Quick-Word-To-LaTeX Documentation

ICPR

April 2022

Abstract

This is the abstract of this document. What you type here will be put in the abstract. If you want this document as a template, you should save it as a `.dotx`, and place it in the Custom Office Templates in your documents folder, assuming you're using Windows. You can also save this as a new style set in **Design.**
Keep in mind that the abstract title, the bolded "**Abstract**", must use the *Abstract Title* style (or is unbolded), and must say `Abstract` exactly, word-for-word, and is case sensitive. Abstracts must be positioned at the start of the document, after the title, date, author, and so on. The abstract ends at the **next** section. It doesn't matter if the abstract is in the *Abstract* style or not.

1   Introduction

This document serves as documentation for some modules provided by the `Quick-word-to-LaTeX` program, which uses Pandoc. The output by the program should look **very similar** to how the content appears in Microsoft Word. I've uploaded the `.docx`, `.pdf`, and `.tex` files for this document. For the PDF, I've uploaded the one generated from Microsoft Word and LaTeX.

To quickly switch font styles if you're on Windows, use CTRL+SHIFT+S, type the style name, and press enter. There are shortcut keywords for some styles, such as `H1,` `H2,` and so on. For Mac OS users, just press `COMMAND-OPTION-1` to switch to heading 1, `COMMAND-OPTION-2` to switch to heading 2, and so on.

2   Caveats

Pandoc isn't perfect, and so is `Quick-word-to-LaTeX`. If you put some things in MS Word that tend to break LaTeX, the document can't be converted properly. The following things in this list are the things you should **avoid.**

What is it	Why doesn't it work?	Chances of breaking the converter
Tables inside tables (UNLESS the table inside is used to define an environment, like *framed,* **and vice versa)**	LaTeX does not like nested tables, and nested tables that aren't used to define environments will confuse the converter.	Always
Not using inline mode for images	This can break Pandoc, `Quick-word-to-LaTeX`, and the LaTeX compiler.	Most of the time

Using the inline code style to write source code	Inline code is analogous to `\texttt`. Source code is analogous to the verbatim environment.	Most of the time
Using CTRL+B or *CTRL+I* in the equation editor	Doing so may confuse Pandoc. Sometimes, when you type something like `sin` into MS Word's equation editor and decide to erase it. If, for some reason, you didn't press CTRL+I and what you're typing in the equation editor is forced upright, just retype the entire equation or delete what you think is causing it.	Sometimes
Creating fake headers by using bolded text, instead of using headers in the style sheet	It works, but it looks ugly, and it won't be recognized as a header.	Does not break the converter, but it looks ugly
Placing Microsoft Word's built-in table of contents into the document	Don't do that. Enable table of contents in the configuration file if you want to do that.	Does not break the converter, but it looks ugly
Making source or inline code lines too long	This creates overfull H-boxes. It's also very bad practice to make code lines extra-long.	Does not break the converter, but it looks ugly.
Changing the font of the MS Word document by using CTRL+A, and changing the font from HOME	This is one way to make the fonts very inconsistent in a Microsoft Word document. The better way to change fonts is by doing to Design > Fonts.	Has nothing to do with `Quick-word-to-LaTeX`. This is just good practice.

You may notice that the table contains bolded text on the left, due to the fact that I selected a preset in the table design tab. If a design causes some parts of the table text to be formatted **bold** or *slanted*, that formatting will **not** carry over to the LaTeX file.

3   Equations

There are three types of equations you can write in Microsoft Word. Equations are created with the ALT+= shortcut:

- **Inline math.** For example, $y = mx + b$ is an inline equation.
- **Display math.** Math that is placed on a new paragraph. The equation below is display math:

$$ax^2 + bx + c$$

- **Aligned / stacked math.** Display math equations stacked on top of each other using SHIFT+ENTER. I'll call them aligned math because `Quick-word-to-LaTeX` will automatically align them, regardless of how they appear in Microsoft Word.

$$\int \sin^3(x)\,dx$$

$$\int \sin(x) \cdot \sin^2(x)\,dx$$

$$\int \sin(x)\,(1 - \cos^2(x))dx$$

$$u = \cos(x)$$

$$du = \sin(x)\,dx$$

$$\int (1 - u^2)\,du$$

$$u - \frac{1}{3}u^3$$

$$\cos(x) - \frac{1}{3}\cos^3(x)$$

When at least two equations are stacked on top of each other with SHIFT+ENTER, they will be aligned. If you want to have more control over alignments, you can try using matrices as equation arrays.

