

高校文実解説冊子

第 *I* 講

書類作成要綱

第 Beta 1 版 / 2022 年 05 月 10 日

第 71 回文化祭実行委員会

文責：麻生（装飾 / 総務 / 庶務 / 広報）

書類作成要綱 概要

【概要】

この解説冊子は、今年度より導入された \TeX を使用した資料・要綱の作り方について解説していきます。

2021 年度 2 月より試行錯誤を繰り返してきた \LaTeX ですが、3 ヶ月の時を経てついに完成版を出すことができました。大変長らくお待たせしました。

まずはみなさんが \TeX に慣れ、以前の Word の半分の時間で効率よく資料が作成できるようになることを目標にしてほしいと思います。日本語が崩壊していて読みにくい箇所も多々あるかとは思いますが、どうか最後までお読みいただけると幸いです。

【目次】

① 導入

- ☐ (1) \TeX の基本
- ☐ (2) 環境構築
- ☐ (3) \LaTeX とは

② 資料用 \LaTeX

- ☐ (1) 共通事項
- ☐ (2) 基本書類
- ☐ (3) 回答用紙
- ☐ (4) リーフレット

③ 冊子用 \LaTeX

- ☐ (1) 要綱・本
- ☐ (2) 複数人で執筆する際は

④ 周辺知識

- ☐ (1) 作成上の注意
- ☐ (2) 各種テンプレート
- ☐ (3) ライセンス

※ この冊子は 25 ページで構成されています。乱丁・落丁がないか今一度ご確認ください。

【本書の使い方】

本書は全 4 回分で構成されています。

【テスト】

復習テスト・確認テスト・総復習テスト等のテストは実施しませんが、4 回を通して少しでも早く書類を作成できるようになることを目標とします。

【ノート】

この冊子は書き込み形式になっています。必要に応じてメモを取るようにしてください。別にノートを用意する必要はありません。

【宿題】

(1) パソコンの環境を整える

第 1 回の宿題は、自分のパソコンで \TeX がかける環境を整えることです。

(2) 冊子内容の復習

第 2～3 回の宿題は、 \LaTeX の機能を知り、自力で書けるようになることです。

(3) 実践

第 4 回の終了後は、各自資料を作ってください。

【作成者】

\LaTeX 及びこの冊子の作成者は、72 期 麻生 (Asosan) です。ご意見・修正依頼等は、Gafe メール (72.asoo.ko-sei@tsukukoma-gafe.org) までお願いします。またわからないことがあれば、いつでも文実 Discord の【# 02a-tex-system】チャンネルにて質問してください。

【バージョン】

書類作成要綱の更新履歴です。5/20 の更新を持って第 1 版は完成となります。

バージョン	更新日時	備考
第 B0.1 版	2022 / 4 / 25	1-2, 2-1 公開
第 B1 版	2022 / 5 / 10	第 3 章以外公開
第 1 版	2022 / 5 / 20	全章完成版公開 (予定)

1-1 T_EXの基本

【概要】の部分で従来の文実 Word フォーマットを T_EX に変えると宣言しましたが、これを読んでいる皆さんの中には「T_EX...? T_EX って何? (kRd)」状態になっている方もいるかと思います。

この章では、「T_EX の概要」について解説していきます。

1 概要

TeX (テフ, テック, 蝶) は、スタンフォード大の数学者である Donald Ervin Knuth 氏が製作した組版システムです。「組版」とは印刷用語で、活字を組んで版 (印刷用データ) を作ることを意味します。決してプログラミング言語ではありません。

TeX には次のような特徴があります。

【T_EX の特徴】

- ・ オープンソースソフトなので、無料で入手でき、自由に中身を調べたり改良したりできます。また商用利用等も自由にできます。
- ・ T_EX は、Windows でも Mac や Linux などの Unix 系 OS でも全く同じ動作をするので、入力と同じなら原理的には全く同じ出力が得られます。
- ・ T_EX への入力はテキスト形式なので、普通のテキストエディタで読み書きでき、再利用・データベース化が容易です。
- ・ 自動ハイフネーション・ペアカーニング (欧文文字幅を自動的に調整)・リガチャ (fi などの合字処理)・孤立行処理等の高度な組版技術が組み込まれています。
- ・ 特に数式の表記に優れています。
- ・ T_EX ファイル単体では数 kB 程度しかなく、非常に軽いです。Discord の 8MB 送信制限にも問題なく対応できます。

2 処理・出力方式

TeX のテキストファイルを作成したら (これは次項で解説します)、このファイルを「出力」して PDF に変換します。出力する際に働くのが「エンジン」です。L^AT_EX には次のような種類があります。

- ・ pL^AT_EX: 長年日本語文書に使用されていた標準
- ・ upL^AT_EX: 上が Unicode 文字に対応
- ・ LuaL^AT_EX: 新しいタイプ①
- ・ pdfL^AT_EX: 新しいタイプ②

各エンジンは内部で次のような処理を行っています。

- ・ p, up: T_EX ファイル → dvi ファイル → PDF 等
- ・ Lua, pdf: T_EX ファイル → PDF (ダイレクト)

B_uL^AT_EX で使用するエンジンは、LuaL^AT_EX です。

3 T_EX ファイルの基本

▶ 構造

① \documentclass[<options>]{jlrq}

② <プリアンブル>

③ \begin{document}

<本文>

④ \end{document}

T_EX ファイルの構造は上のようになっています。

- ① documentclass: その書類の基本設定を指定します (例えば本文の文字サイズや書類の大きさなど)。<options>の部分にはそれぞれの作成物によって違ったものが入ります。B_uL^AT_EX では jlrq というクラスファイルを使用しますが、他に日本語標準のものとして jsarticle 等があります。
- ② プリアンブル: スタイルファイル (後述) の読み込み、自作命令 (マクロという、後述) の定義等を行う場所です。
- ③ 本文: begin document, end document の2つで囲まれた領域に、書類の本文を記述していきます。

▶ テキストを打つ

TeX ファイルにただ文字を打っていただけで通常の文章を打つことができます。

Google ドキュメントの初期画面と同じ状態です。

▶ 命令

書類には文章以外にも、タイトルやリスト、テキストの装飾 (下線等) を使ったり、画像を貼ったりする必要があります。これらを実現するには、T_EX 固有の「命令」を使います。命令には以下の2種類のタイプがあります。

