

TeX 講習会テキスト

2020年9月版

明治大学

生田メディア支援事務室

目次

1	基本	4
1.1	TeX とは何か？	4
1.2	TeX を使うメリット	4
1.3	コンパイルと DVI、PDF の出力	5
1.4	タイプセット	5
1.5	レイアウト	6
1.6	コマンド	6
1.7	ドキュメントクラス	6
1.8	プリアンプル	6
1.9	環境	7
1.10	パッケージ	7
1.11	注意点	7
1.12	サンプルファイル	7
2	実習	8
2.1	はじめに	8
2.2	コメント	8
2.3	字下げ	9
2.4	改行	9
2.5	空白	10
2.6	改ページ	11
2.7	タイトル	11
2.8	見出し	12
2.9	箇条書き	13
2.10	左寄せ・中央揃え・右寄せ	14
2.10.1	左寄せ	14
2.10.2	中央揃え	14
2.10.3	右寄せ	14
2.11	表	15
2.11.1	表の作成	15
2.11.2	表の罫線	15
2.11.3	表の配置	16
2.12	画像	18
2.12.1	画像の挿入	18
2.12.2	画像の配置	19
2.13	数式	20
2.14	参照	21
2.15	脚注	22
2.16	文献	22
2.17	ファイルの読み込み	23
3	その他	24

参考文献	24
A 付録	25
A.1 特殊文字について	25
A.2 array 環境を用いた数式の挿入	25
A.2.1 行列	25
A.2.2 連立方程式	26
A.3 図を並べて表示する	26
A.4 参考文献の記述方法について	28

1 基本

1.1 TeX とは何か？

TeX は組版ソフトです。組版とは文字や図などをレイアウトに従って配置する作業を示します。TeX は組版の作業をコンピュータ上で行うことができます。

1.2 TeX を使うメリット

文章作成において TeX はいくつかのメリットがあります。TeX のメリットは以下の通りです。

- オープンソースなので、誰でも利用できる
- 様々な OS で利用できる
- 数式をとともきれいに書ける
- スタイルファイルを用いることで、学会等で指定されたレイアウトにすることが容易
- ファイル分割ができるため、文書が大きくなっても管理が楽

反対に、デメリットとしては、以下のものが挙げられます。

- コンパイルが必要なため、すぐに作成した文書を確認できるわけではない
- Word のように、Excel 等で作成したグラフをそのまま挿入することはできない

よく Word との比較がされますが、TeX では参考文献や数式の記述、定められたレイアウトでの文書作成においてとても有利なので、論文の作成には TeX をお勧めします。

1.3 コンパイルと DVI、PDF の出力

TeX はテキストデータである TeX ファイルをコンパイルすることで組版を作成することができます。組版は dvi ファイルです。dvi ファイルを pdf 形式に変換することも出来ます。実際に組版を作成する方法を示します。TeX のファイルはファイル名の拡張子を除いた部分をコンパイルに使用します。

コンパイルはコマンドプロンプトを使う例を示しますので、コンパイル対象の TeX ファイルのディレクトリにアクセスしてください。

TeX ファイルから dvi ファイルを作る場合は `platex` の後に半角スペースとファイル名をコマンドプロンプトに打ち込みます。

```
platex ファイル名
```

dvi ファイルから pdf を作る場合は `dvipdfmx` の後に半角スペースとファイル名をコマンドプロンプトに打ち込みます。

```
dvipdfmx ファイル名
```

1.4 タイプセット

TeX ではタイプセットと呼ばれるコマンドを用いることで tex ファイルから様々な設定を読み取り、直接 pdf を得ることができます。タイプセットには、英語のみに対応したものや、日本語でも使えるもの、参考文献の表示に特化したもの¹などがあり、使い分けることで文書作成の幅が広がります。基本的に、日本語文書の作成には、`ptex2pdf` と呼ばれるタイプセットを使用します。

TeX ファイルのコンパイルの流れをまとめると、以下図 1 のようになります。

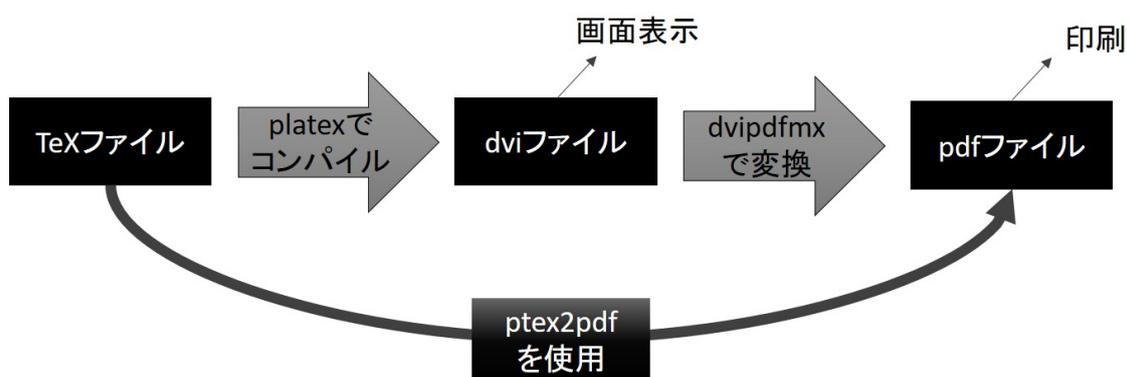


図 1: TeX ファイルのコンパイルの流れ

¹A.4 参照

1.5 レイアウト

TeX のレイアウトは `sty` ファイルというスタイルファイルを使って定義します。スタイルファイルは前述した通り外部で公開されている場合がありますので、今回はスタイルファイルの作成については省略します。また、スタイルファイルによって使える機能や定義も違いますので、個別のスタイルファイルに依存しない基本的な機能だけを講習会で触れます。今回は標準化されている `jarticle` というスタイルファイルを利用します。`jarticle` は論文を書くときに利用されます。また `jbook` というスタイルファイルもありますが、こちらは書籍を書くときに利用されます。今回は `jarticle` を使って勉強します。