3.1 Long equations

A long equation is any equation that is over 110 characters long (by default). This is about the horizontal length of a page. If fractions are used, only the longer part of the fraction will count towards the equation character count. In Microsoft word, long equations will likely be broken automatically.

$$a + b + c + d + e + f + g + h + i + j + k + l + m + n + o$$
$$= q + r + s + t + u + v + w + x + y + z + a + b + c + d + e + f + g + h$$

Because LaTeX can't automatically break equations, I've given the ability for `Quick-word-to-LaTeX` to break long equations apart. The characters that can split equations are:

$$<, >, \le, \ge, =, \wedge, \vee, \subset, \subseteq, \Rightarrow$$

However, if these characters don't appear for a while, then these characters can also split equations:

$$+, -, \times$$

Meaning when equations are split, the second line onwards will always start with one of those above. This is done for **both** display-style and aligned / stacked equations.

3.2 Equation tips

You should check this article out: https://github.com/ICPRplshelp/Quick-word-to-LaTeX-4/wiki/Microsoft-Word-Equation-Syntax/.

Here are some tips extracted from that article:

- Bracket sizing is done automatically.
- Most things done in braces in LaTeX equations are done with parentheses in Microsoft Word.
- Common functions like $\sin(x)$, $\cos(x)$, $\tan(x)$, $\min(x, y)$, $\max(x, y)$ are recognized functions and will not be slanted. Alternatively, you can use `\funcapply` after typing the words of a function to unslant the text. For example: `samplefunction\funcapply(...)`.
- Accents and vectors are done like this: $\vec{a}$ and $\vec{a}$. To type this, do the following keystrokes: `a\vec` then press space twice (to get the one on the right). To get $\vec{a}$, read the article I linked above.

- The syntax for matrices is [\matrix(@@&&)] for a 3×3 matrix. Pressing space afterwards will give you a 3×3 blank matrix for you to work in.
- Equation arrays are matrices without brackets.
- Systems of equations are matrices only with visible braces on the left. Here's an example of how to make one: {\matrix(x+2@x+4)\close. Note that Microsoft Word's \close is like LaTeX's \right. with the period.
- Math blackboard fonts like $\mathbb{Z}$ must be created using $\mathbb{Z}$. If you want to add this to the equation autocorrect, read the article I linked above. For mathcal, it's \scriptO for $\mathcal{O}$.
- Fractions are done automatically, like how Desmos handles them.
- When writing integrals, always hit space twice after typing \int if you want to create a double integral, otherwise Microsoft word will see what you typed as just another crowbar.
- When creating limits, integrals, summations, parentheses/brackets/braces, or some recognized functions, beware of what is inside it or not. See the figure below.

$$\int 324xdx + 323$$
$$\int 324xdx + 333$$
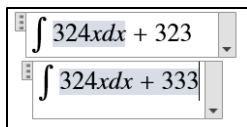
- The equation box on the top has $324xdx$ in the integral, but the equation box on the bottom has $324xdx + 333$ in the integral. Use the right arrow key to leave integral bounds.
- Sometimes, Microsoft word will glitch out and force you to type equation boxes at the bottom of the document. To fix this, copy regular text to the bottom of the document to get you out of this glitch.
- You can set up equation autocorrect entries. Look at the link above to figure out how to.

3.3   Equation fonts

If you don't like MS Word's default math font, Cambria Math, I strongly recommend using TeX Gyre Termes Math. Remember that fonts with the name "Math" at the end are the only fonts that support equations. To change the default font for equations, set it up in the equation options in Equations (your cursor must be inside an equation) > The extra settings within conversions > check the image:
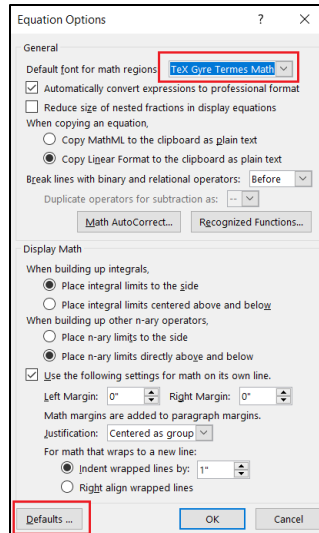
TexGyreTermes looks a lot like Times New Roman. It is also the least buggiest out of all the other Microsoft Word math fonts other than Cambria Math.

