

# The askinclude package

Pablo A. Straub

Heiko Oberdiek\*

<heiko.oberdiek at gmail.com>

2016/05/14 v2.3

## Abstract

This package replaces `\includeonly` by an interactive user interface.

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	User interface . . . . .	2
1.2	Single asking mode . . . . .	4
1.3	Regular expressions . . . . .	4
1.3.1	Method <code>makematch</code> . . . . .	4
1.3.2	Method <code>pdfmatch</code> . . . . .	5
1.4	History . . . . .	6
1.5	New features of version 2 . . . . .	6
1.6	Compatibility to version 1 . . . . .	7
<b>2</b>	<b>Implementation</b>	<b>7</b>
2.1	Header and catcodes . . . . .	7
2.2	Delay action after preamble . . . . .	8
2.3	Write entry in <code>.aux</code> file . . . . .	8
2.4	Remember <code>\include</code> files . . . . .	8
2.5	Redefine <code>\include</code> . . . . .	9
2.6	Store answer and match method in <code>.aux</code> file . . . . .	10
2.7	Support for <code>\@includeonly</code> . . . . .	10
2.8	Stop if <code>\includeonly</code> is active . . . . .	10
2.9	<code>\includeonly</code> . . . . .	11
2.10	Options . . . . .	11
2.11	Strings . . . . .	12
2.12	Set defaults . . . . .	12
2.13	Questions . . . . .	13
2.13.1	Generic question header . . . . .	13
2.13.2	Display files . . . . .	13
2.13.3	Previous answer . . . . .	14
2.13.4	Current regular expression support . . . . .	14
2.13.5	Main question text . . . . .	15
2.13.6	Main question . . . . .	15
2.14	Evaluate answer of main question . . . . .	16
2.15	Ask for each file . . . . .	19

---

\*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

<b>3</b>	<b>Test</b>	<b>20</b>
3.1	Included files	20
3.2	Special test setup	21
3.3	Common start	21
3.4	First run tests	24
3.5	Second run tests	29
3.6	Compatibility for changing from old to new	33
3.7	Regular expressions	34
3.8	Common end	35
<b>4</b>	<b>Installation</b>	<b>35</b>
4.1	Download	35
4.2	Bundle installation	35
4.3	Package installation	35
4.4	Refresh file name databases	36
4.5	Some details for the interested	36
<b>5</b>	<b>Catalogue</b>	<b>37</b>
<b>6</b>	<b>Acknowledgement</b>	<b>37</b>
<b>7</b>	<b>References</b>	<b>37</b>
<b>8</b>	<b>History</b>	<b>38</b>
	[1991/11/05 v0.95]	38
	[1991/11/05 v0.99]	38
	[1991/11/07 v1.0]	38
	[1991/12/13 v1.1]	38
	[1994/11/02 v1.2e]	38
	[2007/10/15 v1.0 (askinc)]	38
	[2007/10/23 v2.0]	38
	[2008/08/11 v2.1]	39
	[2011/12/02 v2.2]	39
	[2016/05/14 v2.3]	39
<b>9</b>	<b>Index</b>	<b>39</b>

# 1 Documentation

## 1.1 User interface

The `\include` feature of L<sup>A</sup>T<sub>E</sub>X allows the division of the text body in separate compilation units. It is page based and stores counter settings in additional auxiliary files. `\includeonly` limits the actually included files to the ones, listed in its argument.

Example:

```

\documentclass{report}

\usepackage{askinclude}

% Don't use \include in the preamble.
% Instead \input can be used or make a package

\begin{document}
\include{title}
\include{introduction}
\include{summary}
\include{index}
\end{document}

```

Package `askinclude` adds an interactive user interface. The user is asked which files he want to include and sets `\includeonly` accordingly.

The first time the package is used the question looks like:

```
*****
*** Package askinclude Question ***
*****

Previous answer (noregexp):
[*]

Regular expressions:
--> [noregexp] disabled
    [makematch] enabled, using package 'makematch'
    [pdfmatch] enabled, using pdfTeX's \pdfmatch

Which files do you want to include?
[foo,bar] comma separated file or pattern list
[*]       all files
[-]       no files
[?]       ask for each file
[]        use the previous answer

\answer=chapter2,summary
```

Square brackets are used as quotation symbols, they are not part of the answer. The answer by the user follows in the last line after `\answer=`.

**[foo,bar]:** A comma separated file list (also one file only) is passed to macro `\includeonly` directly. If regular expression support is enabled, the entries in the list are interpreted as file name patterns that will be matched against the file names in `\include`, see [section 1.3](#).

**[\*]:** This is a shortcut for specifying *all* files.

**[-]:** This means *none* of the files is included.

**[?]:** The switches in single asking mode. The user will be asked at each `\include`. See [section 1.2](#).

**[]:** Just pressing return (empty string) uses the default answer, that was stored in the auxiliary file (`.aux`) of the previous `LATEX`run with package `askinclude`. Without previous runs the default is `[*]` for all files.

**[noregexp]:** Turns regular expression support off.

**[makematch]:** Enables regular expression support package `makematch` provides.

**[pdfmatch]:** It is shown, if pdf`TEX` 1.30 or later is used. It relies on an undocumented feature `\pdfmatch` that may change in future versions of pdf`TEX`.

Package `askinclude` remembers the files of `\include` in the auxiliary file and the given answer. they are presented at the next run at the start of the question:

```
*****
*** Package askinclude Question ***
*****

Files, found by previous run in \include:
(-) title
(-) introduction
(-) chapter1
(+) chapter2
(-) chapter3
```

(+) summary

Previous answer (noregexp):  
[chapter2,summary]

...

(+) and (-) marks files that were and were not included in the previous run. The file list and marks are not always correct. For example, if `\include` statements are added or removed. If the method of regular expression support is different, then the marks only reflects the mode of the previous answer.

## 1.2 Single asking mode

The special answer ‘?’ trigger single asking mode, at the next `\include` the user is asked again:

```
*****  
*** Package askinclude Question ***  
*****
```

Include ‘title’? [y]es, [n]o, [A]ll, [N]one, [D]efault (n):

`\answer=`

Use the letter inside the square brackets as answer. An empty string uses the default answer, the letter in parentheses. The answers are:

[y]: The file is included.

[n]: The file is excluded.

[A]: This file and all following files are included.

[N]: This file and all following files are excluded.

[D]: Use the default answer for this and all following files.

If the file is known from the previous run, then the default is ‘y’ or ‘n’ corresponding to the mark ‘(+)’ or ‘(-)’ in the list of previous files. Otherwise ‘y’ is used as default.

## 1.3 Regular expressions

Three methods are available that can also set by package options:

**noregexp:** Regular expression support is disabled.

**makematch:** Use package `makematch`.

**pdfmatch:** Use pdf $\TeX$ ’s `\pdfmatch` if available.

Without options the value remembered in the auxiliary file is used, otherwise the default is `noregexp`.

### 1.3.1 Method `makematch`

Package `makematch` (part of `qstest`), written by David Kastrup provides pattern matching based on wildcards:

[\*]: The star matches zero or more characters. Thus a lonely star matches anything.

[!]: The exclamation mark at the begin of a pattern negates it, thus the matching files are excluded.

The pattern list is evaluated for a file name from left to right. Always a file name must match the whole pattern. The file is included if it is matched by a pattern and not excluded afterwards. Examples:

Pattern list	File names	Result
foobar	foobar,dummy	foobar
foo*	foobar,xfoobar	foobar
*foo*	xfoobar	xfoobar
f*b*r	foobar	foobar
*	any	any
!foobar	foobar,other	other
!foo*	xfoobar	xfoobar
in*,!*dex	intro,index	intro
!index,in*	intro,index	intro,index
*,!chap2	chap1,chap2,chap3	chap1,chap3

The example in the last line shows, how the behaviour of package `excludeonly` [2] can be simulated.

The documentation of package `makematch` tells the the empty pattern does not match anything. But if you answer the question with an empty string the answer of the previous run is used. Also curly braces are not reliable because the can get stripped during the processing of the answer. Using a pair of empty curly braces or the comma for two empty patterns currently works, but this is specific for the current implementation and may change in future. But the special answer ‘-’ is available.

### 1.3.2 Method `pdfmatch`

In pdfTeX 1.30 the new primitive `\pdfmatch` was introduced. It is not documented yet, because it was not decided if it remains and which implementation of regular expressions should be used. Thus the situation may change in later versions of pdfTeX. Currently (1.30 until 1.40) it uses a library for POSIX 1003.2 regular expressions, see manual page “`regex(7)`”.

If a pattern in the pattern list is invalid, the package prints a warning and removes the pattern from the list. Actually there are two warnings, the other warning is generated by pdfTeX itself.

In opposite to `makematch` the pattern only needs to match the specified part of the file name. For matching the whole name, the symbols ‘^’ and ‘\$’ for begin and end of string must be included.

Additionally an initial exclamation mark ‘!’ implements the same logic as method `makematch`. The match result is inverted, the matched file excluded.

Examples:

Pattern list	File names	Result
foobar	foobar,dummy	foobar
foo.*	foobar,xfoobar	foobar,xfoobar
^foo.*	foobar,xfoobar	foobar
foo\$	foo,foobar	foo
fo?bar	fbar,fobar,foobar	fbar,fobar
fo+bar	fbar,fobar,foobar	fobar,foobar
fo*bar	fbar,fobar,foobar	fbar,fobar,foobar
chap3	chap3,chap32	chap1,chap3,chap32
chap[13]	chap1,chap2,chap3	chap1,chap3
chap(1 3)\$	chap1,chap3,chap33	chap1,chap3
.*	any	any
!foobar	foobar,other	other
in.*,!dex	intro,index	intro
!index,in.*	intro,index	intro,index
.*,!chap2	chap1,chap2,chap3	chap1,chap3
.*,!bla	foo,bla,blabla	foo
.*,!^bla\$	foo,bla,blabla	foo,blabla

The example in the last line shows, how the behaviour of package `excludeonly` [2] can be simulated. As the third last line shows, that sometimes ‘^’ and ‘\$’ can be omitted, but this will not work always, shown in the second last line.

## 1.4 History

Package `askinclude` was invented by Pablo A. Straub in 1991 and maintained until 1994.

In October 2007 it was removed from “TeX Live” because of the no-commercial license [1]. Herbert Voß initiated a search for the author in the newsgroup `comp.text.tex` [3] to ask him to change the license.

It seems that the author could not be reached any more. Joseph Wright suggested a reimplementaion that I (HO) have done as package `askinc`. Soon it was enhanced by feature requests.

Then Ulrike Fischer found email addresses of the original author and Jim Heferon succeeded in contacting him. He allowed a change of the license to LPPL and the transfer of the maintainership to someone else.

Thus I merged both packages as `askinclude` in a new major version 2. The license is LPPL.

## 1.5 New features of version 2

- License is now LPPL.
- A list of `\include` files that are detected by the previous run is printed before the question.
- If the special answer ‘?’ is given, then the user is asked at each `\include`.
- Support for regular expressions.
- The remaining stuff in the `.aux` file does not cause errors, if the package is removed.
- The use of `\includeonly` before the package loading time is detected and respected by taking precedence. After the package is loaded, `\includeonly` is disabled and prints a warning.

## 1.6 Compatibility to version 1

- The user interface of version 1 with its special answers ‘\*’, ‘-’, and ‘’ remain unchanged, but it is extended by version 2 with the special answer ‘?’ and support for regular expressions can be enabled.
- As reimplementing the internal macros have changed of course.
- .aux files that are written by version 1.x will work with version 2.x because \@includeonly is recognized.
- L<sup>A</sup>T<sub>E</sub>X 2.09: The package works as \documentstyle option in compatibility mode of L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. It is possible that it will not work in pure L<sup>A</sup>T<sub>E</sub>X 2.09.