1.6 コマンド

コマンドはコンピュータに指示する命令文のことです。TeX では先頭に `\` 文字を付けてから命令文の名前を記述します。フォントの種類により、`¥` と表示されることもありますが、同様に使用できます。本テキストでは、`\` で統一しています。

```
\命令文の名前
```

1.7 ドキュメントクラス

ドキュメントクラスは文章の種類を示します。文章は論文、レポートや本等がありますが何を書くのかを記述します。例えば論文を書くのであれば `\documentclass{jarticle}` と記述します。このように中括弧の中にはスタイルファイル名を記述します。

```
\documentclass{スタイルファイル名}
```

1.8 プリアンブル

プリアンブルとは序文、序言、前文、前置きという意味です。TeX では事前に機能や設定を使うための読み込みをします。読み込むために記述する領域のことです。具体的には、`\documentclass` から `\begin{document}` の間です。

```
\documentclass{文書クラスの内容}  
この領域がプリアンブルです。  
\begin{document}  
\end{document}
```

1.9 環境

環境とは TeX において文章の表現様式を示します。環境は `begin` コマンドと `end` コマンドを対にして使います。これらのコマンドは中括弧が付きます。中括弧の中には表現様式を示す環境名を記述します。`begin` コマンドから `end` コマンドの中だけ、指定した環境が適用されます。

```
\begin{環境名}  
この中だけ指定した環境が適用されます。  
\end{環境名}
```

1.10 パッケージ

パッケージとは荷物、小包、包装という意味です。TeX では機能を追加するためにパッケージと呼ばれるプログラムの集まりのことを示します。パッケージを読み込むことで拡張機能として使えます。パッケージを読み込むためには `usepackage` コマンドを使います。`usepackage` コマンドは後ろに中括弧が付きます。中括弧の後ろにはパッケージ名を記述します。また、`usepackage` コマンドはプリアンブルに記述します。

```
\usepackage{パッケージ名}
```

1.11 注意点

TeX を扱う前に特に注意して欲しいことがあります。文字の全角と半角についてです。TeX では、全角文字と半角文字を区別します。コマンドに使われている記号は、すべて半角文字になっているため、エラーが解決できない際は全角半角の間違いを疑ってみてください。

1.12 サンプルファイル

講習会を円滑に進行させるため、実習で使うサンプルファイルを用意しています。コンパイルでエラーを吐いて困ったときなどは、サンプルファイルを参考にして比較してみてください。

2 実習

2.1 はじめに

TeX の基本的な構造を記述してみましょう。基本の形は `documentclass` と `document` 環境を使います。 `document` 環境は本文を示します。その中には本文の内容を記述します。

```
\documentclass{jarticle}
\begin{document}
これから TeX を勉強します。まずは好きな様に文章を書いてください。書き終わったら TeX ファイルをコンパイルをしてください。コンパイルが終わったら文章が表示されるか確認してください。
\end{document}
```

出力結果は下記の様になります。

```
これから TeX を勉強します。まずは好きな様に文章を書いてください。書き終わったら TeX
ファイルをコンパイルをしてください。コンパイルが終わったら文章が表示されるか確認して
ください。
```

2.2 コメント

本文とは別に覚えとしてコメントを残すことがあります。TeX では `%` 文字を先頭に入力します。その後に入力した文字はコメントとしてコンピュータに判断されます。コメントは組版に反映されません。

```
\documentclass{jarticle}
\begin{document}
%こちらは "%" が付いているので内容は表示されません。
こちらは表示されます。コメント記号が付いている行との違いを確認してください。コメント記号はコンパイルの対象にはなりません。
\end{document}
```

出力結果は下記の様になります。

```
こちらは表示されます。コメント記号が付いている行との違いを確認してください。コメン
ト記号はコンパイルの対象にはなりません。
```

2.3 字下げ

TeX では段落ごとに字下げが自動的に行われます。

```
\documentclass{jarticle}
```

```
\begin{document}
```

TeX では文章を書き始めると自動で行頭の字下げを行います。そのため字下げを手動で行う必要はありません。

```
\end{document}
```

出力結果は下記の様になります。

TeX では文章を書き始めると自動で行頭の字下げを行います。そのため字下げを手動で行う必要はありません。

2.4 改行

改行をしたい場合は、改行を二度行う、もしくは、改行をしたい箇所で `par` コマンドで行う事ができます。これらの方法では、改行と字下げが同時に行われます。字下げをせずに改行のみ行う場合は、`\\`コマンドを使用します。

```
\documentclass{jarticle}
```

```
\begin{document}
```

プログラミングはコンピュータにプログラムを作成することにより、人間の意図した処理をコンピュータに命令する行為である。

プログラミングの過程で、ソースコードを記述することをコーティングという。`\par` ソースコードをコンピュータが理解できるように変換することを、コンパイルという。`\\TeX` ファイルもコンパイルの必要がある。

コンパイルが完了すると、`dvi` という拡張子の組版ファイルが作成される。

```
\end{document}
```

出力結果は下記の様になります。

プログラミングはコンピュータにプログラムを作成することにより、人間の意図した処理をコンピュータに命令する行為である。プログラミングの過程で、ソースコードを記述することをコーティングという。