【命令の種類】

- ・ 環境：リストなど特定の範囲のレイアウトを変更したいとき、これを実現するために必要となる環境と呼びます。使用するときは該当箇所を`\begin{<環境名>}`と`\end{<環境名>}`で囲います。
- ・ コマンド：「環境」以外の命令のことで、`\<コマンド名>{<情報 1>}{<情報 2>}...` のようにして記述します。

Google ドキュメントのメニューバーでデザインを施すのと同じ要領です。TeX はこれを全てテキスト状態で書くことができるのです。

またこれらの命令をまとめたものを スタイルファイル といい、BunTeX はこのスタイルファイルの一種になります(後述)。

4 TeX で PDF 書類を作成する

TeX で書類を作成する流れは以下の通りです。

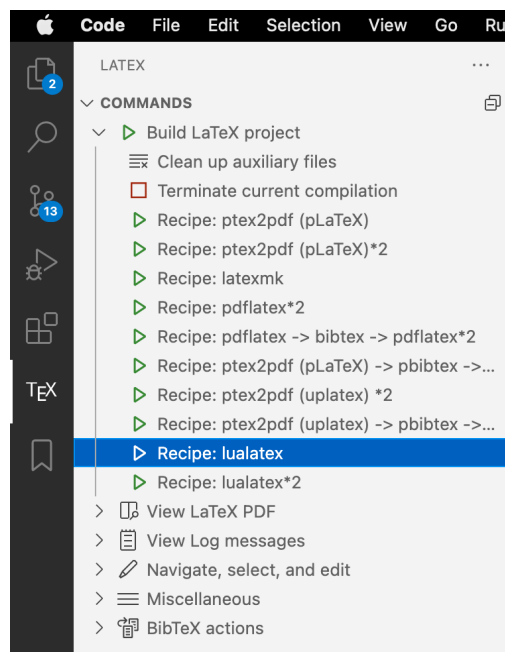
- ① VSCode を開く (インストールについては 1-2 参照)
- ② VSCode でファイルを作りたいフォルダを選択して開く
- ③ 拡張子が `.tex` となるファイルを作成する (これを TeX ファイルという) (Fig1.1 参照)
- ④ ③ で作成した TeX ファイルに文章を書いて書類を作る
- ⑤ LuaTeX を用いて「出力」を行う (下の枠組みを参照)
→ TeX ファイルと同じフォルダに PDF が生成される
- ⑥ ④⑤ を繰り返し行って仕上げていく (VSCode で TeX ファイルと PDF を横並びにしておくとうわかりやすいだろう)
- ⑦ 文章中の間違いやレイアウトのミスがないか校閲を行う
- ⑧ 完成



Fig 4.1 左の丸がファイル作成, 右の丸がフォルダ作成

【VSCode の TeX 操作パネルの使用法】

- (1) 出力する際に発生する、`.aux` や `.log` 等の不要なファイルを一括削除
- (2) 途中で出力を強制停止させる
- (3) pdfLaTeX の出力
- (4) upLaTeX の出力 (BunTeX では使用しない)
- (5) LuaTeX の出力
- (6) LuaTeX の出力を 2 回行う



5 エラーが起きた時は

TeX ファイルの出力の際にエラーが発生する場合があります。VSCode 下側の色付きバーの部分に、赤いバツのマークが出たら出力失敗です (通常はチェックマークがでます)。VSCode 下側の OUTPUT を選択すると、`log` ファイルと呼ばれる、LuaTeX エンジンが動いた際の動作ログを確認することができます。多くの場合、このログの最後の方に「!...」のようにしてエラーが出てきます。

主なエラーの原因は以下の通りです。エラーが出た際は落ち着いてひとつひとつ見直していきましょう。

▶ 定義されていない命令を使っている

BunTeX または TeX で定義されていない命令を使用している可能性があります。命令のスペルミスによるものが多いです。

▷ 解決方法：VSCode で赤く波線が引かれた部分の周辺が該当箇所です。この資料を参照して命令名が合っている

か確認してください。

▶ 括弧の数があっていない

命令の括弧の数があっていない場合もエラーが起きやすいです。

▷ 解決方法：括弧の対応関係を確認してください。

▶ 環境がおかしい

`\begin{}`と`\end{}`が対応していなかったり、環境名が違っていたりするとエラーになります。

▷ 解決方法：環境の対応関係や名称を確認してください。

▶ 使用不可能な文字を使用している

LT_εTeX では# や\ など、そのままでは出力できない半角文字（特殊文字）がいくつかあります。^{*2}

▷ 解決方法：「LaTeX 記号名」等で記号の出し方を検索する。該当する文字を直接使いたい場合は、`\verb|< 記号>` のようにする。

▶ Asosanに聞く

Discord に投げただけであればお答えできます。スクリーンショットとともに送ってください。

また TeX はどの行でエラーになっているかがわからない（わかりにくい）場合があります。その場合エラー箇所を探すのに苦労することになりますので、こまめに出力を行うようにしてください。

6 もっと TeX を知りたい！

今まで紹介してきた知識は TeX のほんの断片的な情報に過ぎません。もっと詳しく TeX について知りたい、あるいは数学式化学式を用いたレポートを作成したいなどの場合には、「改訂第8版 - LaTeX2_ε 美文書作成入門（奥村 春彦先生著）」を適宜参照するとよいでしょう。

ほとんどの基本的な項目をカバーしている本です。文実上本部にもおいてありますが、1冊買って手元に置いておくとう便利です。



^{*2} http://www.ic.daito.ac.jp/~mizutani/tex/special_characters.html を参照するとよい。

1-2 環境構築

この章では、 \TeX 本体、 \LaTeX スタイルファイル、さらにはフォントファイル等のインストール方法について解説していきます。以下の手順でファイル一式をインストールすることで、自分のパソコンで資料やレポートを作れる環境が整います。

【おしながき】

- ☐ (1) \TeX のインストール^{*1}
- ☐ (2) Visual Studio Code の導入
- ☐ (3) \LaTeX スタイルファイルのインストール
- ☐ (4) その他のファイル
- ☐ (5) GitHub

1 \TeX のインストール

まずはあらゆる \TeX ファイルをパソコンでコンパイルするために必要となる環境構築を行います。なお、Windows と Mac で多少手順に違いがありますので、自分のパソコンに合わせて選択してください。

