OneHotkey -- Math Formula Input Simplification Tool

This is a script that simplifies math formula inputs in `OneNote`, `Word` and `PowerPoint` with `AutoHotKey` script, e.g., `\a` for α (`\alpha`).

Demonstration video 1 (Early version):

AutoHotKey增强OneNote公式输入*测试1*哔哩哔哩_bilibili

这是一个用于简化 `OneNote`, `Word` 和 `PowerPoint` 中数学公式输入的 `AutoHotKey` 脚本，例如，`\a` 代表α (`\alpha`)。

This project is still updating. Your suggestions and contributions are welcome.

If the formulas aren't displayed correctly, go to README_EN.pdf.

Table of Contents

How to Use

1. Download and run `symbol_assist.exe`.
2. Input the code of the symbol, then press `Space` to get the symbol. For example, input `\a` and press `Space` to get α.
3. For editting the symbol mapping, please refer to Code Editing Guide. If you need help, go to the AutoHotKey official website.

4. To stop the script, right click the `H` icon in the system tray and select `Exit`.

Symbol Mapping

Overview

The script contains multiple symbol mappings, including Greek letters, math fonts, frequently used letters, and structures. The following is a list of some typical mappings. Make sure that you have entered the formula input mode with `Alt`+`=`.

Code	Output	Category	Source
`\a`	α	lowercase Greek letters	`\alpha`
`\D`	Δ	uppercase Greek letters	`\Delta`
`\R`, `\C`, `\Z`, `\N`	$\mathbb{R}, \mathbb{C}, \mathbb{Z}, \mathbb{N}$	frequently used letters	`\doubleR`, ...
`\do X`, `\sc X`, `\fr X`	$\mathbb{X}, \mathcal{X}, \mathfrak{X}$	fancy letter forms	`\doubleX`, `\scriptX`, `\frakturX`
`\m3`, `\m4`, ...	3 by 3 empty matrix, ...	matrices	`[\matrix(@@&&)]`, ...
`x\h`, `x\~`, `x\d2`	$\hat{x}, \tilde{x}, \ddot{x}$	modifiers	`\hat`, `\tilde`, `\ddot`
`\x`, `\X`, `\sq`, `\pa`, `\eq`	$\cdot, \times, \sqrt{\square}, \parallel, \equiv$	operators	`\cdot`, `\times`, `\sqrt`, `\parallel`, `\equiv`
`\pd`, `\di`, `\inf`	$\partial, \mathrm{d}, \infty$	frequently used symbols	`\partial`, `"d"`, `\infty`
`\ls`	$\begin{smallmatrix}\square\\\square\end{smallmatrix}P$	left super-and-lowerscript	`^_ P`
`\i`, `\j`, `\k`	$\mathrm{i}, \mathrm{j}, \mathrm{k}$	imaginary/quaternion symbols	`"i"`, `"j"`, `"k"`
`\ejw`	$e^{j\omega}$	complex exponential factor	`e^j\omega`

You shall notice that ` `(space) is commonly used, which is the key feature of OneNote formula input. Capital letter code should be inputted with `Shift`, not `CapsLock`.

Full Table

Frequently Used Letters

Code	Output	Source	Code	Output	Source
`\pd`	∂	`\partial{Space}`	`\di`	d	`"d"`
`\inf`	∞	`\infty{Space}`	`\Z`	$\mathbb{Z}$	`\doubleZ{Space}`
`\Q`	$\mathbb{Q}$	`\doubleQ{Space}`	`\R`	$\mathbb{R}$	`\doubleR{Space}`
`\N`	$\mathbb{N}$	`\doubleN{Space}`	`\C`	$\mathbb{C}$	`\doubleC{Space}`
`\J`	$\mathbb{J}$	`\doubleJ{Space}`	`\E`	$\mathbb{E}[]$	`\doubleE{Space}[]{Space}{Left}`

Operators

Code	Output	Source	Code	Output	Source
`\x`	$\cdot$	`\cdot{Space}`	`\X`	$\times$	`\times{Space}`
`\sq`	$\sqrt{\square}$	`\sqrt{Space 2}{Left}`	`\pa`	$\parallel$	`\parallel{Space}`
`\ss`	$\subset$	`\subset{Space}`	`\sse`	$\subseteq$	`\subseteq{Space}`
`\op`	$\oplus$	`\oplus{Space}`	`\ox`	$\otimes$	`\otimes{Space}`
`\od`	$\odot$	`\odot{Space}`	`\dd`	$\ddots$	`\ddots{Space}`
`\cd`	$\cdots$	`\cdots{Space}`	`\vd`	$\vdots$	`\vdots{Space}`
`\map`	$\mapsto$	`\mapsto{Space}`	`\pro`	$\propto$	`\propto{Space}`
`\as`	$\because$	`\because{Space}`	`\so`	$\therefore$	`\therefore{Space}`
`\eq`	$\equiv$	`\equiv`			

Greek Letters

Code	Output	Source	Code	Output	Source
`\a`	α	`\alpha{Space}`	`\b`	β	`\beta{Space}`
`\e`	ε	`\varepsilon{Space}`	`\ve`	ϵ	`\epsilon{Space}`
`\d`	δ	`\delta{Space}`	`\D`	Δ	`\Delta{Space}`
`\s`	σ	`\sigma{Space}`	`\S`	Σ	`\Sigma{Space}`
`\l`	λ	`\lambda{Space}`	`\L`	Λ	`\Lambda{Space}`
`\t`	θ	`\theta{Space}`	`\T`	Θ	`\Theta{Space}`

Code	Output	Source	Code	Output	Source
`\p`	ϕ	`\phi{Space}`	`\P`	Φ	`\Phi{Space}`
`\o`	ω	`\omega{Space}`	`\O`	Ω	`\Omega{Space}`
`\g`	γ	`\gamma{Space}`	`\G`	Γ	`\Gamma{Space}`

- `ve` means variant epsilon. For convenience, `\e` is set to ε and `\ve` is set to ϵ, which is different from their original code.

