

The askinclude package

Pablo A. Straub

Heiko Oberdiek*

<heiko.oberdiek at gmail.com>

2018/11/01 v2.4

Abstract

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

Contents

1	Documentation	2
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
2	Implementation	7
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>

3	Test	20
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
4	Installation	35
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
5	Catalogue	37
6	Acknowledgement	37
7	History	37
	[1991/11/05 v0.95]	37
	[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]	38
	[2011/12/02 v2.2]	38
	[2016/05/14 v2.3]	38
	[2018/11/01 v2.4]	38

1 Documentation

1.1 User interface

The `\include` feature of L^AT_EX 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` [**excludeonly**] 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` [`excludeonly`] 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 [`texlive`]. Herbert Voß initiated a search for the author in the newsgroup `comp.text.tex` [`newsthread`] 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 Hefferon 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^AT_EX 2.09: The package works as \documentstyle option in compatibility mode of L^AT_EX 2_ε. It is possible that it will not work in pure L^AT_EX 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 [2018/11/01 v2.4 Interactive asking of included files]%

    Luatex compat
54 \ifx\pdfextension\@undefined\else
55   \protected\def\pdfflastobj {\numexpr\pdfffeedback lastobj\relax}
56   \protected\def\pdfnames   {\pdfextension names }
57   \protected\def\pdfobj     {\pdfextension obj }
58   \let\pdfoutput           \outputmode
59 \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.

```

60 \AtBeginDocument{%
61   \let\AskInc@AtBeginDocument\@firstofone
62 }
63 \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.

```

64 \def\AskInc@WriteAux#1#2{%
65   \AskInc@AtBeginDocument{%
66     \if@filesw
67       \immediate\write\@mainaux{#1}%
68       #2%
69     \fi
70   }%
71 }

```

2.4 Remember \include files

`\AskInc@CharHash`

```

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

```

`\AskInc@WriteDefAddList`

```

73 \def\AskInc@WriteDefAddList{%
74   \AskInc@WriteAux{%

```

`\AskInc@AddList`

```

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

```



```

83         \string\fi
84         \string\fi
85         \string\fi
86         0 %
87         \string\gdef\AskInc@CharHash1{\AskInc@CharHash2}%
88         \string\else
89         \string\g@addto@macro\AskInc@CharHash1{,\AskInc@CharHash2}%
90         \string\fi
91     }%

92 }{%
93     \global\let\AskInc@WriteDefAddList\relax
94 }%
95 }

\AskInc@WriteDefAddFile

96 \def\AskInc@WriteDefAddFile{%
97     \AskInc@WriteDefAddList
98     \AskInc@WriteAux{%

\AskInc@AddFile

99     \string\providecommand\string\AskInc@AddFile{%
100         \string\AskInc@AddList\string\AskInc@Files
101     }%

102 }{%
103     \global\let\AskInc@WriteDefAddFile\relax
104 }%
105 }

\AskInc@AddFile

106 \AtBeginDocument{%
107     \global\let\AskInc@AddFile\@gobble
108 }

\AskInc@WriteAddFile

109 \def\AskInc@WriteAddFile#1{%
110     \AskInc@WriteDefAddFile
111     \AskInc@WriteAux{%
112         \string\AskInc@AddFile{#1}%
113     }{}%
114 }

\AskInc@Files

115 \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

116 \newcommand{\AskInc@OrgInclude}{}
117 \let\AskInc@OrgInclude\include

\AskInc@AskFile

118 \let\AskInc@AskFile\@gobble

\AskInc@MatchFile

119 \let\AskInc@MatchFile\@gobble

```

`\include`

```
120 \renewcommand*{\include}[1]{%
121   \AskInc@WriteAddFile{#1}%
122   \AskInc@AskFile{#1}%
123   \AskInc@MatchFile{#1}%
124   \AskInc@OrgInclude{#1}%
125 }
```

2.6 Store answer and match method in .aux file

```
126 \AtEndOfPackage{%
127   \AtBeginDocument{%
128     \ifundefined{AskInc@Answer}{%
129       \AskInc@WriteAux{%
130         \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
131       }{}%
132     }%
133     \ifundefined{AskInc@MatchMethod}{%
134       \AskInc@WriteAux{%
135         \string\gdef\string\AskInc@MatchMethod{\AskInc@MatchMethod}%
136       }{}%
137     }%
138   }%
139 }
```

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`.

```
140 \providecommand*{\@includeonly}[1]{%
141   \gdef\AskInc@Answer{#1}%
142   \ifx\AskInc@Answer\@empty
143     \gdef\AskInc@Answer{-}%
144   \fi
145   \global\let\AskInc@MatchMethod\@empty
146 }
```

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.

```
147 \if@partsw
148   \PackageWarningNoLine{askinclude}{%
149     Previous \string\includeonly\space detected,\MessageBreak
150     therefore asking for files is suppressed%
151   }%
```

Dummy options.

```
152 \DeclareOption{makematch}{}%
153 \DeclareOption{pdfmatch}{}%
154 \DeclareOption{noregexp}{}%
155 \ProcessOptions*
```

Preserve `\AskInc@MatchFiles`.

```
156 \ifundefined{AskInc@MatchFiles}{%
157 }{%
158   \ifx\AskInc@MatchMethod\@empty
159     \else
160       \AskInc@WriteAux{%
161         \string\gdef\string\AskInc@MatchFiles{\AskInc@MatchFiles}%
162       }{}%
163     \fi
164   }%
```

```

165 \expandafter\AskInc@AtEnd
166 \fi%

```

2.9 \includeonly

\AskInc@OrgIncludeOnly

```

167 \newcommand{\AskInc@OrgIncludeOnly}{}%
168 \let\AskInc@OrgIncludeOnly\includeonly

```

\includeonly

```

169 \renewcommand*{\includeonly}[1]{%
170 \PackageWarning{askinclude}{%
171 \string\includeonly\space is managed by package `askinclude',%
172 \MessageBreak
173 therefore it is ignored%
174 }%
175 }

```

2.10 Options

\AskInc@StringNoRegExp

```

176 \def\AskInc@StringNoRegExp{noregexp}

```

\AskInc@StringMakeMatch

```

177 \def\AskInc@StringMakeMatch{makematch}

```

\AskInc@StringPdfMatch

```

178 \def\AskInc@StringPdfMatch{pdfmatch}

```

\ifAskInc@PdfMatchAvailable

```

179 \newif\ifAskInc@PdfMatchAvailable
180 \begingroup\expandafter\expandafter\expandafter\endgroup
181 \expandafter\ifx\csname pdfmatch\endcsname\relax
182 \else
183 \AskInc@PdfMatchAvailabletrue
184 \fi