▶ Windows: TeX Liveをインストールする

ダウンロードページ (<https://www.tug.org/texlive/>) にアクセスして「TeXLive2022」をダウンロードし、表示されるダイアログに従ってインストールしてください。ファイルサイズが 4.5GB と非常に大きいので、事前に本体のストレージに余裕があるかを確認しておくようにしてください。

▶ Mac: MacTeXをインストールする

ダウンロードページ (<https://www.tug.org/mactex/>) にアクセスして「MacTeX-2022」をダウンロードし、表示されるダイアログに従ってインストールしてください。

▶ 動作チェック

インストールには 3 時間ほど要します。気長に待ってください。

インストールが終わったら、パソコンのメニューから「コマンドプロンプト」(Mac の場合はターミナル) を起動し、「`lualatex -version`」と打ち込んで実行します。画面にエラーが出ず、バージョン名等がでてきたらインストール成功です。

2 Visual Studio Code の導入

前のセクションで \TeX をコンパイルすることができるようになりましたが、このままでは毎回の書類作成時に「メモ帳を開いて、編集して、コマンドプロンプトで実行」という手順を踏まなければいけないので非常に不便です。そこで Visual Studio Code (以下では VSCode と省略) というエディタを導入することで、気軽に \TeX を書く作業ができるようになります。

▶ VSCodeのダウンロード

公式サイト (<https://code.visualstudio.com/download>) から自分のパソコンにあったバージョンを選択してダウンロード&インストールします。

▶ \LaTeX 拡張機能を入れる

次に拡張機能を入れていきます。VSCode を起動したら、左側のメニューバーの上から 5 番目にある「Extensions」を選択します。検索窓から「Japanese Language Pack」と「LaTeX Workshop」を検索してインストールしてください。

終わったら一度アプリの再起動を行っておきましょう。

▶ \LaTeX 拡張を設定する

先ほどの「拡張機能」の画面に移動し、以下の写真で示す手順で「settings.json」を開きます。

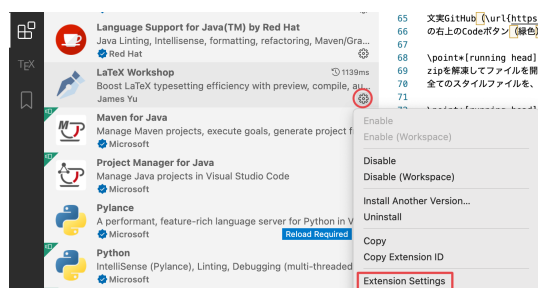


Fig 2.2 赤い部分を順に選択してください

^{*1} <https://qiita.com/passive-radio/items/623c9a35e86b6666b89e> を参照しました。

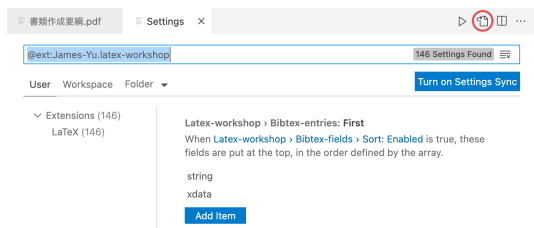


Fig 2.3 赤い丸を押すと setting.json が開きます

そこに GitHub にある「vssetup.txt」のコードをコピー＆ペーストしてください。

3 BunTeX スタイルファイルのインストール

▶ GitHubからインストール

文実 GitHub (<https://github.com/Asosan/BunTeX-System>) の右上の Code ボタン (緑色) から、zip ファイルをダウンロードします。

▶ スタイルファイルの配置

zip を解凍してファイルを開き、「Resources/BunTeX」と「Resources/Style」以下にある全てのスタイルファイルを、自分のパソコンの「/usr/share/texlive/texmf-dist」(Mac の場合は /usr/local/texlive/texmf-local/tex/latex) に配置します。

▶ 反映させる

コマンドプロンプト / ターミナルから、「sudo mktexlsr」を実行して TeX 本体に行われた変更 (今回はスタイルファイルの追加) の反映を行います。終わったらパソコンを再起動しておきましょう。

4 その他のファイル

▶ フォントファイル

資料テンプレートは、さきほど GitHub からダウンロードしたファイルの「Resources/Fonts」の中にあります。

Windows の場合は、パソコンのメニューから設定→個人用設定→フォントの順に選択して、すべてのフォントファイルをドラック&ドロップします。

Mac の場合は、Font Book の中に、まだパソコンの中がないフォントをドラックします (例えばヒラギノ系統はデフォルトでインストールされているので必要ありません)。

これらのフォントの一部には Asosan の Mac から引っ張ってきたものがあります。当然ですが、システムにプリインストールされているフォントを他者に配布することは認められていません。これらのフォントは、文実で TeX を

用いて書類を作る時以外には絶対に使用しないでください。

▶ 資料テンプレート

「Resources/Templates」の中に入っています。各資料の説明は後の章で行います。

5 GitHub

GitHub とは、「Git」というバージョン管理ツールをクラウド上で使用できるようにした Web サービスです。「いつ誰がどこを編集したのか」や「最新のバージョンはどれになるのか」などを明確にすることができ、大変便利です。

GitHub を使用したい場合は、公式サイト (<https://github.com>) よりアカウントを作成し、作成後に文実用 GitHub 組織「TkBunjitsuOfficial」に参加してください。既に組織に所属している人に招待してもらう必要があります。

- ——— ■
- メモ ■

1-3 BunTeX とは

1 概要

BunTeX (文TeX) は「ブンテフ」と読みます。これは文実の資料・要綱のためのスタイルファイル (1-1-3 参照) 群です。

旧文実 Word テンプレートや、鉄緑会のテキストやプリントを参考に、新たに独自のデザインの書類テンプレートを作成しました。

2 経緯

みなさんが LaTeX で文章を書く時、プリアンブル (1-1-3 参照) に何も書かないと LaTeX 標準のデザインで作成されますが、このままでは味気ない書類ができてしまいます。そこでプリアンブルに設定を変更したり、命令 (1-1-3 参照) を組むなどしたりして独自の施す必要が出てきます。ただしこれらの作業を 1 から自分でやろうとすると非常に手間がかかります。

