

1 はじめに

情報リテラシの時間にパワーポイント、ワード等の使い方を学んだと思います。2年生からは情報工学実験 I や論理回路など、レポートを書く機会が多くなります。今日の授業では、レポートを書く際に気をつけてほしいこと、それを守るために、ワードを使うのではなく、 $\text{T}_{\text{E}}\text{X}$ を使う方法を説明したいと思います。

2 レポートを書くとき気をつけてほしいこと

レポートで最も大事なものは、その報告書の中で報告する内容ですが、それ以前に「体裁」が整っていないと、そもそもレポートを読んでももらえないことがあります。レポート、論文を書く際に守ってほしい最低限の体裁を以下に示します。

1. 段落のはじめには、全角 1 字の字下げを行う。
2. フォントの大きさを統一する。
特に節や副節、図の注釈などは狂いやすいので気をつけること
3. 文章中の余白を統一する。段落中は均等割し、右端はそろえること。
4. 図、表番号は相互参照を用いて、間違った番号を引かないこと。
5. ページ番号を付与する

ワードを使ってレポートを書く際にも、気をつけていけば以上のことは守ることができますが、「あとで直せばいいや」と思ってしまうと、途端に体裁の狂ったレポートができあがってしまいます。なかなか、内容を書ききった後に体裁を見直す、というのは忘れてしまいがちです。このような状況を防ぐために、必要な文章のみを記述し、その他の体裁はソフトに任せる方法（マークアップ言語）による方法があります。Web ページの例では XML と XSL がその方式に対応しますが、レポートを書く際には $\text{T}_{\text{E}}\text{X}$ を用いるのが一般的です。今日の授業では $\text{T}_{\text{E}}\text{X}$ の一種である $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$ を使ってレポートを書く方法を学んでもらいます。 $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$ はフリーソフトですので、無料で使うことができます。

3 $\text{pL}_{\text{A}}\text{T}_{\text{E}}\text{X}$ を使うための準備

図書館にある $\text{T}_{\text{E}}\text{X}$ についての本には、たいていインストーラが付属しておりそれを用いてインストールもできますが、ネットでダウンロードした最新版も簡単にインストール

できるようになったので、その方法について説明します。

3.1 pL^AT_EX のインストール (必須)

まずは T_EX インストーラ 3 を以下のサイトからダウンロードしてください。

<http://www.math.sci.hokudai.ac.jp/~abenori/soft/abtexinst.html>

2014 年 2 月 5 日現在の最新版は、「2013/07/25 : 最新版 (0.84r6)」です。ダウンロードし、zip ファイルを解凍した後は、`abtexinst.exe` を実行し、デフォルトで使いやすい設定になるようになっていますので、こだわりがなければ一通り目を通しながらインストールを続行してください。インストールが終われば Windows の再起動が必要です。

3.2 WinShell のインストール (推奨 1)

TeX インストーラを用いれば、TeX をインストールすることができますが、TeX ファイルを記述する際には、専用のソフトウェアを使うと便利です。もちろん、メモ帳や TeraPad といった単純なテキストエディタを用いて記述することもできますが、ここでは、WinShell と Eclipse(TeXlipse) を推奨し、WinShell の設定方法について説明します。TeXlipse の方が補完を行うことができるなど、高機能ですがインストールがやや大変、細かいところに手が届かないなどといった (私的に) 使いにくさがあるので、ここでは紹介しません。使ってみたい人は各自でインストールの方法を調べて試してみてください。

<http://oku.edu.mie-u.ac.jp/~okumura/texwiki/?TeXlipse>

WinShell は以下のサイトの左上に表示されている「WinShell332.exe」をダウンロードすると良いかと思います。2014 年 2 月 5 日現在の最新版は、「3.3.2.6」です。

<http://www.winshell.org/modules/download/>

Windows を再起動した後に、このソフトをデフォルトの設定でインストールすると図 1 のように、WinShell で使用する言語を選択する画面が出てきます。日本語も存在しますので、「Japanese」を選択して OK を押してください。

まずは、demo ワークスペースを試してみても面白いかもしれませんが、今回は必要ありませんので、プロジェクトを閉じます。図 2 のように、demo を選択し、右クリック、プロジェクトを閉じる、と進み、プロジェクトを閉じてください。また、少し WinShell の設定が必要になりますので、メニューバーの「オプション→全般」をクリックして図 3 図 4 の通り、2 箇所の設定を行ってください。