Matrix

Code	Output	Source
`\m4`	4 by 4 empty matrix	`[\matrix(@@@&&&){Space}]{Space}`
`\m3`	3 by 3 empty matrix	`[\matrix(@@&&){Space}]{Space}`
`\m2`	2 by 2 empty matrix	`[\matrix(@&){Space}]{Space}`
`\m`	empty matrix awaiting `&` `@` to set size.	`[]{Space}{Left}\matrix(){Left}`

Modifiers

Code	Output	Source
`\d1`	$\dot{x}$	`\dot{Space 2}`
`\d2`	$\ddot{x}$	`\ddot{Space 2}`
`\d3`	3 dots above	`\dddot{Space 2}`
`\d4`	4 dots above	`\ddddot{Space 2}`
`\~`	$\tilde{x}$	`\tilde{Space 2}`
`\v`	$\vec{x}$	`\vec{Space 2}`
`\h`	$\hat{x}$	`\hat{Space 2}`

Arrows

Code	Output	Source	Code	Output	Source
`\lr`	$\leftrightarrow$	`\leftrightarrow{Space}`	`\lrs`	$\leftrightarrows$	`\leftrightarrows{Enter}{Left}`
`\la`	$\leftarrow$	`\leftarrow{Space}`	`\La`	$\Leftarrow$	`\Leftarrow{Space}`
`\down`	$\downarrow$	`\downarrow{Space}`	`\up`	$\uparrow$	`\uparrow{Space}`
`\ul`	$\nwarrow$	`\nwarrow{Space}`	`\ur`	$\nearrow$	`\nearrow{Space}`
`\dl`	$\swarrow$	`\swarrow{Space}`	`\dr`	$\searrow$	`\searrow{Space}`

Symbols

Code	Output	Source	Code	Output	Source
`\de`	$^\circ$	`\degree{Space}`	`\st`	$\star$	`\star{Space}`

Structures

Code	Output	Source
`\r`	$\{\square$	`\right.{Left}`
`\ceil`	$\lceil\rceil$	`\lceil{Space}\rceil{Space 2}{Left}`
`\floor`	$\lfloor\rfloor$	`\lfloor{Space}\rfloor{Space 2}{Left}`
`\brak`	$\langle\rangle$	`\bra{Space}\ket{Space 2}{Left}`
`\ls`	$\begin{smallmatrix}\square\\\square\end{smallmatrix}P$	`^_ P {Left 4}`
`\fu`	$\mathrm{myfunction}\square$	`\funcapply`

- `\funcapply` is a little different from `\of`. Have a try by yourself!

Prefix for Fancy Letters

Code	Output	Source
`\sc`	$\mathcal{X}$	`\script`
`\do`	$\mathbb{X}$	`\double`
`\fr`	$\mathfrak{X}$	`\fraktur`

- For these mappings, your input should be like `\sc X`.

Recommendations

- Learn more about the math input from this document: UTN28-PlainTextMath-v3.pdf. Page 39~47 is useful.
- Input Unicode characters directly: https://github.com/gtj1/symbol_assist
- Intuitive Vim-like text cursor control: https://github.com/RUSRUSHB/AutoTextCursor

Experimental Features (In folder `experimental/`)

`key_combination.exe`

- Contains: Start formula inputting; Division line; Boxed text; Text block

`rus_hotkey.exe`

- Input Russian alphabets. They can be integrated into formula inputting.
- Format: `\`+Romanized Alphabet+`R`

- e.g., `\dR` generates д, `\DR` generates Д

Code Editing Guide

For editting the mapping, please: Edit `symbol_assist.ahk`, compile it with converter such as `Ahk2Exe` choosing `v1.1.37.01c1 U32 Ahk2Exe.exe` in `Options-Base File (.bin, .exe)`, and run the compiled `.exe` file. You are recommended to learn more about `AutoHotKey` from its website.

The code of `symbol_assist_OneNote.ahk` is very easy to understand, even if you have not learnt about `AutoHotKey`. For newcomers, the explanation of the code is as follows:

Each line of the code is a mapping of the input code to the output symbol. The format is `:(parameters):input::output`. For example, `::\a::\alpha` means that when you input `\a`, the script will output `\alpha`.

I added some parameters `co?`:

Parameter	Meaning
`c`	Case-sensitive. `\a` and `\A` are different.
`o`	Delete the `Space` you entered at the end.
`?`	Output formula even if you have typed something before the code. Otherwise, it will fail in cases like `x\h`