そこで登場するのが、これらの設定や命令を一括にまとめたスタイルファイルです。これを作成者は「文実 TeX」、略して「文 TeX」と名づけて今に至ります。

3 構成

BunTeX は 3 つのスタイルファイルで構成されています。

▶ BunTeXB

要綱等の製本を必要とするような分厚い冊子を作成する際に使用します (B は Booklet の意味)。

要綱にふさわしい表紙や見出し、リストや画像等のデザインを行います。

▶ BunTeXC

どの書類にも共通して必要なスタイルファイルです (C は Common の意味)。

パッケージ (“拡張機能” 的な意味) の追加、欧文和文フォントの設定、テキスト装飾やロゴの定義等を内部で行います。

▶ BunTeXD

会議資料・配布書類・回答用紙・小冊子等を作成する際に使用します (D は Documents の意味)。

それぞれのページスタイルや見出し、さらにはリストや書き込み用空欄等の定義も一括で行うものです。

▶ 使用する際は

これらのスタイルファイルを目的によって以下のように使い分けます。

{KEY}

- ・ 資料用 BunTeX: BunTeXC, BunTeXD (資料・小冊子)
- ・ 冊子用 BunTeX: BunTeXB, BunTeXC (要綱・製本を必要とする冊子)

4 疑問がある場合は

BunTeX について何かわからないことがある場合は、作成者・72 期麻生 (Asosan) の Gafe メール (72.asoo.kosei@tsukukoma-gafe.org)、または文実 Discord の【# 02a-tex-system】チャンネルにて質問してください。



【第2回：資料用 \LaTeX 】

2-1 共通事項

ここからは資料用 \LaTeX の使い方について解説していきます。なお全て書くのは非常に手間なので、一部以下のように略して掲載します。

【省略記号一覧】

[A]: \backslash <コマンド名><文字列>
 [B]: \begin <命令の名前>[<オプション>]{<情報 1>}...
 <コンテンツ>
 \end <命令の名前>
 [C]: \backslash <コマンド名>*[running head]{<文字列>
 [D]: {<コマンド名>
 [E]: \backslash <コマンド名><{<情報 1>}<{<情報 2>}...

※ <>の部分を変えていく感じです。

※ <文字列>には任意の文字列を入れてください。

※ <コンテンツ>には複数行にわたり、他のコマンドを含むものなどが入ります。

※ 専門用語で [A] のことをマクロ、[B] のことを環境と呼びます。

※ *[running head] はつけなくても良い。

1 種類

資料用 \LaTeX で作れる書類は以下の通りです。

【作成可能な書類一覧】

- ・ 定例会資料
- ・ 生徒部提出資料・ 掲示資料
- ・ 解答用紙
- ・ 冊子 / リーフレット（デコ責会議等）

※ デコ責会議の冊子は、各部門が配布する資料類を一冊にまとめたものです。

2 前提

コンパイルは「 \LaTeX 」で行います。プリアンブル（ \LaTeX の基本 - セクション 3 参照）には以下を記述し、buntexc、buntexd の 2 種類のスタイルファイルを読み込みます。

【プリアンブルの記述方法】

```
\documentclass[<options>]{jlreq}

\usepackage{buntexc}
\usepackage{buntexd}
```

```
\begin{document}
```

```
<本文>
```

```
\end{document}
```

※ <options>の部分はテンプレートによって違うものが入る

▶ 改行

\TeX ファイル内で 1 行空白行を挟むと改行されます（通常の改行は出力結果に何の影響も及ぼしません）。ただし 2 行以上空白にしたとしても、出力されるのは改行のみで、空白行は出力されません。空白行を挿入する以外に文章中で強制的に改行したい場合は、その箇所に「 \backslash 」を記述してください。

▶ 余白

余白は縦・横方向それぞれ次のような命令があります。

【横余白】

- ・ ~: 半角空白
- ・ : 全角空白（日本語キーボードでスペースを押す）
- ・ \hspace <長さ>
- ・ \hspace <長さ>
- ・ \hfill : 空白を横に限界まで伸ばす

※ 半角空欄は半角スペースを入れることによっても出力できますが、複数の半角スペースを打っても、出力されるのは 1 つだけです。

【縦余白】

- ・ \vspace <長さ>
- ・ \vspace <長さ>
- ・ \vfill : 空白を縦に限界まで伸ばす

また長さの単位は以下の通りです。長さには自然数だけでなく、負の数や小数を指定することもできます。

【単位】

- ・ mm, cm: お馴染みの SI 単位系
- ・ \zw : 和文文字の横幅と同じ
- ・ \zh : 和文文字の縦幅と同じ

3 見出し

見出しは 3 段階に分かれています。階層に応じて以下のように指定してください。

【見出し 1】

- ・ [A] section：番号がつきます
- ・ [C] sectionb
- ・ [C] sectionc
- ・ [C] sectiond
- ・ [C] sectiont

▷ b～d, t は番号がつきません。若干デザインが変わるだけなので気分を選んでください。

【見出し 2】

[C] point：▶がでて太字下線になります。

【キーポイント】

[A] checking：▷がついて若干インデントが変わります。

4 テキスト装飾

テキストの一部を強調したり、テキストを変えたい場合に使用するコマンド一覧です。フォントのデフォルトはヒラギノ明朝 / ヒラギノゴシック / Palatino です。

【スタイル】

- ・ [A] gothic：囲んだ部分をゴシックに
- ・ [A] boldtext：囲んだ部分を太字に
- ・ [A] uline：アンダーライン
- ・ [A] uwave：波線
- ・ [A] mline：取り消し線
- ・ [A] boldwave：太字かつ波線
- ・ [A] highlighter：マーカーを引きます

【フォントサイズ】

`\selectsize{<フォントサイズ>}{<行送り>}`

※ 単位は pt です

【和文フォント】

- ・ [A] HiraMaruJ：ヒラギノ丸ゴシック
- ・ [A] HiraKakusix：ヒラギノゴシック W6（極太字）
- ・ [A] HuiFont：ふい字フォント
- ・ [A] Kyokasyo：游教科書体

【欧文フォント】

- ・ [A] LatinModern： \TeX 標準
- ・ [A] Helvetica
- ・ [A] HiramIn：ヒラギノ明朝
- ・ [A] HiraMaru：ヒラギノ丸ゴシック
- ・ [A] HiraKakusixE：ヒラギノゴシック W6（極太字）
- ・ [A] GenEiNombre：ノンブル（ページ番号）用フォント

