

# 使用 amsthm 宏包

2.20.3 版, 2017 年 9 月

Publications Technical Group  
American Mathematical Society

## 目录

<b>1</b>	<b>介绍</b>	<b>2</b>
<b>2</b>	<b>建立和输入定理元素</b>	<b>2</b>
2.1	从列表开始的定理 . . . . .	3
<b>3</b>	<b>定理编号</b>	<b>4</b>
3.1	AMS 作者常见问题解答中的相关主题 . . . . .	5
<b>4</b>	<b>为定理类环境更改样式</b>	<b>5</b>
4.1	\theoremstyle 命令 . . . . .	5
4.2	交换数字 . . . . .	6
4.3	新定理的风格 . . . . .	7
<b>5</b>	<b>证明</b>	<b>8</b>
5.1	证明的开始 . . . . .	8
5.2	证明的结束 . . . . .	9
<b>6</b>	<b>已知问题</b>	<b>10</b>
<b>7</b>	<b>其他宏包</b>	<b>10</b>
<b>8</b>	<b>潜在的修改</b>	<b>11</b>
<b>9</b>	<b>提供反馈</b>	<b>11</b>

## 1 介绍

amsthm 包提供了  $\text{\LaTeX}$  的 `\newtheorem` 命令的增强版本, 用于定义类似于定理的环境. 增强的 `\newtheorem` 承认 `\theoremstyle` 规范(在 Mittelbach's 的 `theorem` 包中), 并且有一个 `*` 形式来定义未编号的环境. amsthm 包还定义了一个 `proof` 环境, 该环境可在末尾自动添加 QED 符号. AMS 文档类包含 amsthm 包, 因此这里描述的所有内容也适用于它们.

作为 AMS  $\text{\LaTeX}$  支持环境的一部分, amsthm 遵循 AMS 风格. 这在某些方面与基  $\text{\LaTeX}$  类和上面提到的包提供的样式不同; 如果已知存在差异, 将在这些说明中加以说明. 一个显著的区别是, 在定理或证明结束后, 假设下面一行文本开始一个新的段落, 并且总是缩进, 无论是否存在空白行或 `\par`.

如果 amsthm 包与非 AMS 文档类和 amsmath 包一起使用, amsthm 必须加载在 amsmath 之后, 而不是之前.

由于 amsthm 遵循 AMS 发布样式, 因此可能无法适应与该样式显著不同的格式. 其他定理包确实存在, 并且有不兼容需求的用户在其他地方发布, 建议在 CTAN 上寻找其他可能性. 然而, 准备由 AMS 出版的文件的作者应该遵循 AMS 的风格.

本手册描述了 amsthm 提供的功能. 示例分别在 `thmtest.tex` 文件中给出. 为了更好地理解, 请将输出与输入并排检查.

## 2 建立和输入定理元素

在数学研究的文章和书籍, 定理和证明是最常见的元素之一, 但作者也使用相同的其他一般类: `lemmas`, `propositions`, `axioms`, `corollaries`, `conjectures`, `definitions`, `remarks`, `cases`, `steps` 等等 (见第 5 页). 将这些元素处理为  $\text{\LaTeX}$  环境是很自然的, 但是文档类并不为类定理的元素提供预定义的环境, 因为 (a) 这将使作者难以对自动编号进行必要的控制, 和 (b) 这样的元素种类繁多, 文档类不可能提供所有需要的元素. 取而代之的是命令 `\newtheorem`, 其作用类似于 `\newenvironment`, 它使作者可以很容易地设置特定文档所需的元素.

`\newtheorem` 命令有两个强制参数: 第一个参数是作者想为该元素使用的环境名; 第二个是标题文本. 例如,

```
\newtheorem{lem}{Lemma}
```

表示实例在文档中的

```
\begin{lem} Text text ... \end{lem}
```

将会产生

**Lemma 1.** *Text text ...*

其中标题由指定的文本 “lemma” 和自动生成的数字和标点组成. 注意, 没有固定的文本分配, 例如 `\chaptername` (通常是 “章” 或 “附录”, 取决于上下文); 输出中的标题文本将与您在输入中指定的内容完全一致.