ソースコードをコンピュータが理解できるように変換することを、コンパイルという。`TeX` ファイルもコンパイルの必要がある。

コンパイルが完了すると、`dvi` という拡張子の組版ファイルが作成される。

2.5 空白

TeX では半角スペース、tab は複数入力しても一つ分として出力されてしまいます。強制的にスペースを挿入したい場合、全角スペースを用いるか、`hspace` コマンドを用います。`hspace` コマンドは後ろに中括弧がつき、その中で空白の幅を指定することができます。数値には負の値を用いることも可能です。

縦方向の空白を作りたい場合は、`vspace` コマンドを用います。使い方は `hspace` コマンドと同様です。ただし、`vspace` コマンドは文書の中で行が変わった際に実行されるため、下記の例のようにコマンドを記述した位置で空白ができるわけではない点に注意してください。「`\vspace \`」のように記述することで、イメージした位置に空白を配置できると思います。

`hspace`、`vspace` コマンドで用いることができる単位には以下のようなものがあります。

表 1: 長さの単位一覧

単位	コマンド	単位	コマンド
センチメートル	cm	漢字一文字と同じ幅	zw
ミリメートル	mm	横幅いっぱいの長さ	<code>\hsize</code>
インチ	in	行(列)が埋まるだけの長さ	<code>\fill</code>
ポイント	pt		

```
\documentclass{jarticle}
\begin{document}
プログラミングはコンピュータにプログラムを作成することにより、人間の意図した処理をコンピュータに命令する行為である。
プログラミングの過程で、ソースコードを記述することをコーディングという。\\
\hspace{1zw}ソースコードをコンピュータが理解できるように変換することを、コンパイルという。TeX ファイルもコンパイルの必要がある。
\vspace{2cm}
コンパイルが完了すると、dvi という拡張子の組版ファイルが作成される。
\end{document}
```

出力結果は下記の様になります。

プログラミングはコンピュータにプログラムを作成することにより、人間の意図した処理をコンピュータに命令する行為である。プログラミングの過程で、ソースコードを記述することをコーディングという。

ソースコードをコンピュータが理解できるように変換することを、コンパイルという。TeX ファイルもコンパイルの必要がある。コンパイルが完了すると、dvi という拡張子の組版フ

イルが作成される。

2.6 改ページ

文書作成中に改ページをしたい場合、`newpage` もしくは `clearpage` コマンドで行うことができます。`newpage` では、コマンドが挿入された場所で強制的に改ページします。`clearpage` では、未出力の図表を表示してから改ページします。

```
\documentclass{jarticle}
\begin{document}
ここは 1 ページ目です。
\newpage
ここから 2 ページ目です。
\end{document}
```

出力結果は下記のようになります。

ここは 1 ページ目です。

1

ここから 2 ページ目です。

2

2.7 タイトル

タイトルを TeX で付けるためには、`title` コマンドを使います。`title` コマンドは後ろに中括弧が付きますがその中にタイトルを書きます。

また、同様に著者名と日付も書くことが出来ます。`author` コマンドと `date` コマンドを使います。それぞれ中括弧の中に著者名と日付を書きます。

```
\documentclass{jarticle}
\begin{document}
\title{TeX}
\author{ドナルド・クヌース}
\date{1978 年}
\maketitle
\end{document}
```

出力結果は下記のようになります。

TeX

ドナルド・クヌース

1978 年

2.8 見出し

文章の構造を記述するために見出しを使うことがあります。見出しには以下のような種類があります。

表 2: 見出し一覧

見出し	コマンド	見出し	コマンド
部	part	小々節	subsubsection
節	section	段落	paragraph
小節	subsection	小段落	subparagraph

それぞれ中括弧が付きますが、その中に各見出しの内容を書きます。段落見出しと小段落見出しは見た目上区別は付きません。

```
\documentclass{jarticle}
\begin{document}
\part{日本}
\section{神奈川県}
\subsection{川崎市}
\subsubsection{多摩区}
\paragraph{東三田}
\subparagraph{中央校舎}
\end{document}
```

出力結果は下記のようになります。

```
第 I 部
日本

1 神奈川県

1.1 川崎市

1.1.1 多摩区

東三田

中央校舎
```

2.9 箇条書き

箇条書きしたい場合には `itemize` 環境を使います。 `item` コマンドの後に内容を書くことで箇条書きができます。

```
\documentclass{jarticle}
\begin{document}
日本三大都市を以下に列挙する。
\begin{itemize}
\item 東京
\item 名古屋
\item 大阪
\end{itemize}
\end{document}
```

出力結果は下記の様になります。

日本三大都市を以下に列挙する。

- 東京
- 名古屋
- 大阪

また、番号付きの箇条書きを用いる場合、 `enumerate` 環境を使います。 `itemize` 環境と同様に、 `item` コマンドの後に内容を書くことで箇条書きができます。

```
\documentclass{jarticle}
\begin{document}
日本三大都市を以下に列挙する。
\begin{enumerate}
\item 東京
\item 名古屋
\item 大阪
\end{enumerate}
\end{document}
```

出力結果は下記の様になります。

日本三大都市を以下に列挙する。

1. 東京
2. 名古屋
3. 大阪

2.10 左寄せ・中央揃え・右寄せ

文や画像等の配置について説明します。

2.10.1 左寄せ

左寄せは `flushleft` 環境を使います。その中に記述した内容は左寄せされます。

```
\documentclass{jarticle}
\begin{document}
\begin{flushleft}
今日の天気は快晴である。
\end{flushleft}
\end{document}
```

出力結果は下記の様になります。

```
今日の天気は快晴である。
```

2.10.2 中央揃え

中央揃えは `center` 環境を使います。その中に記述した内容は中央揃えにされます。

```
\documentclass{jarticle}
\begin{document}
\begin{center}
今日の天気は快晴である。
\end{center}
\end{document}
```