5 リスト

[B] の...の部分に `\item` で要素をどんどん並べていくとリストになります。

【リスト】

- ・ [B] reitemize：itemize のインデントを修正したもの
- ・ [B] enumbrackets：(1)(2)...
- ・ [B] enumsquarebrackets：□(1)のように番号にチェックボックスがつく
- ・ [B] enumcircle：①②...
- ・ [B] enumsquare：□囲みの数字

参考

リストの使用例

上の枠囲みの中のリストのコード例を掲載します。

```
\begin{reitemize}
  \item [B] reitemize：itemize のインデントを修正
  したもの
  \item [B] enumbrackets：(1)(2)
  \item [B] enumsquarebrackets：□(1)のように番
  号にチェックボックスがつく
  \item [B] enumcircle：①②
  \item [B] enumsquare：□囲みの数字
\end{reitemize}
```

6 枠囲み

これを使用すると、コンテンツ全体を枠で囲うことができようになります。本文から分けたり、強調したりといった使い方ができます。

【枠囲みの種類】

命令の名前	情報 1	オプション
framebox-simple	タイトル	
framebox-simplified	タイトル	サブタイトル
framebox-key		
framebox-ref	タイトル	
framebox-brackets	タイトル	サブタイトル
framebox-practice	タイトル	

- ・ [B] framebox-simple：単純な枠囲み
- ・ [B] framebox-simplified：上にサブタイトルを追加したもの
- ・ [B] framebox-key：鉄○会の【KEY】
- ・ [B] framebox-ref：参考
- ・ [B] framebox-brackets：鉤括弧風の枠
- ・ [B] framebox-practice：練習問題

7 図・画像

図や画像を入りたい場合は、以下を指定します。

▶ 通常の画像

2 枚横並びまであります。2 段組の場合は、1 枚画像 (70mm) を指定することを推奨します。

【1 枚画像】

- ・ [E] singleimage{<横幅>}{<画像のパス>}：通常
- ・ [E] wrapfig{<横幅>}{<画像のパス>}{<説明>}：テキスト回り込み
- ・ [E] singleimagecap{<横幅>}{<画像のパス>}{<説明>}：キャプション付

【2 枚画像】

- ・ [E] doubleimage{<横幅 1>}{<画像 1 のパス>}{<画像 2 のパス>}：通常
- ・ [E] doubleimagecap{<横幅 1>}{<画像 1 のパス>}{<説明 1>}{<画像 2 のパス>}{<説明 2>}：キャプション付

▶ QRコード

QR コードは 3 枚横並びまでできます。Footnote 等リンクを貼ることを強く推奨します。

【QR コード】

- ・ [E] singleqr{<画像のパス>}{<説明>}：テキスト回り込み
- ・ [E] singleqrc{<画像のパス>}{<説明>}：中央配置
- ・ [E] doubleqr{<画像 1 のパス>}{<説明 1>}{<画像 2 のパス>}{<説明 2>}：2 枚横並び
- ・ [E] tripleqr{<画像 1 のパス>}{<説明 1>}{<画像 2 のパス>}{<説明 2>}{<画像 3 のパス>}{<説明 3>}：3 枚横並び (B5:1 段組, B4:2 段組以上推奨)

8 表

LaTeX の表は、自分で一から書くと非常に手間がかかります。ここでは外部サイトを用いて簡単に表を作成する方法を紹介します。

▶ 表を作成する

Tables Generator (https://www.tablesgenerator.com/latex_tables#) を使用して表を作成していきます。メニューから表のサイズ・枠線・テキスト装飾等を各セル個別に設定することができます (もちろん選択の仕方次第で表全体にスタイルを適用することもできます)。

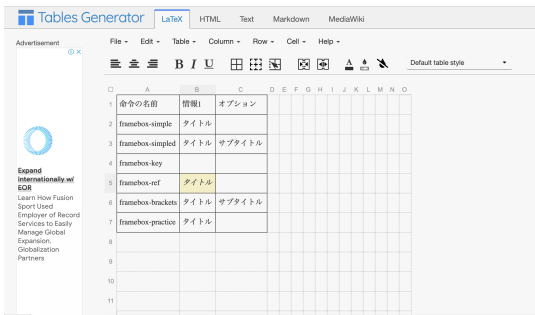


Fig.8.4 Tables Generator の編集画面

▶ 修正

Fig.2.2 のような表のコードが生成されたらこれをコピーして該当箇所貼り付けます。ただしこのままでは使うことができません²、以下のように軽微な修正が必要となります。

Result (click "Generate" to refresh)

```

1 \begin{table}
2 \begin{tabular}{|l|l|l|}
3 \hline
4 命令の名前 & 情報1 & オプション \\ \hline
5 framebox-simple & タイトル & \\ \hline
6 framebox-simplified & タイトル & サブタイトル \\ \hline
7 framebox-key & & \\ \hline
8 framebox-ref & タイトル & \\ \hline
9 framebox-brackets & タイトル & サブタイトル \\ \hline
10 framebox-practice & タイトル & \\ \hline
11 \end{tabular}
12 \end{table}

```

Fig 8.5 Tables Generator

- ① 赤枠の部分に「H」と打ち込みます。
- ② 緑枠の部分の「tabular」を「tabularx」に変更します。
- ③ 紫の部分に{<表の横幅>}を追加します。二段組の場合は基本的に70mmで、枠囲み内では64～66mm、一段組は状況に応じて最大163mmを指定すると良いでしょう。
- ④ 青枠の部分で表の行寄せを変更します。

参考 行寄せのコマンド

青枠に指定するコマンドについて解説します。

- ・ l: 左寄せ・幅は1番長い文字列に合う
- ・ c: 中央寄せ・同上
- ・ r: 右寄せ・同上
- ・ L: 左寄せ・幅は l, c, r で指定したものを除いて、指定された表の幅内で均等に長さをとって配置します。
- ・ C: 中央寄せ・同上
- ・ R: 右寄せ・同上
- ・ | : 区切り線を引く・連続で並べると二重線, 三重線...になる

参考 表の一例

上の「枠囲みの種類」の中で使用した表のコードを掲載しておきます。

```

\begin{table}[H]
\begin{tabularx}{64mm}{|X|c|c|}
\hline
命令の名前 & 情報1 & オプション \\ \hline
framebox-simple & タイトル & \\ \hline
framebox-simplified & タイトル & サブタイトル \\ \hline
framebox-key & & \\ \hline
framebox-ref & タイトル & \\ \hline
framebox-brackets & タイトル & サブタイトル \\ \hline
framebox-practice & タイトル & \\ \hline
\end{tabularx}
\end{table}

