

# 第一回 T<sub>E</sub>X ゼミ

岡田靖男 小野景子 有賀浩

2000年4月10日

# 目次

|              |                                 |           |
|--------------|---------------------------------|-----------|
| <b>第 1 章</b> | <b>TEX の基礎</b>                  | <b>3</b>  |
| 1.1          | TEX の誕生                         | 3         |
| 1.2          | L <sup>A</sup> TEX とは何か         | 3         |
| 1.2.1        | 視覚デザイン                          | 3         |
| 1.2.2        | 論理デザイン                          | 4         |
| 1.3          | TEX を何故用いるのか                    | 4         |
| 1.3.1        | 利点について                          | 4         |
| 1.3.2        | 欠点について                          | 5         |
| <b>第 2 章</b> | <b>L<sup>A</sup>TEX のインストール</b> | <b>6</b>  |
| 2.1          | インストールの前に                       | 6         |
| 2.2          | インストーラの起動                       | 6         |
| 2.2.1        | インストーラの起動の前に                    | 6         |
| 2.2.2        | インストール作業の開始                     | 6         |
| 2.3          | インストールの実行                       | 7         |
| 2.3.1        | インストールするコンポーネントの選択              | 7         |
| 2.3.2        | フォントの解像度の選択                     | 7         |
| 2.3.3        | その他のソフトウェアのインストール               | 7         |
| 2.3.4        | インストールの最終確認                     | 8         |
| 2.4          | インストールの作業の経過                    | 8         |
| 2.4.1        | ファイルをハードディスクにコピーする              | 8         |
| 2.4.2        | 秀丸エディタのインストール                   | 8         |
| <b>第 3 章</b> | <b>PTEX の動作環境</b>               | <b>11</b> |
| 3.1          | PTEX による処理                      | 11        |
| 3.2          | DVIOUT による出力                    | 12        |
| 3.3          | PTEX 環境の完成                      | 12        |
| <b>第 4 章</b> | <b>TEXMAC の設定と利用</b>            | <b>13</b> |
| 4.1          | TEXMAC の環境設定                    | 13        |
| 4.1.1        | 秀丸エディタへの登録                      | 13        |
| 4.2          | TEXMAC による原稿の執筆                 | 14        |
| 4.2.1        | 必須となる設定事項の入力                    | 14        |
| 4.3          | 原稿の執筆                           | 14        |
| 4.4          | TEX による処理とプレビュー                 | 16        |
| 4.4.1        | フォーマットファイルの選択                   | 16        |
| 4.4.2        | TEX による原稿ファイルの処理                | 16        |
| 4.5          | よく使う機能, 便利な機能                   | 17        |

|  |    |
|--|----|
| 第 5 章 課題                                 | 20 |
| 5.1 L <sup>A</sup> T <sub>E</sub> X を使おう | 20 |
| 5.1.1 T <sub>E</sub> XMAC で練習            | 20 |
| 5.2 自己紹介で練習                              | 20 |
| 5.2.1 生い立ち                               | 20 |
| 5.2.2 趣味                                 | 20 |
| 5.2.3 課題について                             | 20 |

# 第1章 $\text{T}_{\text{E}}\text{X}$ の基礎

## 1.1 $\text{T}_{\text{E}}\text{X}$ の誕生

スタンフォード大学の Knuth 博士によって開発された組版用のソフトウェアである  $\text{T}_{\text{E}}\text{X}$  は、高機能で高品位なデスクトップパブリッシング (DTP) 用ソフトとして、主に出版業界や教育・研究分野で広く支持されてきました。

組版という言葉はあまり耳慣れない人もいますが、広辞苑によると、「原稿の指定に従って、順序・字詰・行数・字間・行間・位置などを正しくくみ上げること。植字。」となっています。つまり、簡単に言ってしまうと、印刷物を作るために文字を配置することです。

ところが、当時のコンピュータによる組版結果は、活版印刷によるものと比べて数式などの仕上がりが悪く、Knuth 博士はとても失望したそうです。そこで Knuth 博士は約 4 年をかけ、特に組版が難しいと言われる技術関係の書物を美しく組版できるようなソフトウェアである  $\text{T}_{\text{E}}\text{X}$  を開発したのです。

## 1.2 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ とは何か

$\text{T}_{\text{E}}\text{X}$  の組版機能は主に「プリミティブ」と呼ばれるもっとも原始的で組版指定の根本に関わる命令群によって実現されています。プリミティブとは、いうなれば活字を組みあわせる職人のひとつひとつの動作に対応した  $\text{T}_{\text{E}}\text{X}$  への指示のことです。 $\text{T}_{\text{E}}\text{X}$  の本体に 300 個ほど準備されているのですが、普通に文章を作成するときこれらのプリミティブをいちいち操作するのはまったく非効率です。その上私たち一般人は組版に関して詳しい知識や作法を知っているわけではありませんから、そのような知識に精通した人が我々の組版を手助けしてくれることが望ましいといえます。

そのような組版の機能を実現するのが「マクロ」です。マクロとは、一定の組版処理を行うプリミティブや別のマクロを組み合わせ、新たに作成した命令をいいます。 $\text{T}_{\text{E}}\text{X}$  には、約 300 個のプリミティブと約 600 個の「plain $\text{T}_{\text{E}}\text{X}$ 」マクロという、あわせて 900 個の命令が最初から備わっています。plain $\text{T}_{\text{E}}\text{X}$  とは Knuth 博士が作成した、文書を記述するために必ず必要となるであろう最低限と思われるマクロ集です。

さて、そのような数あるマクロ集のなかで、世界的に最も広く用いられているのが、DEC 社の Lamport 氏が作成した  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  です。現在日本で  $\text{T}_{\text{E}}\text{X}$  によって組版が行われるとき、ほとんどの場合は  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  をベースにしています。

$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  はこれほど広く利用されるようになった理由として、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  における最大の特徴、その文書のデザイン方法に見出すことができます。文書のデザイン方法というあまり耳慣れないかと思いますが、一般的に分けて 2 通りの方法が考えられます。それは「視覚デザイン」と呼ばれる方法と「論理デザイン」と呼ばれる方法です。そして、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  は論理デザインのほうを選択しました。以下で、この 2 つの方法について説明することにしましょう。