## 2 Implementation

### 2.1 Header and catcodes

```
1 \<package>
2 \NeedsTeXFormat{LaTeX2e}
3 \begingroup\catcode61\catcode48\catcode32=10\relax%
4 \catcode13=5 % ^^M
5 \endlinechar=13 %
6 \catcode123=1 % {
7 \catcode125=2 % }
8 \catcode64=11 % @
9 \def\x{\endgroup
10 \expandafter\edef\csname AskInc@AtEnd\endcsname{%
11 \endlinechar=\the\endlinechar\relax
12 \catcode13=\the\catcode13\relax
13 \catcode32=\the\catcode32\relax
14 \catcode35=\the\catcode35\relax
15 \catcode61=\the\catcode61\relax
16 \catcode64=\the\catcode64\relax
17 \catcode123=\the\catcode123\relax
18 \catcode125=\the\catcode125\relax
19 }%
20 }%
21 \x\catcode61\catcode48\catcode32=10\relax%
22 \catcode13=5 % ^^M
23 \endlinechar=13 %
24 \catcode35=6 % #
25 \catcode64=11 % @
26 \catcode123=1 % {
27 \catcode125=2 % }
28 \def\TMP@EnsureCode#1#2{%
29 \edef\AskInc@AtEnd{%
30 \AskInc@AtEnd
31 \catcode#1=\the\catcode#1\relax
32 }%
33 \catcode#1=#2\relax
34 }
35 \TMP@EnsureCode{33}{12}% !
36 \TMP@EnsureCode{39}{12}% '
37 \TMP@EnsureCode{40}{12}% (
38 \TMP@EnsureCode{41}{12}% )
39 \TMP@EnsureCode{42}{12}% *
40 \TMP@EnsureCode{43}{12}% +
41 \TMP@EnsureCode{44}{12}% ,
42 \TMP@EnsureCode{45}{12}% -
43 \TMP@EnsureCode{46}{12}% .
44 \TMP@EnsureCode{47}{12}% /
```

```

45 \TMP@EnsureCode{58}{12}% :
46 \TMP@EnsureCode{62}{12}% >
47 \TMP@EnsureCode{63}{12}% ?
48 \TMP@EnsureCode{91}{12}% [
49 \TMP@EnsureCode{93}{12}% ]
50 \TMP@EnsureCode{96}{12}% '
51 \edef\AskInc@AtEnd{\AskInc@AtEnd\noexpand\endinput}
52 \ProvidesPackage{askinclude}%
53 [2016/05/14 v2.3 Interactive asking of included files]%

    Luatex compat
54 \ifx\pdfextension\@undefined\else
55   \RequirePackage{luatex85}
56 \fi

```

## 2.2 Delay action after preamble

`\AskInc@AtBeginDocument` Some actions such as `.aux` file writing cannot be performed in the preamble. `\AtBeginDocument` delays such action until `\begin{document}`. However `\AtBeginDocument` must not be used after that time (`\@onlypreamble`). Thus `\AskInc@AtBeginDocument` uses `\AtBeginDocument` in the preamble and afterwards executes the code immediately.

```

57 \AtBeginDocument{%
58   \let\AskInc@AtBeginDocument\@firstofone
59 }
60 \providecommand*\AskInc@AtBeginDocument{\AtBeginDocument}

```

## 2.3 Write entry in .aux file

`\AskInc@WriteAux` The first argument is written to the main `.aux` file and the second argument is executed if writing was possible. Remember `\nofiles` prevents writing to auxiliary files.

```

61 \def\AskInc@WriteAux#1#2{%
62   \AskInc@AtBeginDocument{%
63     \if@filesw
64       \immediate\write\@mainaux{#1}%
65       #2%
66     \fi
67   }%
68 }

```

## 2.4 Remember \include files

`\AskInc@CharHash`

```

69 \edef\AskInc@CharHash{\string#}

```

`\AskInc@WriteDefAddList`

```

70 \def\AskInc@WriteDefAddList{%
71   \AskInc@WriteAux{%

```

`\AskInc@AddList`

```

72   \string\providecommand*\string\AskInc@AddList[2]{%
73     \string\ifcase\string\ifx\AskInc@CharHash1\string\@undefined
74       \string\else
75         \string\ifx\AskInc@CharHash1\string\relax
76         \string\else
77         \string\ifx\AskInc@CharHash1\string\@empty
78         \string\else
79         1%
80         \string\fi
81       \string\fi
82     \string\fi

```



```

83         0 %
84         \string\gdef\AskInc@CharHash1{\AskInc@CharHash2}%
85         \string\else
86         \string\g@addto@macro\AskInc@CharHash1{,\AskInc@CharHash2}%
87         \string\fi
88     }%

89 }{%
90     \global\let\AskInc@WriteDefAddList\relax
91 }%
92 }

\AskInc@WriteDefAddFile

93 \def\AskInc@WriteDefAddFile{%
94     \AskInc@WriteDefAddList
95     \AskInc@WriteAux{%

\AskInc@AddFile

96     \string\providecommand\string\AskInc@AddFile{%
97         \string\AskInc@AddList\string\AskInc@Files
98     }%

99 }{%
100     \global\let\AskInc@WriteDefAddFile\relax
101 }%
102 }

\AskInc@AddFile

103 \AtBeginDocument{%
104     \global\let\AskInc@AddFile\@gobble
105 }

\AskInc@WriteAddFile

106 \def\AskInc@WriteAddFile#1{%
107     \AskInc@WriteDefAddFile
108     \AskInc@WriteAux{%
109         \string\AskInc@AddFile{#1}%
110     }{}%
111 }

\AskInc@Files

112 \providecommand*{\AskInc@Files}{}

```

**Summary:** Now `\AskInc@Files` is always defined and either empty or contains a comma separated list of `\include` files of previous run.

The `\include` files are remembered by using `\AskInc@WriteAddFile` at each `\include`.

## 2.5 Redefine `\include`

```

\AskInc@OrgInclude

113 \newcommand{\AskInc@OrgInclude}{}
114 \let\AskInc@OrgInclude\include

\AskInc@AskFile

115 \let\AskInc@AskFile\@gobble

\AskInc@MatchFile

116 \let\AskInc@MatchFile\@gobble

```

`\include`

```
117 \renewcommand*{\include}[1]{%
118   \AskInc@WriteAddFile{#1}%
119   \AskInc@AskFile{#1}%
120   \AskInc@MatchFile{#1}%
121   \AskInc@OrgInclude{#1}%
122 }
```

## 2.6 Store answer and match method in .aux file

```
123 \AtEndOfPackage{%
124   \AtBeginDocument{%
125     \ifundefined{AskInc@Answer}{%
126       \AskInc@WriteAux{%
127         \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
128       }{}%
129     }%
130     \ifundefined{AskInc@MatchMethod}{%
131       \AskInc@WriteAux{%
132         \string\gdef\string\AskInc@MatchMethod{\AskInc@MatchMethod}%
133       }{}%
134     }%
135   }%
136 }
```

## 2.7 Support for \@includeonly

`\@includeonly` Macro `\@includeonly` is defined for compatibility reasons only, if the .aux file was written by versions before 2 of package `askinclude`.

```
137 \providecommand*{\@includeonly}[1]{%
138   \gdef\AskInc@Answer{#1}%
139   \ifx\AskInc@Answer\@empty
140     \gdef\AskInc@Answer{-}%
141   \fi
142   \global\let\AskInc@MatchMethod\@empty
143 }
```

## 2.8 Stop if \includeonly is active

If `\includeonly` is used before, the questions are skipped and the data in the main auxiliary file is preserved.

```
144 \if@partsw
145   \PackageWarningNoLine{askinclude}{%
146     Previous \string\includeonly\space detected,\MessageBreak
147     therefore asking for files is suppressed%
148   }%
149 \fi
```

Dummy options.

```
149 \DeclareOption{makematch}{}%
150 \DeclareOption{pdfmatch}{}%
151 \DeclareOption{noregexp}{}%
152 \ProcessOptions*
```

Preserve `\AskInc@MatchFiles`.

```
153 \ifundefined{AskInc@MatchFiles}{%
154 }{}%
155 \ifx\AskInc@MatchMethod\@empty
156   \else
157     \AskInc@WriteAux{%
158       \string\gdef\string\AskInc@MatchFiles{\AskInc@MatchFiles}%
159     }{}%
160   \fi
161 }
```

```

162 \expandafter\AskInc@AtEnd
163 \fi%

```

## 2.9 \includeonly

\AskInc@OrgIncludeOnly

```

164 \newcommand{\AskInc@OrgIncludeOnly}{}%
165 \let\AskInc@OrgIncludeOnly\includeonly

```

\includeonly

```

166 \renewcommand*{\includeonly}[1]{%
167 \PackageWarning{askinclude}{%
168 \string\includeonly\space is managed by package 'askinclude',%
169 \MessageBreak
170 therefore it is ignored%
171 }%
172 }

```

## 2.10 Options

\AskInc@StringNoRegExp

```

173 \def\AskInc@StringNoRegExp{noregexp}

```

\AskInc@StringMakeMatch

```

174 \def\AskInc@StringMakeMatch{makematch}

```

\AskInc@StringPdfMatch

```

175 \def\AskInc@StringPdfMatch{pdfmatch}

```

\ifAskInc@PdfMatchAvailable

```

176 \newif\ifAskInc@PdfMatchAvailable
177 \begingroup\expandafter\expandafter\expandafter\endgroup
178 \expandafter\ifx\csname pdfmatch\endcsname\relax
179 \else
180 \AskInc@PdfMatchAvailabletrue
181 \fi

182 \DeclareOption{makematch}{%
183 \let\AskInc@MatchOption\AskInc@StringMakeMatch
184 }
185 \ifAskInc@PdfMatchAvailable
186 \DeclareOption{pdfmatch}{%
187 \let\AskInc@MatchOption\AskInc@StringPdfMatch
188 }%
189 \else
190 \DeclareOption{pdfmatch}{%
191 \PackageWarningNoLine{askinclude}{%
192 Option 'pdfmatch' is not available because\MessageBreak
193 your pdfTeX does not provide \string\pdfmatch
194 }%
195 }%
196 \fi
197 \DeclareOption{noregexp}{%
198 \let\AskInc@MatchOption\@empty
199 }
200 \ProcessOptions*

```

## 2.11 Strings

```
\AskInc@CharAsterisk
201 \def\AskInc@CharAsterisk{*}

\AskInc@CharMinus
202 \def\AskInc@CharMinus{-}

\AskInc@CharPlus
203 \def\AskInc@CharPlus{+}

\AskInc@CharQuestion
204 \def\AskInc@CharQuestion{?}

\AskInc@CharExclam
205 \def\AskInc@CharExclam{!}

\AskInc@CharYes
206 \def\AskInc@CharYes{y}

\AskInc@CharNo
207 \def\AskInc@CharNo{n}

\AskInc@CharAll
208 \def\AskInc@CharAll{A}

\AskInc@CharNone
209 \def\AskInc@CharNone{N}

\AskInc@CharDefault
210 \def\AskInc@CharDefault{D}
```

## 2.12 Set defaults

```
\AskInc@SanitizeAnswer
211 \def\AskInc@SanitizeAnswer#1#2{%
212   \begingroup
213   \escapechar='\%
214   \edef\{{\string\}}%
215   \edef\}\{\string\}%
216   \edef\({\string\}%
217   \edef\){\string\}%
218   \edef\*{\string\*}%
219   \edef\+{\string\+}%
220   \edef\?{\string\?}%
221   \edef\|{\string\|}%
222   \edef\~{\string\~}%
223   \edef\${\string\$}%
224   \edef\{\{\string\}%
225   \edef\}\{\string\}%
226   \edef\.\{\string\}%
227   \xdef#1{%
228     \expandafter\expandafter\expandafter\zap@space
229     \expandafter#2\space\@empty
230   }%
231 \endgroup
232 }
```

```

233 \AtBeginDocument{%
234 \ifundefined{AskInc@Answer}{%
235 \global\let\AskInc@Answer\AskInc@CharAsterisk
236 }{%
237 \AskInc@SanitizeAnswer\AskInc@Answer\AskInc@Answer
238 }%
239 \global\let\AskInc@PreviousAnswer\AskInc@Answer
240 \ifundefined{AskInc@MatchMethod}{%
241 \global\let\AskInc@MatchMethod\@empty
242 }{%
243 \global\let\AskInc@PreviousMatchMethod\AskInc@MatchMethod
244 \ifundefined{AskInc@MatchOption}{%
245 \let\AskInc@MatchOption\AskInc@MatchMethod
246 }{%
247 \ifx\AskInc@MatchOption\AskInc@StringPdfMatch
248 \ifAskInc@PdfMatchAvailable
249 \else
250 \let\AskInc@MatchOption\@empty
251 \fi
252 \fi
253 \ifundefined{AskInc@MatchFiles}{%
254 \let\AskInc@MatchFiles\@empty
255 }{%
256 }

```

## 2.13 Questions

### 2.13.1 Generic question header

\AskInc@QuestionHeader

```

257 \def\AskInc@QuestionHeader{%
258 \typeout{}%
259 \typeout{*****}%
260 \typeout{*** Package askinclude Question ***}%
261 \typeout{*****}%
262 \typeout{}%
263 }

```

### 2.13.2 Display files

\AskInc@TypeoutFiles

```

264 \def\AskInc@TypeoutFiles{%
265 \global\let\AskInc@PreviousMatches\@empty
266 \ifx\AskInc@Files\@empty
267 \else
268 \typeout{Files, found by previous run in \string\include:}%
269 \@for\AskInc@File:=\AskInc@Files\do{%
270 \ifx\AskInc@Answer\AskInc@CharAsterisk
271 \let\AskInc@Mark\AskInc@CharPlus
272 \else
273 \let\AskInc@Mark\AskInc@CharMinus
274 \ifx\AskInc@Answer\AskInc@CharMinus
275 \else
276 \ifx\AskInc@MatchMethod\@empty
277 \expandafter\@firstoftwo
278 \else
279 \expandafter\@secondoftwo
280 \fi
281 {\@for\AskInc@Temp:=\AskInc@Answer}%
282 {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
283 \do{%
284 \ifx\AskInc@File\AskInc@Temp