185 \DeclareOption{makematch}{%
186 \let\AskInc@MatchOption\AskInc@StringMakeMatch
187 }
188 \ifAskInc@PdfMatchAvailable
189 \DeclareOption{pdfmatch}{%
190 \let\AskInc@MatchOption\AskInc@StringPdfMatch
191 }%
192 \else
193 \DeclareOption{pdfmatch}{%
194 \PackageWarningNoLine{askinclude}{%
195 Option `pdfmatch' is not available because\MessageBreak
196 your pdfTeX does not provide \string\pdfmatch
197 }%
198 }%
199 \fi
200 \DeclareOption{noregexp}{%
201 \let\AskInc@MatchOption\@empty
202 }
203 \ProcessOptions*

```

2.11 Strings

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

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

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

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

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

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

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

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

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

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

2.12 Set defaults

```
\AskInc@SanitizeAnswer
214 \def\AskInc@SanitizeAnswer#1#2{%
215   \begingroup
216   \escapechar=`\%
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   \edef\{{\string\}%
228   \edef\}{\string\}%
229   \edef\.{\string\}%
230   \xdef#1{%
231     \expandafter\expandafter\expandafter\zap@space
232     \expandafter#2\space\@empty
233   }%
234 \endgroup
235 }
```

```

236 \AtBeginDocument{%
237   \@ifundefined{AskInc@Answer}{%
238     \global\let\AskInc@Answer\AskInc@CharAsterisk
239   }{%
240     \AskInc@SanitizeAnswer\AskInc@Answer\AskInc@Answer
241   }%
242   \global\let\AskInc@PreviousAnswer\AskInc@Answer
243   \@ifundefined{AskInc@MatchMethod}{%
244     \global\let\AskInc@MatchMethod\@empty
245   }{%
246     \global\let\AskInc@PreviousMatchMethod\AskInc@MatchMethod
247     \@ifundefined{AskInc@MatchOption}{%
248       \let\AskInc@MatchOption\AskInc@MatchMethod
249     }{%
250       \ifx\AskInc@MatchOption\AskInc@StringPdfMatch
251         \ifAskInc@PdfMatchAvailable
252         \else
253           \let\AskInc@MatchOption\@empty
254         \fi
255       \fi
256       \@ifundefined{AskInc@MatchFiles}{%
257         \let\AskInc@MatchFiles\@empty
258       }{%
259     }

```

2.13 Questions

2.13.1 Generic question header

\AskInc@QuestionHeader

```

260 \def\AskInc@QuestionHeader{%
261   \typeout{}%
262   \typeout{*****}%
263   \typeout{*** Package askinclude Question ***}%
264   \typeout{*****}%
265   \typeout{}%
266 }

```

2.13.2 Display files

\AskInc@TypeoutFiles

```

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

```

```

288         \let\AskInc@Mark\AskInc@CharPlus
289     \fi
290 }%
291 \fi
292 \fi
293 \typeout{ \@spaces(\AskInc@Mark) \AskInc@File}%
294 \ifx\AskInc@Mark\AskInc@CharPlus
295     \ifx\AskInc@PreviousMatches\@empty
296         \global\let\AskInc@PreviousMatches\AskInc@File
297     \else
298         \expandafter\g@addto@macro
299         \expandafter\AskInc@PreviousMatches{%
300             \expandafter,\AskInc@File
301         }%
302     \fi
303 \fi
304 }%
305 \typeout{}%
306 \fi
307 }

```

2.13.3 Previous answer

\AskInc@TypeoutPreviousAnswer

```

308 \def\AskInc@TypeoutPreviousAnswer{%
309     \typeout{%
310         Previous answer (%
311         \ifx\AskInc@MatchMethod\@empty
312             noregexp%
313         \else
314             \AskInc@MatchMethod
315         \fi
316         ):%
317     }%
318     \typeout{ \@spaces[\AskInc@Answer]]}%
319     \typeout{}%
320 }

```

2.13.4 Current regular expression support

\AskInc@TypeoutMatchMethod

```

321 \def\AskInc@TypeoutMatchMethod{%
322     \typeout{Regular expressions:}%
323     \AskInc@TypeoutMethod\@empty{noregexp}{%
324         \space disabled%
325     }%
326     \AskInc@TypeoutMethod\AskInc@StringMakeMatch{makematch}{%
327         enabled, using package `makematch'%
328     }%
329     \ifAskInc@PdfMatchAvailable
330         \AskInc@TypeoutMethod\AskInc@StringPdfMatch{pdfmatch}{%
331             \space enabled, using pdfTeX's \string\pdfmatch
332         }%
333     \fi
334     \typeout{}%
335 }

```

\AskInc@TypeoutMethod

```

336 \def\AskInc@TypeoutMethod#1#2#3{%
337     \typeout{%
338         \ifx#1\AskInc@MatchOption \space--> \else\@spaces\space\fi
339         [#2] #3% hash-ok

```

```

340 }%
341 }

```

2.13.5 Main question text

\AskInc@TypeoutMainQuestion

```

342 \def\AskInc@TypeoutMainQuestion{%
343 \typeout{Which files do you want to include?}%
344 \typeout{ \@spaces[foo,bar] \space\space
345   comma separated file or pattern list}%
346 \typeout{ \@spaces[*] \@spaces\@spaces all files}%
347 \typeout{ \@spaces[-] \@spaces\@spaces no files}%
348 \typeout{ \@spaces[?] \@spaces\@spaces ask for each file}%
349 }

```

2.13.6 Main question

\AskInc@MainQuestion

```

350 \def\AskInc@MainQuestion{%
351 \AskInc@QuestionHeader
352 \AskInc@TypeoutFiles
353 \AskInc@TypeoutPreviousAnswer
354 \AskInc@TypeoutMatchMethod
355 \AskInc@TypeoutMainQuestion
356 \begingroup
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 \@makeother\|%
371 \@makeother\:%
372 \@makeother\.%
373 \typein[\answer]%
374 { \@spaces[] \space\@spaces\@spaces use the previous answer}%
375 \typeout{}%
376 \AskInc@SanitizeAnswer\AskInc@@Answer\answer
377 \endgroup
378 \let\AskInc@Next\AskInc@MainQuestion
379 \ifx\AskInc@@Answer\AskInc@StringNoRegExp
380 \let\AskInc@MatchOption\@empty
381 \else
382 \ifx\AskInc@@Answer\AskInc@StringMakeMatch
383 \let\AskInc@MatchOption\AskInc@StringMakeMatch
384 \else
385 \ifx\AskInc@@Answer\AskInc@StringPdfMatch
386 \ifAskInc@PdfMatchAvailable
387 \let\AskInc@MatchOption\AskInc@StringPdfMatch
388 \else
389 \PackageWarning{askinclude}{%
390   Method `pdfmatch' is not available because\MessageBreak
391   your pdfTeX does not provide \string\pdfmatch
392 }%
393 \let\AskInc@MatchOption\@empty

```