WinShell の使い方は適宜慣れていってもらえたらと思いますが、図 5 に基本的な使い



図 1 言語選択画面

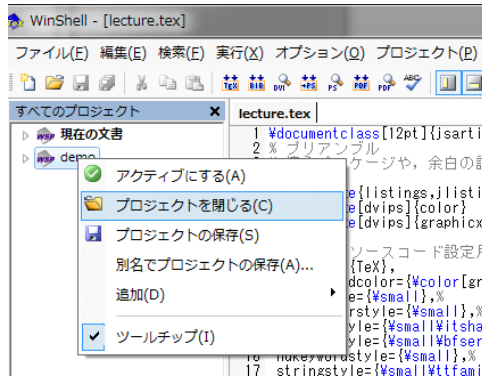


図 2 demo プロジェクトを閉じる

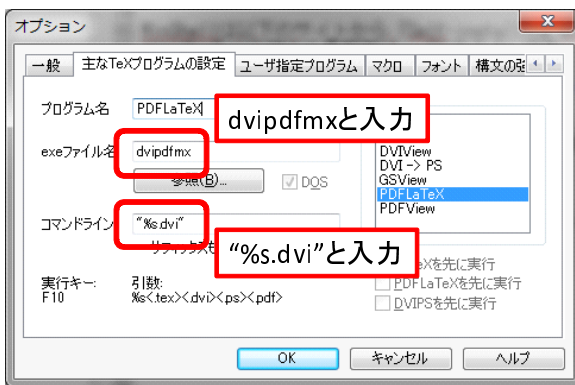


図 3 「主な TeX プログラムの設定」 タブ

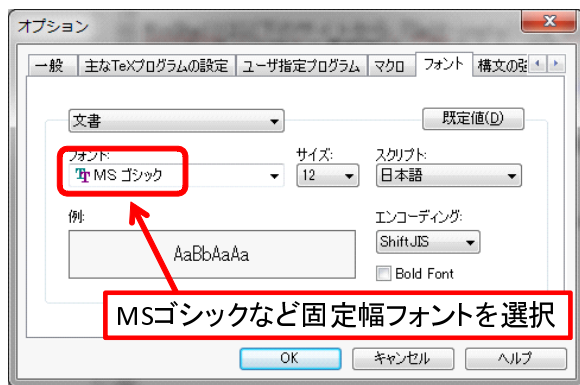


図 4 「フォント」 タブの設定

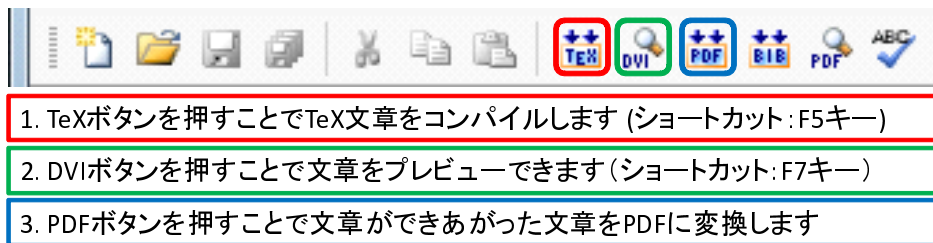


図 5 WinShell の基本ステップ

方を示します。

3.3 他の便利なツール（推奨 2）

3.3.1 図の作成について

WinShell をインストールすれば、文章を書くことは容易になりますが、図の貼り付けがこのままではできません。T_EX 文章中に図を貼り付ける場合、png 形式、pdf 形式など、いろいろな方法で貼り付けることが可能^{*1}ですが、ここでは、PowerPoint で作成した図を EPS 形式（ベクトル形式）に変換し、図として貼り付ける方法を説明します。EPS 形式は T_EX と相性が良く、特別なパッケージをインストールすることなく利用可能です。

パワーポイント、エクセルなどは、図として保存する際に WMF 形式や EWF 形式での保存が可能です。MetafileToEPSConverter というソフトウェアは、それらの形式の画像を EPS 形式に保存するソフトウェアで、パワーポイントでコピーしたデータをきれいに貼り付けることもできます。ダウンロードは以下のサイトで行ってください。