### 1.2.1 視覚デザイン

視覚デザインとは、最近“*What You See Is What You Get* (見たままに出力できる)”，略して“WYSIWYG”と呼ばれており、名前が示すとおりディスプレイ上で編集しているものとまったく同一の印刷出力が得られるデザイン手段です。希望どおりに文字を配置し、自分で整形して、自分が美しいとおもえる出力を手軽に取得できるのです。

しかし、文書の規模が一定以上 (たとえば、画面を何度もスクロールさせなければならない規模) になると、図表や通し番号をつける場合には効率が悪くなる場合があります。このことは皆さんも経験的にわかるとと思います。

WYSIWYG のシステムは、画面上で見ているとおりの出力を得られる点では非常に便利な側面も持っていますが、反面「*What You See Is All You've Got* (見たとおりの出力しかえられず、ミスの有無には関知しない)」というシステムだとして揶揄されることもあります。

## 1.2.2 論理デザイン

視覚デザインに対して、論理デザインは、文章の持つ論理構成にのみ着目して文章をデザインする方法で、一般に「マークアップ方式」などと呼ばれます。他にマークアップ方式を採用している言語として HTML (Hyper Text Markup Language) が挙げられますが、この HTML で記述されている Web ページを思い浮かべてみてください。ある Web ページを表示しているときに Web ブラウザのウィンドウサイズを変更すると、Web ページの外見はウィンドウのサイズにしたがって適切に変化しますが、中央揃えであるとか、段落であるとか、箇条書きであるとか、表であるとか、文章の論理構成にはまったく影響はありません。これは、文章の論理構成に応じて原稿中に埋め込まれた  $\text{\LaTeX}$  のマクロ命令に従って、 $\text{\LaTeX}$  が自動的に体裁を整えて文章を作成するのです。

これが、論理デザインの最大の特徴といえます。論理デザインでは、中央揃えであるとか、段落であるといった文章の論理的な構成をあらかじめ何らかの手段でマークアップしておくことで、体裁に依存しない文章を作成することができるのです。

$\text{\LaTeX}$  の場合、文章の論理的な構成は、あらかじめ  $\text{\LaTeX}$  に用意されている「環境」と呼ばれるマクロ命令を通じて  $\text{\TeX}$  に伝えられます。たとえば、「同レベルの要素を列挙する」論理構成である「箇条書き」の場合、箇条書きの始まりと終わりにそれを宣言する命令を、箇条書きの各要素にもそれとわかる命令を、それぞれ指定することによって、 $\text{\TeX}$  に文章の構造を理解させることができます。

また、 $\text{\LaTeX}$  には、環境のほかにも文章の構成要素(章や節)を指定する命令や、参照用の通し番号をつけたり、その通し番号を参照したりするための命令なども提供されています。

ユーザはこれらの命令を使用して、文章の論理構成や構成要素を  $\text{\TeX}$  に正しく命令を伝えるため、原稿中でただしく命令を使用する必要があります。また、ユーザは、文字の具体的な大きさ、配置などに気を使う必要がなくなりますから、原稿内容に専念することができるため、 $\text{\LaTeX}$  で文章を作成すると必然的に論理構成が整った、よく考えられた文章ができあがるとされています。そのように論理構成や構成要素をもとに組版を行いますから、文章の論理構成や構成要素を的確に表した美しい文書が自然に出来上がります。

## 1.3 $\text{\TeX}$ を何故用いるのか

$\text{\TeX}$  は個人で文書を利用する時は手間が少しかかってしまうが、三木研究室において論文などをグループ利用する時に Word に比べて格段に手間を省く事ができる。これから、実際にどのような利点があるのかについて説明する。

### 1.3.1 利点について

- 文章の体裁に依存できない文章作成が出来る。
  - － 文章のデザイン法としてマークアップ方式をとっている為、文章の論理構成に応じて原稿中に埋め込まれた  $\text{\LaTeX}$  のマクロ命令に従って自動的に文章のレイアウトが決定される。
- フォーマットの統一。
  - － 複数で文章を作成する時、各々で作成した文章を集めてくるだけで統一された文章が出来あがり出来映えがよくなる。
- スタイルファイルの利用。
  - －  $\text{\TeX}$  には文書構造とレイアウトの対応を定義するためのスタイルファイルというものがある。スタイルファイルを変更するだけで、文章のソースを変更することなく、文章のレイアウトを変更することができる。例えば、論文を提出する際にも、スタイルファイルを指定すると各学会の規格に合った論文を瞬時に作ることが出来る。
- レイアウトを考慮しなくてもよい。
  - － 文献の引用文や図表の番号参照などを気にしなくてもよい。
- 再利用性に優れる

### 1.3.2 欠点について

- レイアウトの自由度が小さい。
  - － レイアウトは  $\text{T}_{\text{E}}\text{X}$  が決定するため、文章のレイアウトに個性を発揮する事が出来ない。
- ファイルの扱いが困難。
  - －  $\text{T}_{\text{E}}\text{X}$  は画像と文書は別ファイルになっているためにファイル構成が困難である。又、画像をに EPS ファイルに変換して貼り付けるという作業が必要になる。

## 第2章 L<sup>A</sup>T<sub>E</sub>Xのインストール

### 2.1 インストールの前に

T<sub>E</sub>Xのシステムをハードディスクにインストールするためには、最低でも50Mバイト程度の空き容量が必要となります。また将来的にさまざまなフォントやマクロを追加したり、原稿ファイルを作成したりすることを考えると、最低でも100Mバイト程度の空き容量があるハードディスクを用意したほうがよいでしょう。

### 2.2 インストーラの起動

#### 2.2.1 インストーラの起動の前に

まずインストーラを起動する前に、次の作業を実行してください。

- ・コンピュータを起動させた後、長時間にわたって作業していた場合には、インストールの際にシステムにかかる負担でWindowsが不安定になることがあります。いったんコンピュータを再起動してください。
- ・すべてのアプリケーションソフトを終了させてください。特に、ウイルスチェッカーなどの常駐アプリケーションソフトを利用している場合、常駐を解除からインストールしないと、インストール作業に失敗することがあります。
- ・EzT<sub>E</sub>Xがタスクトレイに常駐している場合には、終了させてください。
- ・もし“常駐秀丸”を利用して、秀丸エディタのアイコンがタスクトレイに表示されているなら、それを終了させてください。