```

```

285         \let\AskInc@Mark\AskInc@CharPlus
286     \fi
287 }%
288 \fi
289 \fi
290 \typeout{ \@spaces(\AskInc@Mark) \AskInc@File}%
291 \ifx\AskInc@Mark\AskInc@CharPlus
292     \ifx\AskInc@PreviousMatches\@empty
293         \global\let\AskInc@PreviousMatches\AskInc@File
294     \else
295         \expandafter\g@addto@macro
296         \expandafter\AskInc@PreviousMatches{%
297             \expandafter,\AskInc@File
298         }%
299     \fi
300 \fi
301 }%
302 \typeout{}%
303 \fi
304 }

```

### 2.13.3 Previous answer

\AskInc@TypeoutPreviousAnswer

```

305 \def\AskInc@TypeoutPreviousAnswer{%
306     \typeout{%
307         Previous answer (%
308         \ifx\AskInc@MatchMethod\@empty
309             noregexp%
310         \else
311             \AskInc@MatchMethod
312         \fi
313         ):%
314     }%
315     \typeout{ \@spaces[\AskInc@Answer]]}%
316     \typeout{}%
317 }

```

### 2.13.4 Current regular expression support

\AskInc@TypeoutMatchMethod

```

318 \def\AskInc@TypeoutMatchMethod{%
319     \typeout{Regular expressions:}%
320     \AskInc@TypeoutMethod\@empty{noregexp}{%
321         \space disabled%
322     }%
323     \AskInc@TypeoutMethod\AskInc@StringMakeMatch{makematch}{%
324         enabled, using package ‘makematch’%
325     }%
326     \ifAskInc@PdfMatchAvailable
327         \AskInc@TypeoutMethod\AskInc@StringPdfMatch{pdfmatch}{%
328             \space enabled, using pdfTeX’s \string\pdfmatch
329         }%
330     \fi
331     \typeout{}%
332 }

```

\AskInc@TypeoutMethod

```

333 \def\AskInc@TypeoutMethod#1#2#3{%
334     \typeout{%
335         \ifx#1\AskInc@MatchOption \space--> \else\@spaces\space\fi
336         [#2] #3% hash-ok

```

```

337 }%
338 }

```

### 2.13.5 Main question text

\AskInc@TypeoutMainQuestion

```

339 \def\AskInc@TypeoutMainQuestion{%
340 \typeout{Which files do you want to include?}%
341 \typeout{ \@spaces[foo,bar] \space\space
342   comma separated file or pattern list}%
343 \typeout{ \@spaces[*] \@spaces\@spaces all files}%
344 \typeout{ \@spaces[-] \@spaces\@spaces no files}%
345 \typeout{ \@spaces[?] \@spaces\@spaces ask for each file}%
346 }

```

### 2.13.6 Main question

\AskInc@MainQuestion

```

347 \def\AskInc@MainQuestion{%
348 \AskInc@QuestionHeader
349 \AskInc@TypeoutFiles
350 \AskInc@TypeoutPreviousAnswer
351 \AskInc@TypeoutMatchMethod
352 \AskInc@TypeoutMainQuestion
353 \begingroup
354 \@makeother\*%
355 \@makeother\-%
356 \@makeother\?%
357 \@makeother\,%
358 \@makeother\!%
359 \@makeother\^%
360 \@makeother\$%
361 \@makeother\(%
362 \@makeother\)%
363 \@makeother\+%
364 \@makeother\[%
365 \@makeother\]%
366 \@makeother\\%
367 \@makeother\|%
368 \@makeother\:%
369 \@makeother\.%
370 \typein[\answer]%
371 { \@spaces[] \space\@spaces\@spaces use the previous answer}%
372 \typeout{}%
373 \AskInc@SanitizeAnswer\AskInc@@Answer\answer
374 \endgroup
375 \let\AskInc@Next\AskInc@MainQuestion
376 \ifx\AskInc@@Answer\AskInc@StringNoRegExp
377 \let\AskInc@MatchOption\@empty
378 \else
379 \ifx\AskInc@@Answer\AskInc@StringMakeMatch
380 \let\AskInc@MatchOption\AskInc@StringMakeMatch
381 \else
382 \ifx\AskInc@@Answer\AskInc@StringPdfMatch
383 \ifAskInc@PdfMatchAvailable
384 \let\AskInc@MatchOption\AskInc@StringPdfMatch
385 \else
386 \PackageWarning{askinclude}{%
387   Method ‘pdfmatch’ is not available because\MessageBreak
388   your pdfTeX does not provide \string\pdfmatch
389 }%
390 \let\AskInc@MatchOption\@empty

```

```

391     \fi
392   \else
393     \ifx\AskInc@@Answer\@empty
394       \global\let\AskInc@@Answer\AskInc@Answer
395     \fi
396     \let\AskInc@Next\AskInc@MainAnswer
397   \fi
398 \fi
399 \fi
400 \AskInc@Next
401 }

```

## 2.14 Evaluate answer of main question

\AskInc@MainAnswer

```

402 \def\AskInc@MainAnswer{%
403   \ifx\AskInc@@Answer\AskInc@CharQuestion
404     \global\let\AskInc@Answer\@empty
405     \global\let\AskInc@MatchMethod\@empty
406     \global\let\AskInc@AskFile\AskInc@DoAskFile
407     \AskInc@OrgIncludeOnly{}%
408   \else
409     \global\let\AskInc@Answer\AskInc@@Answer
410     \ifx\AskInc@@Answer\AskInc@CharAsterisk
411       \global\let\AskInc@MatchMethod\@empty
412     \else
413       \ifx\AskInc@@Answer\AskInc@CharMinus
414         \global\let\AskInc@MatchMethod\@empty
415         \AskInc@OrgIncludeOnly{}%
416       \else
417         \global\let\AskInc@MatchMethod\AskInc@MatchOption
418         \ifx\AskInc@MatchOption\@empty
419           \expandafter\AskInc@OrgIncludeOnly\expandafter{%
420             \AskInc@Answer
421           }%
422         \else
423           \AskInc@MatchInit
424           \AskInc@OrgIncludeOnly{}%
425         \fi
426       \fi
427     \fi
428   \fi
429   \let\AskInc@Include\AskInc@DoInclude
430 }

```

\AskInc@MatchInit

```

431 \def\AskInc@MatchInit{%
432   \RequirePackage{kvsetkeys}[2007/09/29]%
433   \expandafter\comma@normalize\expandafter{\AskInc@Answer}%
434   \global\let\AskInc@MatchList\comma@list

```

\AskInc@MatchFile

```

435 \def\AskInc@MatchFile##1{%
436   \AskInc@DoMatch{##1}{%
437     \AskInc@WriteAddMatch{##1}%
438     \ifx\@partlist\@empty
439       \gdef\@partlist{##1}%
440     \else
441       \g@addto@macro\@partlist{,##1}%
442     \fi
443   }%
444 }%

```



```

445 \ifx\AskInc@MatchMethod\AskInc@StringMakeMatch
446 \AskInc@MakeMatchInit
447 \else
448 \AskInc@PdfMatchInit
449 \fi
450 \let\AskInc@MatchInit\@undefined
451 }

\AskInc@MakeMatchInit
452 \def\AskInc@MakeMatchInit{%
453 \RequirePackage{makematch}%

\AskInc@DoMatch
454 \def\AskInc@DoMatch##1{%
455 \def\AskInc@Temp{%
456 \MakeMatcher[,]\AskInc@Temp
457 }%
458 \expandafter\AskInc@Temp\expandafter{\AskInc@MatchList}%
459 \AskInc@Temp{\##1}\@firstofone\@gobble
460 }%

461 \let\AskInc@MakeMatchInit\@undefined
462 }

\AskInc@PdfMatchInit
463 \def\AskInc@PdfMatchInit{%

\ifAskInc@Found
464 \newif\ifAskInc@Found

\AskInc@DoMatch
465 \def\AskInc@DoMatch##1{%
466 \AskInc@Foundfalse
467 \expandafter\comma@parse@normalized\expandafter
468 {\AskInc@MatchList}\AskInc@CheckMatch{\##1}%
469 \ifAskInc@Found
470 \expandafter\@firstofone
471 \else
472 \expandafter\@gobble
473 \fi
474 }%

\AskInc@CheckMatch
475 \def\AskInc@CheckMatch##1##2{%
476 \AskInc@@CheckMatch##2\@empty\@nil{\##1}%
477 }%

\AskInc@@CheckMatch
478 \def\AskInc@@CheckMatch##1##2\@nil##3{%
479 \def\AskInc@Temp{\##1}%
480 \ifx\AskInc@Temp\AskInc@CharExclam
481 \ifcase\pdfmatch{\##2}{\##3}=1 %
482 \or
483 \AskInc@Foundfalse
484 \else
485 \AskInc@InvalidPattern{\##2}{\##1##2}%
486 \fi
487 \else
488 \ifcase\pdfmatch{\##1##2}{\##3} %
489 \or
490 \AskInc@Foundtrue
491 \else

```

```

492     \AskInc@InvalidPattern{##1##2}{##1##2}%
493     \fi
494     \fi
495 }%

```

`\AskInc@InvalidPattern` Prints a warning with the regular expression and removes it from the list to prevent further warnings for the same expression.

```

496 \def\AskInc@InvalidPattern##1##2{%
497   \edef\AskInc@Temp{##1}%
498   \PackageWarningNoLine{askinclude}{%
499     Invalid pattern '\AskInc@Temp' for \string\pdfmatch
500   }%
501   \edef\AskInc@Temp{##2}%
502   \global\expandafter\let\expandafter\AskInc@MatchList
503   \expandafter\@empty
504   \expandafter\comma@parse@normalized\expandafter
505   {\AskInc@MatchList}\AskInc@FilterPattern
506 }%

```

`\AskInc@FilterPattern`

```

507 \def\AskInc@FilterPattern##1{%
508   \edef\AskInc@Pattern{##1}%
509   \ifx\AskInc@Temp\AskInc@Pattern
510   \else
511     \ifx\AskInc@MatchList\@empty
512       \global\let\AskInc@MatchList\AskInc@Pattern
513     \else
514       \expandafter\g@addto@macro\expandafter\AskInc@MatchList{%
515         \expandafter,\AskInc@Pattern
516       }%
517     \fi
518   \fi
519 }%

520 \let\AskInc@PdfMatchInit\@undefined
521 }

```

`\AskInc@WriteDefAddMatch`

```

522 \def\AskInc@WriteDefAddMatch{%
523   \AskInc@WriteDefAddList
524   \AskInc@WriteAux{%

```

`\AskInc@AddMatch`

```

525     \string\providecommand\string\AskInc@AddMatch{%
526       \string\AskInc@AddList\string\AskInc@MatchFiles
527     }%

528   }{%
529     \global\let\AskInc@WriteDefAddMatch\relax
530   }%
531 }

```

`\AskInc@AddMatch`

```

532 \AtBeginDocument{%
533   \global\let\AskInc@AddMatch\@gobble
534 }

```

`\AskInc@WriteAddMatch`

```

535 \def\AskInc@WriteAddMatch#1{%
536   \AskInc@WriteDefAddMatch
537   \AskInc@WriteAux{%
538     \string\AskInc@AddMatch{#1}%
539   }{%
540 }

```

\AskInc@SaveAnswer

```
541 \def\AskInc@SaveAnswer{%
542   \AskInc@WriteAux{%
543     \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
544   }{}%
545 }
```

## 2.15 Ask for each file

\AskInc@PreviousAnswer

```
546 \@ifundefined{AskInc@PreviousAnswer}{%
547   \let\AskInc@PreviousAnswer\AskInc@CharAsterisk
548 }{}
```

\AskInc@PreviousMatchMethod

```
549 \@ifundefined{AskInc@PreviousMatchMethod}{%
550   \let\AskInc@PreviousMatchMethod\@empty
551 }{}
```

