

# substr.sty\*—Substring Functions with L<sup>A</sup>T<sub>E</sub>X

Harald Harders<sup>†</sup>

March 30, 2010

## Abstract

This documentation of `substr.sty` has been typeset by automatical ASCII/UTF-8 → L<sup>A</sup>T<sub>E</sub>X enhancement provided by `makedoc.sty` and `niceverb.sty`'s “auto mode.”<sup>1</sup> Some extended formatting—kind of sectioning, switches to `\tt`—have been achieved by string replacements specific to the comment text of the package file (see source file `substr.tex`).

`substr.sty` especially demonstrates the rather rare ‘%<sub>␣</sub>’ style of package comments that is somewhat favoured by `makedoc` because this way “true comments” are easily distinguishable from mere “commenting out.”

```
1 %%  
2 %% substr.sty  
3 %%
```

This package provides commands to deal with substrings in strings: Determine if a string contains a substring, count appearances of a substring in a string.

## Commands:

`\IfSubStringInString{substring}{string}{true part}{false part}`  
This command searches `⟨substring⟩` in `⟨string⟩` and executes the `⟨true part⟩` if it is and else the `⟨else part⟩`

`\IfCharInString{char}{string}{true part}{false part}` Actually the same as `\IfSubStringInString`.

`\BehindSubString{substring}{string}` Returns the part of `⟨string⟩` that is on the behind `⟨substring⟩`. Always the first appearance of `⟨substring⟩` is taken.

`\BeforeSubString{substring}{string}` Returns the part of `⟨string⟩` that is on the before `⟨substring⟩`. Always the first appearance of `⟨substring⟩` is taken.

`\CountSubStrings{substring}{string}` Counts the number of appearances of `⟨substring⟩` in `⟨string⟩` and returns it as text.

---

\*See <http://ctan.org/pkg/substr> for more about `substr.sty`.

<sup>†</sup>Joint work with Heiko Oberdiek, one line + documentation tool by Uwe Lück.

<sup>1</sup>See <http://ctan.org/pkg/nicetext> for more on these packages.

`\SubStringsToCounter{counter}{substring}{string}` Counts the number of appearances of `<substring>` in `<string>` and sets the counter `<counter>` to that value.

`\IfBeforeSubStringEmpty{substring}{string}{true part}{false part}` Calls `<true part>` if `<substring>` is equal to the beginning of `<string>`. Else call `<false part>`.

`\IfBehindSubStringEmpty{substring}{string}{true part}{false part}` Calls `<true part>` if `<substring>` is equal to the end of `<string>`. Else call `<false part>`.

### History of this package:

The package arises from a posting of me in the newsgroup `de.comp.text.tex` in which I asked how to find out if a substring is included in a string. Heiko Oberdiek (`oberdiek@ruf.uni-freiburg.de`) posted the commands `\IfSubStringInString` and `\IfCharInString` and suggested to write a command which counts the appearances in a string. So, I wrote the commands `\CountSubStrings` and `\SubStringsToCounter`. After I wrote this package I sent it to Heiko Oberdiek who improved and rewrote many parts of it.

### ChangeLog

2009/10/20 v1.2 Uwe Lück - Fix `\IfSubStringInString` which did not work if the string was contained in the substring

2005/11/29 v1.1 Harald Harders - Add `\IfBeforeSubStringEmpty` and `\IfBehindSubStringEmpty`.

**Copyright** 2000, 2005, 2009 Harald Harders

This program can be redistributed and/or modified under the terms of the L<sup>A</sup>T<sub>E</sub>X Project Public License Distributed from CTAN archives in directory `macros/latex/base/lppl.txt`; either version 1 of the License, or any later version.

