

TASKS 宏包学习手册

原著：Yongxue 2020 年 12 月 20 日

翻译：黄旭华 2022 年 04 月 01 日

目 录

1	动机	2
2	许可证和需要	2
3	如何工作	2
3.1	背景	2
3.2	最基本的	3
3.3	跨越多栏的条目	3
3.4	跨越多栏的条目	6
4	可用的选项	10
5	可用的实例	15
6	自定义标签	16
7	新的 tasks 类 (tasks-like) 环境	17
8	样式化 tasks	18
8.1	tasks 对象	18
8.1.1	可用的选项	18
8.1.2	预定义的实例	19

1 动机

最初的 **TASKS** 是 **ExSheets** 宏包的一部分 [Nie19]。然而，用户告诉我，可将其作为一个独立的宏包，而不必仅仅为了使用 **TASKS** 环境而加载整个 **ExSheets** 宏包。这确实有道理。因此我将该环境被提取到一个 **tasks** 宏包中。那时，**TASKS** 就作为一个独立的宏包分发，也作为 **ExSheets** 捆绑包 (bundle) 的一部分。完全独立的宏包最初的版本是 v0.10。从此，该独立宏包与 **ExSheets** 的关系成为历史。

使用 **tasks** 环境的原因是德国数学教科书（尤其是初中教科书）中有一项不成文的规定，即以横向计算而非纵向计算的栏来组织练习。这就是使用 **tasks** 的主要目的。如果不需要此功能，最好使用传统的 **L^AT_EX** 列表和 **enumitem** 宏包进行定制。

更改

- 不推荐使用 **counter-format** 选项。现在，标签的设置方式与 **enumitem** 中的设置方式非常相似。这也使得列表模板的 **enumerate** 选项变得多余，并相应地被删除。
- 已重命名 **\NewTasks** 命令和 **\RenewTasks** 命令。
- 已删除多项选择列表 (multiple choice lists)。
- 自定义的定义 (custom definitions) 可以放在 **tasks.cfg** 文件中，可自动加载该文件（如果可用）。

2 许可证和需要

根据 **L^AT_EX** 项目公共许可证 (lppl)1.3c 版或更高版本 (<http://www.latex-project.org/lppl.txt>) 的条款，允许复制、分发和/或修改本软件。该软件的状态为“已维护 (maintained)”。

TASKS 需要 **l3kernel** [L3P] 捆绑包、**xparse**1 和 **xtemplate**。

3 如何工作

3.1 背景

TASKS 环境和 **enumerate** 列表有点相似但不相同，不同之处如下：

1. 分页 (pagebreak) 在一个条目 (item) 中不可用，但在两个条目 (item) 间可用。
2. 计数 (enumeration) 默认是 a), b), c)

3. 每次出现条目分隔符 (item separator) 时, `tasks` 环境的主体都会被拆分。因此, 默认的分隔符不是 `\item`, 而是 `\task`, 因此它仅限于该环境。这直接导致...
4. `tasks` 环境不能被嵌套。但是, 它可以嵌套 `itemize` 环境或其它 “真正的 (real)” 列表 (list)。
5. 不能在该环境中使用抄录材料 (verbatim material), 但可以使用 `\string` 命令。
6. `\texttt` 或 `\detokenize`。如果这不起作用, 就不要使用 **TASKS**。

TASKS 环境就是我所说的 “伪环境 (pseudo-environments)” 。这意味着, 与 `environ[Rob14]` 宏包定义的环境一样, 环境主体在被处理之前当作参数被读取。这就是为什么不能在 **TASKS** 列表中使用抄录材料 (verbatim material)。

3.2 最基本的

`\begin{tasks}[<选项>](<栏的数目>)`

类似 `environment` 列表, `\task` 会引入一个条目 (item)。

示例如下:

该环境使用可选参数 (< 栏的数目 >) 来指定使用的栏数。

3.3 跨越多栏的条目

有时, 如果允许一个条目 (item) 跨越多栏, 它可能会派上用场。v0.10 版本的 **TASKS** 支持使用剩余空间的条目, 方法是向 `\task` 命令中添加可选的星号 (optional star):

```
1 % \Sample 该命令包含一些示例文本：
2 % \def\sample{This is some sample text we will use to create a somewhat
3 % longer text spanning a few lines.}
4 % \def\Sample{\sample\ \sample\par\sample}
5 列表前的文本。
6 \begin{tasks}
7 \task \Sample
8 \task \Sample
9 \task \Sample
10 \end{tasks}
11 列表后的文本。
```

a) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

b) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

c) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

列表后的文本。

```
1 \begin{tasks}(2)
2 \task \Sample
3 \task \sample\ \sample
4 \task \sample
5 \task \Sample
6 \task \sample\par\sample
7 \end{tasks}
```

a) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

c) This is some sample text we will use to create a somewhat longer text spanning a few lines.

e) This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

b) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

d) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

3.4 跨越多栏的条目

有时, 如果允许一个条目 (item) 跨越多栏, 它可能会派上用场。v0.10 版本的 **TASKS** 支持使用剩余空间的条目, 方法是向 `\task` 命令中添加可选的星号 (optional star):

```
1 \begin{tasks}(3)
2 \task \sample
3 \task * \sample
4 \task * \sample
5 \task \sample
6 \task \sample
7 \end{tasks}
```

- | | |
|---|---|
| <p>a) This is some sample text we will use to create a somewhat longer text spanning a few lines.</p> | <p>b) This is some sample text we will use to create a somewhat longer text spanning a few lines.</p> |
| <p>c) This is some sample text we will use to create a somewhat longer text spanning a few lines.</p> | |
| <p>d) This is some sample text we will use to create a somewhat longer text spanning a few lines.</p> | <p>e) This is some sample text we will use to create a somewhat longer text spanning a few lines.</p> |

TASKS 还支持在任何情况下跨越所有栏的条目 (item), 方法是向 `\task` 命令中添加可选的感叹号 (optional bang):

```
1 \begin{tasks}(3)
2 \task \sample \task! \sample \task! \sample
3 \task \sample \task \sample
4 \end{tasks}
```

- a) This is some sample
text we will use to cre-
ate a somewhat longer
text spanning a few
lines.
- b) This is some sample text we will use to create a somewhat longer text spanning a
few lines.
- c) This is some sample text we will use to create a somewhat longer text spanning a
few lines.
- | | |
|--|--|
| d) This is some sample
text we will use to cre-
ate a somewhat longer
text spanning a few
lines. | e) This is some sample
text we will use to cre-
ate a somewhat longer
text spanning a few
lines. |
|--|--|

可选的星号 (optional star) 本身有一个带有圆括号的可选参数，您可以在其中指定
条目应该跨越的栏数：

```

1 \settasks{debug}
2 \begin{tasks}(4)
3 \task the first
4 \task the second
5 \task the third
6 \task the fourth
7 \task * (3) the fifth item is way too long for this and needs three
   columns
8 \task the sixth
9 \task the seventh
10 \task * (2) the eighth item is way too long for this and needs two
   columns
11 \task the ninth
12 \task the tenth
13 \end{tasks}

```

a) the first	b) the second	c) the third	d) the fourth
e) (3) the fifth item is way too long for this and needs three columns			
f) the sixth	g) the seventh	h) (2) the eighth item is way too long for this and needs two columns	
i) the ninth	j) the tenth		

如果没有足够的栏数（声明中的是两栏，而您的命令却是 `\task * (3)`），则忽略该参数，跨越实际剩余的栏数（示例中的是两栏）。可选的星号和可选的感叹号都可以与自定义标签 (custom label) 的可选参数结合使用：


```

1 \begin{tasks}(3)
2 \task \sample
3 \task * \sample
4 \task * [(x)] \sample
5 \task \sample
6 \task \sample
7 \end{tasks}

```

-
- a) This is some sample text we will use to create a somewhat longer text spanning a few lines.
- b) This is some sample text we will use to create a somewhat longer text spanning a few lines.
- c) [(x)] This is some sample text we will use to create a somewhat longer text spanning a few lines.
- d) This is some sample text we will use to create a somewhat longer text spanning a few lines.
- e) This is some sample text we will use to create a somewhat longer text spanning a few lines.