\AskInc@DoAskFile

```
552 \def\AskInc@DoAskFile#1{%
553   \edef\AskInc@File{#1}%
554   \ifx\AskInc@PreviousAnswer\AskInc@CharAsterisk
555     \let\AskInc@Default\AskInc@CharYes
556   \else
557     \let\AskInc@Default\AskInc@CharNo
558     \ifx\AskInc@PreviousAnswer\AskInc@CharMinus
559     \else
560       \ifx\AskInc@PreviousMatchMethod\@empty
561         \expandafter\@firstoftwo
562       \else
563         \expandafter\@secondoftwo
564       \fi
565       {\@for\AskInc@Temp:=\AskInc@PreviousAnswer}%
566       {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
567     \do{%
568       \ifx\AskInc@Temp\AskInc@File
569         \let\AskInc@Default\AskInc@CharYes
570       \fi
571     }%
572   \fi
573   \ifx\AskInc@Default\AskInc@CharNo
574     \let\AskInc@Default\AskInc@CharYes
575     \@for\AskInc@Temp:=\AskInc@Files\do{%
576       \ifx\AskInc@Temp\AskInc@File
577         \let\AskInc@Default\AskInc@CharNo
578       \fi
579     }%
580   \fi
581 \fi
582 \let\AskInc@@@QuestionHeader\AskInc@QuestionHeader
583 \AskInc@AskQuestion
584 \ifx\AskInc@AskResult\AskInc@CharYes
585   \ifx\AskInc@Answer\@empty
586     \gdef\AskInc@Answer{#1}%
587   \else
588     \g@addto@macro\AskInc@Answer{, #1}%
589   \fi
590   \AskInc@SaveAnswer
591   \let\@partlist\AskInc@Answer
592 \fi
593 }
```

\AskInc@AskQuestion

```

594 \def\AskInc@AskQuestion{%
595   \let\AskInc@Temp\answer
596   \AskInc@@@QuestionHeader
597   \let\AskInc@@@QuestionHeader\relax
598   \typein[\answer]{%
599     Include '\AskInc@File'? [y]es, [n]o, [A]ll, [N]one, [D]efault %
600     (\AskInc@Default):%
601   }%
602   \typeout{%
603     \let\AskInc@AskResult\answer
604     \let\answer\AskInc@Temp
605     \ifx\AskInc@AskResult\@empty
606       \let\AskInc@AskResult\AskInc@Default
607       \let\AskInc@Next\relax
608     \else
609       \ifx\AskInc@AskResult\AskInc@CharYes
610         \let\AskInc@Next\relax
611       \else
612         \ifx\AskInc@AskResult\AskInc@CharNo
613           \let\AskInc@Next\relax
614         \else
615           \ifx\AskInc@AskResult\AskInc@CharAll
616             \gdef\AskInc@AskQuestion{%
617               \let\AskInc@AskResult\AskInc@CharYes
618             }%
619           \else
620             \ifx\AskInc@AskResult\AskInc@CharNone
621               \gdef\AskInc@AskQuestion{%
622                 \let\AskInc@AskResult\AskInc@CharNo
623               }%
624             \else
625               \ifx\AskInc@AskResult\AskInc@CharDefault
626                 \gdef\AskInc@AskQuestion{%
627                   \let\AskInc@AskResult\AskInc@Default
628                 }%
629               \else
630                 \PackageWarningNoLine{askinclude}{%
631                   Invalid answer [\AskInc@AskResult].\MessageBreak
632                   Answer must be one letter out of [ynAND]%
633                 }%
634               \fi
635             \fi
636           \fi
637           \let\AskInc@Next\AskInc@AskQuestion
638         \fi
639       \fi
640     \fi
641   \AskInc@Next
642 }

643 \AtBeginDocument{\AskInc@MainQuestion}
644 \AskInc@AtEnd%
645 \endpackage

```

## 3 Test

### 3.1 Included files

```

646 \< *a \>
647 \chapter{Chapter A}

```

```

648 \let\filea=Y
649 </a>

650 <*b>
651 \chapter{Chapter B}
652 \let\fileb=Y
653 </b>

654 <*c>
655 \chapter{Chapter C}
656 \let\filec=Y
657 </c>

```

## 3.2 Special test setup

```

658 <*test18>
659 \includeonly{askinclude-b}
660 </test18>

661 <*test20 j test21 j test22>
662 \let\AtBeginDocumentMode=N%
663 </test20 j test21 j test22>

```

## 3.3 Common start

```

664 <*test0>
665 \ifx\CompatibilityMode Y%
666 \documentstyle[filecontents,qstest,askinclude,pdftexcmds]{book}%
667 \else
668 \documentclass{book}%
669 \usepackage{filecontents}%
670 \usepackage{qstest}%
671 \usepackage{pdftexcmds}%
672 <*test1>
673 \makeatletter
674 \def\RestoreCatcodes{%
675 \count@=0 %
676 \loop
677 \edef\RestoreCatcodes{%
678 \RestoreCatcodes
679 \catcode\the\count@=\the\catcode\count@\relax
680 }%
681 \ifnum\count@<255 %
682 \advance\count@\@ne
683 \repeat
684 \def\RangeCatcodeInvalid#1#2{%
685 \count@=#1\relax
686 \loop
687 \catcode\count@=15 %
688 \ifnum\count@<#2\relax
689 \advance\count@\@ne
690 \repeat
691 }%
692 \def\Test#1\EndTest{%
693 \RangeCatcodeInvalid{0}{47}%
694 \RangeCatcodeInvalid{58}{64}%
695 \RangeCatcodeInvalid{91}{96}%
696 \RangeCatcodeInvalid{123}{255}%
697 \catcode'\@=12 %
698 \catcode'\=0 %
699 \catcode'\{=1 %
700 \catcode'\}=2 %
701 \catcode'\#=6 %
702 \catcode'\[=12 %
703 \catcode'\]=12 %
704 \catcode'\%=14 %

```

```

705 \catcode'\ =10 %
706 \catcode13=5 %
707 #1\relax
708 \RestoreCatcodes
709 \makeatother
710 }%
711 \Test
712 </test1>
713 <!*test1>
714 \ifx\AtBeginDocumentMode Y%
715 \AtBeginDocument{%
716 \usepackage{askininclude}[2016/05/14]%
717 }%
718 \else
719 </!test1>
720 \usepackage{askininclude}[2016/05/14]%
721 <!*test1>
722 \fi
723 </!test1>
724 <*test1>
725 \EndTest
726 </test1>
727 \fi
728 \IncludeTests{*}
729 \LogTests{log}{*}{*}
730 \makeatletter
731 \let\org@typein\typein
732 \def\typein[#1]#2{%
733 \ifx\@Stack\@StackEmpty
734 \begin{qstest}{StackNonEmpty}{StackNonEmpty}%
735 \Expect{Error: empty stack}{}%
736 \end{qstest}%
737 \org@typein[#1]{#2}%
738 \else
739 \edef#1{%
740 \unexpanded\expandafter\expandafter\expandafter{%
741 \expandafter\@car\@Stack\@nil
742 }%
743 }%
744 \xdef\@Stack{%
745 \unexpanded\expandafter\expandafter\expandafter{%
746 \expandafter\@cdr\@Stack\@nil
747 }%
748 }%
749 \typeout{%
750 \typeout{\string#1=#1}%
751 \fi
752 }
753 \def\@StackEmpty{\BOTTOM}
754 \let\@Stack\@StackEmpty
755 \newcommand*{\Stack}[1]{%
756 \def\@Stack{#1\BOTTOM}%
757 }
758 \AtEndDocument{%
759 \begin{qstest}{StackUsed}{StackUsed}%
760 \Expect*{%
761 \ifx\@Stack\@StackEmpty
762 \else
763 Error: Unused stack entries%
764 \fi
765 }{}%
766 \end{qstest}%

```

```

767 }
768 \newcommand*\CheckInclude}[2]{%
769   \include{askinclude-#1}%
770   \begin{qstest}{\CheckInclude-#1}{\CheckInclude-#1}%
771     \Expect*{%
772       \expandafter\ifx\csname file#1\endcsname Ytrue\else false\fi
773     }{#2}%
774   \end{qstest}%
775 }
776 \def\@EndChecks{
777   \AtBeginDocument{%
778     \AtEndDocument{%
779       \if@files
780         \immediate\write\@mainaux{\string\@EndChecks}%
781       \fi
782     }%
783 }
784 \newcommand*\CheckAnswer}[1]{%
785   \gdef\AnswerText{#1}%
786   \g@addto@macro\@EndChecks{\@CheckAnswer}%
787 }
788 \def\@CheckAnswer{%
789   \begin{qstest}{\CheckAnswer}{\CheckAnswer}%
790     \Expect*\{\unexpanded\expandafter{\AskInc@Answer}}}%
791     *\{\unexpanded\expandafter{\AnswerText}}}%
792   \end{qstest}%
793 }
794 \newcommand*\CheckMatchMethod}[1]{%
795   \gdef\MatchMethodText{#1}%
796   \g@addto@macro\@EndChecks{\@CheckMatchMethod}%
797 }
798 \def\@CheckMatchMethod{%
799   \begin{qstest}{\CheckMatchMethod}{\CheckMatchMethod}%
800     \Expect*\{\unexpanded\expandafter{\AskInc@MatchMethod}}}%
801     *\{\unexpanded\expandafter{\MatchMethodText}}}%
802   \end{qstest}%
803 }
804 \def\@CheckAux{%
805   \begin{qstest}{\CheckAux}{\CheckAux}%
806     \def\GetFileContents##1{%
807       \pdf@unescapehex{%
808         \pdf@filedump{0}{\pdf@filesize{##1}}{##1}%
809       }%
810     }%
811     \Expect*\{\GetFileContents{\jobname.aux}}}%
812     *\{\GetFileContents{\jobname-aux.aux}}}%
813   \end{qstest}%
814 }
815 \g@addto@macro\@EndChecks{\@CheckAux}
816 \let\org@filecontents\filecontents
817 \def\filecontents{%
818   \catcode'\|=0 %
819   \org@filecontents
820 }
821 \begingroup
822   \endlinechar=-1 %
823   \catcode'\|=0 %
824   \@makeother\#%
825   \@makeother\%
826   \gdef\AddList{%
827     \providecommand*\AskInc@AddList[2]{%
828       \ifcase\ifx#1\@undefined

```

```

829         \else\ifx#1\relax
830         \else\ifx#1\@empty
831         \else1\fi\fi\fi0 %
832         \gdef#1{#2}%
833     \else
834         \g@addto@macro#1{,#2}%
835     \fi
836 }%
837 }%
838 |gdef|AddFile{%
839 \providecommand\AskInc@AddFile{%
840 \AskInc@AddList\AskInc@Files
841 }%
842 }%
843 |gdef|AddMatch{%
844 \providecommand\AskInc@AddMatch{%
845 \AskInc@AddList\AskInc@MatchFiles
846 }%
847 }%
848 |endgroup
849 </test0>

```

### 3.4 First run tests

```

850 <*test1>
851 \Stack{[]}
852 \CheckAnswer{*}
853 \begin{filecontents*}{\jobname.aux}
854 \relax
855 \end{filecontents*}
856 \begin{filecontents*}{\jobname-aux.aux}
857 \relax|space
858 \gdef\AskInc@Answer{*}
859 \gdef\AskInc@MatchMethod{}
860 |AddList
861 |AddFile
862 \AskInc@AddFile{askinclude-a}
863 \@input{askinclude-a.aux}
864 \AskInc@AddFile{askinclude-b}
865 \@input{askinclude-b.aux}
866 \AskInc@AddFile{askinclude-c}
867 \@input{askinclude-c.aux}
868 \@EndChecks
869 \end{filecontents*}
870 \begin{document}
871 \CheckInclude{a}{true}
872 \CheckInclude{b}{true}
873 \CheckInclude{c}{true}
874 </test1>

875 <*test2>
876 \Stack{*}
877 \CheckAnswer{*}
878 \begin{filecontents*}{\jobname.aux}
879 \relax
880 \end{filecontents*}
881 \begin{filecontents*}{\jobname-aux.aux}
882 \relax|space
883 \gdef\AskInc@Answer{*}
884 \gdef\AskInc@MatchMethod{}
885 |AddList
886 |AddFile
887 \AskInc@AddFile{askinclude-a}
888 \@input{askinclude-a.aux}

```



```

889 \AskInc@AddFile{askinclude-b}
890 \@input{askinclude-b.aux}
891 \AskInc@AddFile{askinclude-c}
892 \@input{askinclude-c.aux}
893 \@EndChecks
894 \end{filecontents*}
895 \begin{document}
896 \CheckInclude{a}{true}
897 \CheckInclude{b}{true}
898 \CheckInclude{c}{true}
899 </test2>

900 <*test3>
901 \Stack{-}%
902 \CheckAnswer{-}
903 \begin{filecontents*}{\jobname.aux}
904 \relax
905 \end{filecontents*}
906 \begin{filecontents*}{\jobname-aux.aux}
907 \relax|space
908 \gdef\AskInc@Answer{-}
909 \gdef\AskInc@MatchMethod{}
910 |AddList
911 |AddFile
912 \AskInc@AddFile{askinclude-a}
913 \@input{askinclude-a.aux}
914 \AskInc@AddFile{askinclude-b}
915 \@input{askinclude-b.aux}
916 \AskInc@AddFile{askinclude-c}
917 \@input{askinclude-c.aux}
918 \@EndChecks
919 \end{filecontents*}
920 \begin{document}
921 \CheckInclude{a}{false}
922 \CheckInclude{b}{false}
923 \CheckInclude{c}{false}
924 </test3>

925 <*test4>
926 \Stack{{askinclude-b}}
927 \CheckAnswer{askinclude-b}
928 \begin{filecontents*}{\jobname.aux}
929 \relax
930 \end{filecontents*}
931 \begin{filecontents*}{\jobname-aux.aux}
932 \relax|space
933 \gdef\AskInc@Answer{askinclude-b}
934 \gdef\AskInc@MatchMethod{}
935 |AddList
936 |AddFile
937 \AskInc@AddFile{askinclude-a}
938 \@input{askinclude-a.aux}
939 \AskInc@AddFile{askinclude-b}
940 \@input{askinclude-b.aux}
941 \AskInc@AddFile{askinclude-c}
942 \@input{askinclude-c.aux}
943 \@EndChecks
944 \end{filecontents*}
945 \begin{document}
946 \CheckInclude{a}{false}
947 \CheckInclude{b}{true}
948 \CheckInclude{c}{false}
949 </test4>

950 <*test5>