以上の点を間違いなく実行したら、pT<sub>E</sub>X2.1.8の実行環境を構築するために、さっそくインストーラを起動させましょう。まず、CD-ROMをCD-ROMドライブに挿入してください。次にマイコンピュータでCD-ROMを参照します。“Platex2e1999”と表示されているドライブをクリックして、CD-ROMの中身を見てみましょう。

T<sub>E</sub>XのCD-ROMを立ち上げると図2.1の画面がでるので、setup.exeをクリックする。

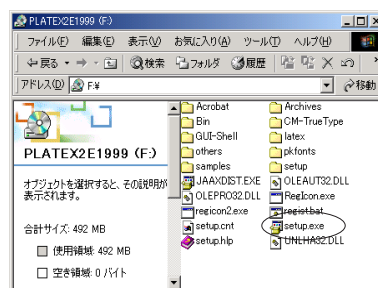


図 2.1: Platex2e1999 の中身

#### 2.2.2 インストール作業の開始

インストーラの起動に成功すると図2.3のような画面が表示されます。インストーラがpT<sub>E</sub>Xの環境を発見できなかった場合には、図2.3のような画面が表示されます。この場合にはインストール開始ボタンを押すとインストール作業を開始できます。は、インストーラがシステムの中にインストール済みのpT<sub>E</sub>Xの環境を発見したときはアンインストールボタンが有効になるまでは、インストーラの起動後、若干の時間を要します。アンインストールボタンが有効になったら、このボタンを押す事で発見したpT<sub>E</sub>X環境を削除することができます。

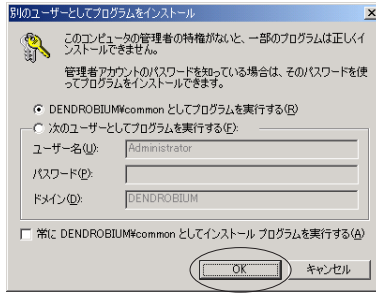


図 2.2: p<sub>T</sub>E<sub>X</sub> の環境を発見した場合

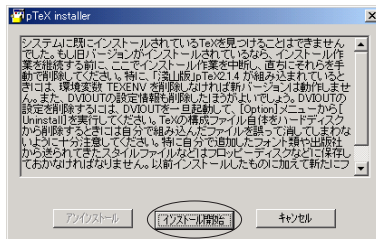


図 2.3: p<sub>T</sub>E<sub>X</sub> の環境を発見できなかった場合

## 2.3 インストールの実行

現在図 2.4 のような画面が表示されているはずですが、これを読み終えたら次へボタンを押して、次の画面に進みます。以降のインストールの作業は、ウィザード形式で進行します。

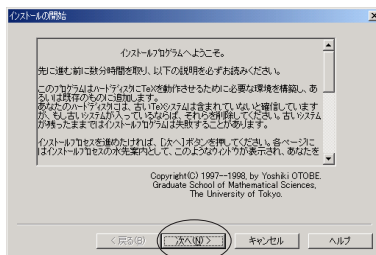


図 2.4: インストール開始

### 2.3.1 インストールするコンポーネントの選択

まず最初に図 2.5 のような画面が表示されます。この画面では、インストールするコンポーネントを選択します。すでにチェックしてあるもの (p<sub>T</sub>E<sub>X</sub>2.1.8 基本セット, L<sup>A</sup>T<sub>E</sub>X2e パッケージ集, dvipsk) だけでいいので「次へ」ボタンを押します。

### 2.3.2 フォントの解像度の選択

図 2.6 では、インストールするフォントの解像度を選択します。ここでは利用するプリンタと同じ解像度にするので、300dpi と 600dpi を選択して「次へ」ボタンを押します。

### 2.3.3 その他のソフトウェアのインストール

インストーラは、p<sub>T</sub>E<sub>X</sub> 本体のほかにも、秀丸エディタや T<sub>E</sub>XMAC など、関連するソフトウェアをインストールすることができます。図 2.7 では、その他のソフトウェアをインストールしたり、T<sub>E</sub>X の環境にあわせて設定したりします。ここでは“秀丸エディタ + T<sub>E</sub>Xmac”のみをチェックします。

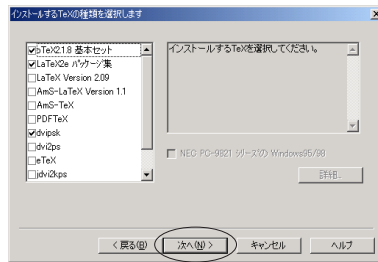


図 2.5: インストールするコンポーネントの選択

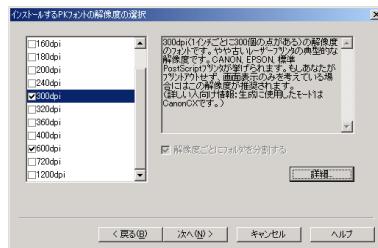


図 2.6: インストールするフォントの解像度の選択

注 すでに秀丸エディタがインストールされている人は、 $\text{T}_{\text{E}}\text{XMAC}$ のみをインストールします。

### 2.3.4 インストールの最終確認

最後に、インストール先となるフォルダを設定します(図 2.8) 起動ドライブ "が"  $c:\$$  "になっていると、"インストール先フォルダ"が"  $c:\$$  "になっていることを確認したら「完了」ボタンを押してください。インストール作業が始まります。

## 2.4 インストールの作業の経過

この節ではインストール作業の経過について説明しています。インストール作業の途中でいくつかの質問をされますので、この節の内容にしたがって設定してください。

### 2.4.1 ファイルをハードディスクにコピーする

ファイルのインストール作業は自動的に進みます。

ファイルのインストール作業はどのように順々に進行しますが、秀丸エディタをインストールするために途中で一時停止します。

### 2.4.2 秀丸エディタのインストール

図 2.7 で"秀丸エディタ +  $\text{T}_{\text{E}}\text{Xmac}$ "というコンポーネントを選択しているので、秀丸エディタのセットアッププログラムが起動します。初めて秀丸エディタをインストールする場合には、はじめの説明の画面で「次へ」ボタンを押してください。この時インストーラは、図 2.9 のようなダイアログボックスを表示しますが、このダイアログボックスはしばらくの間、絶対に操作しないでください。秀丸エディタのインストールが完了しない状態でこのダイアログボックスを操作しても、インストール作業を先に進めることはできません。