如果在本例中使用 `\newtheorem*` 代替 `\newtheorem`, 那么文档中任何引理的数字都不会自动生成. 即使您只有一个引理, 并且不希望对它进行编号, 这种形式的命令也是有用的; 但是, 更常见的情况是, 它用于生成一个通用定理类型的特殊命名变体. 例如, 如果你有一个引理应该得到标题 “Klein’s Lemma”, 然后语句

```
\newtheorem*{KL}{Klein's Lemma}
```

可以让你写

```
\begin{KL} Text text ... \end{KL}
```

得到想要的输出.

有时在定理或引理的标题中需要额外的信息, 通常是因为它是从其他来源引用的. 一个可选参数用于提供以下信息:

```
\begin{lem}[Alinhac-Lerner \cite{a-l}]
```

产生的输出

**Lemma 2.** (Alinhac-Lerner [4]). *Text text ...*

如果引文包含页码, 引用中的括号必须“隐藏”, 以防止右括号被解释为标题可选参数的结尾:

```
\begin{lem}[Alinhac-Lerner {\cite[p.~37]{a-l}}]
```

与输出

**Lemma 2.** (Alinhac-Lerner [4, p. 37]). *Text text ...*

定理通常在文档的其他地方引用. 通常的  $\text{\LaTeX}$  `\label`-`\ref` 机制用于此目的. 但是, 当 `\label` 与标题分开时, 可能会出现这个问题. 因此, 建议将标签立即放在 `\begin{...}[...]` 命令; 把它放在单独的一行应该不会造成任何不利的影响.

## 2.1 从列表开始的定理

有时需要使用一个列表, 用离散的步骤来表述一个定理. 为了避免链接(使用 `enumerate`)和没有吸引力的间距的问题, 建议按文本排列列表.

如果没有提供文本, 列表将跟随同一行的标题. 这有三个副作用: 定理之间的水平空间头和第一项将大得让人无法接受, 一个定理的超链接将不会“活”在 PDF 文件中(尽管它将标志着), 和额外的垂直空间, 应该遵循的列表将会缺席.

避免这些问题的最佳方法是允许列表以新行开始. 实现这一点的一种方法是通过命令 `\leavevmode` 遵循定理头(以及 `\label`, 如果存在的话). (更多信息, 请参见第 4.3.1 节.)但是, 如果定理接近页面的底部, 列表可能会移到下一页的顶部, 留下孤立的标题. 如果发生这种情况, 必须作为异常处理, 在文本为不可更改的后处理; 此时, 推荐的修复方法是调用 `needspace [NDS]` 包, 或插入显式分页符.

另一种在标题后开始新行的方法是提供一个 `\newtheoremstyle{break}`; 定义在下面的第 7 页中给出。像 `\leavevmode` 方法, `break` 定理环境并不完美; 已知的局限性伴随定义。

如果列表**必须**开始在同一行标题, 手动输入第一个项目标识符, 调整: 在标题和标识符之间输入一个普通空格, 如果项目文本需要多行, 则重置左侧空白, 并重置项目计数器(为 `enumerate`), 以便它将以 2(或任何合适的值)开始。

```
\begin{thm}[\langle optional modifier \rangle]
\hangindent\leftmargini % 用于多行项目
\label{thm:xxx}
\textup{(1)} First item ...
\begin{enumerate}
\setcounter{enumi}{1}
\item ...
\item ...
\end{enumerate}
\end{thm}
```

这里显示的项目标识符的样式与 AMS 文档类的默认样式相匹配。对于其他文档类可能会有所不同; `amsthm` 保留文档类默认值。( `\textup` 在使用 AMS 类时应应用于手工编码的项标识符; AMS 样式在这方面符合传统的数学排版, 即使在斜体环境中, 项标识符、交叉引用和类似的元素也始终保持直立。但是, 作者应该始终尝试匹配所使用的文档类的样式。) 在使用 AMS 文档类时, 为了匹配第一级项目计数器的默认样式, 可以使用 `\eqref` 输入引用; 但是, 要注意是否使用了 `cleveref` 这样的宏包, 因为它可能关联错误的元素类型。

