

OneHotkey -- 公式输入简化工具

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

English version of introduction: [README EN.md](#)

这一项目还在开发中, 欢迎你的建议和参与。

目录

- [OneHotkey -- 公式输入简化工具](#)
 - [目录](#)
 - [如何使用](#)
 - [符号映射](#)
 - [概览](#)
 - [完整映射表格](#)
 - [常用字符](#)
 - [运算符](#)
 - [希腊字母](#)
 - [矩阵](#)
 - [修饰符](#)
 - [箭头](#)
 - [其他符号](#)
 - [结构](#)
 - [花体字母](#)
 - [更多建议](#)
 - [映射自定义指南](#)

如何使用

1. 下载并运行[symbol assist OneNote.exe](#)。
2. 输入代码, 然后按 space 键输出符号与公式。例如, 输入 \a, 然后按 space 键, 就会输出 α 。
3. 想要自行修改公式映射, 请参考[映射自定义指南](#)。如果需要帮助, 请前往[AutoHotKey官网](#)。
4. 想要停止脚本, 用鼠标右键点击系统状态栏中的 H 图标, 然后选择 Exit。

符号映射

概览

此脚本包括多种类型的符号映射，包括希腊字母、数学字体、常用字母和结构。以下是其中的一些映射。请确保您已经使用 `Alt +=` 进入了公式输入模式。

输入	输出	分类	实际代码
<code>\a</code>	α	小写希腊字母	<code>\alpha</code>
<code>\D</code>	Δ	大写希腊字母	<code>\Delta</code>
<code>\R, \C, \Z, \N, \J</code>	$\mathbb{R}, \mathbb{C}, \mathbb{Z}, \mathbb{N}, \mathbb{J}$	常用双线体字母	<code>\doubleR, ...</code>
<code>\do x, \sc x, \fr x</code>	$\mathbb{X}, \mathcal{X}, \mathfrak{X}$	花体字母	<code>\doublex, \scriptx, \frakturx</code>
<code>\m3, \m4, ..., \m</code>	3×3空白矩阵, 4×4空白矩阵..., 待定形状矩阵	矩阵	<code>[\matrix{@@&&}], ...</code>
<code>x\h, x\~, x\d2</code>	$\hat{x}, \tilde{x}, \ddot{x}$	修饰符	<code>\hat, \tilde, \ddot</code>
<code>\x, \X, \sq, \pa, \eq</code>	$\cdot, \times, \sqrt{\square}, \parallel, \equiv$	运算符	<code>\cdot, \times, \sqrt, \parallel, \equiv</code>
<code>\pd, \di, \inf</code>	∂, d, ∞	常用符号	<code>\partial, "d", \infty</code>
<code>\ls</code>	$\square P$	左侧上下标	<code>_ P</code>
<code>\i, \j, \k</code>	i, j, k	虚数/四元数符号	<code>"i", "j", "k"</code>

你应该注意到 (空格键)是常用的，这是OneNote公式输入的关键特性。代码中的大写字母应该使用 `Shift` 键输入，而不是 `CapsLock` 键。

完整映射表格

常用字符

输入	输出	源代码	输入	输出	源代码
<code>\pd</code>	∂	<code>\partial{Space}</code>	<code>\di</code>	d	<code>"d"</code>
<code>\inf</code>	∞	<code>\infty{Space}</code>	<code>\R</code>	\mathbb{R}	<code>\doubleR{Space}</code>
<code>\Q</code>	\mathbb{Q}	<code>\doubleQ{Space}</code>	<code>\Z</code>	\mathbb{Z}	<code>\doubleZ{Space}</code>

输入	输出	源代码	输入	输出	源代码
<code>\N</code>	\mathbb{N}	<code>\doubleN{Space}</code>	<code>\C</code>	\mathbb{C}	<code>\doubleC{Space}</code>
<code>\J</code>	\mathbb{J}	<code>\doubleJ{Space}</code>			

运算符

输入	输出	源代码	输入	输出	源代码
<code>\x</code>	\cdot	<code>\cdot{Space}</code>	<code>\X</code>	\times	<code>\times{Space}</code>
<code>\sq</code>	$\sqrt{\square}$	<code>\sqrt{Space 2}{Left}</code>	<code>\pa</code>	\parallel	<code>\parallel{Space}</code>
<code>\ss</code>	\subset	<code>\subset{Space}</code>	<code>\sse</code>	\subseteq	<code>\subseteq{Space}</code>
<code>\op</code>	\oplus	<code>\oplus{Space}</code>	<code>\ox</code>	\otimes	<code>\otimes{Space}</code>
<code>\od</code>	\odot	<code>\odot{Space}</code>	<code>\dd</code>	\ddots	<code>\ddots{Space}</code>
<code>\cd</code>	\cdots	<code>\cdots{Space}</code>	<code>\vd</code>	\vdots	<code>\vdots{Space}</code>
<code>\map</code>	\mapsto	<code>\mapsto{Space}</code>	<code>\pro</code>	\propto	<code>\propto{Space}</code>
<code>\as</code>	\because	<code>\because{Space}</code>	<code>\so</code>	\therefore	<code>\therefore{Space}</code>
<code>\eq</code>	\equiv	<code>\equiv</code>			