出力結果は下記の様になります。

```
今日の天気は快晴である。
```

2.10.3 右寄せ

右寄せは `flushright` 環境を使います。その中に記述した内容は右寄せされます。

```
\documentclass{jarticle}
\begin{document}
\begin{flushright}
今日の天気は快晴である。
\end{flushright}
\end{document}
```

出力結果は下記の様になります。

```
今日の天気は快晴である。
```

2.11 表

ここでは表について説明します。

2.11.1 表の作成

TeX で表を作成するには `tabular` 環境を使います。 `tabular` 環境は中括弧が付きます。中括弧の中には列指定をします。列は3つ指定すれば3列になります。指定方法は `l` と `c` と `r` のいずれかを使います。それぞれ各列を左寄せ、中央、右寄せを示します。例えば `cccc` であれば列は4つ作成して全ての列が中央寄せになります。各行は `\` 文字を2個使って定義します。また、各行列の値は文字と文字の間に `&` 記号を挟むことで値を設定します。

```
\documentclass{jarticle}
\begin{document}
2014年の求人倍率の推移を以下に示す。\\
\begin{tabular}{cccc}
都道府県 & 5月 & 6月 & 7月 \\
北海道 & 0.87 & 0.87 & 0.87 \\
東京都 & 1.55 & 1.57 & 1.60 \\
\end{tabular}
\end{document}
```

出力結果は下記の様になります。

2014年の求人倍率の推移を以下に示す。			
都道府県	5月	6月	7月
北海道	0.87	0.87	0.87
東京都	1.55	1.57	1.60

2.11.2 表の罫線

表の罫線は、縦罫線を `|` (バーティカルバー) で、横罫線を `hline` コマンドで記述します。縦罫線は `tabular` 環境の列指定で引きたい箇所に `|` (バーティカルバー) を記述します。 `|` (バーティカルバー) は、「BackSpace キーの左となりのキー + Shift キー」で入力することができます。横罫線は `tabular` 環境の中で引きたい箇所に `hline` コマンドを記述します。罫線は、 `|` (バーティカルバー) やコマンドを入力した数だけ記述されます。二重線を用いたい場合は、コマンドを連続で二つ記述しましょう。

```

\documentclass{jarticle}
\begin{document}
2014年の求人倍率の推移を以下に示す。\\
\begin{tabular}{|c|c|c|c|}
\hline
都道府県 & 5月 & 6月 & 7月 \\
\hline
北海道 & 0.87 & 0.87 & 0.87 \\
\hline
東京都 & 1.55 & 1.57 & 1.60 \\
\hline
\end{tabular}
\end{document}

```

出力結果は下記のようになります。

2014年の求人倍率の推移を以下に示す。

都道府県	5月	6月	7月
北海道	0.87	0.87	0.87
東京都	1.55	1.57	1.60

2.11.3 表の配置

表を配置するために、`table` 環境を使います。`table` 環境では表の位置指定や表タイトルの作成を行います。表番号も自動的につきます。`table` 環境には角括弧が付きますが、その中に位置を示すオプションを指定します。以下に設定できる配置のオプションを示します。

表 3: 利用可能な配置オプション

オプション	位置	オプション	位置
t	ページ上端	h	できるだけ記述した位置
b	ページ下端	H	必ず記述した位置
p	独立したページ		

配置オプションは複数指定することもできます。`[htp]`とした場合、`[h] → [t] → [p]`の順で図を配置する場所を確保しようとします。`[h]`の位置に図が置けない場合、`[t]`に配置しようとし、それも失敗すると、次は`[p]`の位置に配置しようとする…という動作になります。

タイトルは `caption` コマンドの後ろに、中括弧をつけて記述します。表を中央に表示するには、`centering` コマンドを用います。

```

\documentclass{jarticle}
\begin{document}
\begin{table}[h]
\caption{求人倍率の推移 (2014 年)}
\centering
\begin{tabular}{|c|c|c|c|}
\hline
都道府県 & 5 月 & 6 月 & 7 月 \\
\hline
北海道 & 0.87 & 0.87 & 0.87 \\
\hline
東京都 & 1.55 & 1.57 & 1.60 \\
\hline
\end{tabular}
\end{table}
\end{document}

```

出力結果は下記の様になります。

表 1: 求人倍率の推移 (2014 年)

都道府県	5 月	6 月	7 月
北海道	0.87	0.87	0.87
東京都	1.55	1.57	1.60

2.12 画像

ここでは画像の取り扱いについて説明します。

2.12.1 画像の挿入

TeX で画像を挿入するためには `includegraphics` コマンドを使います。挿入する画像形式は EPS です。EPS という画像形式は拡大や縮小に強く TeX でよく使われます。`includegraphics` コマンドは角括弧と中括弧が付きます。角括弧の中には画像の表示サイズに関するオプションを設定します。例えば、`scale=0.5` と指定すれば画像を 50% に縮小して表示します。中括弧の中には挿入したい画像のパスを指定します。拡大・縮小の比を指定する以外に、画像の幅や高さを数値で指定することもできます。`width=10cm` と指定すれば、10cm の幅で画像を表示できます。`height=10cm` とすれば、10cm の高さで画像を表示できます。`width` と `height` はカンマで区切って併用ができますが、画像の縦横比が変化してしまう可能性があることに注意してください。長さの単位として、cm のほかにも mm や in (インチ) 等も使用できます。興味があれば調べてみてください。