```

9 その他

注釈や URL、文責等の細々した命令集です。

*2 一応そのままでも表は出力されますが、ダサいです。tabular よりは横幅を指定できる tabularx の方が任意性が高いので、基本的には手順に従うようにしてください。

【注釈】

- ・ \footnotemark[N]: 注釈をつけたいテキストの直後に記述する / N には自然数が入る
- ・ \footnotetext[N]{<説明>}: 注釈を書いて出力 / N には footnotemark に対応する自然数が入る

【URL】

- ・ \url{<URL>}: 直接 URL を出力する / 注釈で使うと良い
- ・ \href{<URL>}{<テキスト>}: テキストにリンクを貼る (直接 URL を見ることはできない / 非推奨)

【書類末に使用】

- ・ [D] responsibility{<責任者名>}{<所属>}: 文責 (所属には慣例的に部門名が用いられる)
- ・ [D] allresponsibility{<責任者名>}{<所属>}: 統括文責 (同上)
- ・ [D] distribution{<日付>}: 配布日付
- ・ [D] lastpage: 書類のページ総数

【ロゴ等】

- ・ [D] snowman: 雪だるまの文字
- ・ [D] BunTeX: メインロゴ
- ・ [D] BunTeXC, BunTeXB, BunTeXD
- ・ [D] BunTeXJ: 日本語版のロゴ

参考 URL の使用例

筑駒文化祭\footnotemark[1]は、毎年10月か11月に開催されています。
当日はたくさんのJK・JDが来場します。
...
\footnotetext[1]{\url{https://tsukukoma.bunkasai.info}}を参照のこと}

10 付番方法

今年度より、生徒側の書類管理効率化のため、対外に出す資料にのみ付番をすることになりました。以下に付番のルールを記載します。

デモ責会議の資料のみ付番方法が異なりますので注意し

てください。

▶ 通常の付番方法

【通常の付番】

Xnn-mmxxp

- ・先頭の「X」は、資料の配布先ごとに別のアルファベットを入れます。「E」は生徒部・職員会議（教員）用、「S」は生徒向け配布、「P」は掲示資料、「U」は宇田川先生に提出する資料です。
- ・「nn」には、各部門の識別番号が入ります。
- ・「mm」には、資料の配布月が入ります。
- ・「xx」は、同配布先、同部門、同月の資料を作成順に付番していきます。
- ・回答用紙の場合は「p」の部分に「A」と書きます。他の場合には特に指定する必要はありません。

▷ 例 1：総務から教員 / デコ運営方針について / 5 月の教員提出の総務資料で 3 番目の場合→ E01-0503

▷ 例 2：庶務から生徒向け / 文実通信第 10 号 / 10 月の生徒向けの庶務資料で 5 番目の場合→ S02-1005

▷ 例 3：第 5 回装飾委員会会合 / 9 月の委員会配布の装飾資料で 2 番目の場合→ S12-0902

▶ デコ責会議資料に対する付番方法

【デコ責会議資料に対する付番】

Dnn-ttxxp

- ・先頭の「D」は、デコ責で配布された資料であることを示します。
- ・「nn」には、第 N 回デコ責会議の「 $N \in \mathbb{N}$ 」が 2 桁で入ります。
- ・「tt」には、各部門の識別番号が入ります。
- ・「xx」には、同じ回で同じ部門から配布された資料に対して順番につけます。基本的に若い番号から、説明用資料→提出物説明資料の順（同じ場合は重要度順）に番号を振ってください。
- ・回答用紙の場合は「p」の部分に「A」と書きます。他の場合には特に指定する必要はありません。

▷ 例 1：第 2 回デコ責会議 総務部門 / デコ責広報 / 第 5 回総務配布資料の中で 1 枚目の場合→ D05-0101

▷ 例 2：第 7 回デコ責会議 器材部門 / 特殊器材について / 第 7 回器材配布資料の中で 4 枚目の場合→ D07-0604

▷ 例 3：第 12 回デコ責会議 SCC 部門 / 「当日動線について」に対応する解答用紙 / 第 12 回 SCC 配布資料の中で 2 枚目の場合→ D12-1102A

参考 各部門の番号

番号	部門名	番号	部門名
01	総務	07	電波
02	庶務	08	広報
03	財務	09	企画
04	印刷	10	審査
05	器材	11	SCC
06	電力	12	装飾

※ Discord の番号とは異なります。



【第2回：資料用 B_uT_EX】

2-2 基本書類

5/20に追記します

メモ

2-3 回答用紙

ここからは、デコ責会議や会合のアンケート等に使用する、回答用紙の作成方法を解説していきます。

1 構成

回答用紙は以下のような構成になっています。

【構成】

タイトル → 書類情報 → 署名欄 → 問題 → 解答欄 →
問題 → ... → 解答欄 → (生徒確認欄) → (文実担当欄)
→ 最終チェック欄

※ 括弧はなくてもよい

それぞれを出力するための命令は以下で解説していきます。

2 ページスタイル

テンプレート「answerb4.tex」, 「answerb5.tex」(4-2 参照)にあるコードについて少し説明します。これらはあらかじめ書かれていますので、自分で記述する必要はありません。

【ページスタイル】

- ・ [B] answersheetb4 : B4 回答用紙用の環境で、本文全体をこれで囲う必要がある
- ・ [B] answersheetb5 : B5 回答用紙用の環境、上と同様

3 ヘッダー

回答用紙は、ヘッダーに書類を識別するためのカテゴリ名と番号を記述する必要があります。テンプレート(4-3 参照)の 5, 6 行目の \lefttitle, \righttitle には以下を指定します。

【ヘッダー】

- ・ \lefttitle : 「会議名 or 第~71~回文化祭実行委員会 + (全角空白) + ~配布」(~には部門名が入る)
- ・ \righttitle : 「提出物番号 : ~」(付番方法は 2-1-10 参照)

4 タイトル

回答用紙のタイトルには以下を使用します。

【タイトル】

- ・ [A] answertitle

5 書類情報・署名欄

「書類情報」とは、説明資料番号・提出物番号・提出期限・提出先のことです。また、署名欄はデコ責署名や顧問署名等を一括で出力できます。それぞれ以下のように記述します。