以这种方式输入的初始项不能使用 `\label` 链接, 因为它不与计数器关联; 上面示例中的 `\label` 指的是定理元素, 而不是列表中的项。这是一个重要的概念: `\label` 总是指向最近自动递增的计数器的值; 如果使用 `hyperref` 包, 此计数器也将是超链接的锚。

### 3 定理编号

除了两个强制参数之外, `\newtheorem` 还有两个互斥的可选参数。它们控制编号的顺序和层次结构。编号机制可以这样来考虑:

```
\newtheorem{\langle env name \rangle}{\langle text \rangle}[\langle parent counter \rangle]
\newtheorem{\langle env name \rangle}[\langle shared counter \rangle]{\langle text \rangle}
```

`\langle parent counter \rangle` 相当于 `\numberwithin`; 也就是说, 只要遇到该区段级别, 编号就会重新启动。如果指定了 `\langle shared counter \rangle`, 则使用此计数器的所有定理元素的编号将按顺序进行。请参阅下面更详细的解释。

默认情况下, 每种类定理环境都独立编号。因此, 如果你有三个引理和两个定理, 它们的编号将是这样的: 引理 1, 引理 2, 定理 1, 引理 3, 定理 2。如果您希望引理和定理共享相同的编号序列——引理 1、引理 2、定理 3、引理 4、定理 5——那么您应该使用 `\langle shared counter \rangle` 来表示所需的关系, 如下所示:

```
\newtheorem{thm}{Theorem}
\newtheorem{lem}[thm]{Lemma}
```

第二条语句中的可选参数 [thm] 意味着 lem 环境应该共享 thm 编号序列, 而不是拥有自己的独立序列.

使一个定理环境在一个分段单元内从属编号——例如, 对命题进行编号, 得到第 2 节中的命题 2.1、命题 2.2 等——将父单元的名称用方括号括起来, 放在最后的位置:

```
\newtheorem{prop}{Proposition}[section]
```

使用可选参数 [section], 当父计数器 section 增加时, prop 计数器将被重置为 0, 并且命题标题将预置 section 数.

如果任何定理元素是按 section 编号的, 并且(在书中)某一 chapter 中的第一个这样的元素出现在第一 section 之前, 编号将从上一 chapter 继续. 在这种情况下, 通过在受影响的元素之前调用这个命令来重置计数器:

```
\setcounter{thm}{0}
```

注意, 定理编号不是由 amsmath 的 \numberwithin 工具的相同方法完成的, 因此试图使用该方法将定理编号与等式编号关联起来将不会按预期的方式工作. AMS 作者 FAQ(关于“*Theorems in AMS- $\LaTeX$* ”的一节)中给出了一种实现此功能的方法以及其他一些变体.

### 3.1 AMS 作者常见问题解答中的相关主题

- “数字”定理用字母代替数字
- 重新开始每个 chapter 的编号, 但不包括 chapter 编号作为定理编号的一部分
- 当 chapters 有 section 时, 定理编号被 section 取代, 但只有当 chapters 没有 section 时, 才被 chapter 取代
- 具有相同编号序列的定理编号和 equations

## 4 为定理类环境更改样式

### 4.1 \theoremstyle 命令

amsthm 宏包支持当前定理样式的概念, 它决定了给定的 \newtheorem 命令将生成什么. 这三种定理样式提供了——plain、definition 和 remark——根据它们的相对重要性指定不同的视觉强调程度. 这种排版处理的细节可能因文档类而异, 但通常 plain 样式产生斜体文本, 而其他两种样式产生罗马体文本. 提供了这些默认设置:

- plain: 斜体文本, 上面和下面额外的空间;
- definition: 直立文字, 在上面和下面留出额外的空间;
- remark: 直立文字, 上面或下面没有额外的空间.