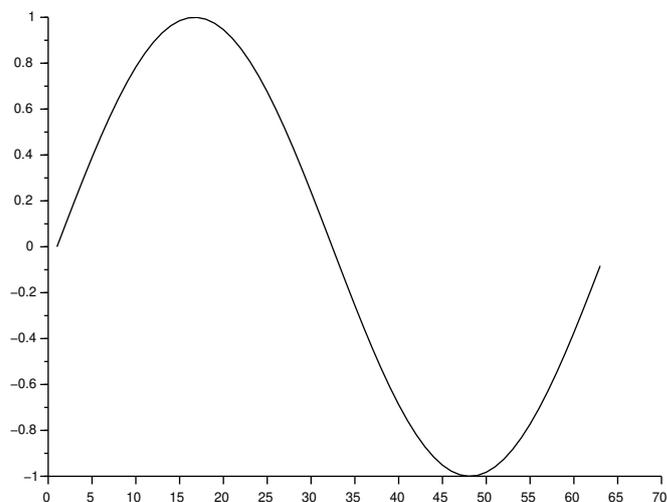
なお、`includegraphics` コマンドを使うためには `graphicx` パッケージを使用します。使用するには `\usepackage[dvipdfm]{graphicx}` を指定してください。ここではパッケージの詳細の説明は省きます。

```
\documentclass{jarticle}
\usepackage[dvipdfm]{graphicx}
\begin{document}
画像を挿入します。

\includegraphics[scale=0.5]{./sample.eps}
\end{document}
```

出力結果は下記のようになります。

画像を挿入します。



2.12.2 画像の配置

図の配置や、図タイトルをつけるためには、`figure` 環境を使います。基本的な使い方は、2.11.3 表の配置で説明した `table` 環境と同様です。角括弧の中に位置オプションを指定します。`centering` コマンドは表示内容を中央に揃えます。

```
\documentclass{jarticle}
\usepackage[dvipdfmx]{graphicx}
\begin{document}
画像を図として挿入します。
\begin{figure}[t]
\centering
\includegraphics[width=9cm]{./sample.eps}
\caption{正弦のグラフ}
\end{figure}
\end{document}
```

出力結果は下記の様になります。画像の位置オプションを「t」としているため、文章の上に図が表示されています。

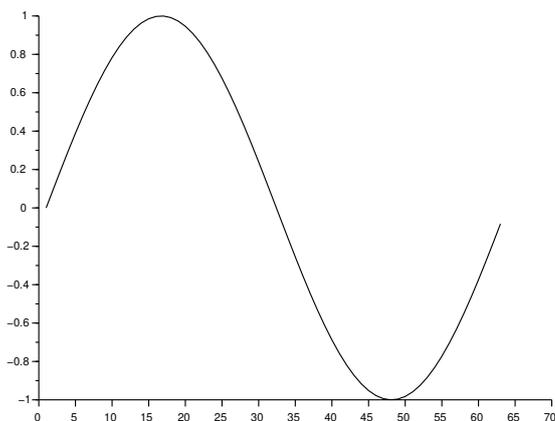


図 1: 正弦のグラフ

画像を図として挿入します。

2.13 数式

数式を挿入したい場合、TeX では `equation` 環境を使います。`equation` 環境に数式を記述すると、数式が別行の中心に数式番号付きで挿入されます。

数式が複数行にわたる場合、`eqnarray` 環境を用いて改行のコマンド`\\`で表示できます。数式間で縦の位置を合わせたい箇所を`&` `&`で囲むことで、きれいに表示することができます。一部数式番号を表示したくない場合は、`nonumber` コマンドを入力することで実現できます。また、環境名を `eqnarray*` とすることで、環境内に記述したすべての数式の番号を非表示にすることができます。

数式を別行ではなく文章中に表示させたい場合、数式を`$` `$`もしくは`\` `\`で囲みます。

```
\documentclass{jarticle}
\begin{document}
二次方程式 $3x^2-12x-15=0$ を解く。
\begin{eqnarray}
3x^2-12x-15 &=& 3(x^2-4x-5) \ \text{\nonumber} \\
&=& 3(x-5)(x+1)
\end{eqnarray}
よって、
\begin{eqnarray*}
3(x-5)(x+1)=0
\end{eqnarray*}
したがって、答えは
\begin{equation}
x=5,-1
\end{equation}
となる。
\end{document}
```

出力結果は下記の様になります。

二次方程式 $3x^2 - 12x - 15 = 0$ を解く。

$$\begin{aligned} 3x^2 - 12x - 15 &= 3(x^2 - 4x - 5) \\ &= 3(x - 5)(x + 1) \end{aligned} \tag{1}$$

よって、

$$3(x - 5)(x + 1) = 0$$

したがって、答えは

$$x = 5, -1 \tag{2}$$

となる。

2.14 参照

参照に関して、2.12.2 を例に説明をします。2.12.2 の図番号を本文中から参照することを考えます。

まず参照したい対象に、ラベルを設定します。ラベルは参照したい番号を出力する内容の後ろに `label` コマンドを使います。`label` コマンドは中括弧が付きます。中括弧の中に対象を表す一意の名前を記述します。また、ラベルの定義は見た目では表示されません。

`label` コマンドでラベル付けた図を本文中で参照するには、`ref` コマンドを使います。`ref` コマンドは中括弧が付きます。中括弧の中に参照する対象のラベルを記述します。参照する対象が示す番号が表示されます。

表や数式、章番号であっても、同様の操作でその番号を参照することができます。

```
\documentclass{jarticle}
\usepackage[dvipdfmx]{graphicx}
\begin{document}
画像を図\ref{sin}に示す。
\begin{figure}[h]
\centering
\includegraphics[scale=0.5]{./sample.eps}
\caption{正弦のグラフ}
\label{sin}
\end{figure}
\end{document}
```