<http://wiki.lyx.org/Windows/MetafileToEPSConverter>

3.3.2 PDF リーダーについて

現在、PDF を開くソフトウェアとしては Adobe Reader が一般的に多く使われていますが、PDF を開いているときにそのファイルを更新しようとしてもロックがかかっており PDF を変更できません。さらに少し動作が重いという欠点もあります。SumatraPDF というソフトウェアは軽くてファイルを開くときにロックを取得しないので、T_EX と相性が良いという特長があります。ダウンロードは以下のサイトで行ってください。2014 年 2 月 5 日現在の最新版は、「SumatraPDF-2.4」です。

<http://blog.kowalczyk.info/software/sumatrapdf/>

4 T_EX 文章のサンプル

さて、演習室の Windows 環境には今年度から pL^AT_EX がインストールされましたので、早速 WinShell を利用して pL^AT_EX 文章を作ってみましょう。まずは WinShell を起動して、以下の文章を PDF から貼り付けて F5 を押してコンパイルしてください。

^{*1} jpg, png などの図はスクリーンショットなどの画像に適したビットマップ形式として保存されます。回路やグラフなどの図はベクトル形式で保存、貼り付けた方がきれいに出力することができます。ビットマップ形式とベクトル形式の簡単な例としては、Adobe Photoshop や gimp で作るデータはビットマップ形式に向いていて、Adobe Illustrator や InkScape で作るデータはベクトル形式に向いています。

ソースコード 1 L^AT_EX ソースファイルのテンプレート

```
1 \documentclass[12pt]{jsarticle} % 12pt ぐらいが見やすいと思います
2 % 以下はコメントです
3 \begin{document}
4 \section*{アブストラクト}
5 報告する内容についての概要を示します.
6
7 \section{目的}
8 \pLaTeX を使ったレポートの書き方を学んでもらいます.
9
10 \section{数式}
11 数式が式 \ref{equ:example-equation} のようにきれいに出力できます.
12 複数行だってお手の物です (式 \ref{equ:example-eqnarray}).
13
14 \begin{equation}\label{equ:example-equation} % equation 環境は1行の式
15 \sum_{i=1}^n i = 1 + 2 + \dots + n
16 \end{equation}
17
18 \begin{eqnarray}
19 (a + b + c)^2 &=& \{(a+b)+c\}^2 \nonumber \\
20 &=& (a+b)^2 + 2(a+b)c + c^2 \label{equ:example-eqnarray}
21 \end{eqnarray}
22
23 \end{document}
```

アブストラクト

報告する内容についての概要を示します.

1 目的

pL^AT_EX を使ったレポートの書き方を学んでもらいます.

2 数式

数式が式 1 のようにきれいに出力できます. 複数行だってお手の物です (式 2).

$$\sum_{i=1}^n i = 1 + 2 + \dots + n \quad (1)$$

$$\begin{aligned} (a + b + c)^2 &= (a + b) + c^2 \\ &= (a + b)^2 + 2(a + b)c + c^2 \end{aligned} \quad (2)$$

図 6 出力結果

このソースコードを解説していきます.

- 1 行目 ここでは、jsarticle というパッケージを利用し、12pt を基本の文字サイズとすること定めています。12pt 以外にも 10pt, 11pt が利用可能ですが、レポートとしては 12pt が見やすく良いと思います。jsarticle 以外にも jsbook, jreport などが利用可能ですが、レポートとしては jsarticle を利用すると良いと思います。
- 2 行目 % という命令についての説明です。% 以降の文字はコメントとしてレポート本文中には現れません。% ここはまだ未完成 などといった形で、自分が分かりやすくなるように適宜挿入すると良いです。
- 3 行目 この行以降が文章の始まり、という命令です
- 4 行目 section(節) を定義しますが、* をつけることによって節番号を出力していません。アブストラクト(概要)には通常節番号をつけないため、このような命令を利用する必要があります。* には他にも強制余白生成など色々な意味がありますが、ここでは節番号を出力しない、という意味です。
- 5 行目 単純に文章を記述していきます。文章中の 1 回の改行は無視されます。2 回改行を入れることで段落を変更することができます。段落を変える場合は、説明する内容が大幅に変わるときなどに利用してください(1 行ごとに段落を変えたりすることはありません)。
- 7 行目 4 行目に似ていますが、* が付いていませんので、節番号が出力されます。
- 8 行目 あまり使うことはありませんが、\pLaTeX という T_EX の命令を使っています。T_EX では、\ (円マーク、バックスラッシュ) 以降の半角文字はすべて T_EX 用の命令として扱われますので、\pLaTeX の直後には半角スペースが必要です。
- 10 行目 7 行目と同じく、節を定義しています。節番号は自動的に加算されますので間違えることはありません。
- 14 行目 ここでは、数式を記述しています。`\begin{equation}` とは、equation (式) を `\end{equation}` の間に記入しますよ～という宣言です。この、`\begin` ではじまり、`\end` で終わるブロックのことを**環境**と呼びます。
- 他にも、`\label{equ:example-equation}` という命令が使われています。これは、英語の名前通り、この式に `equ:example-equation` という名前をつけています。11 行目と 12 行目で `\ref{equ:example-equation}` という形で、同じ名前を利用していますが、こちらはこの名前を持つ式の番号を教えてください、という命令ですので、出力結果を見ると「式 1」という形で出力されています。この名前をつけて番号を参照する方法を**相互参照**と呼びます。図や表にも相互参照を利用することができるので、効果的に利用することで図、表番号の間違いを無くすことができます。