在 v0.9 (2013/04/07) 版本中可以使用以下命令手动强制新条目行 (new item line):

`\startnewitemline`

在 `tasks` 环境中引入新行。

虽然这样做有效, 但也需要小心一点, 因为条目 (item) 的宽度不会改变, 这意味着为了使用全宽度 (full width), 您必须使用 `\rlap` 之类的技巧, 而这意味着条目文本 (item text) 可能会超出页边。

```

1 \begin{tasks}(4)
2 \task the first
3 \task the second
4 \task the third
5 \task the fourth
6 \task \rlap{the fifth item is way too long for this so we start a new row
   }
7 \startnewitemline
8 \task the sixth
9 \task the seventh
10 \task \rlap{the eighth item also is too long} \startnewitemline
11 \task the ninth
12 \task the tenth
13 \end{tasks}

```

a) the first b) the second c) the third d) the fourth
e) the fifth item is way too long for this so we start a new row
f) the sixth g) the seventh h) the eighth item also is too long
i) the ninth j) the tenth

4 可用的选项

TASKS 宏包没有任何包选项 (package option)。

不过, `tasks` 环境有许多选项, 可以使用 `setup` 命令设置 `tasks` 环境的以下选项:

`\settasks<options>`

用于 `tasks` 环境的 `setup` 命令如下:

`style = <instance>` (初始为空)

选择 `instance`。更多信息请参见第 8.1 节。

`label-format = <code>` (初始为空)

可用于对标签 (labels) 的格式如 `\bfseries` 进行设置。接受该条目作为强制参数 (mandatory argument) 时使用该 `code`。

`label = <code>` 默认为 `\alph *`)

设置自定义标签 (custom label)。* 被 `task` 替换。这在很大程度上受到 `enumitem` 的 [Bez19] 标签选项的启发。

`ref = <code>` (初始为空)

New Works 类似标签 (label), 但是通过设置 `\the<counter>` (默认设置为 `\thetask`) 来设置引用的输出 (output of the reference)。

`label-width = <dim>` 默认为 1em

设置条目标签 (item labels) 的宽度。

`label-offset = <dim>` 默认为 0.3333em

设置偏移 (offset), 例如, 标签 (label) 和条目 (item) 之间的距离。

`item-format = <code>` (初始为空)

可用于对条目 (items) 的格式如 `\bfseries` 进行设置。接受该条目作为强制参数 (mandatory argument) 时使用该 code。

`item-indent = <dim>` 默认为 2.5em

条目 (item) 的缩进, 包括标签 (label) 和标签偏移 (label-offset) 的水平距离。

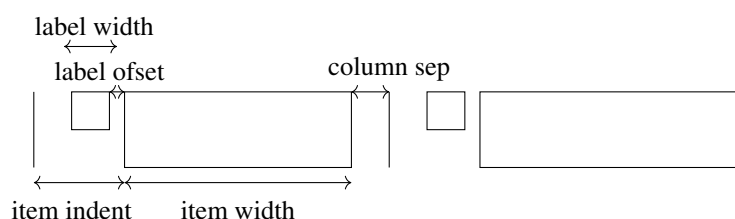


图 1: 所用长度的示意图

如果: $\text{indent} = \text{labelwidth} + \text{label-offset}$

即: 条目缩进 (item indent) = 标签宽度 (label width) + 标签偏移 (label-offset)

那么: 标签 (label) 将与上面的文本块 (textblock) 对齐 (如果设置了 `label-align = {left}`)。请参见上述图 1 查看可用长度及其设置。

`column-sep = <dim>` 默认为 0pt

栏间距 (column sep), 即两栏之间的水平间距。

`label-align = left | right | center` 默认为 left

标签对齐 (label align), 设定标签 (label) 在标签盒子 (label-box) 中的对齐方式, 标签盒子的宽度由 `label-width` 设定。

`before-skip = {<skip>}` 默认为 0pt

设置列表前附加的垂直空白 (before skip)。

`after-skip = {<skip>}` 默认为 0pt

设置列表后附加的垂直空白 (after skip)。

`after-item-skip = {<skip>}` 默认: 1ex plus 1ex minus 1ex

设置条目 (items) 的行 (rows) 之间附加的垂直空白。

`resume = true|false` 默认为 false

从以前的 tasks 环境中恢复计数 (enumeration)。为了正确使用这个选项，您不应该混合使用不同的 tasks 环境，因为这些环境都会计算它们的条目。

`start = {<integer>}` 默认为 1

设置列表开始计数的起始值 (starting value)。

`counter = {<counter>}` 默认为 task

设置用于计算条目 (items) 的计数器 (counter)。

`debug = true|false` 默认为 false

`debug` (调试) 如果设置为 `true`，在 tasks 环境中 `\fboxsep` 将设置为 0pt，`\fbox` 将在标签盒子 (label boxes) 和条目盒子 (item boxes) 周围绘制框架 (frame)。

现在，列表与上面的相同，但有三栏和一个不同的标签：

```

1 \begin{tasks}[label=(\roman * ),label-width=4ex](2)
2 \task \Sample
3 \task \sample
4 \task \sample
5 \task \Sample
6 \end{tasks}

```

() This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

() This is some sample text we will use to create a somewhat longer text spanning a few lines.

