文など、原稿を文章のかたまりとして取り扱い、行や段落、ページに分割していきます。後者は表などを組版するときを使用し、このモード内では改行は起こりません。これらの他に L<sup>A</sup>T<sub>E</sub>X ではもう一つの動作モードがあります。それが数式を扱うときに用いる「数式モード」です。この章ではこの「数式モード」についての説明を行います。

### 2.2 基本的な数式環境

L<sup>A</sup>T<sub>E</sub>X には様々なバリエーションの数式を出力することができるように、いくつかの数式環境が用意されています。ここでは比較的使用頻度の高いものについて説明します。

#### 2.2.1 math 環境

math 環境は本文中に数式を出力する環境です。たとえば次のように表示したい場合に使用します。

例) 二次関数  $y = x^2$  のグラフは放物線である

二次関数`\begin{math}y=x^2\end{math}`のグラフは放物線である

しかし本文中でいちいち math 環境を使用していると、文章のつながり等がわかりにくくなるので、通常は数式を `$と$` や、`¥(と ¥)` で囲んで使用します。

二次関数`$y=x^2$`のグラフは放物線である

二次関数`\(y=x^2\)`のグラフは放物線である

#### 2.2.2 displaymath 環境

displaymath 環境は、数式を別行立てで出力する環境です。別行立ての数式は行の中央に出力されます。またこれも math 環境と同様に `¥[と ¥]` で数式を囲めば displaymath 環境として出力されます。

例) 数式を displaymath 環境で出力してみる

$$y = x^2$$

数式を displaymath 環境で出力してみる`\begin{displaymath}y=x^2\end{displaymath}`

数式を displaymath 環境で出力してみる`\[y=x^2\]`

#### 2.2.3 equation 環境

equation 環境は基本的に displaymath 環境と同じですが、数式に参照用の番号が伴います。

例) 数式を equation 環境で出力してみる

$$y = ax^2 + bx + c \tag{2.1}$$

数式を equation 環境で出力してみる`\begin{equation}y=ax^2+bx+c\end{equation}`

## 2.2.4 eqnarray 環境

複数の連続する式に番号を付けたい場合は eqnarray 環境を使用します。もし番号を付けたくない場合は環境名を eqnarray\* としてください。

例) 数式を eqnarray 環境で出力してみる

$$y = ax^2 + bx + c \quad (2.2)$$

$$y = \sin x \quad (2.3)$$

$$y = \cos x \quad (2.4)$$

$$y = e^x \quad (2.5)$$

$$y = \log x \quad (2.6)$$

数式を eqnarray 環境で出力してみる

```
\begin{eqnarray}
y = ax^2+bx+c \\
y = \sin x \\
y = \cos x \\
y = e^x \\
y = \log x
\end{eqnarray}
```

複数の式を扱う場合は、各式の最後に ~~\\~~ の改行命令を書きます。これは eqnarray 環境では自動的に改行を行わないために、ユーザーが改行命令を用いて改行を行う必要があるからです。

出力結果を見てもわかる通り、数式が右揃えで表示されています。これを左揃えたい場合は、= を & 記号で囲みます。

例) 数式を eqnarray 環境で出力し、= で揃えてみる

$$y = ax^2 + bx + c \quad (2.7)$$

$$y = \sin x \quad (2.8)$$

$$y = \cos x \quad (2.9)$$

$$y = e^x \quad (2.10)$$

$$y = \log x \quad (2.11)$$

数式を eqnarray 環境で出力し、= で揃えてみる

```
\begin{eqnarray}
y &=& ax^2+bx+c \\
y &=& \sin x \\
y &=& \cos x \\
y &=& e^x \\
y &=& \log x
\end{eqnarray}
```

eqnarray 環境を使って、数式を途中で改行することもできます。次のような出力を得ることができます。

例)

$$z = xy \quad (2.12)$$

$$y = x^2 + xy - y^2 + x - y - 1 \quad (2.13)$$



```

\begin{eqnarray}
z &=& xy \\\
y &=& x^2+xy-y^2 \nonumber \\\
&& \& \{ \} +x-y-1
\end{eqnarray}

```

3 行目に”\nonumber”という記述があります。これは eqnarray 環境では各行に番号をつけてしまいがちですが、3 行目と 4 行目は単に改行されているだけでありひとつの式であるため、番号を付けられると不自然になります。そのためここでは数式番号の出力を抑制する命令である”\nonumber”を使用しています。

次に 4 行目に { } が記述されていることに注意してください。これは T<sub>E</sub>X は数式中の空白を自動的に調節するので、例えば”+1”と”a+1”では”+”のあとの空白が微妙に違います。そのため T<sub>E</sub>X にはっきりと単項演算子か二項演算子かを伝える必要があります。ここでは何も出力しない項として { } を付けることにより、T<sub>E</sub>X に二項演算子であることを伝えています。{ } を利用すれば  $ax_{n+1}^{ax^2+bx+c}$  といった出力も可能です。この表示は”\$ax\_{n+1}\{ \}^{\{ax^2+bx+c\}}\$”から得られます。