如果没有给出 \theoremstyle 命令, 则使用的样式为 plain. 要指定不同的样式, 请将您的 \newtheorem 命令划分为组, 并在每个组前面加上适当的 \theoremstyle. 一些例子:

```

\theoremstyle{plain}% default
\newtheorem{thm}{Theorem}[section]
\newtheorem{lem}[thm]{Lemma}
\newtheorem{prop}[thm]{Proposition}
\newtheorem*{cor}{Corollary}
\newtheorem*{KL}{Klein's Lemma}

\theoremstyle{definition}
\newtheorem{defn}{Definition}[section]
\newtheorem{exmp}{Example}[section]
\newtheorem{xca}[exmp]{Exercise}

\theoremstyle{remark}
\newtheorem*{rem}{Remark}
\newtheorem*{note}{Note}
\newtheorem{case}{Case}

```

下面的列表总结了通常与每种定理风格相关的结构类型。

plain	Theorem, Lemma, Corollary, Proposition, Conjecture, Criterion, Assertion
definition	Definition, Condition, Problem, Example, Exercise, Algorithm, Question, Axiom, Property, Assumption, Hypothesis
remark	Remark, Note, Notation, Claim, Summary, Acknowledgment, Case, Conclusion

## 4.2 交换数字

一种常见的样式变体是将定理号放在标题的开头而不是结尾, 例如“1.4 定理”而不是“定理 1.4”。由于这种变化通常应用于所有 `\theoremstyle`, 所以它是通过在应该受到影响的 `\newtheorem` 语句列表的开头放置一个 `\swapnumbers` 命令来实现的。例如:

```

\swapnumbers
\theoremstyle{plain}
\newtheorem{thm}{Theorem}[section]
\theoremstyle{remark}
\newtheorem{rem}{Remark}

```

当 `amsthm` 与非 AMS 文档类一起使用, 并且交换了数字时, 数字后面不设置句点; 这与“基本” $\text{\LaTeX}$  样式不同。在 AMS 作者 FAQ [AF] 中会给出一个解决方案。

在某些 AMS 文档类中, 交换的数字的样式与节标题的样式相匹配; 这可能与定理标题的其他部分不同。

### 4.3 新定理的风格

如果三个预定义的样式证明不足够, amsthm 包提供了一个 `\newtheoremstyle` 命令来帮助创建定制样式. 下面的示例演示了 `\newtheoremstyle` 命令的使用.

```
\newtheoremstyle{note}% <名字>
  {3pt}%           <上面的空间>1
  {3pt}%           <下面的空间>1
  {}%             <主体的字体>
  {}%             <缩进量>2
  {\itshape}%     <定理头字体>
  {:}%            <定理头后标点符号>
  {.5em}%         <定理头后的空间>3
  {}%            <定理头规范 (可以为空, 表示“正常”) >
```

<sup>1</sup> 上下空格: 它们是相等的, 但是使用 AMS 文档类时的默认值不同于 `\usepackage{amsthm}`. 对于 AMS 文档类, 这是 `.5\baselineskip ± .2\baselineskip`; 对于作为单独宏包使用的 amsthm, 它是 `\topsep` 的本地值. 如果这些参数为空, 则使用当前默认值.

<sup>2</sup> 缩进量: 空 = 没有缩进, `\parindent` = 正常段落缩进

<sup>3</sup> 定理头后的空间: `{ }` = 正常字间的空间; `\newline` = line break

以与预定义的方法相同的方式应用新的定理样式.

下面是一些作者所要求的样式.

#### 4.3.1 break 定理风格

这个样式将在定理标题之后中断, 并开始一个新行.

```
\newtheoremstyle{break}%
  {}{}%
  {\itshape}{}%
  {\bfseries}{}% % 注意, 最后的标点省略了。
  {\newline}{}%
```

这种样式可用于以列表开头的定理. 当与 `enumerate` 和 AMS 文档类一起使用时, 所有项都被正确标记并将链接. 但是, 垂直间距需要帮助; 定义之间的冲突会阻止第一项在新行中开始——它看起来几乎与以枚举列表开头的默认定理相同. 为了修复这个问题, 可以这样开始这个定理:

```
\begin{breakthm}[...]
\leavevmode \vspace{-\baselineskip}
\begin{enumerate}
```

`\leavevmode` 单独会在定理头后面留下一个完整的空行.

还有一个问题可能会出现: 如果定理开始于页面的末尾, 那么列表可能会被分割为一个新页面, 留下一个孤立的标题. 在最后一篇文章中要注意这一点; 然后调用 `needspace [NDS]` 包, 或插入显式分页符.