出力結果は下記のようになります。

画像を図 1 に示す。

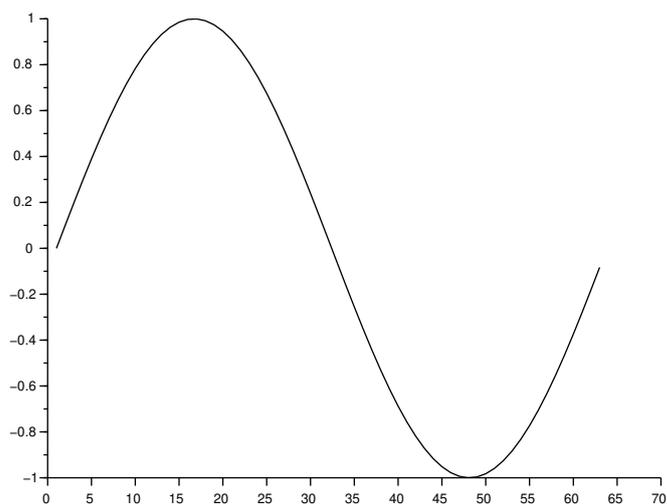


図 1: 正弦のグラフ

2.15 脚注

脚注を付けるには脚注を付ける対象の後ろに `footnote` コマンドを使います。 `footnote` は後ろに中括弧が付きますが、その中に脚注の内容を書きます。脚注番号の形式は、 `\renewcommand` で `\thefootnote` の定義を変更することで変えることができます。 `{\renewcommand\thefootnote{\roman{footnote}}` とすることでローマ数字に、 `{\renewcommand\thefootnote{\alph{footnote}}` とすることでアルファベットに変更できます。

```
\documentclass{jarticle}
\begin{document}
\renewcommand\thefootnote{\alph{footnote}}
スヌーピー\footnote{アメリカの漫画家、チャールズ・モンロー・シュルツが1950年から書き始めた漫画『ピーナッツ』に登場するビーグル犬のこと}はいつもかわいい。
\end{document}
```

出力結果は下記の様になります。

```
スヌーピーaはいつもかわいい。
aアメリカの漫画家、チャールズ・モンロー・シュルツが1950年から書き始めた漫画『ピーナッツ』に登場するビーグル犬のこと
```

2.16 文献

TeX で参考文献の一覧を作成するには、 `thebibliography` 環境を使います。中括弧内の数字は、文献が9件以下であれば9、99件以下であれば99と書きます。

文献情報は一件ずつ `bibitem` コマンドに続いて記述します。中括弧には本文中での参照名を記述します。ここで記述した順に文献番号が振られるため、順番に注意してください。

本文中で文献を参照する場合、 `cite` コマンドを用いて、 `bibitem` で設定した参照名を呼び出します。なお、 `cite` コマンドを使うためには `cite` パッケージを使用します。使用するには `\usepackage{cite}` を指定してください。

```
\documentclass{jarticle}
\usepackage{cite}
\begin{document}
C++設計者による書籍\cite{cpplang}が発行されている。
\begin{thebibliography}{9}
\bibitem{cpplang} Bjarne Stroustrup. "The C++ Programming Language". Addison-Wesley, 1985.
\end{thebibliography}
\end{document}
```

出力結果は下記の様になります。

C++設計者による書籍 [1] が発行されている。

参考文献

[1] Bjarne Stroustrup. "The C++ Programming Language". Addison-Wesley, 1985.

2.17 ファイルの読み込み

TeX では他のファイルを読み込むことができます。例えば、長い文書をいくつかのファイルに分けて作成しそれらを読み込むことによって、文書の管理が容易になり、章や節の入れ替え等の作業を楽に行うことができます。

ファイルを読み込むには `input` コマンドを使います。input コマンドは中括弧が付きます。中括弧の中には読み込むファイル名を記述します。例えば、客観の定義と主観の定義をそれぞれ別のファイルに分けて記述するとします。各ファイルはその内容を記述します。そしてメインの文章から読み込みます。

```
\documentclass{jarticle}
\begin{document}
\input{./sample_child_01.tex}
\input{./sample_child_02.tex}
\end{document}
```

例えば、客観の定義について書いた `sample_child_01.tex` というファイルを下記の様に記述し、

`sample_child_01.tex`

客観とは第三者の立場から観察し考えること。

主観の定義について書いた `sample_child_02.tex` というファイルは下記の様に記述すると、

`sample_child_02.tex`

主観とはそのひとりのものの見方である。

出力結果は下記の様になります。今回用いている例では、すべてのファイルが同じフォルダにある前提ですが、ない場合でも tex ファイルが存在するフォルダのパスを `input` コマンドで指定することで、同様に参照することが可能です。

客観とは第三者の立場から観察し考えること。 主観とはそのひとりのものの見方である。

3 その他

今後勉強する上で詳細な情報が欲しい場合は書籍 [1][2] やインターネットに情報があります。下の参考文献に記載してあるものはどれもおすすめです、わからないことがあった場合はぜひ参照してみてください。

参考文献

- [1] 豊沢聡. LATEX の精義—フリーの文書作成ツールを使って論文を書く. カットシステム, 2009.
- [2] 奥村晴彦, 黒木裕介. [改訂第 7 版] LaTeX2 ϵ 美文書作成入門. 技術評論社, 2017.
- [3] LaTeX コマンド集. <http://www.latex-cmd.com/>. 2020/07/22 アクセス.
- [4] TeX Wiki. <https://texwiki.texjp.org/>. 2020/07/22 アクセス.