() This is some sample text we will use to create a somewhat longer text spanning a few lines.

() This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

让我们在一个问题 (question) 中使用它, 例如, 在 xsim 宏包的 exercise 环境 [Nie20] 中:

```
1 % 由于这些设置是局部的(local)，以下设置在该示例以外将丢失；
2 \settasks{
3 label = \theexercise.\arabic * ,
4 item-indent = 2em ,
5 label-width = 2em ,
6 label-offset = 0pt
7 }
8 \begin{exercise}
9 I have these two tasks for you. Shall we begin?
10 \begin{tasks}(2)
11 \task The first task: easy!
12 \task The second task: even more so!
13 \end{tasks}
14 \end{exercise}
15 \begin{solution}[print]
16 Now, let's see\ldots\ ah, yes:
17 \begin{tasks}
18 \task This is the first solution. Told you it was easy.
19 \task This is the second solution. And of course you knew that!
20 \end{tasks}
21 \end{solution}
```

I have these two tasks for you. Shall we begin?

1. The first task: easy!
2. The second task: even more so!

Now, let's see... ah, yes:

1. This is the first solution. Told you it was easy.
2. This is the second solution. And of course you knew that!

最后，让我们看看 debug（调试）选项的作用（您可以在第 8 页上看到）：

```

1 \settasks{debug}
2 \begin{tasks}(2)
3 \task \Sample
4 \task \Sample
5 \end{tasks}

```

a) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

b) This is some sample text we will use to create a somewhat longer text spanning a few lines. This is some sample text we will use to create a somewhat longer text spanning a few lines.

This is some sample text we will use to create a somewhat longer text spanning a few lines.

5 可用的实例

目前，tasks 对象 (tasks object) 还有三个其他可用的实例：

itemize 使用 `\labelitemi` 作为标签。

enumerate 用 1., 2., ... 计数 (enumerates) 条目 (items)。

```

1 \begin{tasks}[style=itemize](2)
2 \task that's just how\ldots
3 \task \ldots we expected
4 \end{tasks}
5 \begin{tasks}[style=enumerate](2)
6 \task that's just how\ldots
7 \task \ldots we expected
8 \end{tasks}

```

- that's just how...

- ... we expected

1. that's just how...

2. ... we expected

6 自定义标签

如果要更改列表中的单个标签 (single label), 可以使用 `\task` 命令的可选参数。这将临时覆盖默认标签 (default label)。

```
1 \begin{tasks}[style=itemize](2)
2 \task that's just how\ldots
3 \task[+] \ldots we expected
4 \task that's just how\ldots
5 \task \ldots we expected
6 \end{tasks}
```

- that's just how... + ...we expected
- that's just how... • ...we expected

您已经看到了标签选项 (label option) 的示例。

label = <code> 默认为 \alph *)

用于设置列表的标签 (label)。其中 * 常由括号内的当前计数器名称替换。它可以包含 **\bfseries** 之类的格式化指令，但使用下面的标签格式化 (label-format) 选项更简洁。

label-format = <code> (初始为空)

这一点尤其如此，因为在您通常不希望有格式化命令 (formatting instructions) 的地方设置 `\the<counter>`。解决这个问题的另一种方法是使用下面的 `ref` 选项：

ref = `<code>` (初始为空)

该选项设置 `\the hcounteri` (`\thetask` 在默认设置里)。

```
1 \begin{tasks}[label=\arabic * .,ref=\arabic * ]
2 \task first item
3 \task second item \label{foo}
4 \end{tasks} See item~\ref{foo} without dot.
```

- a) first item
- b) second item

See item **b)** without dot.