As a warning, if you use custom fonts, always export MS Word documents using "Microsoft Print to PDF." Otherwise, if you export as PDF from File > Export, issues can arise, namely fonts not being properly embedded.

4   Inline and Source code

If you want `inline` or source code in a Word document, consider using the WordTeX template or the Pandoc template. You should be able to google both.

This document makes use of the [WordTeX](#) template, so you can use this document as a template.

You can either save this document as a template and put it in the MS Word templates folder, **or** you can save this document's style by going to **Design,** opening the list of style sets, and saving the style set for this document.
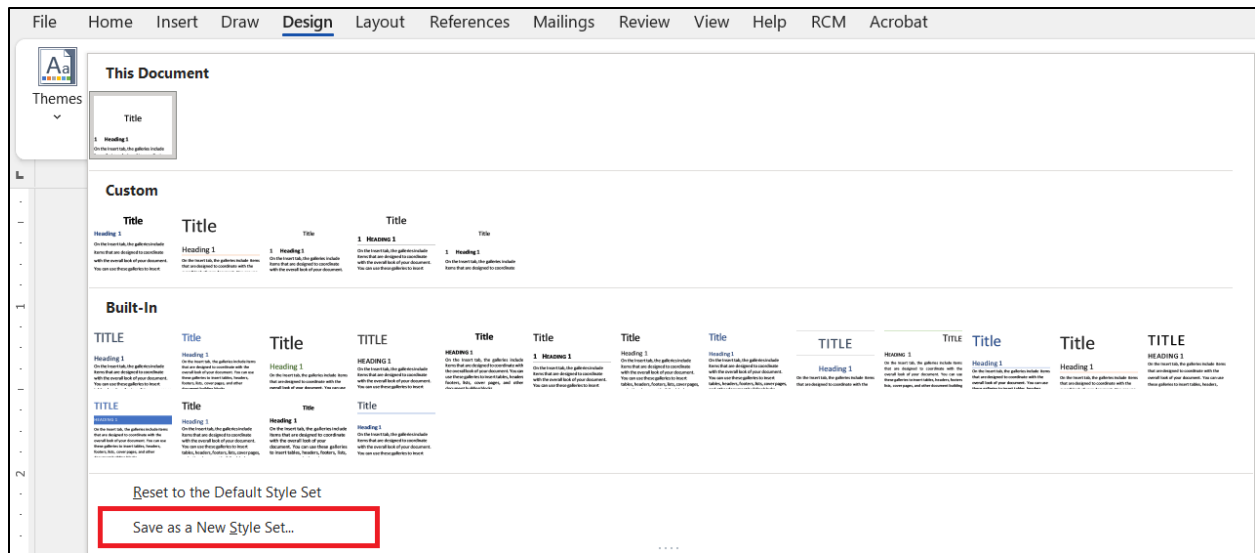
After opening one of the templates / styles sets that support code blocks, you can initiate a code block by choosing it from the styles pane.

Alternatively, you can use CTRL+SHIFT+S (Windows only) and type "Inline code" to change the style to code blocks. Or you can use macros. You can press CTRL+SPACE to revert back to normal text when typing Inline code, or by switching the style back to normal.

CTRL+SHIFT+S is useful if you want to quickly type headings. The WordTeX template, which this document is based on, has shortcuts: You can type H1 in the style bar to apply the heading 1 style very quickly. You can create footnotes with ALT+CTRL+F.[1]

For source code, on a new paragraph, set the style to source code. You can then paste in your code, **which must be formatted as plain text.**
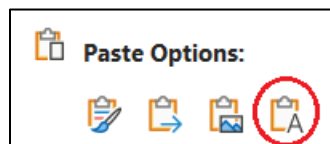
Pandoc does not recognize TABS used in Word documents, so always use four spaces to indent code. To revert back from source code to normal text, switch the style back to normal using CTRL+SHIFT+S (type "Normal" and press enter. This shortcut is Windows only; Mac OS users must set this up themselves). CTRL+SPACE does not work for source code because the paragraph's style is set to source code.

---

[1] Footnote.

Note: You may safely type LaTeX-like code in source code – it will not break.