```

```

951 \Stack{{askinclude-a,askinclude-c}}
952 \CheckAnswer{askinclude-a,askinclude-c}
953 \begin{filecontents*}{\jobname.aux}
954 \relax
955 \end{filecontents*}
956 \begin{filecontents*}{\jobname-aux.aux}
957 \relax|space
958 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
959 \gdef\AskInc@MatchMethod{}
960 |AddList
961 |AddFile
962 \AskInc@AddFile{askinclude-a}
963 \@input{askinclude-a.aux}
964 \AskInc@AddFile{askinclude-b}
965 \@input{askinclude-b.aux}
966 \AskInc@AddFile{askinclude-c}
967 \@input{askinclude-c.aux}
968 \@EndChecks
969 \end{filecontents*}
970 \begin{document}
971 \CheckInclude{a}{true}
972 \CheckInclude{b}{false}
973 \CheckInclude{c}{true}
974 </test5>

975 <*test6>
976 \Stack{?yny}
977 \CheckAnswer{askinclude-a,askinclude-c}
978 \begin{filecontents*}{\jobname.aux}
979 \relax
980 \end{filecontents*}
981 \begin{filecontents*}{\jobname-aux.aux}
982 \relax|space
983 \gdef\AskInc@Answer{}
984 \gdef\AskInc@MatchMethod{}
985 |AddList
986 |AddFile
987 \AskInc@AddFile{askinclude-a}
988 \gdef\AskInc@Answer{askinclude-a}
989 \@input{askinclude-a.aux}
990 \AskInc@AddFile{askinclude-b}
991 \@input{askinclude-b.aux}
992 \AskInc@AddFile{askinclude-c}
993 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
994 \@input{askinclude-c.aux}
995 \@EndChecks
996 \end{filecontents*}
997 \begin{document}
998 \CheckInclude{a}{true}
999 \CheckInclude{b}{false}
1000 \CheckInclude{c}{true}
1001 </test6>

1002 <*test7>
1003 \Stack{?yA}
1004 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1005 \begin{filecontents*}{\jobname.aux}
1006 \relax
1007 \end{filecontents*}
1008 \begin{filecontents*}{\jobname-aux.aux}
1009 \relax|space
1010 \gdef\AskInc@Answer{}
1011 \gdef\AskInc@MatchMethod{}
1012 |AddList

```

```

1013 |AddFile
1014 \AskInc@AddFile{askinclude-a}
1015 \gdef\AskInc@Answer{askinclude-a}
1016 \@input{askinclude-a.aux}
1017 \AskInc@AddFile{askinclude-b}
1018 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1019 \@input{askinclude-b.aux}
1020 \AskInc@AddFile{askinclude-c}
1021 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1022 \@input{askinclude-c.aux}
1023 \@EndChecks
1024 \end{filecontents*}
1025 \begin{document}
1026 \CheckInclude{a}{true}
1027 \CheckInclude{b}{true}
1028 \CheckInclude{c}{true}
1029 \</test7>

1030 \<*test8>
1031 \Stack{?yD}
1032 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1033 \begin{filecontents*}{\jobname.aux}
1034 \relax
1035 \end{filecontents*}
1036 \begin{filecontents*}{\jobname-aux.aux}
1037 \relax|space
1038 \gdef\AskInc@Answer{}
1039 \gdef\AskInc@MatchMethod{}
1040 |AddList
1041 |AddFile
1042 \AskInc@AddFile{askinclude-a}
1043 \gdef\AskInc@Answer{askinclude-a}
1044 \@input{askinclude-a.aux}
1045 \AskInc@AddFile{askinclude-b}
1046 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1047 \@input{askinclude-b.aux}
1048 \AskInc@AddFile{askinclude-c}
1049 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1050 \@input{askinclude-c.aux}
1051 \@EndChecks
1052 \end{filecontents*}
1053 \begin{document}
1054 \CheckInclude{a}{true}
1055 \CheckInclude{b}{true}
1056 \CheckInclude{c}{true}
1057 \</test8>

1058 \<*test9>
1059 \Stack{?yN}
1060 \CheckAnswer{askinclude-a}
1061 \begin{filecontents*}{\jobname.aux}
1062 \relax
1063 \end{filecontents*}
1064 \begin{filecontents*}{\jobname-aux.aux}
1065 \relax|space
1066 \gdef\AskInc@Answer{}
1067 \gdef\AskInc@MatchMethod{}
1068 |AddList
1069 |AddFile
1070 \AskInc@AddFile{askinclude-a}
1071 \gdef\AskInc@Answer{askinclude-a}
1072 \@input{askinclude-a.aux}
1073 \AskInc@AddFile{askinclude-b}
1074 \@input{askinclude-b.aux}

```

```

1075 \AskInc@AddFile{askinclude-c}
1076 \@input{askinclude-c.aux}
1077 \@EndChecks
1078 \end{filecontents*}
1079 \begin{document}
1080 \CheckInclude{a}{true}
1081 \CheckInclude{b}{false}
1082 \CheckInclude{c}{false}
1083 </test9>

1084 <*test10>
1085 \Stack{?{N}}
1086 \CheckAnswer{askinclude-a}
1087 \begin{filecontents*}{\jobname.aux}
1088 \relax
1089 \end{filecontents*}
1090 \begin{filecontents*}{\jobname-aux.aux}
1091 \relax|space
1092 \gdef\AskInc@Answer{}
1093 \gdef\AskInc@MatchMethod{}
1094 |AddList
1095 |AddFile
1096 \AskInc@AddFile{askinclude-a}
1097 \gdef\AskInc@Answer{askinclude-a}
1098 \@input{askinclude-a.aux}
1099 \AskInc@AddFile{askinclude-b}
1100 \@input{askinclude-b.aux}
1101 \AskInc@AddFile{askinclude-c}
1102 \@input{askinclude-c.aux}
1103 \@EndChecks
1104 \end{filecontents*}
1105 \begin{document}
1106 \CheckInclude{a}{true}
1107 \CheckInclude{b}{false}
1108 \CheckInclude{c}{false}
1109 </test10>

1110 <*test11>
1111 \Stack{?xxyxxN}
1112 \CheckAnswer{askinclude-a}
1113 \begin{filecontents*}{\jobname.aux}
1114 \relax
1115 \end{filecontents*}
1116 \begin{filecontents*}{\jobname-aux.aux}
1117 \relax|space
1118 \gdef\AskInc@Answer{}
1119 \gdef\AskInc@MatchMethod{}
1120 |AddList
1121 |AddFile
1122 \AskInc@AddFile{askinclude-a}
1123 \gdef\AskInc@Answer{askinclude-a}
1124 \@input{askinclude-a.aux}
1125 \AskInc@AddFile{askinclude-b}
1126 \@input{askinclude-b.aux}
1127 \AskInc@AddFile{askinclude-c}
1128 \@input{askinclude-c.aux}
1129 \@EndChecks
1130 \end{filecontents*}
1131 \begin{document}
1132 \CheckInclude{a}{true}
1133 \CheckInclude{b}{false}
1134 \CheckInclude{c}{false}
1135 </test11>

```

### 3.5 Second run tests

```
1136 <*test12>
1137 \Stack{ {} }
1138 \CheckAnswer{askinclude-b}
1139 \begin{filecontents*}{\jobname.aux}
1140 \gdef\AskInc@Answer{askinclude-b}
1141 \gdef\AskInc@MatchMethod{}
1142 |AddList
1143 |AddFile
1144 \AskInc@AddFile{askinclude-a}
1145 \AskInc@AddFile{askinclude-b}
1146 \AskInc@AddFile{askinclude-c}
1147 \end{filecontents*}
1148 \begin{filecontents*}{\jobname-aux.aux}
1149 \relax|space
1150 \gdef\AskInc@Answer{askinclude-b}
1151 \gdef\AskInc@MatchMethod{}
1152 |AddList
1153 |AddFile
1154 \AskInc@AddFile{askinclude-a}
1155 \@input{askinclude-a.aux}
1156 \AskInc@AddFile{askinclude-b}
1157 \@input{askinclude-b.aux}
1158 \AskInc@AddFile{askinclude-c}
1159 \@input{askinclude-c.aux}
1160 \@EndChecks
1161 \end{filecontents*}
1162 \begin{document}
1163 \CheckInclude{a}{false}
1164 \CheckInclude{b}{true}
1165 \CheckInclude{c}{false}
1166 </test12>

1167 <*test13>
1168 \Stack{ {} }
1169 \CheckAnswer{askinclude-b}
1170 \begin{filecontents*}{\jobname.aux}
1171 \gdef\AskInc@Answer{askinclude-b}
1172 \gdef\AskInc@MatchMethod{}
1173 |AddList
1174 |AddFile
1175 \AskInc@AddFile{askinclude-a}
1176 \AskInc@AddFile{askinclude-b}
1177 \end{filecontents*}
1178 \begin{filecontents*}{\jobname-aux.aux}
1179 \relax|space
1180 \gdef\AskInc@Answer{askinclude-b}
1181 \gdef\AskInc@MatchMethod{}
1182 |AddList
1183 |AddFile
1184 \AskInc@AddFile{askinclude-a}
1185 \@input{askinclude-a.aux}
1186 \AskInc@AddFile{askinclude-b}
1187 \@input{askinclude-b.aux}
1188 \AskInc@AddFile{askinclude-c}
1189 \@input{askinclude-c.aux}
1190 \@EndChecks
1191 \end{filecontents*}
1192 \begin{document}
1193 \CheckInclude{a}{false}
1194 \CheckInclude{b}{true}
1195 \CheckInclude{c}{false}
1196 </test13>
```

```

1197 <*test14>
1198 \Stack{[]}
1199 \CheckAnswer{*}
1200 \begin{filecontents*}{\jobname.aux}
1201 \gdef\AskInc@Answer{*}
1202 \gdef\AskInc@MatchMethod{}
1203 |AddList
1204 |AddFile
1205 \AskInc@AddFile{askinclude-a}
1206 \AskInc@AddFile{askinclude-b}
1207 \end{filecontents*}
1208 \begin{filecontents*}{\jobname-aux.aux}
1209 \relax|space
1210 \gdef\AskInc@Answer{*}
1211 \gdef\AskInc@MatchMethod{}
1212 |AddList
1213 |AddFile
1214 \AskInc@AddFile{askinclude-a}
1215 \@input{askinclude-a.aux}
1216 \AskInc@AddFile{askinclude-b}
1217 \@input{askinclude-b.aux}
1218 \AskInc@AddFile{askinclude-c}
1219 \@input{askinclude-c.aux}
1220 \@EndChecks
1221 \end{filecontents*}
1222 \begin{document}
1223 \CheckInclude{a}{true}
1224 \CheckInclude{b}{true}
1225 \CheckInclude{c}{true}
1226 </test14>

1227 <*test15>
1228 \Stack{?D}
1229 \CheckAnswer{askinclude-a,askinclude-c}
1230 \begin{filecontents*}{\jobname.aux}
1231 \gdef\AskInc@Answer{askinclude-a}
1232 \gdef\AskInc@MatchMethod{}
1233 |AddList
1234 |AddFile
1235 \AskInc@AddFile{askinclude-a}
1236 \AskInc@AddFile{askinclude-b}
1237 \end{filecontents*}
1238 \begin{filecontents*}{\jobname-aux.aux}
1239 \relax|space
1240 \gdef\AskInc@Answer{}
1241 \gdef\AskInc@MatchMethod{}
1242 |AddList
1243 |AddFile
1244 \AskInc@AddFile{askinclude-a}
1245 \gdef\AskInc@Answer{askinclude-a}
1246 \@input{askinclude-a.aux}
1247 \AskInc@AddFile{askinclude-b}
1248 \@input{askinclude-b.aux}
1249 \AskInc@AddFile{askinclude-c}
1250 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1251 \@input{askinclude-c.aux}
1252 \@EndChecks
1253 \end{filecontents*}
1254 \begin{document}
1255 \CheckInclude{a}{true}
1256 \CheckInclude{b}{false}
1257 \CheckInclude{c}{true}
1258 </test15>