A 付録

ここでは、TeX を使う上で応用と言える内容について記載します。

A.1 特殊文字について

TeX には特殊文字と呼ばれる記号があります。具体的には、& や { 等でコマンドに用いられるものが多いです²。これらの記号を、文中にそのまま表示したい場合、直前に \ をつけて入力することで実現できます。中には、直前に \ をつけても出力できない文字もあります。その場合、\verb コマンドを用います。 \verb コマンドの後ろに、出力したい文字を | (バーティカルバー) で挟んで入力することで、表示できます。

```
\{ \& \% \_ \verb|\| \verb|^|
```

出力結果は下記ようになります。

```
{ & % - \ ^
```

A.2 array 環境を用いた数式の挿入

A.2.1 行列

ここでは、行列の出力をします。出力には、array 環境を用います。基本的な使い方は、2.11 で紹介した tabular 環境と同様ですが、こちらは表ではなく、数式の描画に用いられます。中括弧が付き、l、c、r を用いて列数の指定をすることができます。

次に、行列をくくる括弧についてです。括弧を挿入したい範囲を \left と \right コマンドで挟みます。各コマンドの直後に括弧を入力することで、自動的に大きさが調整された括弧が出力されます。

```
\begin{equation}
A = \left(
  \begin{array}{lcr}
    -a & b & c \\
    d & -e & -f \\
    -g & h & -i
  \end{array}
\right)
\end{equation}
```

出力結果は下記ようになります。

$$A = \left(\begin{array}{lcr} -a & b & c \\ d & -e & -f \\ -g & h & -i \end{array} \right) \quad (1)$$

²特殊文字にはさまざまなものがあります。以下のリンクに一覧があります「<http://www.latex-cmd.com/special/special.html>」。

A.2.2 連立方程式

続いて、array 環境を用いて連立方程式を出力する方法を説明します。連立方程式では、列が一つと考え、array 環境の中括弧の中は「1」一文字だけです。式左側に中括弧を入力したいですが、中括弧は前述の特殊文字です。そのため、\{と入力する必要があります。また、式の右側には括弧を表示させる必要がありませんが、\right コマンドを省くとエラーが出ます。\\left と \\right はセットで用いなければならない決まりがあります。そこで、\\right コマンドの後ろにドットを入力します。\\right. と記述することで、右側に何も表示されなくなります。

```
\begin{equation}
\left\{
\begin{array}{l}
2x + 4y = 10 \\
x + 3y = 6
\end{array}
\right.
\end{equation}
```

出力結果は下記のようにになります。

$$\left\{ \begin{array}{l} 2x + 4y = 10 \\ x + 3y = 6 \end{array} \right. \quad (1)$$

A.3 図を並べて表示する

ここでは、図を並べて表示する方法を説明します。図を左右に並べるには、minipage 環境を使います。\\columnwidth は版面の幅を表し、0.4\\columnwidth は版面の 0.4 倍の幅を意味します。版面の幅の 0.4 倍の minipage を作成して、左右に並べています。

また、2.12.1 画像の挿入で説明した、includegraphics コマンドを使用しているため、graphicx パッケージが必要です。\\usepackage[dvipdfm]{graphicx}を指定してください。

```
\begin{figure}[h]
\centering
\begin{minipage}{0.4\columnwidth}
\includegraphics[width=\columnwidth]{sample.eps}
\caption{左の図}
\end{minipage}
\begin{minipage}{0.4\columnwidth}
\includegraphics[width=\columnwidth]{sample.eps}
\caption{右の図}
\end{minipage}
\end{figure}
```

出力結果は下記のようにになります。

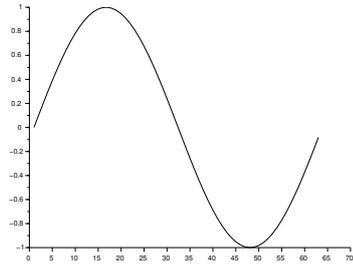


図 1:左の図

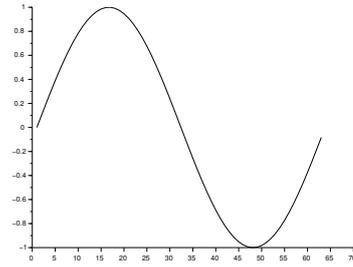
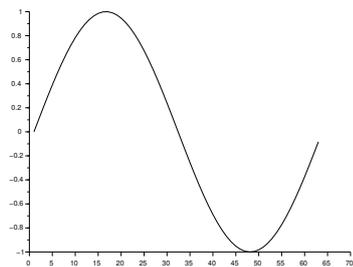


図 2:右の図

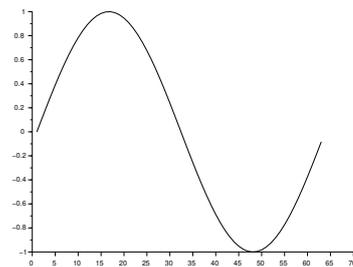
上記の例では、図番号を別々にしましたが、関連のある図を一つの図番号にまとめて、(a)、(b)、…とラベリングすることも可能です。その場合、subcaption パッケージが必要になるため、プリアンブルに `\usepackage{subcaption}` と追加してください。

```
\begin{figure}[h]
\centering
\begin{minipage}{0.4\columnwidth}
\includegraphics[width=\columnwidth]{サンプル.eps}
\subcaption{左の図}
\end{minipage}
\begin{minipage}{0.4\columnwidth}
\includegraphics[width=\columnwidth]{サンプル.eps}
\subcaption{右の図}
\end{minipage}
\caption{図を並べる}
\end{figure}
```

