



链滴

Markdown 中 ****** 加粗失效问题解析

作者: [88250](#)

原文链接: <https://ld246.com/article/1597581380183>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Markdown 原文：

****故：恒无，欲也以观其眇。恒有，欲也以观其所微**

****闭合****部分可以形成加粗这是共识，但加上前后内容的话会有细节上的出入。

开始的 ****** 在规范里定义为“左边界序列”，结束的 ****** 定义为“右边界序列”。在这个例子中我们简理解为开始标记符/结束标记符就行，因为不存在嵌套 ****** _ 嵌套情况。

开始标记符的前后必须满足：

1. 不能后跟空白
2. 标点符号
 1. 不能后跟标点符号
 2. 如果后跟标点符号，则前面必须有空白

结束标记符的规则和开始标记符相反：

1. 前面不能是空白
2. 标点符号
 1. 前面不能是标点符号
 2. 如果前面是标点符号，则后面必须有空白

结合给的例子看：****故：恒无，欲也以观其眇。恒有，欲也以观其所微**之所以不能加粗，是因为结束标记符 ****** 前面跟了标点符号，但后面又不是空白。

<https://spec.commonmark.org/0.29/#emphasis-and-strong-emphasis>

这部分写成代码就是：

```
afterIsWhitespace := lex.IsUnicodeWhitespace(tokenAfter)
afterIsPunct := unicode.IsPunct(tokenAfter) || unicode.IsSymbol(tokenAfter)
if (lex.ItemAsterisk == token && '~' == tokenAfter) || (lex.ItemTilde == token && '*' == tokenAfter) {
    afterIsPunct = false
}
beforeIsWhitespace := lex.IsUnicodeWhitespace(tokenBefore)
beforeIsPunct := unicode.IsPunct(tokenBefore) || unicode.IsSymbol(tokenBefore)
if (lex.ItemAsterisk == token && '~' == tokenBefore) || (lex.ItemTilde == token && '*' == tokenBefore) {
    beforeIsPunct = false
}

isLeftFlanking := !afterIsWhitespace && (!afterIsPunct || beforeIsWhitespace || beforeIsPunct)
isRightFlanking := !beforeIsWhitespace && (!beforeIsPunct || afterIsWhitespace || afterIsPunct)
var canOpen, canClose bool
if lex.ItemUnderscore == token {
    canOpen = isLeftFlanking && (!isRightFlanking || beforeIsPunct)
    canClose = isRightFlanking && (!isLeftFlanking || afterIsPunct)
} else {
    canOpen = isLeftFlanking
    canClose = isRightFlanking
}
```

所以如果需要加粗的话，需要在这个位置插入一个空格：

```
**故：恒无，欲也以观其眇。 **恒有，欲也以观其所徼
```

```
**故：恒无，欲也以观其眇。 ** 恒有，欲也以观其所徼
```

从 Markdown 规范定义来看，对类似中文非空格分词的语言（CJK）支持有点不够友好了，谁让这范不是咱们自己定义的呢.....