```

394     \fi
395   \else
396     \ifx\AskInc@@Answer\@empty
397       \global\let\AskInc@@Answer\AskInc@Answer
398     \fi
399     \let\AskInc@Next\AskInc@MainAnswer
400   \fi
401 \fi
402 \fi
403 \AskInc@Next
404 }

```

2.14 Evaluate answer of main question

\AskInc@MainAnswer

```

405 \def\AskInc@MainAnswer{%
406   \ifx\AskInc@@Answer\AskInc@CharQuestion
407     \global\let\AskInc@Answer\@empty
408     \global\let\AskInc@MatchMethod\@empty
409     \global\let\AskInc@AskFile\AskInc@DoAskFile
410     \AskInc@OrgIncludeOnly{}%
411   \else
412     \global\let\AskInc@Answer\AskInc@@Answer
413     \ifx\AskInc@@Answer\AskInc@CharAsterisk
414       \global\let\AskInc@MatchMethod\@empty
415     \else
416       \ifx\AskInc@@Answer\AskInc@CharMinus
417         \global\let\AskInc@MatchMethod\@empty
418         \AskInc@OrgIncludeOnly{}%
419       \else
420         \global\let\AskInc@MatchMethod\AskInc@MatchOption
421         \ifx\AskInc@MatchOption\@empty
422           \expandafter\AskInc@OrgIncludeOnly\expandafter{%
423             \AskInc@Answer
424           }%
425         \else
426           \AskInc@MatchInit
427           \AskInc@OrgIncludeOnly{}%
428         \fi
429       \fi
430     \fi
431   \fi
432   \let\AskInc@Include\AskInc@DoInclude
433 }

```

\AskInc@MatchInit

```

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

```

\AskInc@MatchFile

```

438 \def\AskInc@MatchFile##1{%
439   \AskInc@DoMatch{##1}{%
440     \AskInc@WriteAddMatch{##1}%
441     \ifx\@partlist\@empty
442       \gdef\@partlist{##1}%
443     \else
444       \g@addto@macro\@partlist{,##1}%
445     \fi
446   }%
447 }%

```



```

448 \ifx\AskInc@MatchMethod\AskInc@StringMakeMatch
449 \AskInc@MakeMatchInit
450 \else
451 \AskInc@PdfMatchInit
452 \fi
453 \let\AskInc@MatchInit\@undefined
454 }

\AskInc@MakeMatchInit
455 \def\AskInc@MakeMatchInit{%
456 \RequirePackage{makematch}%

\AskInc@DoMatch
457 \def\AskInc@DoMatch##1{%
458 \def\AskInc@Temp{%
459 \MakeMatcher[,]\AskInc@Temp
460 }%
461 \expandafter\AskInc@Temp\expandafter{\AskInc@MatchList}%
462 \AskInc@Temp{##1}\@firstofone\@gobble
463 }%

464 \let\AskInc@MakeMatchInit\@undefined
465 }

\AskInc@PdfMatchInit
466 \def\AskInc@PdfMatchInit{%

\ifAskInc@Found
467 \newif\ifAskInc@Found

\AskInc@DoMatch
468 \def\AskInc@DoMatch##1{%
469 \AskInc@Foundfalse
470 \expandafter\comma@parse@normalized\expandafter
471 {\AskInc@MatchList}\AskInc@CheckMatch{##1}%
472 \ifAskInc@Found
473 \expandafter\@firstofone
474 \else
475 \expandafter\@gobble
476 \fi
477 }%

\AskInc@CheckMatch
478 \def\AskInc@CheckMatch##1##2{%
479 \AskInc@@CheckMatch##2\@empty\@nil{##1}%
480 }%

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

```

```

495     \AskInc@InvalidPattern{##1##2}{##1##2}%
496     \fi
497     \fi
498 }%

```

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

```

499 \def\AskInc@InvalidPattern##1##2{%
500   \edef\AskInc@Temp{##1}%
501   \PackageWarningNoLine{askinclude}{%
502     Invalid pattern `\'AskInc@Temp' for \'string\'pdfmatch
503   }%
504   \edef\AskInc@Temp{##2}%
505   \global\expandafter\let\expandafter\AskInc@MatchList
506   \expandafter\@empty
507   \expandafter\comma@parse@normalized\expandafter
508   {\AskInc@MatchList}\AskInc@FilterPattern
509 }%

```

`\AskInc@FilterPattern`

```

510 \def\AskInc@FilterPattern##1{%
511   \edef\AskInc@Pattern{##1}%
512   \ifx\AskInc@Temp\AskInc@Pattern
513     \else
514     \ifx\AskInc@MatchList\@empty
515       \global\let\AskInc@MatchList\AskInc@Pattern
516     \else
517       \expandafter\g@addto@macro\expandafter\AskInc@MatchList{%
518         \expandafter,\AskInc@Pattern
519       }%
520     \fi
521   \fi
522 }%

523 \let\AskInc@PdfMatchInit\@undefined
524 }

```

`\AskInc@WriteDefAddMatch`

```

525 \def\AskInc@WriteDefAddMatch{%
526   \AskInc@WriteDefAddList
527   \AskInc@WriteAux{%

```

`\AskInc@AddMatch`

```

528     \string\providecommand\string\AskInc@AddMatch{%
529       \string\AskInc@AddList\string\AskInc@MatchFiles
530     }%

531   }{%
532     \global\let\AskInc@WriteDefAddMatch\relax
533   }%
534 }

```

`\AskInc@AddMatch`

```

535 \AtBeginDocument{%
536   \global\let\AskInc@AddMatch\@gobble
537 }

```

`\AskInc@WriteAddMatch`

```

538 \def\AskInc@WriteAddMatch#1{%
539   \AskInc@WriteDefAddMatch
540   \AskInc@WriteAux{%
541     \string\AskInc@AddMatch{#1}%
542   }{%
543 }

```

\AskInc@SaveAnswer

```
544 \def\AskInc@SaveAnswer{%
545   \AskInc@WriteAux{%
546     \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
547   }{}%
548 }
```

2.15 Ask for each file

\AskInc@PreviousAnswer

```
549 \@ifundefined{AskInc@PreviousAnswer}{%
550   \let\AskInc@PreviousAnswer\AskInc@CharAsterisk
551 }{}
```

\AskInc@PreviousMatchMethod

```
552 \@ifundefined{AskInc@PreviousMatchMethod}{%
553   \let\AskInc@PreviousMatchMethod\@empty
554 }{}
```

\AskInc@DoAskFile