```

```

1259 <*test16>
1260 \Stack{?A}
1261 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1262 \begin{filecontents*}{\jobname.aux}
1263 \gdef\AskInc@Answer{askinclude-a}
1264 \gdef\AskInc@MatchMethod{}
1265 |AddList
1266 |AddFile
1267 \AskInc@AddFile{askinclude-a}
1268 \AskInc@AddFile{askinclude-b}
1269 \end{filecontents*}
1270 \begin{filecontents*}{\jobname-aux.aux}
1271 \relax|space
1272 \gdef\AskInc@Answer{}
1273 \gdef\AskInc@MatchMethod{}
1274 |AddList
1275 |AddFile
1276 \AskInc@AddFile{askinclude-a}
1277 \gdef\AskInc@Answer{askinclude-a}
1278 \@input{askinclude-a.aux}
1279 \AskInc@AddFile{askinclude-b}
1280 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1281 \@input{askinclude-b.aux}
1282 \AskInc@AddFile{askinclude-c}
1283 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1284 \@input{askinclude-c.aux}
1285 \@EndChecks
1286 \end{filecontents*}
1287 \begin{document}
1288 \CheckInclude{a}{true}
1289 \CheckInclude{b}{true}
1290 \CheckInclude{c}{true}
1291 </test16>

1292 <*test17>
1293 \Stack{?N}
1294 \CheckAnswer{}
1295 \begin{filecontents*}{\jobname.aux}
1296 \gdef\AskInc@Answer{askinclude-a}
1297 \gdef\AskInc@MatchMethod{}
1298 |AddList
1299 |AddFile
1300 \AskInc@AddFile{askinclude-a}
1301 \AskInc@AddFile{askinclude-b}
1302 \end{filecontents*}
1303 \begin{filecontents*}{\jobname-aux.aux}
1304 \relax|space
1305 \gdef\AskInc@Answer{}
1306 \gdef\AskInc@MatchMethod{}
1307 |AddList
1308 |AddFile
1309 \AskInc@AddFile{askinclude-a}
1310 \@input{askinclude-a.aux}
1311 \AskInc@AddFile{askinclude-b}
1312 \@input{askinclude-b.aux}
1313 \AskInc@AddFile{askinclude-c}
1314 \@input{askinclude-c.aux}
1315 \@EndChecks
1316 \end{filecontents*}
1317 \begin{document}
1318 \CheckInclude{a}{false}
1319 \CheckInclude{b}{false}
1320 \CheckInclude{c}{false}

```

```

1321 </test17>
1322 <*test18>
1323 \Stack{}
1324 \CheckAnswer{askinclude-a}
1325 \begin{filecontents*}{\jobname.aux}
1326 \gdef\AskInc@Answer{askinclude-a}
1327 \gdef\AskInc@MatchMethod{}
1328 |AddList
1329 |AddFile
1330 \AskInc@AddFile{askinclude-a}
1331 \AskInc@AddFile{askinclude-b}
1332 \end{filecontents*}
1333 \begin{filecontents*}{\jobname-aux.aux}
1334 \relax|space
1335 \gdef\AskInc@Answer{askinclude-a}
1336 \gdef\AskInc@MatchMethod{}
1337 |AddList
1338 |AddFile
1339 \AskInc@AddFile{askinclude-a}
1340 \@input{askinclude-a.aux}
1341 \AskInc@AddFile{askinclude-b}
1342 \@input{askinclude-b.aux}
1343 \AskInc@AddFile{askinclude-c}
1344 \@input{askinclude-c.aux}
1345 \@EndChecks
1346 \end{filecontents*}
1347 \begin{document}
1348 \CheckInclude{a}{false}
1349 \CheckInclude{b}{true}
1350 \CheckInclude{c}{false}
1351 </test18>
1352 <*test19>
1353 \Stack{?N}
1354 \CheckAnswer{}
1355 \begin{filecontents*}{\jobname.aux}
1356 \gdef\AskInc@Answer{askinclude-a}
1357 \gdef\AskInc@MatchMethod{}
1358 |AddList
1359 |AddFile
1360 \AskInc@AddFile{askinclude-a}
1361 \AskInc@AddFile{askinclude-b}
1362 \end{filecontents*}
1363 \begin{filecontents*}{\jobname-aux.aux}
1364 \relax|space
1365 \gdef\AskInc@Answer{}
1366 \gdef\AskInc@MatchMethod{}
1367 |AddList
1368 |AddFile
1369 \AskInc@AddFile{askinclude-a}
1370 \@input{askinclude-a.aux}
1371 \AskInc@AddFile{askinclude-b}
1372 \@input{askinclude-b.aux}
1373 \AskInc@AddFile{askinclude-c}
1374 \@input{askinclude-c.aux}
1375 \@EndChecks
1376 \end{filecontents*}
1377 \ifx\AtBeginDocumentMode Y%
1378 \AtBeginDocument{\includeonly{askinclude-b}}}%
1379 \else
1380 \includeonly{askinclude-b}%
1381 \fi
1382 \begin{document}

```



```

1383 \CheckInclude{a}{false}
1384 \CheckInclude{b}{false}
1385 \CheckInclude{c}{false}
1386 </test19>

```

### 3.6 Compatibility for changing from old to new

```

1387 <*test20>
1388 \Stack{[]}
1389 \CheckAnswer{*}
1390 \begin{filecontents*}{\jobname.aux}
1391 \@includeonly{*}
1392 \end{filecontents*}
1393 \begin{filecontents*}{\jobname-aux.aux}
1394 \relax |space
1395 \gdef\AskInc@Answer{*}
1396 \gdef\AskInc@MatchMethod{}
1397 |AddList
1398 |AddFile
1399 \AskInc@AddFile{askinclude-a}
1400 \@input{askinclude-a.aux}
1401 \AskInc@AddFile{askinclude-b}
1402 \@input{askinclude-b.aux}
1403 \AskInc@AddFile{askinclude-c}
1404 \@input{askinclude-c.aux}
1405 \@EndChecks
1406 \end{filecontents*}
1407 \begin{document}
1408 \CheckInclude{a}{true}
1409 \CheckInclude{b}{true}
1410 \CheckInclude{c}{true}
1411 </test20>

1412 <*test21>
1413 \Stack{[]}
1414 \CheckAnswer{-}
1415 \begin{filecontents*}{\jobname.aux}
1416 \@includeonly{}
1417 \end{filecontents*}
1418 \begin{filecontents*}{\jobname-aux.aux}
1419 \relax |space
1420 \gdef\AskInc@Answer{-}
1421 \gdef\AskInc@MatchMethod{}
1422 |AddList
1423 |AddFile
1424 \AskInc@AddFile{askinclude-a}
1425 \@input{askinclude-a.aux}
1426 \AskInc@AddFile{askinclude-b}
1427 \@input{askinclude-b.aux}
1428 \AskInc@AddFile{askinclude-c}
1429 \@input{askinclude-c.aux}
1430 \@EndChecks
1431 \end{filecontents*}
1432 \begin{document}
1433 \CheckInclude{a}{false}
1434 \CheckInclude{b}{false}
1435 \CheckInclude{c}{false}
1436 </test21>

1437 <*test22>
1438 \Stack{[]}
1439 \CheckAnswer{askinclude-a,askinclude-c}
1440 \begin{filecontents*}{\jobname.aux}
1441 \@includeonly{askinclude-a,askinclude-c}
1442 \end{filecontents*}

```

```

1443 \begin{filecontents*}{\jobname-aux.aux}
1444 \relax |space
1445 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1446 \gdef\AskInc@MatchMethod{}
1447 |AddList
1448 |AddFile
1449 \AskInc@AddFile{askinclude-a}
1450 \@input{askinclude-a.aux}
1451 \AskInc@AddFile{askinclude-b}
1452 \@input{askinclude-b.aux}
1453 \AskInc@AddFile{askinclude-c}
1454 \@input{askinclude-c.aux}
1455 \@EndChecks
1456 \end{filecontents*}
1457 \begin{document}
1458 \CheckInclude{a}{true}
1459 \CheckInclude{b}{false}
1460 \CheckInclude{c}{true}
1461 \</test22>

```

### 3.7 Regular expressions

```

1462 \<*test23>
1463 \Stack{{\makematch}{*,!askinclude-b}}
1464 \CheckAnswer{*,!askinclude-b}
1465 \CheckMatchMethod{\makematch}
1466 \begin{filecontents*}{\jobname.aux}
1467 \relax
1468 \end{filecontents*}
1469 \begin{filecontents*}{\jobname-aux.aux}
1470 \relax |space
1471 \gdef\AskInc@Answer{*,!askinclude-b}
1472 \gdef\AskInc@MatchMethod{\makematch}
1473 |AddList
1474 |AddFile
1475 \AskInc@AddFile{askinclude-a}
1476 |AddMatch
1477 \AskInc@AddMatch{askinclude-a}
1478 \@input{askinclude-a.aux}
1479 \AskInc@AddFile{askinclude-b}
1480 \@input{askinclude-b.aux}
1481 \AskInc@AddFile{askinclude-c}
1482 \AskInc@AddMatch{askinclude-c}
1483 \@input{askinclude-c.aux}
1484 \@EndChecks
1485 \end{filecontents*}
1486 \begin{document}
1487 \CheckInclude{a}{true}
1488 \CheckInclude{b}{false}
1489 \CheckInclude{c}{true}
1490 \</test23>

1491 \<*test24>
1492 \Stack{{\pdfmatch}{.,!askinclude-b}}
1493 \CheckAnswer{.,!askinclude-b}
1494 \CheckMatchMethod{\pdfmatch}
1495 \begin{filecontents*}{\jobname.aux}
1496 \relax
1497 \end{filecontents*}
1498 \begin{filecontents*}{\jobname-aux.aux}
1499 \relax |space
1500 \gdef\AskInc@Answer{.,!askinclude-b}
1501 \gdef\AskInc@MatchMethod{\pdfmatch}
1502 |AddList

```

```

1503 |AddFile
1504 \AskInc@AddFile{askinclude-a}
1505 |AddMatch
1506 \AskInc@AddMatch{askinclude-a}
1507 \@input{askinclude-a.aux}
1508 \AskInc@AddFile{askinclude-b}
1509 \@input{askinclude-b.aux}
1510 \AskInc@AddFile{askinclude-c}
1511 \AskInc@AddMatch{askinclude-c}
1512 \@input{askinclude-c.aux}
1513 \@EndChecks
1514 \end{filecontents*}
1515 \begin{document}
1516 \CheckInclude{a}{true}
1517 \CheckInclude{b}{false}
1518 \CheckInclude{c}{true}
1519 \end{test24}

```

### 3.8 Common end

```

1520 \end{test0}
1521 \end{document}
1522 \end{test0}

```

## 4 Installation

### 4.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/askinclude.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/askinclude.pdf](#) Documentation.

**Bundle.** All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

*TDS* refers to the standard “A Directory Structure for T<sub>E</sub>X Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

### 4.2 Bundle installation

**Unpacking.** Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

**Script installation.** Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

### 4.3 Package installation

**Unpacking.** The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain T<sub>E</sub>X:

```
tex askinclude.dtx
```

---

<sup>1</sup><http://ctan.org/pkg/askinclude>

**TDS.** Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

<code>askinclude.sty</code>	→ <code>tex/latex/oberdiek/askinclude.sty</code>
<code>askinclude.pdf</code>	→ <code>doc/latex/oberdiek/askinclude.pdf</code>
<code>test/askinclude-a.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-a.tex</code>
<code>test/askinclude-b.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-b.tex</code>
<code>test/askinclude-c.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-c.tex</code>
<code>test/askinclude-test1.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test1.tex</code>
<code>test/askinclude-test2.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test2.tex</code>
<code>test/askinclude-test3.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test3.tex</code>
<code>test/askinclude-test4.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test4.tex</code>
<code>test/askinclude-test5.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test5.tex</code>
<code>test/askinclude-test6.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test6.tex</code>
<code>test/askinclude-test7.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test7.tex</code>
<code>test/askinclude-test8.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test8.tex</code>
<code>test/askinclude-test9.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test9.tex</code>
<code>test/askinclude-test10.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test10.tex</code>
<code>test/askinclude-test11.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test11.tex</code>
<code>test/askinclude-test12.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test12.tex</code>
<code>test/askinclude-test13.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test13.tex</code>
<code>test/askinclude-test14.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test14.tex</code>
<code>test/askinclude-test15.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test15.tex</code>
<code>test/askinclude-test16.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test16.tex</code>
<code>test/askinclude-test17.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test17.tex</code>
<code>test/askinclude-test18.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test18.tex</code>
<code>test/askinclude-test19.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test19.tex</code>
<code>test/askinclude-test20.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test20.tex</code>
<code>test/askinclude-test21.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test21.tex</code>
<code>test/askinclude-test22.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test22.tex</code>
<code>test/askinclude-test23.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test23.tex</code>
<code>test/askinclude-test24.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test24.tex</code>
<code>askinclude.dtx</code>	→ <code>source/latex/oberdiek/askinclude.dtx</code>

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

## 4.4 Refresh file name databases

If your  $\text{T}_{\text{E}}\text{X}$  distribution ( $\text{t}_{\text{e}}\text{T}_{\text{E}}\text{X}$ ,  $\text{m}_{\text{i}}\text{k}_{\text{T}}\text{E}_{\text{X}}$ , ...) relies on file name databases, you must refresh these. For example,  $\text{t}_{\text{e}}\text{T}_{\text{E}}\text{X}$  users run `texhash` or `mktextlsr`.

