

文書作成システム L^AT_EX 入門

1 準備

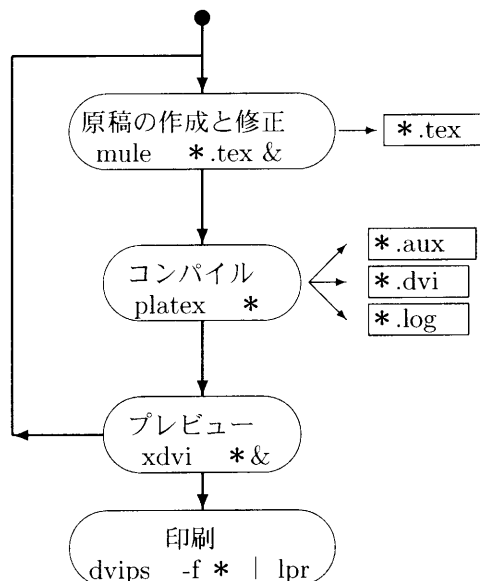
Linux を起動する。ホームディレクトリで `lftp` を使ってサーバ 157.118.89.2 の TG-Local/tsuchi/DTP07 の下にあるファイルを持ってくる (`cd lftp 157.118.89.2 cd TG-Local/tsuchi mirror DTP07 exit`)。DTP07 というディレクトリができていないはずなのでそこに移動し (`cd DTP07`)、`cubic0.ps`、`example0.tex`、`example1.tex`、`latex0.dvi`、`latexmath.dvi`、`sample.dvi` というファイルがあることを確認する (`ls`)。

`xdvi latex0&` とすればこのプリントと同じものが画面でも見れる。その 5 ページ以降にさらに進んだ L^AT_EX の使いかたが書いてあるので参考にしてほしい。

2 L^AT_EX の基本

L^AT_EX(ラテフ)は Donald Knuth (クヌース) 氏が作った T_EX(テフ) という文書作成システムに Lesli Lamport (ランポート) 氏がマクロを書き加えて使いやすくしたものである。ワープロと違い、L^AT_EX を使うには `mule` 等のエディターで `example0.tex` のように `tex` という拡張子をもつ TeX のソースファイルを作り、`'platex example0'` としてコンパイルする。すると `example0.dvi` のように `dvi` という拡張子を持ったファイルが出来る。次に、これを `xdvi` というプレビューアで画面に表示し、必要があれば、エディターでソースファイルを修正してコンパイルするという作業を繰り返す。

L^AT_EX による文書作成の流れ (下の図の * には具体的なファイル名が入る)



`xdvi *&` と `mule *.tex &` は
最初の一回だけ実行すればよい

`platex *` としたときエラーが出たり
思いどおりの結果にならないときは
`mule` でソースファイル (*.tex) を
修正して、再び `platex *` を実行する

ソースファイルは以下のような形式である。

```
\documentclass[a4paper]{jarticle}
    プリアンブル (preamble)
\begin{document}
    本文
\end{document}
```

`\documentclass` の後の `[]` の中には `a4paper`, `a4book`, `b5paper`, `twocolumn`, `10pt`, `12pt` 等が、`{ }` の中には `article`, `jarticle`, `tarticle` 等が指定できる。

プリアンブルには印刷領域の大きさの変更やマクロの定義等のコマンドを書く。例えば、
`\setlength{\textheight}{23cm}`
とすれば、本文領域の高さが 23cm になる。
`%` で始まる行はコメントとして無視される。

例題 先ず、`'platex example0'` を実行した後、`example0.aux`, `example0.dvi`, `example0.log` というファイルができていることを確認しなさい (`ls -l ex*`)。

次に `'xdvi example0&'` を実行してみなさい。(注:以後 `platex` を実行した後に `xdvi example0&` を実行する必要はない。) さらに `mule` でファイル `example0.tex` を開き (`mule example0.tex&`), 一行目

```
\documentclass[a4paper]{jarticle}
```

の `[a4paper]` を `[a4paper, twocolumn]`, `[b5book, 12pt]` 等と書き直して上書き保存し、`kterm` でコンパイル (`platex example0`) してどのような変化が起こったかみなさい。また、`{jarticle}` を `{tarticle}` と書き直してコンパイルしてみなさい。

2 行目から 5 行目¹ までの `%` を取って、行末尾の `{ }` の中の数値を変えてコンパイルしてみなさい。

10 行目の `\author{ }` の `{ }` の中を自分の名前に書き換えてコンパイルしてみなさい。11 行目の先頭に `%` を付けてコンパイルしてみなさい。15 行目 `'%\tableofcontents'` の先頭の `%` を取って二回コンパイルしてみなさい。

コンパイル (`platex` を実行) したとき、ソースファイルの中に間違いがあれば、次のようなエラーメッセージが表示される。

! Undefined control sequence.

1.19 \hh

?