```
555 \def\AskInc@DoAskFile#1{%
556   \edef\AskInc@File{#1}%
557   \ifx\AskInc@PreviousAnswer\AskInc@CharAsterisk
558     \let\AskInc@Default\AskInc@CharYes
559   \else
560     \let\AskInc@Default\AskInc@CharNo
561     \ifx\AskInc@PreviousAnswer\AskInc@CharMinus
562     \else
563       \ifx\AskInc@PreviousMatchMethod\@empty
564         \expandafter\@firstoftwo
565       \else
566         \expandafter\@secondoftwo
567       \fi
568       {\@for\AskInc@Temp:=\AskInc@PreviousAnswer}%
569       {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
570     \do{%
571       \ifx\AskInc@Temp\AskInc@File
572         \let\AskInc@Default\AskInc@CharYes
573       \fi
574     }%
575   \fi
576   \ifx\AskInc@Default\AskInc@CharNo
577     \let\AskInc@Default\AskInc@CharYes
578     \@for\AskInc@Temp:=\AskInc@Files\do{%
579       \ifx\AskInc@Temp\AskInc@File
580         \let\AskInc@Default\AskInc@CharNo
581       \fi
582     }%
583   \fi
584 \fi
585 \let\AskInc@@@QuestionHeader\AskInc@QuestionHeader
586 \AskInc@AskQuestion
587 \ifx\AskInc@AskResult\AskInc@CharYes
588   \ifx\AskInc@Answer\@empty
589     \gdef\AskInc@Answer{#1}%
590   \else
591     \g@addto@macro\AskInc@Answer{, #1}%
592   \fi
593   \AskInc@SaveAnswer
594   \let\@partlist\AskInc@Answer
595 \fi
596 }
```

\AskInc@AskQuestion

```

597 \def\AskInc@AskQuestion{%
598   \let\AskInc@Temp\answer
599   \AskInc@@@QuestionHeader
600   \let\AskInc@@@QuestionHeader\relax
601   \typein[\answer]{%
602     Include ``\AskInc@File'? [y]es, [n]o, [A]ll, [N]one, [D]efault %
603     (\AskInc@Default):%
604   }%
605   \typeout{%}
606   \let\AskInc@AskResult\answer
607   \let\answer\AskInc@Temp
608   \ifx\AskInc@AskResult\@empty
609     \let\AskInc@AskResult\AskInc@Default
610     \let\AskInc@Next\relax
611   \else
612     \ifx\AskInc@AskResult\AskInc@CharYes
613       \let\AskInc@Next\relax
614     \else
615       \ifx\AskInc@AskResult\AskInc@CharNo
616         \let\AskInc@Next\relax
617       \else
618         \ifx\AskInc@AskResult\AskInc@CharAll
619           \gdef\AskInc@AskQuestion{%
620             \let\AskInc@AskResult\AskInc@CharYes
621           }%
622         \else
623           \ifx\AskInc@AskResult\AskInc@CharNone
624             \gdef\AskInc@AskQuestion{%
625               \let\AskInc@AskResult\AskInc@CharNo
626             }%
627           \else
628             \ifx\AskInc@AskResult\AskInc@CharDefault
629               \gdef\AskInc@AskQuestion{%
630                 \let\AskInc@AskResult\AskInc@Default
631               }%
632             \else
633               \PackageWarningNoLine{askinclude}{%
634                 Invalid answer [\AskInc@AskResult].\MessageBreak
635                 Answer must be one letter out of [ynAND]%
636               }%
637             \fi
638           \fi
639         \fi
640       \let\AskInc@Next\AskInc@AskQuestion
641     \fi
642   \fi
643 \fi
644 \AskInc@Next
645 }

646 \AtBeginDocument{\AskInc@MainQuestion}
647 \AskInc@AtEnd%
648 \</package>

```

3 Test

3.1 Included files

```

649 <*a>
650 \chapter{Chapter A}

```

```

651 \let\filea=Y
652 </a>

653 <*b>
654 \chapter{Chapter B}
655 \let\fileb=Y
656 </b>

657 <*c>
658 \chapter{Chapter C}
659 \let\filec=Y
660 </c>

```

3.2 Special test setup

```

661 <*test18>
662 \includeonly{askinclude-b}
663 </test18>

664 <*test20 j test21 j test22>
665 \let\AtBeginDocumentMode=N%
666 </test20 j test21 j test22>

```

3.3 Common start

```

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

```

```

708 \catcode\ =10 %
709 \catcode13=5 %
710 #1\relax
711 \RestoreCatcodes
712 \makeatother
713 }%
714 \Test
715 </test1>
716 <!*test1>
717 \ifx\AtBeginDocumentMode Y%
718 \AtBeginDocument{%
719 \usepackage{askinclude}[2018/11/01]%
720 }%
721 \else
722 </!test1>
723 \usepackage{askinclude}[2018/11/01]%
724 <!*test1>
725 \fi
726 </!test1>
727 <*test1>
728 \EndTest
729 </test1>
730 \fi
731 \IncludeTests{*}
732 \LogTests{log}{*}{*}
733 \makeatletter
734 \let\org@typein\typein
735 \def\typein[#1]#2{%
736 \ifx\@Stack\@StackEmpty
737 \begin{qstest}{StackNonEmpty}{StackNonEmpty}%
738 \Expect{Error: empty stack}{}%
739 \end{qstest}%
740 \org@typein[#1]{#2}%
741 \else
742 \edef#1{%
743 \unexpanded\expandafter\expandafter\expandafter{%
744 \expandafter\@car\@Stack\@nil
745 }%
746 }%
747 \xdef\@Stack{%
748 \unexpanded\expandafter\expandafter\expandafter{%
749 \expandafter\@cdr\@Stack\@nil
750 }%
751 }%
752 \typeout{%
753 \typeout{\string#1=#1}%
754 \fi
755 }
756 \def\@StackEmpty{\BOTTOM}
757 \let\@Stack\@StackEmpty
758 \newcommand*{\Stack}[1]{%
759 \def\@Stack{#1\BOTTOM}%
760 }
761 \AtEndDocument{%
762 \begin{qstest}{StackUsed}{StackUsed}%
763 \Expect*{%
764 \ifx\@Stack\@StackEmpty
765 \else
766 Error: Unused stack entries%
767 \fi
768 }{}%
769 \end{qstest}%

```

```

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

```

```

832         \else\ifx#1\relax
833         \else\ifx#1\@empty
834         \else1\fi\fi\fi0 %
835         \gdef#1{#2}%
836     \else
837         \g@addto@macro#1{,#2}%
838     \fi
839 }%
840 }%
841 |gdef|AddFile{%
842     \providecommand\AskInc@AddFile{%
843         \AskInc@AddList\AskInc@Files
844     }%
845 }%
846 |gdef|AddMatch{%
847     \providecommand\AskInc@AddMatch{%
848         \AskInc@AddList\AskInc@MatchFiles
849     }%
850 }%
851 |endgroup
852 </test0>

```

3.4 First run tests