定义了另外两个在某些情况下可能有用的命令:

```
\tasksifmeasuringTF{<true>}{<false>}
```


该命令在标签内部使用，检查标签的排版是否用于测量其宽度，或者是否为“真正的 (for real)”排版。这里还有 `\tasksifmeasuringT` 变量 (variants) 和 `\tasksifmeasuringF` 变量。

`\tasklabel`

用于保存当前标签文本 (label text)。

7 新的 tasks 类 (tasks-like) 环境

可以添加与 tasks 环境类似的自定义环境 (custom environments)。

`\NewTasksEnvironment[<options>]{<name>}[<separator>](<cols>)`

使用名为 separator 的分隔符定义名为 name 的新环境。分隔符 separator 默认为，栏数 cols 默认为 1。选项 options 在第 4 节中已描述。

`\RenewTasksEnvironment[<options>]{<name>}[<separator>](<cols>)`

用 `\NewTasksEnvironment` 修改 (renew) 前面定义的环境。

tasks 环境定义如下：

```
1 \NewTasksEnvironment{tasks}
```

分隔符 (separator) 不必是控制序列 (control sequence)：

```
1 % preamble:
2 % \usepackage{fontawesome}
3 \NewTasksEnvironment[label=\faThumbsOUp,label-width=15pt]{done}[ * ]
4 \begin{done}
5 * First task
6 * Second task
7 \end{done}
```

♣ First task

♠ Second task

✓ First task

♡ Second task

虽然这看起来很方便，甚至很好，但我强烈建议不要使用与命令序列 (command sequence) 不同的东西。记住，每当出现分隔符 (separator) 时，条目 (items) 都会被拆分。因此，为了在一个条目 (items) 中使用分隔符 (separator) (例如，这里是一个带星号的命令变体)，它必须隐藏在中括号中。这可以避免你使用一个还未定义的命令序列。

请记住，分隔符 (separator) 仍然有一个可选的星号参数 (optional star argument) (参见 4)、一个可选的感叹号参数 (optional bang argument) 和标准可选参数 (standard optional argument)。使用 * 将阻止可选的星号参数。

```

1 % preamble:
2 % \usepackage{fontawesome}
3 \NewTasksEnvironment[label=\faThumbsOUp,label-width=15pt]{done}[ * ]
4 \begin{done}(3)
5 * First task
6 * Second task
7 * ! Third task spanning the full width available
8 * Fourth task
9 \end{done}

```

▽ First task

»» Second task

8 样式化 tasks

TASK 使用 xtemplate 宏包为列表声明其它实例 (additional instances)。

8.1 tasks 对象

由 tasks 定义的对象 (object) 是 “tasks” 对象。这一次，定义的一个模板（同样是“默认”）有四个实例可用。

8.1.1 可用的选项

本节仅列出定义 “default” 模板的实例时可用的选项。下面的内容给出了一些例子。

```
1 \DeclareTemplateInterface{tasks}{default}{3}{  
2 % option : type = default  
3 label : tokenlist = \alph * ) ,  
4 indent : length = 2.5em ,  
5 label-format : tokenlist ,  
6 label-width : length = 1em ,  
7 label-offset : length = .3333em ,  
8 after-item-skip : skip = 1ex plus 1ex minus 1ex  
9 }
```

8.1.2 预定义的实例

这一次相当简洁:

```
1 % alphabetize: a) b) c)  
2 \DeclareInstance{tasks}{alphabetize}{default}{}  
3 % itemize  
4 \DeclareInstance {tasks} {itemize} {default}{  
5 label-width = 1.125em ,  
6 label = \labelitemi} % enumerate:  
7 \DeclareInstance {tasks} {enumerate} {default}  
8 {label = \arabic * . }
```

参考文献

- [1] Javier Bezos. enumitem. Version 3.9. June 20, 2019. url: <https://ctan.org/pkg/enumitem>.
- [2] The L^AT_EX3 Project Team. l3kernel. Sept. 19, 2019. url: <https://www.ctan.org/pkg/l3kernel/>.
- [3] Clemens Niederberger. exsheets. Version 0.21k. Sept. 30, 2019. url: <https://ctan.org/pkg/exsheets>.
- [4] Clemens Niederberger. xsim. Version 0.19a. Mar. 19, 2020. url: <https://ctan.org/pkg/xsim>.
- [5] Will Robertson. environ. Version 0.3. May 4, 2014. url: <https://ctan.org/pkg/environ>.