2009-10-20 Harald Harders `h.harders@tu-bs.de`

```

4  \ProvidesPackage{substr}[2009/10/20 v1.2 Handle substrings]
5  % expands the first and second argument with
6  % \protected@edef and calls #3 with them:
7  \newcommand\su@ExpandTwoArgs[3]{%
8  \protected@edef\su@SubString{#1}%
9  \protected@edef\su@String{#2}%
10 \expandafter\expandafter\expandafter#3%
11 \expandafter\expandafter\expandafter{%
12   \expandafter\su@SubString\expandafter
13   }\expandafter{\su@String}%
14 }

```

tests if #1 in #2. If yes execute #3, else #4

```

15 \newcommand*\IfSubStringInString[2]{%
16   \su@ExpandTwoArgs{#1}{#2}\su@IfSubStringInString
17 }
18 \newcommand*\su@IfSubStringInString[2]{%
19   \def\su@compare##1#1##2\@nil{%
20     \def\su@param{##2}%
21     \ifx\su@param\@empty
22       \expandafter\@secondoftwo
23     \else
24       \expandafter\@firstoftwo
25     \fi
26   }%
27   \su@compare#2\@nnil#1\@nil
28 }

```

tests if #1 in #2. If yes execute #3, else #4

```

29 \newcommand\IfCharInString{}
30 \let\IfCharInString\IfSubStringInString

```

returns the part of the string behind the found substring

```

31 \newcommand*\BehindSubString[2]{%
32   \su@ExpandTwoArgs{#1}{#2}\su@BehindSubString
33 }
34 \newcommand*\su@BehindSubString[2]{%
35   \def\su@rest##1#1##2\@nil{##2}%
36   \IfSubStringInString{#1}{#2}{\su@rest#2\@nil}{}%
37 }

```

returns the part of the string before the found substring

```

38 \newcommand*\BeforeSubString[2]{%
39   \su@ExpandTwoArgs{#1}{#2}\su@BeforeSubString
40 }
41 \newcommand*\su@BeforeSubString[2]{%
42   \def\su@rest##1#1##2\@nil{##1}%
43   \IfSubStringInString{#1}{#2}{\su@rest#2\@nil}{#2}%
44 }

```

calls #3 if part of string before substring is empty, otherwise calls #4.

```

45 \newcommand*\IfBeforeSubStringEmpty[2]{%
46   \su@ExpandTwoArgs{#1}{#2}\su@IfBeforeSubStringEmpty
47 }
48 \newcommand*\su@IfBeforeSubStringEmpty[4]{%
49   \def\su@rest##1#1##2\@nil{##1}%
50   \IfSubStringInString{#1}{#2}{%
51     \edef\su@resta{\su@rest#2\@nil}%
52     \ifx\@empty\su@resta #3\else #4\fi
53   }{#4}%
54 }

```

calls #3 if part of string after substring is empty, otherwise calls #4.

```

55 \newcommand*\IfBehindSubStringEmpty[2]{%
56   \su@ExpandTwoArgs{#1}{#2}\su@IfBehindSubStringEmpty
57 }
58 \newcommand*\su@IfBehindSubStringEmpty[4]{%
59   \def\su@rest##1#1##2\@nil{##2}%
60   \IfSubStringInString{#1}{#2}{%
61     \edef\su@resta{\su@rest#2\@nil}%
62     \ifx\@empty\su@resta #3\else #4\fi
63   }{#4}%
64 }

counter for counting appearances

65 \newcounter{su@anzahl}
66 % #1: String
67 % #2: Substring
68 % #3: Counter
69 \newcommand*\su@StringSubstringCounter[3]{%
70   \su@IfSubStringInString{#2}{#1}{%
71     \stepcounter{#3}%
72     \def\su@rest##1#1##2\@nil{##2}%
73     \expandafter\su@StringSubstringCounter\expandafter
74     {\su@rest#1\@nil}{#2}{#3}%
75   }{}%
76 }
77 \newcommand*\CountSubStrings[2]{%
78   \su@ExpandTwoArgs{#1}{#2}\su@CountSubStrings
79 }
80 \newcommand*\su@CountSubStrings[2]{%
81   \setcounter{su@anzahl}{0}%
82   \su@StringSubstringCounter{#2}{#1}{su@anzahl}%
83   \thesu@anzahl
84 }
85 % #1: counter
86 % #2: substring
87 % #3: string
88 \newcommand*\SubStringsToCounter[3]{%
89   \su@ExpandTwoArgs{#2}{#3}\su@SubStringsToCounter{#1}%
90 }
91 % #1: substring
92 % #2: string
93 % #3: counter
94 \newcommand*\su@SubStringsToCounter[3]{%
95   \setcounter{#3}{0}%
96   \su@StringSubstringCounter{#2}{#1}{#3}%
97 }
98 \endinput

```

EOF