```

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

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

```



```

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

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

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

953 <*test5>

```

```

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

978 <*test6>
979 \Stack{?yny}
980 \CheckAnswer{askinclude-a,askinclude-c}
981 \begin{filecontents*}{\jobname.aux}
982 \relax
983 \end{filecontents*}
984 \begin{filecontents*}{\jobname-aux.aux}
985 \relax|space
986 \gdef\AskInc@Answer{}
987 \gdef\AskInc@MatchMethod{}
988 |AddList
989 |AddFile
990 \AskInc@AddFile{askinclude-a}
991 \gdef\AskInc@Answer{askinclude-a}
992 \@input{askinclude-a.aux}
993 \AskInc@AddFile{askinclude-b}
994 \@input{askinclude-b.aux}
995 \AskInc@AddFile{askinclude-c}
996 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
997 \@input{askinclude-c.aux}
998 \@EndChecks
999 \end{filecontents*}
1000 \begin{document}
1001 \CheckInclude{a}{true}
1002 \CheckInclude{b}{false}
1003 \CheckInclude{c}{true}
1004 </test6>

1005 <*test7>
1006 \Stack{?yA}
1007 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1008 \begin{filecontents*}{\jobname.aux}
1009 \relax
1010 \end{filecontents*}
1011 \begin{filecontents*}{\jobname-aux.aux}
1012 \relax|space
1013 \gdef\AskInc@Answer{}
1014 \gdef\AskInc@MatchMethod{}
1015 |AddList

```

```

1016 |AddFile
1017 \AskInc@AddFile{askinclude-a}
1018 \gdef\AskInc@Answer{askinclude-a}
1019 \@input{askinclude-a.aux}
1020 \AskInc@AddFile{askinclude-b}
1021 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1022 \@input{askinclude-b.aux}
1023 \AskInc@AddFile{askinclude-c}
1024 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1025 \@input{askinclude-c.aux}
1026 \@EndChecks
1027 \end{filecontents*}
1028 \begin{document}
1029 \CheckInclude{a}{true}
1030 \CheckInclude{b}{true}
1031 \CheckInclude{c}{true}
1032 </test7>

1033 <*test8>
1034 \Stack{?yD}
1035 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1036 \begin{filecontents*}{\jobname.aux}
1037 \relax
1038 \end{filecontents*}
1039 \begin{filecontents*}{\jobname-aux.aux}
1040 \relax|space
1041 \gdef\AskInc@Answer{}
1042 \gdef\AskInc@MatchMethod{}
1043 |AddList
1044 |AddFile
1045 \AskInc@AddFile{askinclude-a}
1046 \gdef\AskInc@Answer{askinclude-a}
1047 \@input{askinclude-a.aux}
1048 \AskInc@AddFile{askinclude-b}
1049 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1050 \@input{askinclude-b.aux}
1051 \AskInc@AddFile{askinclude-c}
1052 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1053 \@input{askinclude-c.aux}
1054 \@EndChecks
1055 \end{filecontents*}
1056 \begin{document}
1057 \CheckInclude{a}{true}
1058 \CheckInclude{b}{true}
1059 \CheckInclude{c}{true}
1060 </test8>

1061 <*test9>
1062 \Stack{?yN}
1063 \CheckAnswer{askinclude-a}
1064 \begin{filecontents*}{\jobname.aux}
1065 \relax
1066 \end{filecontents*}
1067 \begin{filecontents*}{\jobname-aux.aux}
1068 \relax|space
1069 \gdef\AskInc@Answer{}
1070 \gdef\AskInc@MatchMethod{}
1071 |AddList
1072 |AddFile
1073 \AskInc@AddFile{askinclude-a}
1074 \gdef\AskInc@Answer{askinclude-a}
1075 \@input{askinclude-a.aux}
1076 \AskInc@AddFile{askinclude-b}
1077 \@input{askinclude-b.aux}

```

```

1078 \AskInc@AddFile{askinclude-c}
1079 \@input{askinclude-c.aux}
1080 \@EndChecks
1081 \end{filecontents*}
1082 \begin{document}
1083 \CheckInclude{a}{true}
1084 \CheckInclude{b}{false}
1085 \CheckInclude{c}{false}
1086 \</test9>

1087 \<*test10>
1088 \Stack{?{N}}
1089 \CheckAnswer{askinclude-a}
1090 \begin{filecontents*}{\jobname.aux}
1091 \relax
1092 \end{filecontents*}
1093 \begin{filecontents*}{\jobname-aux.aux}
1094 \relax|space
1095 \gdef\AskInc@Answer{}
1096 \gdef\AskInc@MatchMethod{}
1097 |AddList
1098 |AddFile
1099 \AskInc@AddFile{askinclude-a}
1100 \gdef\AskInc@Answer{askinclude-a}
1101 \@input{askinclude-a.aux}
1102 \AskInc@AddFile{askinclude-b}
1103 \@input{askinclude-b.aux}
1104 \AskInc@AddFile{askinclude-c}
1105 \@input{askinclude-c.aux}
1106 \@EndChecks
1107 \end{filecontents*}
1108 \begin{document}
1109 \CheckInclude{a}{true}
1110 \CheckInclude{b}{false}
1111 \CheckInclude{c}{false}
1112 \</test10>

1113 \<*test11>
1114 \Stack{?xxyxxN}
1115 \CheckAnswer{askinclude-a}
1116 \begin{filecontents*}{\jobname.aux}
1117 \relax
1118 \end{filecontents*}
1119 \begin{filecontents*}{\jobname-aux.aux}
1120 \relax|space
1121 \gdef\AskInc@Answer{}
1122 \gdef\AskInc@MatchMethod{}
1123 |AddList
1124 |AddFile
1125 \AskInc@AddFile{askinclude-a}
1126 \gdef\AskInc@Answer{askinclude-a}
1127 \@input{askinclude-a.aux}
1128 \AskInc@AddFile{askinclude-b}
1129 \@input{askinclude-b.aux}
1130 \AskInc@AddFile{askinclude-c}
1131 \@input{askinclude-c.aux}
1132 \@EndChecks
1133 \end{filecontents*}
1134 \begin{document}
1135 \CheckInclude{a}{true}
1136 \CheckInclude{b}{false}
1137 \CheckInclude{c}{false}
1138 \</test11>

```

3.5 Second run tests

```
1139 <*test12>
1140 \Stack{{}}
1141 \CheckAnswer{askinclude-b}
1142 \begin{filecontents*}{\jobname.aux}
1143 \gdef\AskInc@Answer{askinclude-b}
1144 \gdef\AskInc@MatchMethod{}
1145 |AddList
1146 |AddFile
1147 \AskInc@AddFile{askinclude-a}
1148 \AskInc@AddFile{askinclude-b}
1149 \AskInc@AddFile{askinclude-c}
1150 \end{filecontents*}
1151 \begin{filecontents*}{\jobname-aux.aux}
1152 \relax|space
1153 \gdef\AskInc@Answer{askinclude-b}
1154 \gdef\AskInc@MatchMethod{}
1155 |AddList
1156 |AddFile
1157 \AskInc@AddFile{askinclude-a}
1158 \@input{askinclude-a.aux}
1159 \AskInc@AddFile{askinclude-b}
1160 \@input{askinclude-b.aux}
1161 \AskInc@AddFile{askinclude-c}
1162 \@input{askinclude-c.aux}
1163 \@EndChecks
1164 \end{filecontents*}
1165 \begin{document}
1166 \CheckInclude{a}{false}
1167 \CheckInclude{b}{true}
1168 \CheckInclude{c}{false}
1169 </test12>