### 4.3.2 注释字段加粗的定理

如果带有括号的注释特别重要，可能需要将注释字段设置为粗体。这种变体会使得定理编号，这是 `\newtheorem*` 提供不了的选择。

```
\newtheoremstyle{bfnote}%
  {}{}%
  {\itshape}{}%
  {\bfseries}{.}%
  { }%
  {\thmname{#1}\thmnumber{ #2}\thmnote{ (#3)}}
```

括号可以从周围的注释使用粗体的形式的这种琐碎的改编省略。

```
\newtheoremstyle{noparens}%
  {}{}%
  {\itshape}{}%
  {\bfseries}{.}%
  { }%
  {\thmname{#1}\thmnumber{ #2}\thmnote{ #3}}
```

### 4.3.3 其他的风格

例如, 请参阅文件 `thmtest.tex`.

## 5 证明

由 `amsthm` 包提供的一个预定义的 `proof` 环境产生带有适当间隔和标点符号的标题“proof”。证明的环境主要是针对简短的证明, 长度不超过一页或两页; 较长的证明通常最好在文档中作为单独的 `\section` 或 `\subsection` 进行。

### 5.1 证明的开始

证明环境的一个可选参数允许您用不同的名称替换标准“Proof”。如果你想证明标题是, 说, “Proof of the Main Theorem”, 然后写

```
\begin{proof}[Proof of the Main Theorem]
```

由于 `proof` 环境的定义方式(作为一个 `LATEX` list), 如果替换的“name”比一个输出行长, 则不会中断; 这是要修理的。与此同时, 在序言中重新定义将会达到预期的效果。

```
\makeatletter
\renewenvironment{proof}[1][\proofname]{\par
  \pushQED{\qed}%
```



```

\normalfont \topsep6\p@\@plus6\p@\relax
\trivlist
\item\relax
    {\itshape
      #1\@addpunct{.}}\hspace\labelsep\ignorespaces
}%
\popQED\endtrivlist\@endpfalse
}
\makeatother

```

以列表开头的证明可以用与定理相同的方法来处理;参见第 2.1 节.

## 5.2 证明的结束

QED 符号  $\square$  会自动附加到 `proof` 环境的末尾. 要替换不同的证明结束符号, 请使用 `\renewcommand` 重新定义命令 `\qedsymbol`. 对于作为 subsection 或 section 而不是在 `proof` 环境中完成的长证明, 您可以使用 `\qed` 获取符号和前面通常的空格;只有当它遵循普通文本时, 它才能正常工作.

在 `\end{proof}` 之前(或显式 `\qed` 之前)的空白行禁用放置 QED 符号的机制, 甚至可能允许将该符号设置在新页面的顶部.

如果一个 `proof` 环境的最后一部分是一个 `displayed` 的等式或列表环境或类似性质的东西, 那么 QED 符号的放置可能会有问题. 在这种情况下, 在 QED 符号应该出现的地方放置一个 `\qedhere` 命令, 例如,

```

\begin{proof}
...
\begin{equation}
G(t)=L\gamma!\,,t^{\{-\gamma\}+t^{\{-\delta\}}\eta(t) \ \qedhere
\end{equation}
\end{proof}

```

当与 `amsmath` 包一起使用时, 版本 2 或更高版本, `\qedhere` 将把 QED 符号右对齐;在早期的版本中, 符号与文本或 `display` 的末尾之间的间隔为 `\quad`.

如果 `\qedhere` 在方程式中产生错误消息, 请尝试使用 `\mbox{\qedhere}` 代替.

`\qedhere` 不适用于 `eqnarray` 或输入 `$$` 的方程;由 `amsmath` 和 `\[ ... \]` 定义的显示环境为了这个目的, 已经特别制作了无编号方程的形式.

当验证以列表环境结束时, 将 `\qedhere` 作为 `\end{enumerate}` 或 `\end{itemize}` 之前的最后一项.

如果, 由于某种原因, 需要从证明的结尾省略 QED 符号, 可以通过在 `\end{proof}` 之前包含此定义来抑制它:

```
\renewcommand{\qedsymbol}{}{}
```

如果要从文档的所有证明中省略符号, 请将重新定义放在序言中.

如果在 `proof` 环境之外需要 QED 符号, 请使用 `\qed` 而不是 `\qedhere`.

## 6 已知问题

以下列出的问题供 amsthm 下次更新时注意.