【書類情報・署名欄】

- ・ [E] answersheetinfo{<説明書類番号>}{<提出物番号>}{<締切 (5/27 のような形式で) >}{<締切曜日>}{<提出先>}
- ・ [D] groupinfo : 団体名, デコ責署名 2 名分, 顧問署名
- ・ [D] groupinfooad : 団体名, デコ責署名 2 名分のみ

6 問題

回答用紙では各セクションごとに、問題欄と回答欄を設けます。ここでは問題欄で使用できる命令を説明します。

▶ 前提

問題欄に配置するコンテンツは、「[B] question」で囲った上で、そのセクションのタイトルを指定する必要があります。

参考 問題欄と回答欄の構成例

(タイトル, 書類情報, 署名欄)

```
% セクション 1
\begin{question}{<セクション 1 のタイトル>}
  (セクション 1 の問題)
\end{question}
\begin{answer}
  (セクション 1 の回答欄)
\end{answer}

% セクション 2
\begin{question}{<セクション 2 のタイトル>}
  (セクション 2 の問題)
\end{question}
\begin{answer}
  (セクション 2 の回答欄)
```

```
\end{answer}

% セクション 3
\dots

(確認欄等)
```

▶ リスト

問題作成に使用するリストは通常のものとは異なります。以下の 3 種類を使用してください。ただし、セクションに対して問題が 1 つしかない場合には使用する必要はありません。

リストの使用方法是 2-1-5 を参照してください

【リスト】

- [B] enumbracketsans : (1)(2)...
- [B] enumcircleans : ①②...
- [B] enumsquareans : □ 囲みの数字

▶ 問題欄が必要な場合

問題欄が必要なく、回答欄のみでよい場合は以下のようになります。

【問題欄が必要な場合】

```
\begin{question}{<セクションのタイトル>}
% 一行開ける！
\end{question}
\begin{answer}
\dots
\end{answer}
```

7 回答欄

次に問題の項目に対する回答欄を作ります。短文回答、長文回答、マークシート等を指定できます。

▶ 前提

回答欄に配置するコンテンツは、「[B] answer」で囲います。これを行うと、セクションの末尾に確認欄が自動的に出力されます。前項の例を参照してください。

▶ リスト・表

リストは前項、表は 2-1-7 を参照して作成してください。表の横幅は 150mm 程度が良いでしょう。

▶ 記述式回答欄

短文・長文それぞれ以下のように指定します。短文回答の場合、複数下線を並べて複数行回答にすることもできます。

【短文回答】

- `\kasen[]{} :` 右端まで下線を出力する。{}内に文字を打つと、下線の上にあらかじめ文字を書くことができる。
- `\kasen[width=<長さ>]{} :` 指定した長さ分の下線を出力する。

【長文回答】

- `\writebox[height=<長さ>]{} :` 書き込みボックスを出力する。{}内に文字を打つと、ボックス内にあらかじめ文字を書くことができる。

▶ 選択式回答欄

マークシートによる選択式回答欄です。表との相性が非常に良いです。

【長文回答】

- `{\egg{<1 桁の半角数字>}} :` 数字入りのマークシート
- `{\egg{~}} :` 数字なしのマークシート
- `{\eggg{<1 桁の半角数字>}} :` あらかじめ塗りつぶされたマークシート (数字あり)
- `{\eggg{~}} :` あらかじめ塗りつぶされたマークシート (数字なし)
- `\markegg{<長さ>}{<1 桁の半角数字>}{<説明>} :` 1 行に複数のマークシートを掲載したい場合に、各項目の長さ・番号・説明を一度に指定できる命令

参考

マークシートと表の組み合わせ例

```
\begin{table}[H]
\centering
\begin{tabularx}{150mm}{|c|C|C|C|C|C|}\hline
日付 (第 1 週・第 2 週) & 5/23 & 5/24 & \\\hline
第 1 週 / 15:30~ & \egg{~} & \egg{~} & \\\hline
第 1 週 / 16:00~ & \egg{~} & \egg{~} & \\\hline
...
\end{tabularx}
\end{table}
```


8 確認欄系統

書類の最後には確認欄を出力します。確認欄は 4 種類あります。最終チェック欄はマストで出力する必要が、checkcolumn と confirmationcolumn は任意です。

また scoring はセクションの途中で確認欄を出力したい場合に使用します（回答欄の途中に配置してください）。

【確認欄】

- ・ \scoring{<タイトル>}：セクションの末尾以外の任意の位置に配置し、確認欄を出力
- ・ \checkcolumn：生徒向けの提出前の確認欄。書類の抜け落ちがないか、ボールペンで記入しているか等の項目がある。
- ・ \confirmationcolumn{<部門名>}：部門担当者用の備考欄等を出力する。
- ・ \finalcheck：文実用の最終チェック欄

■ — ■
メモ
■ — ■

【第2回：資料用 Bu_μT_EX】

2-4 リーフレット

5/20に追記します

メモ

3-1 要綱・本

要綱や長い本を作成する際にはこの「冊子用 $\text{B}_\text{u}\text{T}_{\text{E}}\text{X}$ 」を使用します。以下では各機能・マクロ、さらには本作成の Tips について説明していきます。この中にはみなさんが使用する必要のないマクロも含まれていることと思います。この章の最後によく使用するマクロを一覧にまとめておきましたので、必要に応じて参照してください。

1 前提

要綱・本作成の際に使用するスタイルファイルは、「 $\text{B}_\text{u}\text{T}_{\text{E}}\text{X}\text{C}$ 」と「 $\text{B}_\text{u}\text{T}_{\text{E}}\text{X}\text{B}$ 」です。また、コンパイルには資料用同様「 $\text{Lua}_{\text{E}}\text{T}_{\text{E}}\text{X}$ 」を使用します。^{*1}

5/20 に追記します「3-2 複数人で執筆する際は」も同様

メモ

^{*1} ちなみに鉄○会のテキストは $\text{up}_{\text{L}}\text{A}_{\text{E}}\text{X}$ を使っているみたいです。時代遅れですね。

【第4回：周辺知識】

4-1 作成上の注意

5/20に追記します

メモ

【第4回：周辺知識】

4-2 各種テンプレート・使用例

この章では、GitHub 上の「BunTeX-System/Resources/Templates」内にある、各種テンプレートについて解説していきます。資料を作る際は基本的にこれらのテンプレートをダウンロードして作成するようにしてください。