## 2.3 数式の参照

数式を文中で参照するには、”\label”命令と”\ref”命令を使用します。

例) (2.14) をオイラーの公式と呼ぶ

$$e^{ix} = \cos x + i \sin x \quad (2.14)$$

```

(\ref{first}) をオイラーの公式と呼ぶ
\begin{equation}
e^{ix} = \cos x + i \sin x \label{first}
\end{equation}

```

## 2.4 文字列の扱い

数式モードに記述された文字列は原則としてすべて数式と見なされ、数式用に用意された特殊なイタリック体で出力されます。例えば” $\textit{diff}(a)$ ”という出力を得たい場合に、普通に”\$diff(a)\$”と記述すると、それぞれ個別な文字と見なされ” $\textit{diff}(a)$ ”と出力されてしまいます。この時本文中のイタリック体と同じ出力を得たい場合は”\mathit{diff}(a)”と記述します。数式モード中で書体を変更する命令を次に挙げておきます。

Table 2.1 数式モード中で書体を変更するための命令

書体名	命令と使用例	出力
ローマン体	\mathrm{A B C a b c 1 2 3}	ABCabc123
ボールド体	\mathbf{A B C a b c 1 2 3}	<b>ABCabc123</b>
サンセリフ体	\mathsf{A B C a b c 1 2 3}	ABCabc123
イタリック体	\mathit{A B C a b c 1 2 3}	<i>ABCabc123</i>
タイプライタ体	\mathtt{A B C a b c 1 2 3}	ABCabc123
明朝体	\mathmc{A B a b 1 2 あ 亜}	ABab12 あ亜
ゴシック体	\mathgt{A B a b 1 2 あ 亜}	ABab12 あ亜

## 2.5 空白

数式モードでは、T<sub>E</sub>X や L<sup>A</sup>T<sub>E</sub>X の命令の区切りを示す以外の半角空白は無視されます。つまり、”x+1”、”x<sub>+</sub>+1”、”x<sub>++</sub>+1”は、すべて同じ出力となります。空白をあけるには次の命令を使用します。

Table 2.2 数式モード内での空白制御命令

命令	空白の種類	例
$\backslash$	半角の空白	a b
$\backslash$ quad	全角の空白	a b
$\backslash$ quad	2 個の全角空白	a b
$\backslash$ ,	細スペース ( 全角の 1/6 )	a b
$\backslash$ >	中スペース ( 全角の 2/9 )	a b
$\backslash$	大スペース ( 全角の 5/18 )	a b
$\backslash$ !	負の細スペース ( 全角の -1/6 )	a b

## 2.6 添え字

数式モードでは文字に添え字をつけることができます。下添え字は” $\sub$ ”記号、上添え字は” $\sup$ ”記号に続けて書きます。この 2 つは同時に指定することができます。また  $a_{j+2}^{i+5}$  のように、添え字が式のような複数の文字列からなる場合には” $\$a_{\{j+2\}}^{\{i+5\}}\$$ ”のように上添え字部と下添え字部を { } とでくります。

添え字にさらに添え字をつけたい場合、例えば  $a^{x^y}$  のように出力したい場合は” $\$a^{x^y}\$$ ”のように記述するとエラーとなります。これは「 $a$  の  $x^y$  乗、つまり  $a^{x^y}$ 」なのか、「 $a^x$  の  $y$  乗、つまり  $a^{xy}$ 」なのかははっきりと区別しなければならないからです。区別するためには、これも { } でまとまりを表します。

Table 2.3 添え字の例

出力	ソース
$a_j^i$	$\$a^i_j\$$
$a_j^i$	$\$a_j^i\$$
$a_{j+2}^{i+5}$	$\$a_{\{j+2\}}^{\{i+5\}}\$$
$a^{x^y}$	$\$a^{\{x^y\}}\$$
$a^{xy}$	$\$\{a^x\}^y\$$
$a^{xy}$	$\$a^{\{xy\}}\$$
$a^x y$	$\$a^{xy}\$$

## 2.7 分数

$\mathrm{T}_{\mathrm{E}}\mathrm{X}$  における分数の書き方について説明します。 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$  において分数を書くには  $\backslash$ frac 命令を使用します。第 1 引数に分子を第 2 引数に分母を指定し、書式は  $\backslash$ frac{第 1 引数}{第 2 引数} となります。

例)

$$y = \frac{1}{x+1}$$

```
\[
y = \frac{1}{x+1}
\]
```

文中で  $\backslash$ frac 命令を使い、分数を書くと次のように表示されてしまいます

例) 分数関数  $y = \frac{x}{x+1}$  について考えてみましょう

```
分数関数$y=\frac{x}{x+1}$について考えてみましょう
```