1170 <*test13>
1171 \Stack{{}}
1172 \CheckAnswer{askinclude-b}
1173 \begin{filecontents*}{\jobname.aux}
1174 \gdef\AskInc@Answer{askinclude-b}
1175 \gdef\AskInc@MatchMethod{}
1176 |AddList
1177 |AddFile
1178 \AskInc@AddFile{askinclude-a}
1179 \AskInc@AddFile{askinclude-b}
1180 \end{filecontents*}
1181 \begin{filecontents*}{\jobname-aux.aux}
1182 \relax|space
1183 \gdef\AskInc@Answer{askinclude-b}
1184 \gdef\AskInc@MatchMethod{}
1185 |AddList
1186 |AddFile
1187 \AskInc@AddFile{askinclude-a}
1188 \@input{askinclude-a.aux}
1189 \AskInc@AddFile{askinclude-b}
1190 \@input{askinclude-b.aux}
1191 \AskInc@AddFile{askinclude-c}
1192 \@input{askinclude-c.aux}
1193 \@EndChecks
1194 \end{filecontents*}
1195 \begin{document}
1196 \CheckInclude{a}{false}
1197 \CheckInclude{b}{true}
1198 \CheckInclude{c}{false}
1199 </test13>
```

```

1200 <*test14>
1201 \Stack{[]}
1202 \CheckAnswer{*}
1203 \begin{filecontents*}{\jobname.aux}
1204 \gdef\AskInc@Answer{*}
1205 \gdef\AskInc@MatchMethod{}
1206 |AddList
1207 |AddFile
1208 \AskInc@AddFile{askinclude-a}
1209 \AskInc@AddFile{askinclude-b}
1210 \end{filecontents*}
1211 \begin{filecontents*}{\jobname-aux.aux}
1212 \relax|space
1213 \gdef\AskInc@Answer{*}
1214 \gdef\AskInc@MatchMethod{}
1215 |AddList
1216 |AddFile
1217 \AskInc@AddFile{askinclude-a}
1218 \@input{askinclude-a.aux}
1219 \AskInc@AddFile{askinclude-b}
1220 \@input{askinclude-b.aux}
1221 \AskInc@AddFile{askinclude-c}
1222 \@input{askinclude-c.aux}
1223 \@EndChecks
1224 \end{filecontents*}
1225 \begin{document}
1226 \CheckInclude{a}{true}
1227 \CheckInclude{b}{true}
1228 \CheckInclude{c}{true}
1229 </test14>

1230 <*test15>
1231 \Stack{?D}
1232 \CheckAnswer{askinclude-a,askinclude-c}
1233 \begin{filecontents*}{\jobname.aux}
1234 \gdef\AskInc@Answer{askinclude-a}
1235 \gdef\AskInc@MatchMethod{}
1236 |AddList
1237 |AddFile
1238 \AskInc@AddFile{askinclude-a}
1239 \AskInc@AddFile{askinclude-b}
1240 \end{filecontents*}
1241 \begin{filecontents*}{\jobname-aux.aux}
1242 \relax|space
1243 \gdef\AskInc@Answer{}
1244 \gdef\AskInc@MatchMethod{}
1245 |AddList
1246 |AddFile
1247 \AskInc@AddFile{askinclude-a}
1248 \gdef\AskInc@Answer{askinclude-a}
1249 \@input{askinclude-a.aux}
1250 \AskInc@AddFile{askinclude-b}
1251 \@input{askinclude-b.aux}
1252 \AskInc@AddFile{askinclude-c}
1253 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1254 \@input{askinclude-c.aux}
1255 \@EndChecks
1256 \end{filecontents*}
1257 \begin{document}
1258 \CheckInclude{a}{true}
1259 \CheckInclude{b}{false}
1260 \CheckInclude{c}{true}
1261 </test15>

```

```

1262 <*test16>
1263 \Stack{?A}
1264 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1265 \begin{filecontents*}{\jobname.aux}
1266 \gdef\AskInc@Answer{askinclude-a}
1267 \gdef\AskInc@MatchMethod{}
1268 |AddList
1269 |AddFile
1270 \AskInc@AddFile{askinclude-a}
1271 \AskInc@AddFile{askinclude-b}
1272 \end{filecontents*}
1273 \begin{filecontents*}{\jobname-aux.aux}
1274 \relax|space
1275 \gdef\AskInc@Answer{}
1276 \gdef\AskInc@MatchMethod{}
1277 |AddList
1278 |AddFile
1279 \AskInc@AddFile{askinclude-a}
1280 \gdef\AskInc@Answer{askinclude-a}
1281 \@input{askinclude-a.aux}
1282 \AskInc@AddFile{askinclude-b}
1283 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1284 \@input{askinclude-b.aux}
1285 \AskInc@AddFile{askinclude-c}
1286 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1287 \@input{askinclude-c.aux}
1288 \@EndChecks
1289 \end{filecontents*}
1290 \begin{document}
1291 \CheckInclude{a}{true}
1292 \CheckInclude{b}{true}
1293 \CheckInclude{c}{true}
1294 </test16>

1295 <*test17>
1296 \Stack{?N}
1297 \CheckAnswer{}
1298 \begin{filecontents*}{\jobname.aux}
1299 \gdef\AskInc@Answer{askinclude-a}
1300 \gdef\AskInc@MatchMethod{}
1301 |AddList
1302 |AddFile
1303 \AskInc@AddFile{askinclude-a}
1304 \AskInc@AddFile{askinclude-b}
1305 \end{filecontents*}
1306 \begin{filecontents*}{\jobname-aux.aux}
1307 \relax|space
1308 \gdef\AskInc@Answer{}
1309 \gdef\AskInc@MatchMethod{}
1310 |AddList
1311 |AddFile
1312 \AskInc@AddFile{askinclude-a}
1313 \@input{askinclude-a.aux}
1314 \AskInc@AddFile{askinclude-b}
1315 \@input{askinclude-b.aux}
1316 \AskInc@AddFile{askinclude-c}
1317 \@input{askinclude-c.aux}
1318 \@EndChecks
1319 \end{filecontents*}
1320 \begin{document}
1321 \CheckInclude{a}{false}
1322 \CheckInclude{b}{false}
1323 \CheckInclude{c}{false}

```