さて、はじめの説明の画面で「次へ」ボタンを押すと、画面が表示されます。"秀丸エディタ +  $\text{T}_{\text{E}}\text{Xmac}$ "で説明したフォルダと同一のフォルダが表示されていることを確認してから「次へ」ボタンを押してください。ここでは「いいえ」を選択して「次へ」ボタンを押してください。

すると秀丸エディタの常駐確認の画面になっているはずですが、秀丸エディタは常駐させて利用することもできますが、ここでは常駐させないこととして話を進めますので<sup>1</sup>、そのように設定してから「次へ」ボタンを押してください。

<sup>1</sup> 秀丸エディタは常駐させなくても十分早いですし、常駐しないほうがシステム全体も安定します。どうしても常駐させたい人は常駐させても構いません。

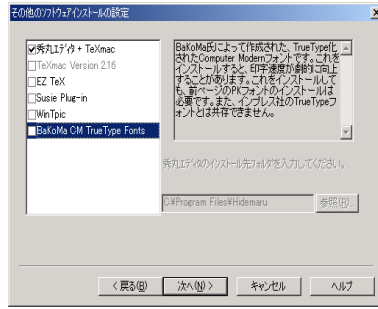


図 2.7: その他のソフトウェアのインストール

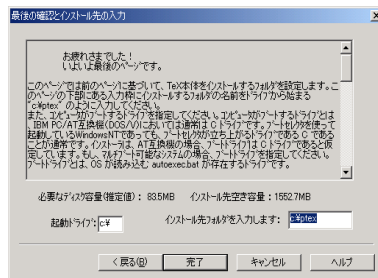


図 2.8: インストールの最終確認

ここでは、デフォルトのキー配置を選択しますが、後からキー配置を変更しなければなりませんので、ここではそのままの選択で「次へ」ボタンを押してください。

エクスプローラから  $\text{T}_\text{E}\text{X}$  の原稿ファイルを開くと秀丸エディタが起動してそのファイルを表示してくれると便利です。このように、任意のファイルと秀丸エディタとを関連付けたいときには「はい」を選択して「次へ」ボタンを押して、次に拡張子を登録します。画面には関連付けられている拡張子の一覧があります (図 2.10)

ここで「追加」ボタンを押して図 2.11 の画面の記入欄に“ tex ”と入力してください。

「OK」ボタンを押すと先ほどの画面の一覧に、関連付けされた拡張子として“ tex ”が追加されます (図 2.12)

ここで「次へ」ボタンを押してください。すると秀丸エディタの登録の画面が表示されます。ここでは出てきたおりのチェックで構わないので、そのまま「次へ」ボタンを押してください。次に「完了」ボタンを押すと、秀丸エディタのセットアッププログラムは終了します。以上で秀丸エディタのインストールは完了です。秀丸エディタインストール終了のを示すダイアログボックスの「はい」ボタンを押し、インストール作業を続けます。

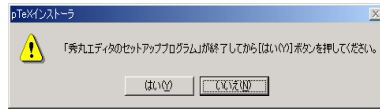


図 2.9: 秀丸エディタの入力待ちメッセージ

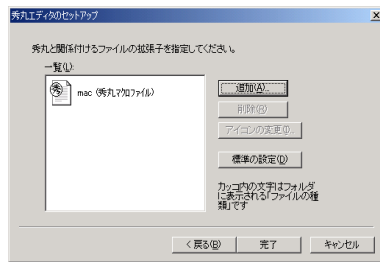


図 2.10: 関連付けの設定

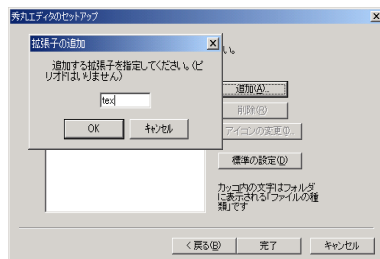


図 2.11: 拡張子の入力

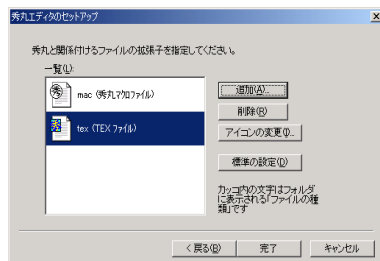


図 2.12: 13

## 第3章 P<sub>T</sub>E<sub>X</sub>の動作環境

コンピュータが再起動すると、T<sub>E</sub>X が利用できる状態になっています。ここでは、インストールを済ませた T<sub>E</sub>X の動作を確認します。また今後、T<sub>E</sub>X をインストールしたフォルダを `c:\ptex` だと仮定します。

### 3.1 P<sub>T</sub>E<sub>X</sub>による処理

インストールされた T<sub>E</sub>X 環境は、GUI シェルによって統合的に操作できるようになっています。GUI シェルとは、T<sub>E</sub>X の実行ファイル群を内部で動作させ、画面上にその動作状況を表示する特殊な制御プログラムです。

この章では、GUI シェルを利用して、サンプルとして用意されている原稿ファイルを実際に処理し、その動作と利用方法を確認します。サンプルの原稿ファイルは `c:\ptex\sample` フォルダにインストールされています。

デスクトップ上に GUI シェルのアイコンが用意されているので、そのアイコンにドラッグ&ドロップする説明します。

まず、Windows の「マイコンピュータ」を開いて、C ドライブ、ptex フォルダ、sample フォルダと順にたどり、`c:\ptex\sample` フォルダを表示させます。

この中から `testlatex2e.tex` ファイルを探し、そのファイルを GUI シェルのウィンドウやアイコンにドラッグ&ドロップします。ドラッグ&ドロップとは、マウスで任意のファイルをつかんできて引きずり、離すという操作です。具体的には、任意のファイルにマウスカーソルを合わせて左ボタンを押し、そのボタンを押したまま指定された場所までマウスカーソルを移動させ、ボタンを離すことを指します。この場合には、GUI シェルのウィンドウ上まで `testlatex2e.tex` ファイルをドラッグし、GUI シェルにドロップします。