これでは分数が小さすぎて見にくいという場合は，“分子 / 分母”のようにするか，`\displaystyle` という命令を使います．`\displaystyle` 命令は，本文中の数式を別行立ての数式と同じスタイルで出力するための命令です．

例) 分数関数  $y = x/x + 1$  について考えてみましょう

分数関数 $y=x/x+1$ について考えてみましょう

例) 分数関数  $y = \frac{x}{x+1}$  について考えてみましょう

分数関数 $\displaystyle y=\frac{x}{x+1}$ について考えてみましょう

## 2.8 様々な数式

ここでは使用頻度の高そうな数式の例を挙げます．数式を書く際の参考にしてください．

$\int_a^b f(x) dx$	<code>\int_a^b f(x)\,dx</code>
$y = \cos \alpha + \sin \beta + \tan \gamma$	<code>y = \cos \alpha + \sin \beta + \tan \gamma</code>
$df = \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy$	<code>df = \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy</code>
$\int_0^1 f(x) dx = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{k=0}^{n-1} f\left(\frac{k}{n}\right)$	<code>\displaystyle \int_0^1 f(x)\,dx = \lim_{n \to \infty} \frac{1}{n} \sum_{k=0}^{n-1} f\left(\frac{k}{n}\right)</code>
$e^{\sqrt{-1} nx} = \cos nx + \sqrt{-1} \sin nx$	<code>e^{\sqrt{-1}\,nx} = \cos nx + \sqrt{-1}\,\sin nx</code>

## 第3章 BIB<sub>T</sub>E<sub>X</sub>

そろそろみなさんも研究テーマが決まり、論文を書く機会も増えてくると思います。そのとき役立つものとしてぜひ知っていただきたいものに BIB<sub>T</sub>E<sub>X</sub> というものがあります。そこで、この章では BIB<sub>T</sub>E<sub>X</sub> について解説していこうと思います。

### 3.1 BIB<sub>T</sub>E<sub>X</sub> について

まず、BIB<sub>T</sub>E<sub>X</sub> についてですが、そのまえに5月に行われた月例発表会のレジュメについて少し思い出してください。本文中に参考文献という欄を皆さん作られたと思います。このときは個々にきめられたテーマであったのでこの参考文献をもう一度使うことはないと思いますが、現在は研究テーマもきまって同じ参考文献を用いることが増えてくると思います。ところが毎回同じ参考文献を書くのは非常に面倒であり、効率が良いとはいえません。そのときに役立つのが BIB<sub>T</sub>E<sub>X</sub> です。BIB<sub>T</sub>E<sub>X</sub> は T<sub>E</sub>X の本文とは別に文献のデータベースファイルを作成し、そこから本文に引用するというものです。

### 3.2 BIB<sub>T</sub>E<sub>X</sub> の使用

それでは実際に BIB<sub>T</sub>E<sub>X</sub> をつかってみましょう。まずは文献データベースファイルの作成です。後ほど設定しますがマクロを用いて図 3.3 のメニューを用いて文献の形式の選択をします。この図を見ればわかると思いますが、文献によって使用する形式がちがいます。横に解説が書いてあるのでそれを見て、参考文献の形式を選んでください。例としてここでは「book」を選んでみます。すると、図 3.1 のような画面が表示されます。

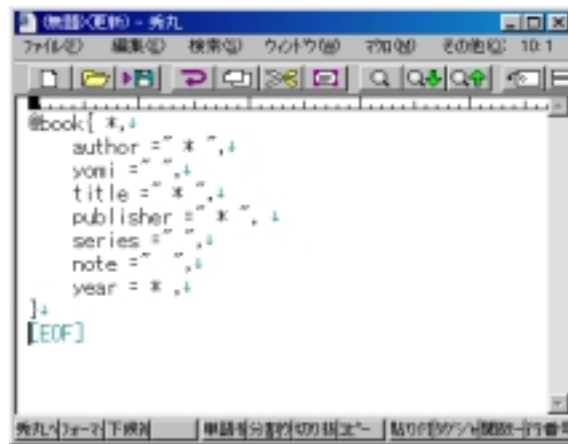


Fig. 3.1 データベースファイル作成

「book」についての属性を聞かれるのでここでそれぞれ入力してください。ただし「\*」がついているところは必ず書いてください。その他のところは書かなくてもかまいません。また属性の意味は後ほど説明します！「@book」の横の属性のない「\*」は参照名（ラベル）です。そして入力ができたら、拡張子を「.bib」にして保存してください。これでデータベースファイルの作成は終わりです。