Here's an example:

```python
# python
def open_file(file: str, allow_exceptions: bool = False) -> str:
    """Return file contents of any plain text file in the directory file.
    """
    if not allow_exceptions:
        with open(file, encoding='UTF-8') as f:
            file_text = f.read()
        return file_text
    else:
        try:
            with open(file, encoding='UTF-8') as f:
                file_text = f.read()
            return file_text
        except FileNotFoundError:
            return ''
```

4.1 Code highlighting

Code highlighting is very similar to the markdown syntax. Add the language on the **very first** word in your source code block. You may put #, //, or -- in front of the programming language. The languages supported by the `minted` package are the languages that are supported. It is **not** case-sensitive.

5 Figure Numbering and Labels

This module is when you want to reference figures, tables, or equations in a document. You can label the following things:

- Display-style equations
- Images
- Tables

SHIFT+ENTER is not used on this document other than in display-style equations.

5.1 Labeling Display-style equations

To number an equation, enter the equation followed by a #, followed by the equation numbering. For example, the following keystrokes produce the equation below: 9+10#(10)

$$1 + 12 \tag{1}$$

You can number equations however you like, but it is good practice to wrap them around a bracket. If you do, then the equation number is the text you've entered within the brackets. In this case, the equation number is 1. If you don't, then the equation number is the text you've added to the equation comment.

If I want to reference this equation, I will type "equation 1" or "Equation 1." The program will search for all instances of the phrase "equation 1" and make the 1 clickable.

You should not have any duplicate labels. This means each equation's number must be unique. `Quick-word-to-LaTeX` will try to make equations **look exactly how they appear in Microsoft Word.**

As a warning, equation labels should be alphanumeric with decimals. The same goes for all other labels. Failing to do so will remove the label from the equation. For example, $\left(\frac{4}{3}\right)$ as a label is unacceptable, and the label will be discarded (treated as if it were never there). Note that if an equation is numbered in Microsoft Word, it will stop having automatic line breaks, so avoid excessively long single-line equations if you decide to label them.

Long equations will have their numberings / labeling dropped before being split, because Microsoft Word does not give automatic line breaks to long equations. If "overfull h-boxes" appear in Microsoft Word, it means that you'll have to change something.

Equation labeling also works for aligned equations. For example:

$$1 + 2 + 8 \tag{2}$$
$$3 + 4 + 4 \tag{3}$$
$$4 + 5 + 3$$
$$7 + 8 + 2 \tag{4}$$

To show that labeling works, I'll type the following: equation 2, equation 3, and equation 4. Note that issues like these can appear:

$$1 + 2 + 3 \tag{5}$$
$$4 + 6 + 7$$
$$8 + 9 + 10 \tag{6}$$

If that happens, select the equations that are flushed left and align them to the center (CTRL+E) for Windows users, or COMMAND-E for Mac OS users.

5.2   Labeling Images

To add a label to an image, do the following:

Figure 1: This is an image.

To reference an image, type "figure 1." "Figure 1" also works. References work like equations. The image caption is `This is an image.` Denote the end of an image caption by creating a new paragraph. You should always caption images (however, the program will not crash if you don't).

You should not have any duplicate labels. This means multiple figures with the same label are not allowed.

Labels to images are optional. As a warning, image labels should be alphanumeric with decimals. The same goes for all other labels. Failing to do so will result in the label not being read.

It does not matter whether the image is centered or not. The same applies to the caption. Pandoc cannot tell the difference between center-aligned text and left-aligned text.

*"Figure 1: This is an image" should be below the image. No other text should be interrupting it, but you may press "enter" as many times as you want below the image before typing "Figure 1: This is an image." All that matters is that no other text is placed between. Also, **do not format the figure numbering in any way (other than coloring it).***

5.3   Labeling tables

Suppose I want to label this table.

Trial	Variable 1	Variable 2
1	3	5
2	4	6

Table 1: This is a table.

To reference this table, type "table 1." "Table 1" also work. References work like equations and images. Denote the end of a table caption by creating a new paragraph. You should always caption tables (however, the program will not crash if you don't).

You should not have any duplicate labels. This means multiple tables with the same label are not allowed.

Labels to tables are optional. Labels to images are optional. As a warning, image labels should be alphanumeric with decimals. The same goes for all other labels. Failing to do so will result in the label not being read.