GUI シェルにファイルが正しくドロップされると、次の様に T<sub>E</sub>X 処理が始まります。しばらく待っていると、処理が終了した事を示す次のようなメッセージが表示されますので、確認してください。この時点で、原稿がファイルの組版結果である DVI ファイル (`testlatex2e.dvi`) が生成されています。`testlatex2e.tex` ファイルはサンプルファイルなので、正常な動作環境が整っているならば、正常に処理が完了し、図 3.2 の画面が表示されるはずですが、正常に処理が完了して図 3.2 のメッセージが表示されたら「OK」ボタンを押してメッセージを消します。

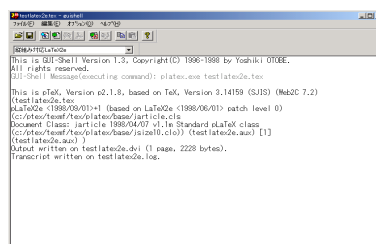


図 3.1: GUI シェルで T<sub>E</sub>X の処理を行っているところ

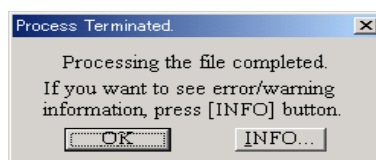


図 3.2: 処理終了のメッセージ

### 3.2 DVIOOUT による出力

次にメニューから「ファイル」-「DVIOOUT」を選択します。すると、こんどは DVIOOUT が起動し、組版結果を表示しようとしています。

しかし、 $\text{T}_\text{E}_\text{X}$  をインストールしたばかりの段階では、この DVI ファイルを表示するために必要な、ある解像度のフォント (cmmmi12) が不足しています。DVIOOUT には、任意の解像度のフォントが不足した時に、そのフォントを自動作成する機能が搭載されています。

不足しているフォントの作成が終了すると、図 3.3 のような DVIOOUT で組版結果が表示されます。

組版結果を DVIOOUT で問題無く表示できるようであれば、 $\text{T}_\text{E}_\text{X}$ 、GUI シェル、DVIOOUT のインストールと設定は完了しています。

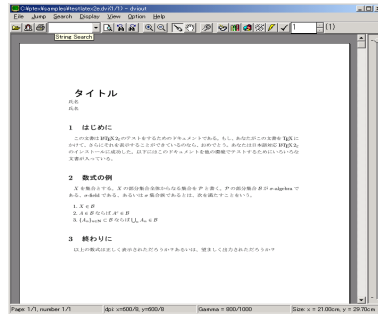


図 3.3: DVIOOUT による表示例

### 3.3 $\text{P}_\text{T}_\text{E}_\text{X}$ 環境の完成

以上で、GUI シェルの動作は確認できました。基本的な利用方法についても、理解できたはずですが、これで、任意の原稿ファイル进行处理し、出力できる環境が入手できた事になります。しかし、いくらか処理環境と出力環境が構築できても、原稿が存在しなければ意味がありません。そこで、次に秀丸エディタによる統合環境を利用して、原稿を作成する方法について概説します。

## 第4章 T<sub>E</sub>XMACの設定と利用

この節では、秀丸エディタと T<sub>E</sub>XMAC を利用した総合環境の設定方法と利用について解説します。T<sub>E</sub>XMAC とは、秀丸エディタ上で T<sub>E</sub>X や L<sup>A</sup>T<sub>E</sub>X の命令をメニュー形式で入力したり、入力した命令を補完したり、作成した原稿を T<sub>E</sub>X で処理して、DIVOUT や GsView で出力できるようにするものです。

### 4.1 T<sub>E</sub>XMAC の環境設定

#### 4.1.1 秀丸エディタへの登録

まず、秀丸エディタ(以下秀丸)を起動します。秀丸を起動したらメニュー画面から「その他」「キー割り当て」を選択します。図 4.1 のようなキー操作に対して機能を割り当てるためのダイアログボックスが表示されますので、図 4.2 に示すように「読み込み」を選んでください。キー割り当てを記憶したファイルがリスト形式で表示されます。図 4.3 でわかるようにキー割り当てを記録したファイルは全部で 9 つあるが、今回インストールした L<sup>A</sup>T<sub>E</sub>X の入力のためのキー定義ファイルは「def-tm98.key」と「old-tm98.key」と「vz-tm98.key」の 3 つです。T<sub>E</sub>XMAC の登録は、T<sub>E</sub>XMAC に対応したキー定義ファイルを読み込むことで実現されます。ここでは、「def-tm98.key」を選択し「OK」を押してください。以下の説明も「def-tm98.key」ファイルを選択したものととして、話を進めます。

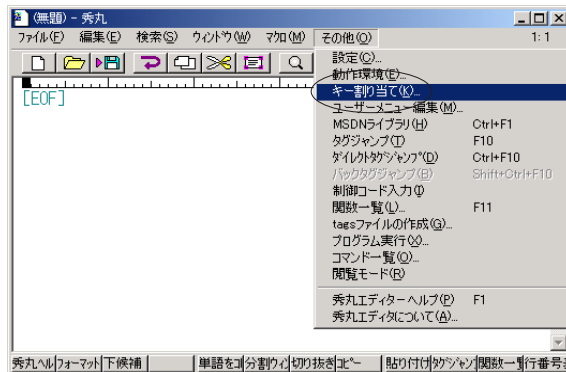


図 4.1: 秀丸エディタにキー割り当てを登録その 1

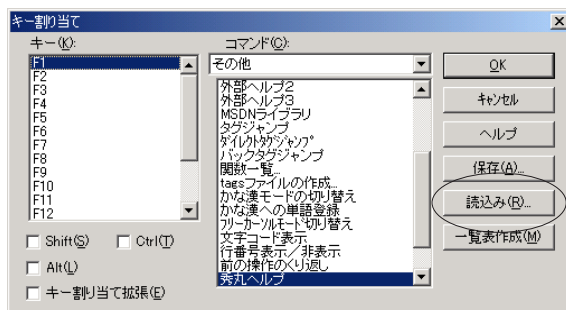


図 4.2: 秀丸エディタにキー割り当てを登録その 2

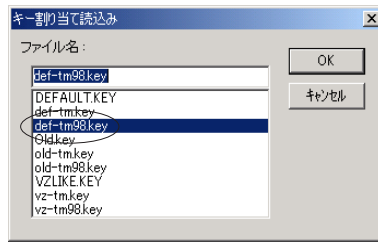


図 4.3: 秀丸エディタにキー割り当てを登録その 3

## 4.2 TeXMAC による原稿の執筆

では、実際に TeXMAC を使用して、原稿を作成し、TeX で処理する方法についてみてみることにしましょう。

### 4.2.1 必須となる設定事項の入力

秀丸エディタを起動したら、最初に「Ctrl」+「¥」キーを押します。すると、図 4.4 のような画面が表示されます。ここで、文書クラスやクラスオプション、パッケージなど、 $\LaTeX$  で必須となる項目を設定することができます。表示されたメニューから「この設定で実行」を選んだら「Enter」キーを押します。すると図 4.5 のように表示されます。入力された文字列のうち、「¥」の記号で始まる文字列は TeX や  $\LaTeX$  の命令で、「%」の記号以降はコメントです。図 4.5 で入力された命令群には  $\LaTeX$  の原稿を作成する上で、必ず記述しておかなければならない必須の命令が含まれています。