それでは早速 bibtex を用いて T<sub>E</sub>X の文章を作ってみましょう。例として用いる文献データベースファイルを

```
@article{ pata,\nauthor = "魔夜峰央 ",\nyomi = " mayamineo",
```

```

title =" パタリロ",
journal ="花と夢コミックス ",
volume =" 1 巻",
series =" 白泉社文庫",
year = 1994 ,
}

```

のように設定し，ファイル名を「 patalliro.bib 」とします．実際の  $\text{T}_{\text{E}}\text{X}$  の文章中では

```

\documentclass[a4paper,10pt]{jarticle}
\pagestyle{plain}

%%%%%%%% TEXT START %%%%%%%%%

\begin{document}
パタリロはマリネラ王国の国王です．\cite{pata}

```

```

\bibliographystyle{jplain}
\bibliography{patalliro}
\end{document}

```

とします．このように，引用したい場所に `\cite{ ( 参照名 ) }` という書式で記述します．また，`\end{document}` の前に

```

\bibliographystyle{jplain}
\bibliography{patalliro}

```

と記述します．`\bibliography{ }` の中には文献データベースファイル名が入ります．もちろんここでは参照名は「 patalliro 」です．また拡張子の `.bib` は必要ありません．この状態で，いつも通りに文書をコンパイルすると図 3.2 のように出力されます．以上が， $\text{BIB}_{\text{T}_{\text{E}}\text{X}}$  を用いた参考文献の引用の概要です．



Fig. 3.2 出力結果

### 3.3 $\text{BIB}_{\text{T}_{\text{E}}\text{X}}$ マクロの設定

$\text{BIB}_{\text{T}_{\text{E}}\text{X}}$  の使用についてですが，まず文献のデータがないと意味がないので，データベースファイルの作成の方法を解説したいと思います．ところが，文献によって使用する設定の形式が異なっているうえに，形式の属性の量も多いため入力が少々面倒です．そこで，この面倒な入力を簡略化するために秀丸エディタ用のマクロが作成されていますので，まずこの  $\text{BIB}$  マクロをインストールしてください． $\text{BIB}$  マクロは

<http://mikilab.doshisha.ac.jp/dia/seminar/latex/doc/bib2.html>

からダウンロードできます。そしてこの bib0.mac を秀丸エディタのマクロフォルダにコピーしてください。標準の設定では、マクロフォルダは、

C:\Program Files\Hidemaru\macros

です。マクロが正しくインストールされたかどうかを確認するためにマクロを実行してみます。秀丸エディタを起動し、メニューのマクロ (M) - マクロ実行 (X) を選択します。すると、マクロ実行というウィンドウが表示されます。ここで、先ほどコピーした bib0.mac を選択し「OK」をクリックしてください。正しく動作すれば、図 3.3 ようなメニューが表示されます。

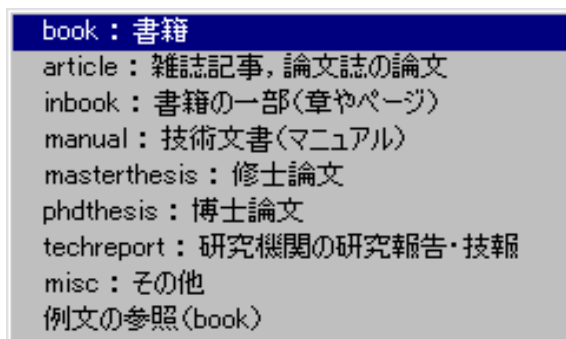


Fig. 3.3 bib マクロ形式のメニュー

以上で BIB<sub>T</sub>E<sub>X</sub> マクロの設定はおわりです。

### 3.4 属性の解説

属性についてのそれぞれの意味を以下の表にまとめてみました。参考文献データベースファイル作成のときの参考にしてください。

Table 3.1: 属性の説明

属性	解説
address	publisher (発行元) の住所
annotate	メモ, 注釈. 通常の文献スタイルでは無視されます.
author	文献の著者
booktitle	書籍の表題名
chapter	章や節などの番号
crossref	文献リスト中で相互参照される文献のラベル
edition	書籍の版数. 第 2 版や Second のように指定する
editor	編集者名
howpublished	出版形態が特殊な場合に, その出版形態を説明する
institution	技報などを出版した, 研究所などの名前
journal	雑誌名
key	author や editor が存在しない文献において 並べ替えを行うときに利用される文字列
month	刊行月
note	文献に対する注意事項
number	新聞, 雑誌などの号数.
organization	会議などを主催した団体や, 文書の編纂を行った団体
pages	ページ番号. "13" や "74-93", "256+" のように指定する.
publisher	発行元名. 出版社名.
school	論文が提出された学校名

Table 3.1: 属性の説明

series	シリーズ名
title	表題名
type	@techreport の種別
volume	巻数
year	刊行年．まだ刊行されていない場合は執筆された年．