- `\qedhere` 的问题.
  - 当 `\qedhere` 与无星号的多行显示结构一起使用时, 最后一行没有编号. (只有数字在左边时, 这才有意义; `\qedhere` 不适用于右边编号的显示.)
  - 当 `\qedhere` 在 `split` 环境中时, 显示的那部分不被编号, `\label` 被抑制. 如果 `\qedhere` 被移到 `split` 之外, 数字会出现, 但是 `split` 的内容会向左移动, 并且 QED 符号可能会缺席.
  - 在对齐中, 如果没有 `&`, 则方框不会对齐到右边框.
  - 对于单行方程, 将同时显示数字和方框, 但将方程向左移动.
- amsthm 固有的其他问题.
  - 当定理以列表开始时, 超链接第一项.
- 与 AMS 文档类交互的问题. 纠正将在文档类上进行, 而不是在 amsthm 中.
  - 在 `amsbook` 中, 当一个定理出现在一个 chapter 的第一个编号的 section 之前时, 定理编号从上一个 chapter 继续.

## 7 其他宏包

有些包提供了附加的定理功能, 带有识别 amsthm 的存在并针对其进行调整的选项.

- `mdframed` [MDF] 提供了使用各种形状和颜色的框架来启动定理(和其他元素)的能力. 如果同时使用 amsthm, 则应该在其后加载.
- `thmtools` [THT] 为定理规范提供了另一种 key-value 语法, 以及不同的表示样式(预定义的和用户可定义的). QED 符号可以应用于任何定理元素, 而不只是用于证明. 还提供了一个可适应的 `\listoftheorems`.

有关详细信息, 请参阅包装手册. 两个包都包含在 TeX Live 中, 所以可以通过 `texdoc <package>` 访问.

一些提供潜在有用特性的包具有与 AMS 文档类或 amsthm 实现定理相冲突的限制. 以下是一些我们知道的.

- `paralist` 覆盖在定理内 `enumerate` 环境中 `item` 数字应用 `\upshape` 的 amsthm 中的设置; 数字变成斜体, 这是不希望看到的.
- `wrapfig`, 它不与列表共存. amsthm 中的定理和证明都基于 `\trivlist`. 使用这种方法的插入可能导致后面的段落按插入的宽度缩进, 即使插入的深度不允许这样做.
- `floatflt` 和 `pycinpar` 有同样的限制.

## 8 潜在的修改

来自用户的反馈和 AMS 出版物的经验指出了几个方面, 增加的灵活性可以使 `amsthm.sty` 的用户和基于 AMS 集的“派生”文档类的开发人员受益.

这些建议比较容易实施:

- 将硬编码的字体规范(例如, `\itshape`)更改为允许修改的命令(例如, `\proofheadingfont`).
- 为其他元素(如标点符号)提供命令, 而不是硬编码的值.

下面的建议, 虽然有价值, 但需要更多的工作, 包括重新设计现有代码:

- 扩展 `\qedhere` 对定理类元素和证明的适用性.
- 提供一种方法来忽略 `proof` 的定义, 以避免与一些非 AMS 文档类的名称冲突.
- 考虑用于指定 `\newtheoremstyle` 选项的 key-value 机制.
- 提供一种以列表开始的新定理样式.

## 9 提供反馈

在这个包中发现的错误报告应该提交给

`tech-support@ams.org`

为了获得最好的结果, 报告应该附带一个简短的、可编译的示例.

另一种方法是使用  $\text{\LaTeX}$  bugs 报告机制, 指定 AMS-LaTeX 作为 bug 类别.

## 参考文献

- [AF] AMS Author FAQ, <http://www.ams.org/authors/author-faq>
- [MDF] The `mdframed` package, Marco Daniel and Elke Schubert, 2013/07/01, v1.9b,  
<http://mirror.ctan.org/macros/latex/contrib/mdframed>
- [NDS] The `needspace` package, Peter Wilson, 2010/09/12, v1.3d,  
<http://mirror.ctan.org/macros/latex/contrib/needspace>
- [THT] `Thmtools` Users' Guide, Ulrich M. Schwarz, 2014/04/21 v66,  
<http://mirror.ctan.org/macros/latex/exptl/thmtools>