```

1324 </test17>
1325 <*test18>
1326 \Stack{}
1327 \CheckAnswer{askinclude-a}
1328 \begin{filecontents*}{\jobname.aux}
1329 \gdef\AskInc@Answer{askinclude-a}
1330 \gdef\AskInc@MatchMethod{}
1331 |AddList
1332 |AddFile
1333 \AskInc@AddFile{askinclude-a}
1334 \AskInc@AddFile{askinclude-b}
1335 \end{filecontents*}
1336 \begin{filecontents*}{\jobname-aux.aux}
1337 \relax|space
1338 \gdef\AskInc@Answer{askinclude-a}
1339 \gdef\AskInc@MatchMethod{}
1340 |AddList
1341 |AddFile
1342 \AskInc@AddFile{askinclude-a}
1343 \@input{askinclude-a.aux}
1344 \AskInc@AddFile{askinclude-b}
1345 \@input{askinclude-b.aux}
1346 \AskInc@AddFile{askinclude-c}
1347 \@input{askinclude-c.aux}
1348 \@EndChecks
1349 \end{filecontents*}
1350 \begin{document}
1351 \CheckInclude{a}{false}
1352 \CheckInclude{b}{true}
1353 \CheckInclude{c}{false}
1354 </test18>
1355 <*test19>
1356 \Stack{?N}
1357 \CheckAnswer{}
1358 \begin{filecontents*}{\jobname.aux}
1359 \gdef\AskInc@Answer{askinclude-a}
1360 \gdef\AskInc@MatchMethod{}
1361 |AddList
1362 |AddFile
1363 \AskInc@AddFile{askinclude-a}
1364 \AskInc@AddFile{askinclude-b}
1365 \end{filecontents*}
1366 \begin{filecontents*}{\jobname-aux.aux}
1367 \relax|space
1368 \gdef\AskInc@Answer{}
1369 \gdef\AskInc@MatchMethod{}
1370 |AddList
1371 |AddFile
1372 \AskInc@AddFile{askinclude-a}
1373 \@input{askinclude-a.aux}
1374 \AskInc@AddFile{askinclude-b}
1375 \@input{askinclude-b.aux}
1376 \AskInc@AddFile{askinclude-c}
1377 \@input{askinclude-c.aux}
1378 \@EndChecks
1379 \end{filecontents*}
1380 \ifx\AtBeginDocumentMode Y%
1381 \AtBeginDocument{\includeonly{askinclude-b}}}%
1382 \else
1383 \includeonly{askinclude-b}%
1384 \fi
1385 \begin{document}

```



```

1386 \CheckInclude{a}{false}
1387 \CheckInclude{b}{false}
1388 \CheckInclude{c}{false}
1389 </test19>

```

3.6 Compatibility for changing from old to new

```

1390 <*test20>
1391 \Stack{[]}
1392 \CheckAnswer{*}
1393 \begin{filecontents*}{\jobname.aux}
1394 \@includeonly{*}
1395 \end{filecontents*}
1396 \begin{filecontents*}{\jobname-aux.aux}
1397 \relax |space
1398 \gdef\AskInc@Answer{*}
1399 \gdef\AskInc@MatchMethod{}
1400 |AddList
1401 |AddFile
1402 \AskInc@AddFile{askinclude-a}
1403 \@input{askinclude-a.aux}
1404 \AskInc@AddFile{askinclude-b}
1405 \@input{askinclude-b.aux}
1406 \AskInc@AddFile{askinclude-c}
1407 \@input{askinclude-c.aux}
1408 \@EndChecks
1409 \end{filecontents*}
1410 \begin{document}
1411 \CheckInclude{a}{true}
1412 \CheckInclude{b}{true}
1413 \CheckInclude{c}{true}
1414 </test20>

1415 <*test21>
1416 \Stack{[]}
1417 \CheckAnswer{-}
1418 \begin{filecontents*}{\jobname.aux}
1419 \@includeonly{}
1420 \end{filecontents*}
1421 \begin{filecontents*}{\jobname-aux.aux}
1422 \relax |space
1423 \gdef\AskInc@Answer{-}
1424 \gdef\AskInc@MatchMethod{}
1425 |AddList
1426 |AddFile
1427 \AskInc@AddFile{askinclude-a}
1428 \@input{askinclude-a.aux}
1429 \AskInc@AddFile{askinclude-b}
1430 \@input{askinclude-b.aux}
1431 \AskInc@AddFile{askinclude-c}
1432 \@input{askinclude-c.aux}
1433 \@EndChecks
1434 \end{filecontents*}
1435 \begin{document}
1436 \CheckInclude{a}{false}
1437 \CheckInclude{b}{false}
1438 \CheckInclude{c}{false}
1439 </test21>

1440 <*test22>
1441 \Stack{[]}
1442 \CheckAnswer{askinclude-a,askinclude-c}
1443 \begin{filecontents*}{\jobname.aux}
1444 \@includeonly{askinclude-a,askinclude-c}
1445 \end{filecontents*}

```

```

1446 \begin{filecontents*}{\jobname-aux.aux}
1447 \relax |space
1448 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1449 \gdef\AskInc@MatchMethod{}
1450 |AddList
1451 |AddFile
1452 \AskInc@AddFile{askinclude-a}
1453 \@input{askinclude-a.aux}
1454 \AskInc@AddFile{askinclude-b}
1455 \@input{askinclude-b.aux}
1456 \AskInc@AddFile{askinclude-c}
1457 \@input{askinclude-c.aux}
1458 \@EndChecks
1459 \end{filecontents*}
1460 \begin{document}
1461 \CheckInclude{a}{true}
1462 \CheckInclude{b}{false}
1463 \CheckInclude{c}{true}
1464 \</test22>

```

3.7 Regular expressions

```

1465 \<*test23>
1466 \Stack{{makematch}{*,!askinclude-b}}
1467 \CheckAnswer{*,!askinclude-b}
1468 \CheckMatchMethod{makematch}
1469 \begin{filecontents*}{\jobname.aux}
1470 \relax
1471 \end{filecontents*}
1472 \begin{filecontents*}{\jobname-aux.aux}
1473 \relax |space
1474 \gdef\AskInc@Answer{*,!askinclude-b}
1475 \gdef\AskInc@MatchMethod{makematch}
1476 |AddList
1477 |AddFile
1478 \AskInc@AddFile{askinclude-a}
1479 |AddMatch
1480 \AskInc@AddMatch{askinclude-a}
1481 \@input{askinclude-a.aux}
1482 \AskInc@AddFile{askinclude-b}
1483 \@input{askinclude-b.aux}
1484 \AskInc@AddFile{askinclude-c}
1485 \AskInc@AddMatch{askinclude-c}
1486 \@input{askinclude-c.aux}
1487 \@EndChecks
1488 \end{filecontents*}
1489 \begin{document}
1490 \CheckInclude{a}{true}
1491 \CheckInclude{b}{false}
1492 \CheckInclude{c}{true}
1493 \</test23>