希腊字母

输入	输出	源代码	输入	输出	源代码
<code>\a</code>	α	<code>\alpha{Space}</code>	<code>\b</code>	β	<code>\beta{Space}</code>
<code>\e</code>	ε	<code>\varepsilon{Space}</code>	<code>\ve</code>	ϵ	<code>\epsilon{Space}</code>
<code>\d</code>	δ	<code>\delta{Space}</code>	<code>\D</code>	Δ	<code>\Delta{Space}</code>
<code>\s</code>	σ	<code>\sigma{Space}</code>	<code>\S</code>	Σ	<code>\Sigma{Space}</code>
<code>\l</code>	λ	<code>\lambda{Space}</code>	<code>\L</code>	Λ	<code>\Lambda{Space}</code>
<code>\t</code>	θ	<code>\theta{Space}</code>	<code>\T</code>	Θ	<code>\Theta{Space}</code>
<code>\p</code>	ϕ	<code>\phi{Space}</code>	<code>\P</code>	Φ	<code>\Phi{Space}</code>
<code>\o</code>	ω	<code>\omega{Space}</code>	<code>\O</code>	Ω	<code>\Omega{Space}</code>
<code>\g</code>	γ	<code>\gamma{Space}</code>	<code>\G</code>	Γ	<code>\Gamma{Space}</code>

- `\ve` 指的是epsilon的变体(variant epsilon)。为了方便，`\e` 被设置为 ε ，`\ve` 被设置为 ϵ 。这和它们的原始代码相反。

矩阵

输入	输出	源代码
<code>\m4</code>	4×4空白矩阵	<code>[\matrix{@@@&&}{Space}]{Space}</code>
<code>\m3</code>	3×3空白矩阵	<code>[\matrix{@@&}{Space}]{Space}</code>
<code>\m2</code>	2×2空白矩阵	<code>[\matrix{@&}{Space}]{Space}</code>
<code>\m</code>	空白矩阵，输入 & @ 来设置大小	<code>[]{Space}{Left}\matrix(){Left}</code>

修饰符

输入	输出	源代码
<code>\d1</code>	\dot{x}	<code>\dot{Space 2}</code>
<code>\d2</code>	\ddot{x}	<code>\ddot{Space 2}</code>
<code>\d3</code>	上方有3个点	<code>\dddot{Space 2}</code>
<code>\d4</code>	上方有4个点	<code>\ddddot{Space 2}</code>
<code>\~</code>	\tilde{x}	<code>\tilde{Space 2}</code>
<code>\v</code>	\vec{x}	<code>\vec{Space 2}</code>
<code>\h</code>	\hat{x}	<code>\hat{Space 2}</code>

箭头

输入	输出	源代码	输入	输出	源代码
<code>\lr</code>	\leftrightarrow	<code>\leftrightarrow{Space}</code>	<code>\lrs</code>	\Leftrightarrow	<code>\leftrightarrows{Enter}{Left}</code>
<code>\la</code>	\leftarrow	<code>\leftarrow{Space}</code>	<code>\La</code>	\Leftarrow	<code>\Leftarrow{Space}</code>
<code>\down</code>	\downarrow	<code>\downarrow{Space}</code>	<code>\up</code>	\uparrow	<code>\uparrow{Space}</code>
<code>\ul</code>	\nwarrow	<code>\nwarrow{Space}</code>	<code>\ur</code>	\nearrow	<code>\nearrow{Space}</code>
<code>\dl</code>	\swarrow	<code>\swarrow{Space}</code>	<code>\dr</code>	\searrow	<code>\searrow{Space}</code>

其他符号

输入	输出	源代码	输入	输出	源代码
<code>\de</code>	°	<code>\degree{Space}</code>	<code>\st</code>	★	<code>\star{Space}</code>

结构

输入	输出	源代码
<code>\r</code>	$\{\square\}$	<code>\right.{Left}</code>
<code>\ceil</code>	\lceil	<code>\lceil{Space}\rceil{Space 2}{Left}</code>
<code>\floor</code>	\lfloor	<code>\lfloor{Space}\rfloor{Space 2}{Left}</code>
<code>\brak</code>	$\langle \rangle$	<code>\bra{Space}\ket{Space 2}{Left}</code>
<code>\ls</code>	$\square P$	<code>^_ P {Left 4}</code>
<code>\fu</code>	myfunction \square	<code>\funcapply</code>

- `\funcapply` 与 `\of` 有些许不同。建议自行尝试一下。

花体字母

输入	输出	源代码
<code>\sc</code>	\mathcal{X}	<code>\script</code>
<code>\do</code>	\mathbb{X}	<code>\double</code>
<code>\fr</code>	\mathfrak{X}	<code>\fraktur</code>

- 对于以上代码，你实际应当输入的内容形如 `\sc X`。

更多建议

- 了解更多关于公式输入的内容：[UTN28-PlainTextMath-v3.pdf](#)。第39~47页非常有用。
- 直接输入Unicode数学符号：https://github.com/gtj1/symbol_assist
- 符合直觉的类Vim文本光标控制：<https://github.com/RUSRUSHB/AutoTextCursor>

映射自定义指南

想要自定义映射，请编辑 `symbol_assist.ahk`，使用 Ahk2Exe 或者其他转换器编译，然后运行编译后的 `.exe` 文件。建议你从[AutoHotKey官网](#)了解更多关于 AutoHotKey 的知识。

`symbol_assist_OneNote.ahk` 中的代码非常容易理解，即使你没有学习过 AutoHotKey。对于新手，代码的解释如下：

代码的每一行都是一个输入到符号的映射。格式为 `:(参数):输入::输出`。例如，`::\a::\alpha` 意味着当你输入 `\a` 时，脚本会输出 `\alpha`。

我添加了一些参数 `co?`：

参数	作用
<code>c</code>	区分大小写，比如 <code>\a</code> 和 <code>\A</code> 是不同的。

参数	作用
o	删除你在代码末尾输入的 空格键。
?	即使代码紧跟在数字或字母后面，它也能正常输出。否则，在像是 x\h 的情况下它不会输出。