18 行目 `eqnarray` 環境では複数行にまたがる式の展開に用いる環境です。出力結果のように、`=` の位置をそろえて出力することが簡単に実現することができます。詳しい使い方は、使うときに調べてみてください。

文章中に簡単な数式を記述する必要がある場合は、`$` で囲うことにより実現することができます (インライン数式と呼びます)。例えば、`$(a + b)^2 = a^2 + 2ab + b^2$` と記述すると、 $(a + b)^2 = a^2 + 2ab + b^2$ と表示することができます。ただし、分数や総和記号などを用いる場合には用いない方が良いでしょう。例えば、15 行目を文章中に`$$\sum_{i=1}^n i = 1 + 2 + \dots + n$` というように記入すると、 $\sum_{i=1}^n i = 1 + 2 + \dots + n$ と少しカッコ悪く表示されます。インライン数式モードでもきれいに出力する方法はありますが、ここでは紹介しません。`equation` 環境などを利用してください。

では、もう少し複雑な数式について考えてみます。以下は三重大の奥村先生 (日本の \TeX の権威) のページ (TeX Wiki: <http://oku.edu.mie-u.ac.jp/~okumura/texwiki/>) に載っているサンプルです。

$$\frac{\pi}{2} = \left(\int_0^\infty \frac{\sin x}{\sqrt{x}} dx \right)^2 = \sum_{k=0}^\infty \frac{(2k)!}{2^{2k} (k!)^2} \frac{1}{2k+1} = \prod_{k=1}^\infty \frac{4k^2}{4k^2 - 1}$$

解説もそのまま引用させていただきます。^{*2}

`\frac{分子}{分母}` は分数です。`\pi` は π です。`\left(\dots \right)` は中身に応じて大きさが変わる括弧です。`\int` は積分記号です。`\int_{下限}^{上限}` のようにして定積分にできます。`\infty` は ∞ です。

`\sin` は `sin` と出力します。単に数式中で `sin x` と書くと \sim のように全部が数式用イタリック体 (傾いた文字) になってしまいます。数式中では、変数はイタリック体にしますが、`sin` のような演算子はイタリック体にしません。`\sin x` と書けば $\sin x$ のようになります。同様に、`\cos`, `\tan`, `\log` などが使えます。

`\sqrt{x}` は \sqrt{x} と出力します。`\sqrt[3]{x}` と書けば 3 乗根になります。

`\sum`, `\prod` は和の記号 Σ と積の記号 Π です。`\sum_{下限}^{上限}` のように上限・下限が書き込めます。

^{*2} TeX Wiki:TeX 入門/簡単な数式 (2)

5 体裁を変更するコマンド

あまり利用する機会はないと思いますが、体裁を変更するコマンドについても述べておきます。T_EX 文章は基本的に左寄せですが、表紙を作成する際などには中央寄せ、右寄せにしたい場合もあります。その場合、`center` 環境と `flushright` 環境を用いると良いです。`center` 環境については図表の際に用いるので、ここでは右寄せ環境のサンプルを挙げておきます。

```
\begin{flushright}
```

この文章は右寄せされます。名前や日付などを記入する際に利用します。

```
\end{flushright}
```

この文章は右寄せされます。名前や日付などを記入する際に利用します。

同じように、文字の大きさを変更する場合は表 1 の環境を利用可能です。`tiny`, `scriptsize`, `footnotesize`, `normalsize`, `large`, `Large`, `LARGE`, `huge`, `Huge`.