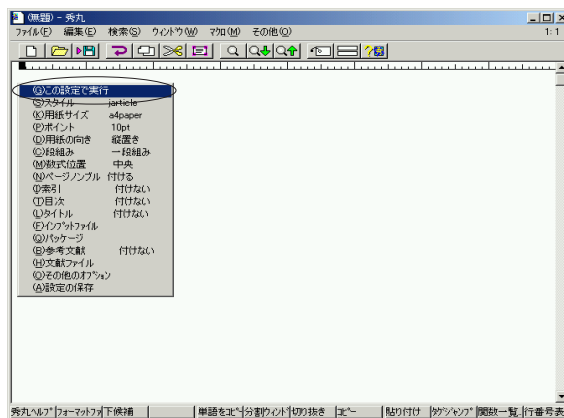


図 4.4: sset2e.mac の起動

## 4.3 原稿の執筆

すでに、 $\LaTeX$  で文章を作成する際に必須の命令群を入力しました。次は、いよいよ本文を記述します。秀丸エディタ上のカーソルは ¥begindocument と入力されている行の次の行に移動しているはずです。もしこの位置になれば、矢印を用いて、その位置へ移動してください。

$\LaTeX$  で文章を作成するためには、本文以外にいくつかの命令を記述しなければなりません。TeXMAC は、面倒な命令語の入力を助けてくれます。

簡単な例文を TeXMAC の機能を利用して作成してみましょう。

まず「Ctrl」+「[」キーを押して texmac98.mac というマクロを起動します。図 4.6 のようなメニューが表示されますので「見出し」に移動して「Enter」キーを押します。すると、章や節などの、文章の見出しを記述するための命令を選択するメニューが開きます(図 4.7)。この中から「節の見出しに移動して Enter」キーを押します。

この操作によって、秀丸エディタにはセクション(節)を開始するための、¥section 命令が入力されます。この時点でカーソルは ¥section 命令の直後の「{」と「}」の間に表示されているはずです。

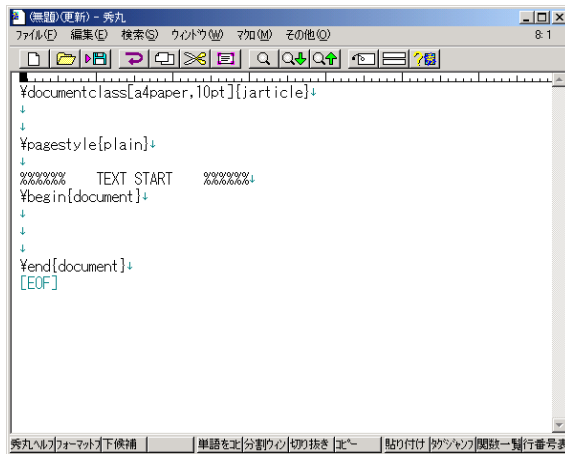


図 4.5: sset2e.mac による自動入力

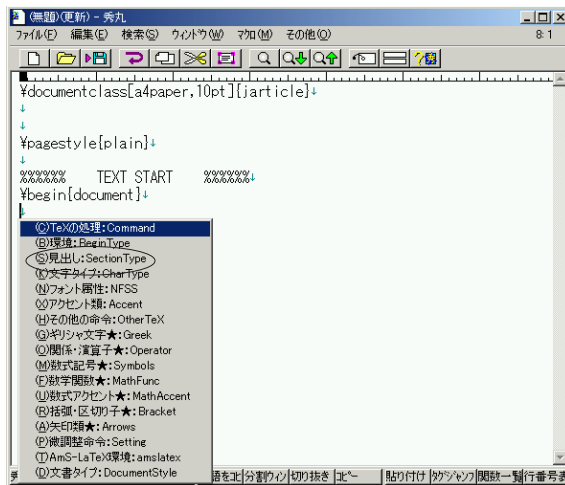


図 4.6: texmac98.mac のメニュー

$\text{\LaTeX}$  では、命令にとって必須となる引数を「{ 」と「 }」の間にはさみ、また、状況に応じて指定可能なオプション引数を「 [ 」と「 ] 」で挟んで、それぞれ指定する約束になっています。つまりこの場合、 $\%section$  命令の引数を「{ 」と「 }」の間に入力すればよいことになります。 $\%section$  命令はセクションを開始するための命令ですから、ここでは「{ 」と「 }」の間にセクションの見出しを入れればよいということです。ここでは、仮に「はじめに」と入力してみましょう。また、改行して「 $\text{\LaTeX}$  ゼミは必修ゼミです」と入力してください。(図 4.8)

以上で示したように、texmac98.mac を用いると、 $\text{\LaTeX}$  の命令を大きな分類の中から選択し、秀丸エディタに入力することができるようになります。このマクロを用いると命令を忘れてしまった場合なども、メニューに表示される簡単な説明からその命令を探しエディタ上に入力できるので便利です。

しかしながら、いちいちメニューを順番にたどらなければならないのでは面倒ですし、文章の作成効率も低下してしまいます。そこで、 $\text{\TeX}$ MAC には、図 4.9 に上げたような、rtex\*.mac というマクロが用意されています。tex\*.mac は「Ctrl」+「F」「D」「E」「K」、のいずれかを押すことで、起動できます。