! の後がエラーの種類であり、1.19 はエラーの位置である。この場合、19 行目の `\hh` という命令が定義されていないという意味である。最後に ? が出て入力をうながしている。ここで `Enter key` を押せば、エラーを無視して処理を続行するが、`x` を押せば、**処理を中断**する。


本文中の段落を変えるには空白行を入れる。改行するには `\` を入れる。`example0.tex` の 20 行目から 25 行目までとプレビューを見比べればわかるであろう。

¹カーソルが何行目にあるかは `mule` の画面の下のモードラインの中央から少し左に表示されている。カーソルを 10 行下に移動するには `Esc` 1 0 ↓ と入力すればよい (`Esc` キーはキーボード左上にある)。

3 文字のスタイルとサイズ

文字の大きさと字体はかなり自由に変えられる。特に説明しなくても、ソースファイル example0.tex の 30 行目から 35 行目までとプレビューを見比べればわかるであろう。

数文字だけイタリックにしたければ `\it 数文字` とすればよいし、何行かにわたってイタリックにしたければ、その部分を `\begin{it}` と `\end{it}` で囲めばよい。

演習 1 `p***.tex` というファイルを作り (***) には自分の学生番号の下 3 桁を入れる)、それをコンパイル (`platex p***`)、プレビュー (`xdvi p*** &`) したときに `sample.dvi(xdvi sample&`  で表示) の 1. のように表示されるようにしなさい。尚、今後の**演習**はすべて `p***.tex` というファイルの中に作りなさい。

4 数式

TEX はもともと数式を簡単に美しく印刷するために作られたので、複雑な数式の表示は L^AT_EX の得意とするところである。文章中に数式を入れるには `$` と `$` の間に数式を書くためのコマンドを記述する。一行を使って大きく表示させるには `$$` と `$$` の間に記述する。

例えば、 $\frac{a+b}{2}$ と表示させるには `‘\frac{a+b}{2}’` と書き、 \sqrt{ab} と表示させるには `‘\sqrt{ab}’` と書けばよい。また、

$$\frac{a+b}{2} \geq \sqrt{ab}$$

と表示させるには `‘$$ \frac{a+b}{2} \geq \sqrt{ab} $$’` と書けばよい。example0.tex の中にいろいろな数式の例を書いてあるので、それを参考にしてほしい。

行列等のように複数の数式を表のように並べたいときは `array` 環境を使う。example0.tex の 78 行目から 107 行目を参考にしてほしい。数式についてさらに詳しいことは `latexmath.dvi` をみてほしい (`xdvi latexmath &`)。

演習 2 `sample.dvi` の 2. の数式を表示させなさい。

5 表

表を作成するには `tabular` 環境を使う。最初に簡単な例で説明する。ソースファイルに

```
\begin{tabular}{cc|c|}
一行一列 & 一行二列 & 一行三列 \\
\hline
二行二列 & 二行二列 & 二行三列 \\
\end{tabular}
```

と書いてコンパイルすれば、右上のように表示される。

`\begin{tabular}` の後の `{ }` の中には `l, c, r |` が指定できる。l, c, r はそれぞれ左詰め、中央寄せ、右詰めに意味する。列と列の間に縦罫線を引きたければ `|` を指定する。行と行の間に横罫線を引きたければ `\hline` を入れる。項目の間は `&` で区切り、行の終には `\\` を書く。

例題 example0.tex の 110 行目の `\begin{tabular}` の後の `{|cc|cc|}` を `{|r||lrr|}` 等と書き換えてコンパイルしてみて、結果がどうなるかみなさい。

複数の欄をぶち抜いて文字を表示したいときは `\multicolumn` コマンドを使う。このコマンドの一般形は `\multicolumn{ぶち抜く欄の数}{項目の様式}{テキスト}` である。例えば、右下のように表示させたいければ、

```
\begin{tabular}{|c|c|c|c|}
```

```
\hline
```

```
一列 & \multicolumn{3}{c}{二列と三列と四列} \\
```

```
\hline
```

```
一列 & 二列 & 三列 & 四列 \\
```

```
\hline
```

一列	二列と三列と四列		
一列	二列	三列	四列
一列と二列		三列と四列	

```
\multicolumn{2}{|c|}{一列と二列} & \multicolumn{2}{|c|}{三列と四列} \\
```

```
\hline
```

```
\end{tabular}
```

とすればよい。

演習 3 example0.tex の 125 行目から 150 行目までを `p***.tex` に コピー し、さらにそれを修正して `sample.dvi` の 3. のように表示させなさい。

6 図

図を描くには `picture` 環境を使う。example0.tex の 155 行目の `\setlength{\unitlength}{1mm}` は単位長の設定をしている。156 行目の `\begin{picture}(60,80)` の `(60,80)` は横 60 縦 80 の枠を作り、その中に図を描画するというのである。157 行目の `\put(40,75){\circle*{2}}` は左下を原点として `x=40, y=75` の位置に半径 2 の円を描き、中を塗りつぶすという命令である。`*`を付けなければ、中を塗りつぶさない。

例題 example0.tex の 155 行目の `\setlength{\unitlength}{1mm}` の `1mm` を `2mm` と書き換えてコンパイルしてみなさい。次に、`x=60, y=50` の位置に半径 5 の円を描きなさい。

7 補足

このプリントでは説明していないが、化学式を書くこと、脚注や側注を付けること、一部分だけ縦書にすること、箇条書き、右寄せ、文字を反転及び回転して表示することもできる。それらの例を `'example1.tex'` に載せてあるので参考にしてほしい。

レポート提出 1/9 までにこのプリントの演習 (1., 2., 3. は必ずやること) の解答を載せた TeX のソースファイル (`p***.tex`) を添付した E-mail を

tsuchi@cs.tohoku-gakuin.ac.jp

へ送ること。その際、件名はレポート提出とし、本文に学生番号と氏名を書くこと