表 1 フォントの大きさ

小さくする	<code>\tiny</code>	<code>\scriptsize</code>	<code>\footnotesize</code>		
	約 50%	約 70%	約 80%		
大きくする	<code>\large</code>	<code>\Large</code>	<code>\LARGE</code>	<code>\huge</code>	<code>\Huge</code>
	約 120%	約 140%	約 170%	約 210%	約 250%

また、文章中の一部の文字だけを大きくしたい場合は、`{中括弧}` で囲うことにより一部だけのフォントの変更が可能です。`{\large てすと}` と記述した場合、てすとと表示されます。同じ方法により、太文字 (ボールド体) `\bf` , 英字は斜体 (イタリック: *Italic*) `\it` なども利用することができます。

6 図, 表の利用

ソースコード 2 図と表を挿入

- ```
1 \documentclass[12pt]{jsarticle} % 12pt ぐらいが見やすいと思います
2 % \begin{document}までのこの区間はプリアンプルと言います
```



```

3 \usepackage[dvips]{graphicx} % 図を利用するために利用
4 \begin{document} % ここから文章を記入する
5
6 \begin{figure}[htbp]
7 \begin{center}
8 \resizebox{.4\textwidth}{!}{\includegraphics{half_adder.eps}}
9 \caption{半加算器}\label{fig:half-adder}
10 \end{center}
11 \end{figure}
12
13 \begin{table}[htbp]
14 \begin{center}
15 \caption{表のテスト}\label{tab:table-test}
16 \begin{tabular}{l|cr}
17 タイトル & 列1 & 列2 \\ \hline
18 行1 & あああ & ううう \\
19 行2 & いいい & えええ
20 \end{tabular}
21 \end{center}
22 \end{table}
23
24 \end{document}

```

## 6.1 図の利用

図を作成するためには、パワーポイントなどを利用すればいいですが、演習室には WMF2EPS がインストールされていませんので、ホームページから EPS ファイルをダウンロードして利用してください。

<http://www.info.nara-k.ac.jp/~iwata/lecture/2011/il-w01-half-adder.eps>

まず、ソースコード 2 の 3 行目を見てください。ここでは、`graphicx` パッケージの `dvips` というオプションを利用する、と書かれています。これは図を使うときのおまじないとしてプリアンブルに記入してください。プリアンブルとは、`\documentclass` の行以降から `\begin{document}` までの行のことを指します。このプリアンブルには、作成する文章の体裁や、利用するパッケージなどを記入します。余白の設定なども行うことができますが、特に指定されない場合、`jsarticle` のデフォルトを用いると良いと思います。

実際に、図を貼り付ける命令は 6 行目～11 行目までとなっています。8 行目では、貼り付ける図の大きさ（ここでは余白を除くページ全体の 0.4 倍の大きさ）を指定し、読み込みたいファイル名を `\includegraphics{xxx.eps}` という形で記入します。さらに、9

行目では図のキャプション（図や表を端的に説明する文章）と相互参照を作成しています。

## 6.2 表の利用

13 行目から 22 行目は、表の書き方を示しています。図と異なるのは、15 行目において図では図の下に書いてあったキャプションが、表の上にかかれていることです。これは、キャプションは図の下に、表については上に入れるという決まりがあるからです。命令としては図の時と同じ、キャプションと相互参照を挿入しているだけです。

16 行目では、表の書式設定を行い、何列必要なのか、また左寄せ (l)、中央寄せ (c)、右寄せ (r) するのか、罫線をどこに引くのかを決めています。lcr のうち 1 つ書くごとに列が増えていきます。| マークが入っている列と列の間に罫線を引くという命令です。そのため、この例では 1 列目と 2 列目の間に罫線が引かれます。

17 行目では 1 行目の内容を記述しています。列の区切りは & マークを用います。3 行を指定した場合、3 行きちんと全部記述しないとエラーになってしまいます。& と & の間はスペースだけで良いので入れておくとエラーにはなりません。複数列を 1 マスに統合したい場合は、\multicolumn 命令を使えば実現できますがここでは紹介しません。Excel2LaTeX などの外部ツールを用いた方が楽だと思います。

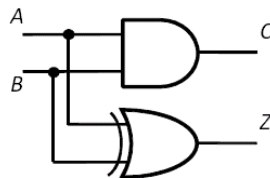


図 1 半加算器

表 1 表のテスト

| タイトル | 列 1 | 列 2 |
|------|-----|-----|
| 行 1  | あああ | ううう |
| 行 2  | いいい | えええ |