ここで、これまでに記述した原稿を、保存することにします。保存するためには、秀丸エディタの「ファイル」-「名前をつけて保存」を選択してください。すると、ダイアログボックスが表示されるので、適当なファイル名を入力します。このとき、ファイルの拡張子を「tex」にするようにしてください。ダイアログボックスの「保存」ボタンを押すと、ディスクに原稿ファイルが保存されます。

これでひとまず原稿ファイルが完成しました。

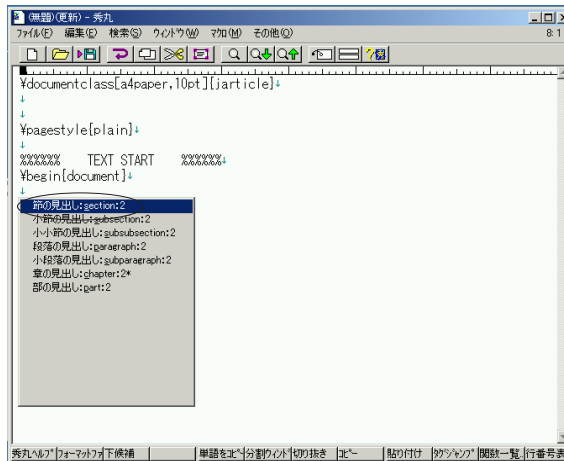


図 4.7: 見出しを選択すると表示されるメニュー

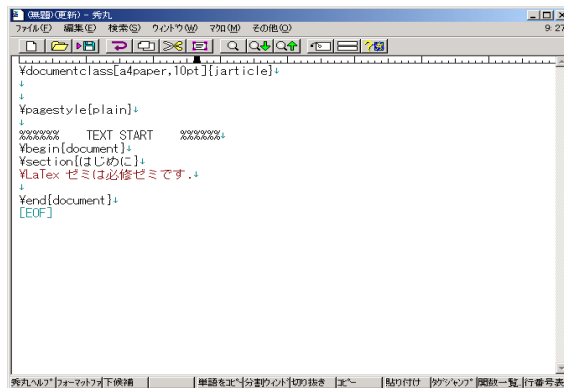


図 4.8: 例文を打ち込んでみる

## 4.4 TeXによる処理とプレビュー

### 4.4.1 フォーマットファイルの選択

原稿ファイルが作成できたら、それを TeX で処理します。このとき、原稿ファイルに応じてフォーマットファイルを選択しなければなりません。フォーマットファイルとは、TeX の設定ファイルのことです。TeX はフォーマットファイルを切り替えることでさまざまな特徴を持った組版システムへと変化します（LaTeX とは、TeX の本体に LaTeX のフォーマットファイルを組み合わせただけです）。

### 4.4.2 TeXによる原稿ファイルの処理

作成した原稿ファイルを TeX で処理してみましょう。「Ctrl」+「]」キーを押して、texmac98.mac を起動します。このマクロは、TeX や DIVOUT といった関連モジュールを秀丸上から操作するためのものです。

texmac98.mac を起動すると、図 4.10 のようなメニューが表示されるので、「現在のファイルのプレビュー」を選択してください。すると、図 4.11 のようなメッセージが表示されます。「はい」を選択すると、TeX で処理した原稿ファイルのプレビューがプレビューアと呼ばれるソフト（ここでは DIVOUT がこれにあたる）で表示されます。図 4.12 は先ほど作成した原稿のプレビューです。

原稿ファイルの保存をしないで上記の操作を行った場合でも、「名前を付けて保存」のダイアログボックスが表示されますので、このときにファイルを保存してもかまいません。また、このときに限り、先ほど述べたファイルの拡張子を tex にする作業をとばしても図 4.13 のようなメニューが表示され、「はい」を選択することで自動的に拡張子を tex にしてくれます。

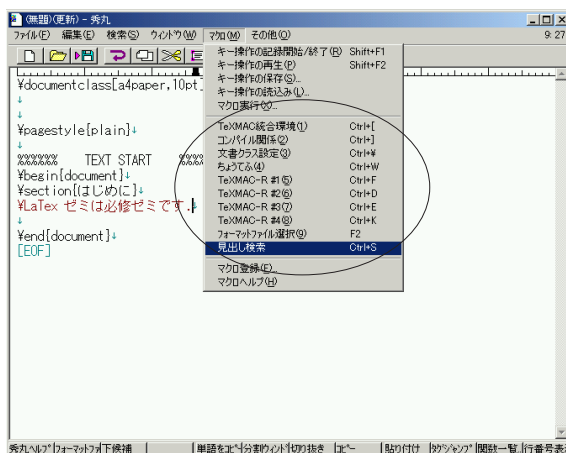


図 4.9: デフォルトのキー割り当て

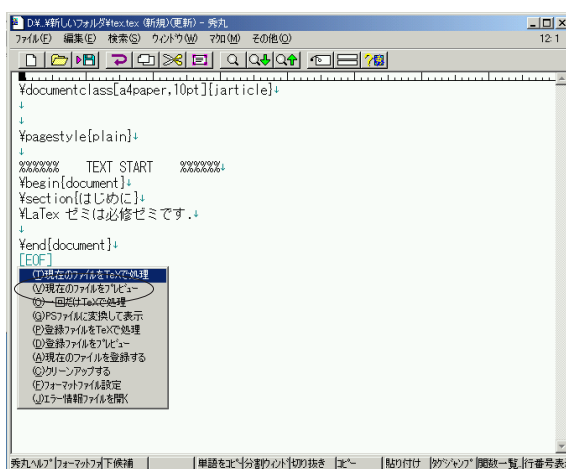


図 4.10: texcom98.mac によるメニュー画面

#### 4.5 よく使う機能, 便利な機能

$\text{T}_\text{E}\text{X}$  や  $\text{L}\text{A}\text{T}_\text{E}\text{X}$  の原稿ファイルは, 本項で示したような作業を繰り返すことで作成できます. したがって,  $\text{T}_\text{E}\text{X}\text{M}\text{A}\text{C}$  のような総合環境を利用することで,  $\text{T}_\text{E}\text{X}$  の原稿の作成は非常に容易になります.