## 4.5 Some details for the interested

**Unpacking with  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .** The `.dtx` chooses its action depending on the format:

**plain  $\text{T}_{\text{E}}\text{X}$ :** Run `docstrip` and extract the files.

**$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ :** Generate the documentation.

If you insist on using  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  for `docstrip` (really, `docstrip` does not need  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{askinclude.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

**Generating the documentation.** You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfL<sup>A</sup>T<sub>E</sub>X:

```
pdflatex askinclude.dtx
bibtex askinclude.aux
makeindex -s gind.ist askinclude.idx
pdflatex askinclude.dtx
makeindex -s gind.ist askinclude.idx
pdflatex askinclude.dtx
```

## 5 Catalogue

The following XML file can be used as source for the [T<sub>E</sub>X Catalogue](#). The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `askinclude.xml`.

```
1523 <*catalogue>
1524 <?xml version='1.0' encoding='us-ascii'?>
1525 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
1526 <entry datestamp='$Date$' modifier='$Author$' id='askinclude'>
1527   <name>askinclude</name>
1528   <caption>Interactive use of \includeonly</caption>
1529   <authorref id='auth:straub'/>
1530   <authorref id='auth:oberdiek'/>
1531   <copyright owner="Pablo A. Straub" year='1991-1994'/>
1532   <copyright owner="Heiko Oberdiek" year='2007,2008,2011'/>
1533   <license type='lppl1.3'/>
1534   <version number='2.3'/>
1535   <description>
1536     The package asks the user which files to put in a \includeonly
1537     command. There is provision for answering &#x201C;same as last
1538     time&#x201D; or &#x201C;all files&#x201D;.
1539   <p/>
1540   The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
1541 </description>
1542 <documentation details='Package documentation'
1543   href='ctan:/macros/latex/contrib/oberdiek/askinclude.pdf'/>
1544 <ctan file='true' path='/macros/latex/contrib/oberdiek/askinclude.dtx'/>
1545 <miktex location='oberdiek'/>
1546 <texlive location='oberdiek'/>
1547 <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
1548 </entry>
1549 </catalogue>
```

## 6 Acknowledgement

**Tom Scavo** suggested saving the include file list from the previous run and a simplified user interface.

**Joseph Wright** suggested questions for each `\include`.

## 7 References

- [1] Robin Fairbairns. *[tex-live] licence on includeonly.sty*. 2007-10-13. URL: <http://www.tug.org/pipermail/tex-live/2007-October/014928.html> (visited on 2011-12-02).

- [2] Daniel H. Luecking and Donald Arseneau. *The excludeonly package*. 2003-03-14. URL: <http://mirror.ctan.org/macros/latex/contrib/excludeonly/excludeonly.pdf> (visited on 2011-12-02).
- [3] Herbert Voß. *askinclude.sty*. Thread in newsgroup [comp.text.tex](http://groups.google.com/group/comp.text.tex/browse_thread/thread/728a3e73c04c6789/349e31865c40d00c); Message-ID: fev72m\$b35\$02\$2@news.t-online.com. 2007-10-15. URL: [http://groups.google.com/group/comp.text.tex/browse\\_thread/thread/728a3e73c04c6789/349e31865c40d00c](http://groups.google.com/group/comp.text.tex/browse_thread/thread/728a3e73c04c6789/349e31865c40d00c) (visited on 2011-12-02).

## 8 History

### [1991/11/05 v0.95]

- First beta-test version (asked two questions).

### [1991/11/05 v0.99]

- Second beta-test version (asks one question; uses ‘\*’ and ‘-’).

### [1991/11/07 v1.0]

- Changed prompts (sent to [comp.text.tex](http://groups.google.com/group/comp.text.tex)): “Interactive inclusion of files in LaTeX”<sup>2</sup>

### [1991/12/13 v1.1]

- *Unknown*.

### [1994/11/02 v1.2e]

- Two small changes to run under LaTeX2e [RF].
- Additional documentation [TRS].

### [2007/10/15 v1.0 (askinc)]

- Reimplementation, published as askinc in newsgroup [comp.text.tex](http://groups.google.com/group/comp.text.tex): “Re: askinclude.sty”<sup>3</sup>
- Listing of included files, collected from the previous run before the question.
- No error message by .aux file if package is removed.

### [2007/10/23 v2.0]

- The next version of askinc becomes version 2.0 of askinclude.
- ‘?’ added for questioning each file.
- \includeonly has precedence until package loading. Then it is disabled by a warning.
- Documentation rewritten/added.
- Tests added.

---

<sup>2</sup>Url: <http://groups.google.com/group/comp.text.tex/msg/34b770a7c277e79c>

<sup>3</sup>Url: <http://groups.google.com/group/comp.text.tex/msg/2d638b84a858fdc8>

[2008/08/11 v2.1]

- Code is not changed.
- URLs updated.

[2011/12/02 v2.2]

- Bibliography rewritten for biblatex.

[2016/05/14 v2.3]

- Luatex compatibility

## 9 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

Symbols	
<code>\!</code> .....	358
<code>\#</code> .....	701, 824
<code>\\$</code> .....	223, 360
<code>\%</code> .....	704
<code>\(</code> .....	216, 361
<code>\)</code> .....	217, 362
<code>\*</code> .....	218, 354
<code>\+</code> .....	219, 363
<code>\,</code> .....	357
<code>\-</code> .....	355
<code>\.</code> .....	226, 369
<code>\:</code> .....	368
<code>\?</code> .....	220, 356
<code>\@</code> .....	697
<code>\@CheckAnswer</code> .....	786, 788
<code>\@CheckAux</code> .....	804, 815
<code>\@CheckMatchMethod</code> .....	796, 798
<code>\@EndChecks</code> .....	776,
	780, 786, 796, 815, 868, 893,
	918, 943, 968, 995, 1023, 1051,
	1077, 1103, 1129, 1160, 1190,
	1220, 1252, 1285, 1315, 1345,
	1375, 1405, 1430, 1455, 1484, 1513
<code>\@Stack</code> .....	733, 741, 744, 746, 754, 756, 761
<code>\@StackEmpty</code> .....	733, 753, 754, 761
<code>\@car</code> .....	741
<code>\@cdr</code> .....	746
<code>\@empty</code> .....	77,
	139, 142, 155, 198, 229, 241,
	250, 254, 265, 266, 276, 292,
	308, 320, 377, 390, 393, 404,
	405, 411, 414, 418, 438, 476,
	503, 511, 550, 560, 585, 605, 830
<code>\@firstofone</code> .....	58, 459, 470
<code>\@firstoftwo</code> .....	277, 561
<code>\@for</code> .....	269, 281, 282, 565, 566, 575
<code>\@gobble</code> ..	104, 115, 116, 459, 472, 533
<code>\@ifundefined</code> .....	125, 130,
	153, 234, 240, 244, 253, 546, 549
<code>\@includeonly</code> ..	137, 1391, 1416, 1441
<code>\@input</code> .....	863, 865, 867,
	888, 890, 892, 913, 915, 917,
	938, 940, 942, 963, 965, 967,
	989, 991, 994, 1016, 1019, 1022,
	1044, 1047, 1050, 1072, 1074,
	1076, 1098, 1100, 1102, 1124,
	1126, 1128, 1155, 1157, 1159,
	1185, 1187, 1189, 1215, 1217,
	1219, 1246, 1248, 1251, 1278,
	1281, 1284, 1310, 1312, 1314,
	1340, 1342, 1344, 1370, 1372,
	1374, 1400, 1402, 1404, 1425,
	1427, 1429, 1450, 1452, 1454,
	1478, 1480, 1483, 1507, 1509, 1512
<code>\@mainaux</code> .....	64, 780
<code>\@makeother</code> .....	354, 355, 356, 357, 358,
	359, 360, 361, 362, 363, 364,
	365, 366, 367, 368, 369, 824, 825
<code>\@ne</code> .....	682, 689
<code>\@nil</code> .....	476, 478, 741, 746
<code>\@partlist</code> .....	438, 439, 441, 591
<code>\@secondoftwo</code> .....	279, 563
<code>\@spaces</code> .....	290,
	315, 335, 341, 343, 344, 345, 371
<code>\@undefined</code> ..	54, 73, 450, 461, 520, 828
<code>\[</code> .....	214, 364, 702
<code>\]</code> .....	213, 366, 698, 825
<code>\{</code> .....	224, 699
<code>\}</code> .....	225, 700
<code>\]</code> .....	215, 365, 703
<code>\^</code> .....	222, 359
<code>\ </code> .....	221, 367, 818, 823
<code>\_</code> .....	705
A	
<code>\advance</code> .....	682, 689
<code>\answer</code> ...	370, 373, 595, 598, 603, 604
<code>\AnswerText</code> .....	785, 791
<code>\AskInc@@Answer</code> ...	373, 376, 379,
	382, 393, 394, 403, 409, 410, 413

\AskInc@@CheckMatch	476, <u>478</u>	\AskInc@Default	555, 557, 569, 573, 574, 577, 600, 606, 627
\AskInc@@QuestionHeader	582, 596, 597	\AskInc@DoAskFile	406, <u>552</u>
\AskInc@AddFile	96, <u>103</u> , 109, 839, 862, 864, 866, 887, 889, 891, 912, 914, 916, 937, 939, 941, 962, 964, 966, 987, 990, 992, 1014, 1017, 1020, 1042, 1045, 1048, 1070, 1073, 1075, 1096, 1099, 1101, 1122, 1125, 1127, 1144, 1145, 1146, 1154, 1156, 1158, 1175, 1176, 1184, 1186, 1188, 1205, 1206, 1214, 1216, 1218, 1235, 1236, 1244, 1247, 1249, 1267, 1268, 1276, 1279, 1282, 1300, 1301, 1309, 1311, 1313, 1330, 1331, 1339, 1341, 1343, 1360, 1361, 1369, 1371, 1373, 1399, 1401, 1403, 1424, 1426, 1428, 1449, 1451, 1453, 1475, 1479, 1481, 1504, 1508, 1510	\AskInc@DoInclude	429
\AskInc@AddList	72, 97, 526, 827, 840, 845	\AskInc@DoMatch	436, <u>454</u> , <u>465</u>
\AskInc@AddMatch	525, <u>532</u> , 538, 844, 1477, 1482, 1506, 1511	\AskInc@File	269, 284, 290, 293, 297, 553, 568, 576, 599
\AskInc@Answer	127, 138, 139, 140, 235, 237, 239, 270, 274, 281, 315, 394, 404, 409, 420, 433, 543, 585, 586, 588, 591, 790, 858, 883, 908, 933, 958, 983, 988, 993, 1010, 1015, 1018, 1021, 1038, 1043, 1046, 1049, 1066, 1071, 1092, 1097, 1118, 1123, 1140, 1150, 1171, 1180, 1201, 1210, 1231, 1240, 1245, 1250, 1263, 1272, 1277, 1280, 1283, 1296, 1305, 1326, 1335, 1356, 1365, 1395, 1420, 1445, 1471, 1500	\AskInc@Files	97, <u>112</u> , 266, 269, 575, 840
\AskInc@AskFile	115, 119, 406	\AskInc@FilterPattern	505, <u>507</u>
\AskInc@AskQuestion	583, <u>594</u>	\AskInc@Foundfalse	466, 483
\AskInc@AskResult	584, 603, 605, 606, 609, 612, 615, 617, 620, 622, 625, 627, 631	\AskInc@Foundtrue	490
\AskInc@AtBeginDocument	57, 62	\AskInc@Include	429
\AskInc@AtEnd	29, 30, 51, 162, 644	\AskInc@InvalidPattern	485, 492, <u>496</u>
\AskInc@CharAll	208, 615	\AskInc@MainAnswer	396, <u>402</u>
\AskInc@CharAsterisk	201, 235, 270, 410, 547, 554	\AskInc@MainQuestion	347, 643
\AskInc@CharDefault	210, 625	\AskInc@MakeMatchInit	446, <u>452</u>
\AskInc@CharExclam	205, 480	\AskInc@Mark	271, 273, 285, 290, 291
\AskInc@CharHash	69, 73, 75, 77, 84, 86	\AskInc@MatchFile	116, 120, <u>435</u>
\AskInc@CharMinus	202, 273, 274, 413, 558	\AskInc@MatchFiles	158, 254, 282, 526, 566, 845
\AskInc@CharNo	207, 557, 573, 577, 612, 622	\AskInc@MatchInit	423, <u>431</u>
\AskInc@CharNone	209, 620	\AskInc@MatchList	434, 458, 468, 502, 505, 511, 512, 514
\AskInc@CharPlus	203, 271, 285, 291	\AskInc@MatchMethod	132, 142, 155, 241, 243, 245, 276, 308, 311, 405, 411, 414, 417, 445, 800, 859, 884, 909, 934, 959, 984, 1011, 1039, 1067, 1093, 1119, 1141, 1151, 1172, 1181, 1202, 1211, 1232, 1241, 1264, 1273, 1297, 1306, 1327, 1336, 1357, 1366, 1396, 1421, 1446, 1472, 1501
\AskInc@CharQuestion	204, 403	\AskInc@MatchOption	183, 187, 198, 245, 247, 250, 335, 377, 380, 384, 390, 417, 418
\AskInc@CharYes	206, 555, 569, 574, 584, 609, 617	\AskInc@Next	375, 396, 400, 607, 610, 613, 637, 641
\AskInc@CheckMatch	468, <u>475</u>	\AskInc@OrgInclude	113, 121
		\AskInc@OrgIncludeOnly	164, 407, 415, 419, 424
		\AskInc@Pattern	508, 509, 512, 515
		\AskInc@PdfMatchAvailabletrue	180
		\AskInc@PdfMatchInit	448, <u>463</u>
		\AskInc@PreviousAnswer	239, <u>546</u> , 554, 558, 565
		\AskInc@PreviousMatches	265, 292, 293, 296
		\AskInc@PreviousMatchMethod	243, <u>549</u> , 560
		\AskInc@QuestionHeader	257, 348, 582
		\AskInc@SanitizeAnswer	211, 237, 373
		\AskInc@SaveAnswer	541, 590
		\AskInc@StringMakeMatch	174, 183, 323, 379, 380, 445
		\AskInc@StringNoRegExp	173, 376
		\AskInc@StringPdfMatch	175, 187, 247, 327, 382, 384
		\AskInc@Temp	281, 282, 284, 455, 456, 458, 459, 479, 480, 497, 499, 501, 509, 565, 566, 568, 575, 576, 595, 604
		\AskInc@TypeoutFiles	264, 349