出力結果は下記ようになります。



(a) 左の図



(b) 右の図

図 1:図を並べる

A.4 参考文献の記述方法について

参考文献の記述方法について、文献が多く管理が大変な場合、文献情報を外部データとして作成しておく方法があります。外部データは bib ファイルと呼ばれるファイルに必要な情報を記述します。ここでは、sample.bib というファイルを新規作成します。例えば書籍情報は@book と記述し中括弧を加えます。中括弧の中には参照名を記述した後に、著者名、タイトル、出版社、出版年をそれぞれカンマで区切って記述します。イコールの左辺は上記種類を示し右辺で具体的な内容を記述します。

```
sample.bib

@book{cpplang,
author = "Bjarne Stroustrup",
title = "The C++ Programming Language",
publisher = "Addison-Wesley",
year = 1985
}
```

tex ファイルに移ります。文献を読み込むためには bibliographystyle コマンドを使います。bibliographystyle コマンドは中括弧が付きます。中括弧の中には文献スタイルを記述します。文献スタイルにより表示が変わります。

次に bibliography コマンドを使います。bibliography コマンドは中括弧が付きます。中括弧の中には文献の外部データのファイルパスを記述します。

実際に参照されると文献情報として表示されます。

```
sample.tex

C++設計者による書籍\cite{cpplang}が発行されている。
\bibliographystyle{jplain}
\bibliography{./sample.bib}
```

続いて、コンパイルの方法について説明します。bib ファイルを用いる場合、コンパイルを特殊な手順で実行しなければなりません。TeXworks からコンパイルする場合、タイプセットの設定が必要になります。TeXworks の「編集」タブから、「設定」を開き、「タイプセット」タブを選択すると図 1 のような画面が出ます。「タイプセットの方法」欄の「+」ボタンを押して、図 2 のように入力します。以上の操作で、図 3 のように、タイプセットを設定できるようになります。

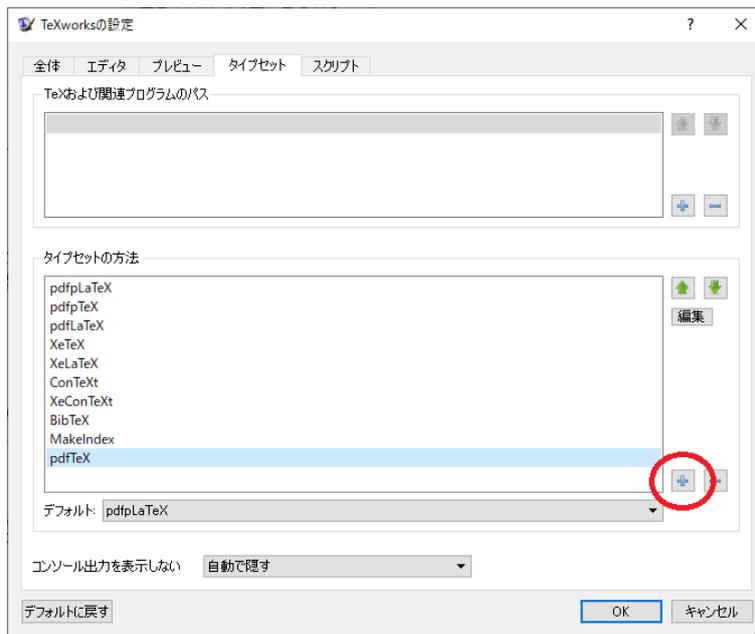


図 1: タイプセットの設定 1

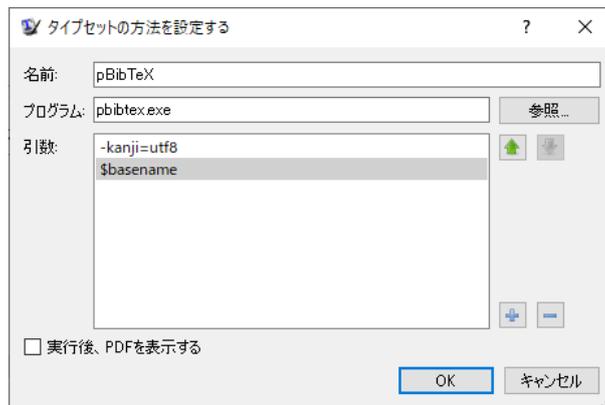


図 2: タイプセットの設定 2

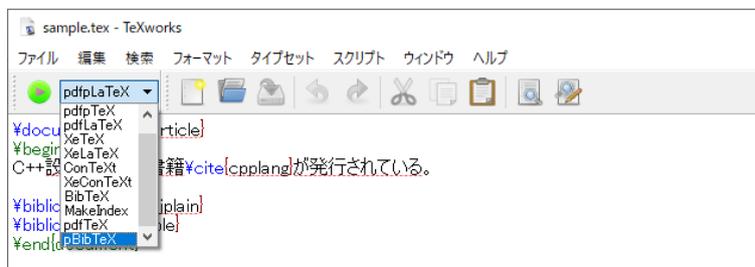


図 3: タイプセットの設定 3

以上の設定が完了したら、あとは以下の手順で sample.tex をタイプセットを変更しながらコンパイルします。

1. 「ptex2pdf」で実行

この手順により、sample.aux というファイルが作成され、`\cite` コマンドで引用されているキーワードや、`\bibliographystyle` コマンドで指定されている BibTeX スタイルファイル名、`\bibliography` コマンドで指定されている BibTeX データベースファイル名等の情報が保存されます。

2. 「pBibTeX」で実行

ここで、先の手順で作成された sample.aux ファイルを pBibTeX が読み込み、参照されている文献のデータを抜粋して、指定されたスタイルの参考文献を sample.bbl に作成します。

3. 「ptex2pdf」で実行

ここで、先ほど作成された sample.bbl が読み込まれ、参考文献が表示されます。

4. 「ptex2pdf」で実行

ここで、表示された参考文献から、引用箇所が処理されて本文中に文献番号が表示されます。

最終的に、以下の様に出力されます。

C++設計者による書籍 [1] が発行されている。

参考文献

[1] Bjarne Stroustrup. "The C++ Programming Language". Addison-Wesley, 1985.