なおテンプレート中のコードは基本的に変更する必要はありませんが、`<...>`の部分は資料によって任意のテキストを設定する必要がありますので、適宜変更するようにしてください。

またこれらのテンプレートを用いて作成した資料の一例を、「BunTeX-System/Resources/Examples」に掲載しています。2, 3 章で解説した各命令を使用し、実際に配布した実例です。必要に応じて参照してください。

1 基本書類

基本書類は種類によって以下のように使い分けます。

▶ 定例会資料・付属委員会資料

これら2種類はB4版で作成します。資料担当者が作成したメインファイルの中に、`\input{<ファイルのパス>}`で各議題提供者が作った議題ごとの \TeX ファイルを読み込んでいきます。見出しには基本的に、番号付きのsectionを使用してください。

【定例会・付属委員会 - テンプレート】

- ・メイン：**normalb4.tex**（資料担当者が作成）
- ・各議題：**agenda.tex**（議題提供者が作成 / 資料担当者がinputで読み込み）

※ 見出しにはsectionを使用する

【定例会・付属委員会 - 実例】

- ・第2回装飾委員会合資料 (**normalb4**) .tex

▶ 全校掲示資料・生徒配布資料・教員提出資料

これらは全てB5版で作成します。見出しには基本的に、番号がつかないsectionb, sectionc, sectiondを使用してください。

【全校掲示・生徒配布・教員提出】

normalb5.tex を用いる

※ 見出しにはsectionb, sectionc, sectiondを使用する

2 解答用紙

解答用紙にはB4版とB5版があります。推奨サイズはB4です。

【解答用紙 - テンプレート】

- ・B4：**answerb4.tex**（基本的に推奨）
- ・B5：**answerb5.tex**

【解答用紙 - 実例】

- ・B4：**第2次中間報告書・教室希望調査書 (answerb4) .tex**（基本的に推奨）
- ・B5：**面談希望調査書 (answerb5) .tex**

3 リーフレット

デコ責会議やその他補足冊子の作成に使用するリーフレットは、以下のように複数のテンプレートがあります。定例会同様、メインファイルに各資料の \TeX ファイルを読み込んでいく形式を取ります。

冊子の構成等については2-4を参照してください。

【リーフレット - テンプレート】

- ・メイン：**leaflet.tex**
- ・表表紙：**leaflet-coverf.tex**（デフォルトではデコ責会議15回分の爪インデックスがついている）
- ・裏表紙：**leaflet-coverb.tex**（同上）
- ・連絡リーフ：**leaflet-coverb.tex**（必ず冊子の見開きページにつける）
- ・各資料：**leaflet-doc.tex**（各資料の担当者はこれを編集すること）
- ・冊子訂正：**leaflet-correct.tex**（冊子訂正が付属する場合はこれを使用すること）

【リーフレット - 実例】

- ・第1回デコ責会議資料 (**leaflet**) .tex
- ・書類作成要綱 (**leaflet**) .tex

4 要綱系

5 結合・分割

直接 \LaTeX に関係があるわけではありませんが、以下のテンプレートを使用することにより、PDF を結合したり分割したりすることができます。結合とは例えば「B5 2 枚を B4 1 枚」に、分割とは「B4 1 枚を B5 2 枚」にすることを言います。

テンプレートの 1 行目に対象となる PDF ファイル名を記述します。この際、分割する対象となる PDF ファイル名は英語にする必要があることに注意してください。ただしこの場合のみ、コンパイルに使用するのは \pdfLaTeX です。

【結合・分割】

- ・ 隣り合う 2 枚を結合：**combine.tex**
- ・ 半分に分割：**cut.tex**

※ 分割する対象となる PDF ファイル名は英語にしなければならないことに注意

メモ

4-3 ライセンス

1 $\text{B}\mu\text{T}\text{E}\text{X}$ のライセンス



$\text{B}\mu\text{T}\text{E}\text{X}$ はクリエイティブ・コモンズ 表示 - 非営利 - 継承 4.0 国際 ライセンスの下に提供されています。以下のライセンスに反しない限り無償で利用できます。またここでの「 $\text{B}\mu\text{T}\text{E}\text{X}$ 」は、 $\text{B}\mu\text{T}\text{E}\text{X}\text{B}$ ・ $\text{B}\mu\text{T}\text{E}\text{X}\text{C}$ ・ $\text{B}\mu\text{T}\text{E}\text{X}\text{D}$ と、各種テンプレートからなる一連のファイル群であると規定します。

▶ BY-表示

文実以外の団体で $\text{B}\mu\text{T}\text{E}\text{X}$ を使用したい場合は、文書に適切なクレジット（著作者：Asosan）を示す必要があります。

▶ NC-非営利

いかなる場合であっても、営利目的で $\text{B}\mu\text{T}\text{E}\text{X}$ を頒布することは認められていません。

▶ SA-継承

今後 $\text{Lua}\text{L}\text{A}\text{T}\text{E}\text{X}$ のバージョンアップ等に合わせて $\text{B}\mu\text{T}\text{E}\text{X}$ を修正する必要がある場合は、適切に行って構いません。ただし加工した場合には、元の $\text{B}\mu\text{T}\text{E}\text{X}$ と同じライセンスを持ち、原則としてそれに遵守して頒布を行う必要があります。

また、2023/3月までの改変は原則として禁止します。

2 フォントのライセンス

1-2 でも言及した通り、配布したフォントの一部には Asosan の Mac から直接引っ張ってきたものがあります。当然ですが、システムにプリインストールされているフォントを他者に配布することは認められていません。具体的には以下のフォントです。

- ・ Helvetica Neue (欧)
- ・ HiraKakuProN W1-W9 (和/欧)
- ・ HiraMaruProN (和/欧)
- ・ HiraMinProN (和/欧)
- ・ Kyokasho (和)

- ・ Palatino (欧)

これらのフォントは、文実で TEX を用いて書類を作る時以外には絶対に使用しないでください。

また、これ以外のフォントはフリーフォントになります。

以下に一覧を掲載しておきます。

- ・ GenEiNombre
- ・ HuiFont
- ・ LatinModern
- ・ Mikachan

各フォントはそれぞれのガイドラインに従って適切に使用してください。



氏名：_____：