<code>\AskInc@TypeoutMainQuestion</code>	339, 352	<code>\CheckMatchMethod</code>	794, 1465, 1494
<code>\AskInc@TypeoutMatchMethod</code>	318, 351	<code>\comma@list</code>	434
<code>\AskInc@TypeoutMethod</code>	320, 323, 327, 333	<code>\comma@normalize</code>	433
<code>\AskInc@TypeoutPreviousAnswer</code>	305, 350	<code>\comma@parse@normalized</code>	467, 504
<code>\AskInc@WriteAddFile</code>	106, 118	<code>\CompatibilityMode</code>	665
<code>\AskInc@WriteAddMatch</code>	437, 535	<code>\count@</code>	675, 679, 681, 682, 685, 687, 688, 689
<code>\AskInc@WriteAux</code>	61, 71, 95, 108, 126, 131, 157, 524, 537, 542	<code>\csname</code>	10, 178, 772
<code>\AskInc@WriteDefAddFile</code>	93, 107	<b>D</b>	
<code>\AskInc@WriteDefAddList</code>	70, 94, 523	<code>\DeclareOption</code>	149, 150, 151, 182, 186, 190, 197
<code>\AskInc@WriteDefAddMatch</code>	522, 536	<code>\do</code>	269, 283, 567, 575
<code>\AtBeginDocument</code>	57, 60, 103, 124, 233, 532, 643, 715, 777, 1378	<code>\documentclass</code>	668
<code>\AtBeginDocumentMode</code>	662, 714, 1377	<code>\documentstyle</code>	666
<code>\AtEndDocument</code>	758, 778	<b>E</b>	
<code>\AtEndOfPackage</code>	123	<code>\end</code>	736, 766, 774, 792, 802, 813, 855, 869, 880, 894, 905, 919, 930, 944, 955, 969, 980, 996, 1007, 1024, 1035, 1052, 1063, 1078, 1089, 1104, 1115, 1130, 1147, 1161, 1177, 1191, 1207, 1221, 1237, 1253, 1269, 1286, 1302, 1316, 1332, 1346, 1362, 1376, 1392, 1406, 1417, 1431, 1442, 1456, 1468, 1485, 1497, 1514, 1521
<b>B</b>		<code>\endcsname</code>	10, 178, 772
<code>\begin</code>	734, 759, 770, 789, 799, 805, 853, 856, 870, 878, 881, 895, 903, 906, 920, 928, 931, 945, 953, 956, 970, 978, 981, 997, 1005, 1008, 1025, 1033, 1036, 1053, 1061, 1064, 1079, 1087, 1090, 1105, 1113, 1116, 1131, 1139, 1148, 1162, 1170, 1178, 1192, 1200, 1208, 1222, 1230, 1238, 1254, 1262, 1270, 1287, 1295, 1303, 1317, 1325, 1333, 1347, 1355, 1363, 1382, 1390, 1393, 1407, 1415, 1418, 1432, 1440, 1443, 1457, 1466, 1469, 1486, 1495, 1498, 1515	<code>\endinput</code>	51
<code>\BOTTOM</code>	753, 756	<code>\endlinechar</code>	5, 11, 23, 822
<b>C</b>		<code>\EndTest</code>	692, 725
<code>\catcode</code>	3, 4, 6, 7, 8, 12, 13, 14, 15, 16, 17, 18, 21, 22, 24, 25, 26, 27, 31, 33, 679, 687, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 818, 823	<code>\escapechar</code>	213
<code>\chapter</code>	647, 651, 655	<code>\Expect</code>	735, 760, 771, 790, 800, 811
<code>\CheckAnswer</code>	784, 852, 877, 902, 927, 952, 977, 1004, 1032, 1060, 1086, 1112, 1138, 1169, 1199, 1229, 1261, 1294, 1324, 1354, 1389, 1414, 1439, 1464, 1493	<b>F</b>	
<code>\CheckInclude</code>	768, 871, 872, 873, 896, 897, 898, 921, 922, 923, 946, 947, 948, 971, 972, 973, 998, 999, 1000, 1026, 1027, 1028, 1054, 1055, 1056, 1080, 1081, 1082, 1106, 1107, 1108, 1132, 1133, 1134, 1163, 1164, 1165, 1193, 1194, 1195, 1223, 1224, 1225, 1255, 1256, 1257, 1288, 1289, 1290, 1318, 1319, 1320, 1348, 1349, 1350, 1383, 1384, 1385, 1408, 1409, 1410, 1433, 1434, 1435, 1458, 1459, 1460, 1487, 1488, 1489, 1516, 1517, 1518	<code>\filea</code>	648
<b>D</b>		<code>\fileb</code>	652
<code>\CheckMatchMethod</code>	794, 1465, 1494	<code>\filec</code>	656
<code>\comma@list</code>	434	<code>\filec@ntents</code>	816, 817
<code>\comma@normalize</code>	433	<b>G</b>	
<code>\comma@parse@normalized</code>	467, 504	<code>\g@addto@macro</code>	86, 295, 441, 514, 588, 786, 796, 815, 834
<code>\CompatibilityMode</code>	665	<code>\gdef</code>	84, 127, 132, 138, 140, 158, 439, 543, 586, 616, 621, 626, 785, 795, 832, 858, 859, 883, 884, 908, 909, 933, 934, 958, 959, 983, 984, 988, 993, 1010, 1011, 1015, 1018, 1021, 1038, 1039, 1043, 1046, 1049, 1066, 1067, 1071, 1092, 1093, 1097, 1118, 1119, 1123, 1140, 1141, 1150, 1151, 1171, 1172, 1180, 1181, 1201, 1202, 1210, 1211, 1231, 1232, 1240, 1241, 1245, 1250, 1263, 1264, 1272, 1273, 1277, 1280, 1283, 1296, 1297, 1305, 1306, 1326, 1327, 1335, 1336, 1356, 1357, 1365, 1366, 1395, 1396, 1420, 1421, 1445, 1446, 1471, 1472, 1500, 1501
<code>\count@</code>	675, 679, 681, 682, 685, 687, 688, 689	<code>\GetFileContents</code>	806, 811, 812
<code>\csname</code>	10, 178, 772		

<b>I</b>		<b>P</b>	
<code>\if@filesw</code>	63, 779	<code>\PackageWarning</code>	167, 386
<code>\if@partsw</code>	144	<code>\PackageWarningNoLine</code>	145, 191, 498, 630
<code>\ifAskInc@Found</code>	464, 469	<code>\pdf@filedump</code>	808
<code>\ifAskInc@PdfMatchAvailable</code>	176, 185, 248, 326, 383	<code>\pdf@filesize</code>	808
<code>\ifcase</code>	73, 481, 488, 828	<code>\pdf@unescapehex</code>	807
<code>\ifnum</code>	681, 688	<code>\pdfextension</code>	54
<code>\ifx</code>	54, 73, 75, 77, 139, 155, 178, 247, 266, 270, 274, 276, 284, 291, 292, 308, 335, 376, 379, 382, 393, 403, 410, 413, 418, 438, 445, 480, 509, 511, 554, 558, 560, 568, 573, 576, 584, 585, 605, 609, 612, 615, 620, 625, 665, 714, 733, 761, 772, 828, 829, 830, 1377	<code>\pdfmatch</code>	193, 328, 388, 481, 488, 499
<code>\immediate</code>	64, 780	<code>\ProcessOptions</code>	152, 200
<code>\include</code>	114, 117, 268, 769	<code>\providecommand</code>	60, 72, 96, 112, 137, 525, 827, 839, 844
<code>\includeonly</code>	146, 165, 166, 659, 1378, 1380, 1528, 1536	<code>\ProvidesPackage</code>	52
<code>\IncludeTests</code>	728	<b>R</b>	
<b>J</b>		<code>\RangeCatcodeInvalid</code>	684, 693, 694, 695, 696
<code>\jobname</code>	811, 812, 853, 856, 878, 881, 903, 906, 928, 931, 953, 956, 978, 981, 1005, 1008, 1033, 1036, 1061, 1064, 1087, 1090, 1113, 1116, 1139, 1148, 1170, 1178, 1200, 1208, 1230, 1238, 1262, 1270, 1295, 1303, 1325, 1333, 1355, 1363, 1390, 1393, 1415, 1418, 1440, 1443, 1466, 1469, 1495, 1498	<code>\renewcommand</code>	117, 166
<b>L</b>		<code>\repeat</code>	683, 690
<code>\LogTests</code>	729	<code>\RequirePackage</code>	55, 432, 453
<code>\loop</code>	676, 686	<code>\RestoreCatcodes</code>	674, 677, 678, 708
<b>M</b>		<b>S</b>	
<code>\makeatletter</code>	673, 730	<code>\space</code>	146, 168, 229, 321, 328, 335, 341, 371
<code>\makeatother</code>	709	<code>\Stack</code>	755, 851, 876, 901, 926, 951, 976, 1003, 1031, 1059, 1085, 1111, 1137, 1168, 1198, 1228, 1260, 1293, 1323, 1353, 1388, 1413, 1438, 1463, 1492
<code>\MakeMatcher</code>	456	<b>T</b>	
<code>\MatchMethodText</code>	795, 801	<code>\Test</code>	692, 711
<code>\MessageBreak</code>	146, 169, 192, 387, 631	<code>\the</code>	11, 12, 13, 14, 15, 16, 17, 18, 31, 679
<b>N</b>		<code>\TMP@EnsureCode</code>	28, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50
<code>\NeedsTeXFormat</code>	2	<code>\typein</code>	370, 598, 731, 732
<code>\newcommand</code>	113, 164, 755, 768, 784, 794	<code>\typeout</code>	258, 259, 260, 261, 262, 268, 290, 302, 306, 315, 316, 319, 331, 334, 340, 341, 343, 344, 345, 372, 602, 749, 750
<code>\newif</code>	176, 464	<b>U</b>	
<b>O</b>		<code>\unexpanded</code>	740, 745, 790, 791, 800, 801
<code>\org@filec@ntents</code>	816, 819	<code>\usepackage</code>	669, 670, 671, 716, 720
<code>\org@typein</code>	731, 737	<b>W</b>	
<b>P</b>		<code>\write</code>	64, 780
<b>R</b>		<b>X</b>	
<b>S</b>		<code>\x</code>	9, 21
<b>T</b>		<b>Z</b>	
<b>U</b>		<code>\zap@space</code>	228