*"Table 1: This is a table" should be below the image. No other text should be interrupting it, but you may press "enter" as many times as you want below the image before typing Table 1: This is a table" All that matters is that no other text is placed between. Also, **do not format the table numbering in any way (other than coloring it).***

5.4   Summary

Labelable object	How to label it (declaration)	How to reference it
Equation	$$\int f(x)dx \qquad (2)$$ Note that equations won't be labeled properly if placed in tables, so this one will look weird in the LaTeX file. It's not good practice to do this.	By typing "equation 2".
Image / Figure	[IMAGE] Image 2: This is an image.	By typing "figure 2".
Table	[TABLE] Table 2: This is the table.	By typing "table 2".

CTRL+F the words in quotation marks to see what will be affected.

6   Environments

Environments (mostly theorems) can be mimicked using Word or can be written in a way that it will be recognized by `Quick-word-to-LaTeX`.

6.1   Tables

You can use tables to mimic LaTeX environments. For the purposes of this document, we'll be using theorem-like environments. The following table will be translated to the following LaTeX code, assuming you're using the standard config:

Theorem
The name of the theorem
The text that goes in the theorem

Produces the following code:

```
# TeX
\begin{theorem}[The name of the theorem]
    The text that goes in the theorem
\end{theorem}
```

The text is color-coded here so you'll know exactly what maps to where. Also, it does not matter if the word "**Theorem**" is **bolded** or *slanted.*

The following

Theorem
The text that goes in the theorem

Produces the following code:

```
# TeX
\begin{theorem}
    The text that goes in the theorem
\end{theorem}
```

The following environments are supported by default (NOT case-sensitive):

- Theorem
- Lemma
- Corollary
- Proposition
- Conjecture
- Remark
- Note
- Claim
- Definition
- Condition
- Problem
- Example

You can define other environments in the configuration file. However, that is very advanced. Any table that does not use any of the environments in the list, unless specified in the config, will not be converted to an environment.

You can nest environments in other environments, or environments in tables.

6.2   Framing

Any 1x1 table will be converted to a `framed` environment. This means the following table will appear as the following code:

This text is framed. I can also frame equations: $$33 + 43 = 99$$

Produces the following code:

```
# TeX
\begin{framed}
    This text is framed. I can also frame equations:
    \[33+43=99\]
\end{framed}
```

6.3   Proofs

You can type proofs like how they appear in LaTeX. Here's an example:

Proof. Left to the reader. ■

The substring "*Proof.*" Indicates the start of a proof, and the box ■ indicates the end of a proof. The substring "*Proof.*" **MUST** be at the start of a new paragraph. (The word "*Proof.*" **must** be slanted, but it can be in the Normal style.)

The box ■ **MUST** be the QED symbol **inside** an equation. It must be this specific Unicode character, found here:
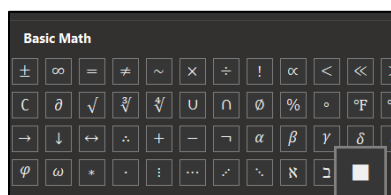
There is no default math autocorrect for this box. You should add it yourself into the equation autocorrect. Select and copy this black square, go to the equations tab in MS Word (your cursor must be within an equation for the equations tab to show), click the bottom right of "conversions," and add the black box to the equation autocorrect.

You may nest proofs in $\mathtt{framed}$ environments or other theorems, but I don't see any reason why to. Proofs may not be placed inside tables that are not used to create environments.

After a proof starts, this program looks for the nearest equation with a black square and ends the proof after that equation, and also removes the black square it detected. Multiple equations stacked with SHIFT+ENTER will be treated as one equation. If it cannot detect a black square, then it will end the proof at the next header or the end of the current environment; whichever comes first. For example, if you begin a proof in a framed environment, the proof will always end at the end of the framed environment if there are no black squares stopping it.

Standalone black squares will just show up like regular black squares.

6.4   LaTeX-like Theorems

The following text from MS Word in the frame produces the following LaTeX code:

> **Definition (The term of the definition).** This is the text within the definition.
> $$1 + 2 = 3$$
> this is some text within the definition.
> This text is no longer in the definition.

```
# TeX
\begin{definition}[The term of the definition]
    This is the text within the definition.
    \[1 + 2 = 3\]
    this is some text within the definition.
\end{definition}
```

```
This text is no longer in the definition.
```

The syntax is

> **Env (Arguments).** Description

The body of the environment ("Description", in this case) **cannot start as bolded text**. This will mess with the environment detection.