$\text{T}_\text{E}\text{X}\text{M}\text{A}\text{C}$  の各マクロを起動するキーの操作が覚えにくいというのであれば, 「Ctrl」 + 「[」キーをまず覚えてください. このキー操作によって, 各種命令の入力補助とモジュール操作を可能にする `texmac98.mac` というマクロが起動します. また「Ctrl」 + 「]」キーは作成したファイルをコンパイルするときに使うので, 覚えておくと便利です. さらにもう1つ, `SuperTeX.MAC` (ちよてふ) の使い方についても紹介しておきます. これは少しコマンドを覚えてきたときにとっても便利な機能です. 具体例で示します. まず,

```
\sec
```

と入力してこのマクロを起動するとします. このマクロは「Ctrl」 + 「w」キーを押すと起動します. すると,

```
section
```

```
sec
```

という二つのメニューが開きますので, 実際に入力したいものを選んでください. これが, もし

```
\sect
```

まで入力した段階でマクロを起動したなら,

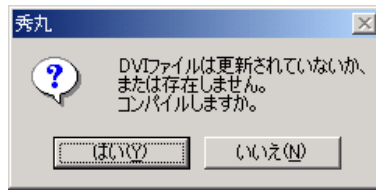


図 4.11: 作成したファイルのコンパイル

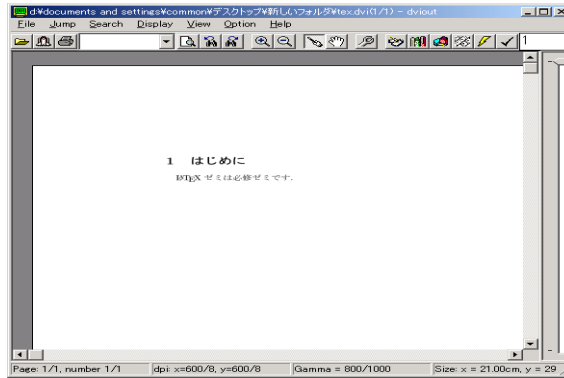


図 4.12: 例文のプレビュー

```
\section
```

といきなり確定されます。

```
\s
```

の段階でマクロを起動したとすると、s から始まる多くの命令のリストが表示されます(図 4.14)。その中から、望みのものを選択してください。

さらに、この場合(1文字だけ入力して起動)に限り辞書の学習機能を備えています。例えば、\s と入力してこのマクロを起動し、sharp を選択したとします。すると、次に同じように起動すると、sharp がメニューの先頭に位置しているはずでず。

また、beg と入力してマクロを起動したとします。すると、多くのメニューが表示されるとおもいます。これは、入力したい命令が begin であると判断したため、TEX.MAC の 'begin{}タイプ' のメニューが表示されているわけです。そこで、abstract を選択したとすれば

```
\begin{abstract}
```

```
\end{abstract}
```

と表示され、真ん中の空行にカーソルが移動します。

最後に、

```
\begin{flushr
```

の状態マクロを起動したとします。すると、いきなり

```
\begin{flushright}
```

```
\end{flushright}
```

と表示されます。

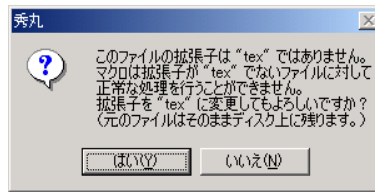


図 4.13: 拡張子を TeX にする

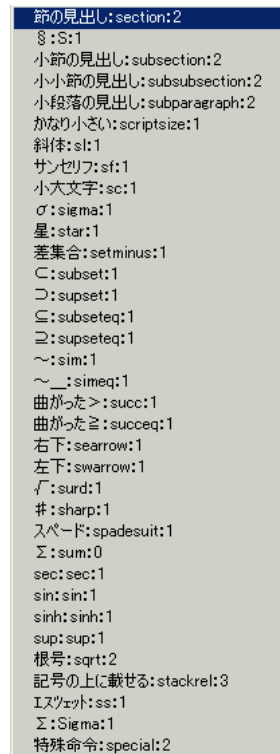


図 4.14: s から始まる命令のリスト

最後にこのマクロを使用するにあたって注意事項を述べておきます。 \記号を忘れたまま、命令の一部を入力してこのマクロを起動すると、マクロは \記号を求めて、ファイルの先頭方向へ走っていきます。そして、 \記号を発見したとしても、それは決して入力者の意図通りの結果にはなりません。ご注意ください。また、 \begin の {} 内の補完に際しては、マクロとしては {} を探しているだけです。例えば、 \section{f1} と入力して間違っこのマクロを起動すると、おかしなことになってしまいますので、その場合は ESC キーを押してください。また、 \beg{abst} とかいて、 \begin 命令自体も abstract も共に補完してもらおうというのは甘い考えです。

以上が TeXMAC を利用した原稿ファイル作成の概要です。

## 第5章 課題

### 第一回 L<sup>A</sup>T<sub>E</sub>X ゼミ課題

名前を書こう today

#### 5.1 L<sup>A</sup>T<sub>E</sub>X を使おう

##### 5.1.1 T<sub>E</sub>XMAC で練習

種々の

書体や

大きさを  
使いこなそう.

#### 5.2 自己紹介で練習

##### 5.2.1 生い立ち

私は,1977年8月27日生まれの22歳です. 奈良智弁学園高校出身です.

##### 5.2.2 趣味

- ・ 車を走らせる<sup>1</sup>
- ・ 映画を見る

##### 5.2.3 課題について

次のページのソースを参照し, マクロを使って簡単な自己紹介文章を作ってください.

---

<sup>1</sup>私の愛車はシルビアです.

```

\documentclass[a4paper,10pt]{jarticle}

\title{第一回\LaTeX ゼミ課題}
\author{名前を書こう}
\date{today}
\pagestyle{plain}

%%%%%    TEXT START    %%%%%
\begin{document}
\maketitle
\section{\LaTeX を使おう}
\subsection{\TeX MAC で練習}

\textrm{{\tiny 種々の}}\
\textit{{\small 書体や}}\
\textgt{{\LARGE 大きさを}}\
\textbf{{\Huge 使いこなそう。}}

\section{自己紹介で練習}
\subsection{生い立ち}
私は,1977年8月27日生まれの22歳です。奈良智弁学園高校出身です。
\subsection{趣味}
\begin{itemize}
\item {車を走らせる}\footnote{私の愛車はシルビアです。}
\item {映画を見る}
\end{itemize}
\subsection{課題について}
このページのソースを参照し,マクロを使って簡単な自己紹介文章を作ってください。

\end{document}

```