The end of the environment will be positioned

- On the next paragraph, **only if:**
  - First character of the next paragraph is uppercase

Equations will not cause an environment defined like this to end. Also, lists and equations will not break these types of environments (even if the first letter in them is capital).

The dash may be the dash that is typed (-) or the emdash (–).

Also, these syntaxes are also supported:

> **Definition** (Term)**.** Text within the environment (bold period at the end).
> **Definition (Term).** Text within the environment (all bold).
> **Definition** (Term). Text within the environment (no bold).
> **Definition.** Text within the environment (plain).
> **Definition – Term:** Text within the environment.

Don't worry if spaces are bolded – if a space character is adjacent to **both** a **bolded** character and a normal character, the space will be considered a normal character.

You cannot define these environments inside

- Lists like these.

However, you can define them in any table.

6.5   Quote

Quote environments can be placed using Microsoft Word's built in Quote (specifically this one) style:

> This is a quote.

Note: if you're not using this template, you may find the quote style to be slanted by default. The text will not be slanted in the LaTeX file that is output by `Quick-word-to-LaTeX`. If the text in the quote is not slanted by default, then *slanting* specific parts of the quote will slant only the parts you slanted in the LaTeX file. The same applies to default styling caused by using Microsoft Word's tables.

7   Citations

Citations can be typed such that it will be recognized by `Quick-word-to-LaTeX`. The bibliography style is default, but you can always change it in the preamble.

7.1   Setting up the bibliography

You can set up the bibliography in **either** of the two ways. Both ways require a level 1 heading named `Bibliography` in the document, with no bolding or *slanting*:

1. Copy-pasting what you would put in the `.bib` file inside the bibliography region (**must be named** `Bibliography` **word-for-word, with no formatting, and must be a level 1 header)** in the Microsoft Word document, **which must be formatted as source code, and the formatting must be continuous (format may not change back to Normal mid-way)**. You can see this done in the document if you scroll down to the bibliography section.
2. Having a `.bib` file ready. The program will prompt you to select one if:
   a. You did not enter the bibliography entries in the Microsoft Document, as stated above in bold, and
   b. There is a level 1 heading named `Bibliography` with no formatting. Then, everything in the bibliography section will be erased (up until the next level 1 heading).

7.2   In-text citations

To do an in-text citation assuming you have the bibliography set up, there are three ways to do so. Note that the author tag is `sampleArticle` in these examples. The text I've highlighted in orange is considered variable text – text not highlighted in orange must remain as-is.

1. **Citing the author only.** For example: "LaTeX is a software system for document preparation" (sampleArticle). The syntax is the author tag surrounded by parentheses.
2. **Citing the author and the page number.** For example: "LaTeX is a software system for document preparation" (sampleArticle, p. 1). The syntax is the author tag surrounded by parentheses, but after the author tag, place a comma, a space, the lowercase letter p followed by a period, another space, and the page number.
3. **Citing multiple authors.** For the purposes of this document, we'll be citing the same author multiple times: "LaTeX is a software system for document preparation" (sampleArticle, sampleArticle, sampleArticle). Page numbers are not supported here. The syntax is multiple author tags in the region surrounded by parentheses, each separated by a comma **and** precisely one space character.

8   Bibliography

```
@misc{sampleArticle,
   author = "{Wikipedia contributors}",
   title = "LaTeX --- {Wikipedia}{,} The Free Encyclopedia",
   year = "2022",
   howpublished = "\url{https://en.wikipedia.org/w/index.php?title=LaTeX&old
id=1083550063}",
   note = "[Online; accessed 21-April-2022]"
}
```

All text by this point for the rest of this section will not appear in the LaTeX document, because it is within the bibliography section. This paragraph has the `Normal` style, so it is not part of the bibliography anyways. Note that the bibliography must be continuous source code. If I pasted `.bib`-like text below, it will not be converted.

9   Further readings

Here are some readings that I find particularly useful:

- https://www.andrew.cmu.edu/user/twildenh/wordtex/WordTeXPaper.pdf
- https://www.youtube.com/watch?v=jlX_pThh7z8/
- https://unicode.org/notes/tn28/UTN28-PlainTextMath-v3.pdf
- https://en.wikibooks.org/wiki/Typing_Mathematics_in_Microsoft_Word
- https://support.microsoft.com/en-us/office/linear-format-equations-using-unicodemath-and-latex-in-word-2e00618d-b1fd-49d8-8cb4-8d17f25754f8