1494 \<*test24>
1495 \Stack{{pdfmatch}{.,!askinclude-b}}
1496 \CheckAnswer{.,!askinclude-b}
1497 \CheckMatchMethod{pdfmatch}
1498 \begin{filecontents*}{\jobname.aux}
1499 \relax
1500 \end{filecontents*}
1501 \begin{filecontents*}{\jobname-aux.aux}
1502 \relax |space
1503 \gdef\AskInc@Answer{.,!askinclude-b}
1504 \gdef\AskInc@MatchMethod{pdfmatch}
1505 |AddList

```

```

1506 |AddFile
1507 \AskInc@AddFile{askinclude-a}
1508 |AddMatch
1509 \AskInc@AddMatch{askinclude-a}
1510 \@input{askinclude-a.aux}
1511 \AskInc@AddFile{askinclude-b}
1512 \@input{askinclude-b.aux}
1513 \AskInc@AddFile{askinclude-c}
1514 \AskInc@AddMatch{askinclude-c}
1515 \@input{askinclude-c.aux}
1516 \@EndChecks
1517 \end{filecontents*}
1518 \begin{document}
1519 \CheckInclude{a}{true}
1520 \CheckInclude{b}{false}
1521 \CheckInclude{c}{true}
1522 \</test24>

```

3.8 Common end

```

1523 \<*test0>
1524 \end{document}
1525 \</test0>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[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_EX 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_EX:

```
tex askinclude.dtx
```

¹<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 pdf \LaTeX :

```
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 \$\TeX\$ 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`.

```
1526 <*catalogue>
1527 <?xml version='1.0' encoding='us-ascii'?>
1528 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
1529 <entry datestamp='$Date$' modifier='$Author$' id='askinclude'>
1530   <name>askinclude</name>
1531   <caption>Interactive use of \includeonly</caption>
1532   <authorref id='auth:straub'/>
1533   <authorref id='auth:oberdiek'/>
1534   <copyright owner="Pablo A. Straub" year='1991-1994'/>
1535   <copyright owner="Heiko Oberdiek" year='2007,2008,2011'/>
1536   <license type='lppl1.3'/>
1537   <version number='2.4'/>
1538   <description>
1539     The package asks the user which files to put in a \includeonly
1540     command. There is provision for answering &#x201C;same as last
1541     time&#x201D; or &#x201C;all files&#x201D;.
1542   </p>
1543   The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
1544 </description>
1545 <documentation details='Package documentation'
1546   href='ctan:/macros/latex/contrib/oberdiek/askinclude.pdf'/>
1547 <ctan file='true' path='/macros/latex/contrib/oberdiek/askinclude.dtx'/>
1548 <miktex location='oberdiek'/>
1549 <texlive location='oberdiek'/>
1550 <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip'/>
1551 </entry>
1552 </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 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`): “Interactive inclusion of files in LaTeX”²

[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`: “Re: askinclude.sty”³
- 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.

[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

[2018/11/01 v2.4]

- Remove luatex85 package dependency

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

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

8 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	
\!	361
\#	704, 827
\\$	226, 363
\%	707
\(219, 364
\)	220, 365
*	221, 357
\+	222, 366
\,	360
\-	358
\.	229, 372
\:	371
\?	223, 359
\@	700
\@CheckAnswer	789, 791
\@CheckAux	807, 818
\@CheckMatchMethod	799, 801
\@EndChecks	779,
	783, 789, 799, 818, 871, 896,
	921, 946, 971, 998, 1026, 1054,
	1080, 1106, 1132, 1163, 1193,
	1223, 1255, 1288, 1318, 1348,
	1378, 1408, 1433, 1458, 1487, 1516
\@Stack	736, 744, 747, 749, 757, 759, 764
\@StackEmpty	736, 756, 757, 764
\@car	744
\@cdr	749
\@empty	80,
	142, 145, 158, 201, 232, 244,
	253, 257, 268, 269, 279, 295,
	311, 323, 380, 393, 396, 407,
	408, 414, 417, 421, 441, 479,
	506, 514, 553, 563, 588, 608, 833
\@firstofone	61, 462, 473
\@firstoftwo	280, 564
\@for	272, 284, 285, 568, 569, 578
\@gobble	107, 118, 119, 462, 475, 536
\@ifundefined	128, 133,
	156, 237, 243, 247, 256, 549, 552
\@includeonly	140, 1394, 1419, 1444
\@input	866, 868, 870,
	891, 893, 895, 916, 918, 920,
	941, 943, 945, 966, 968, 970,
	992, 994, 997, 1019, 1022, 1025,
	1047, 1050, 1053, 1075, 1077,
	1079, 1101, 1103, 1105, 1127,
	1129, 1131, 1158, 1160, 1162,
	1188, 1190, 1192, 1218, 1220,
	1222, 1249, 1251, 1254, 1281,
	1284, 1287, 1313, 1315, 1317,
	1343, 1345, 1347, 1373, 1375,
	1377, 1403, 1405, 1407, 1428,
	1430, 1432, 1453, 1455, 1457,
	1481, 1483, 1486, 1510, 1512, 1515
\@mainaux	67, 783
\@makeother	357, 358, 359, 360, 361,
	362, 363, 364, 365, 366, 367,
	368, 369, 370, 371, 372, 827, 828
\@ne	685, 692
\@nil	479, 481, 744, 749
\@partlist	441, 442, 444, 594
\@secondoftwo	282, 566
\@spaces	293,
	318, 338, 344, 346, 347, 348, 374
\@undefined	54, 76, 453, 464, 523, 831
\[217, 367, 705
\]	216, 369, 701, 828
\{	227, 702
\}	228, 703
\]	218, 368, 706
\^	225, 362
\	224, 370, 821, 826
\	708
A	
\advance	685, 692
\answer	373, 376, 598, 601, 606, 607
\AnswerText	788, 794
\AskInc@@Answer	376, 379, 382,
	385, 396, 397, 406, 412, 413, 416
\AskInc@@CheckMatch	479, <u>481</u>
\AskInc@@QuestionHeader	585, 599, 600
\AskInc@AddFile	99, <u>106</u> , 112, 842,
	865, 867, 869, 890, 892, 894,
	915, 917, 919, 940, 942, 944,
	965, 967, 969, 990, 993, 995,
	1017, 1020, 1023, 1045, 1048,
	1051, 1073, 1076, 1078, 1099,
	1102, 1104, 1125, 1128, 1130,
	1147, 1148, 1149, 1157, 1159,
	1161, 1178, 1179, 1187, 1189,
	1191, 1208, 1209, 1217, 1219,
	1221, 1238, 1239, 1247, 1250,
	1252, 1270, 1271, 1279, 1282,
	1285, 1303, 1304, 1312, 1314,
	1316, 1333, 1334, 1342, 1344,
	1346, 1363, 1364, 1372, 1374,
	1376, 1402, 1404, 1406, 1427,
	1429, 1431, 1452, 1454, 1456,
	1478, 1482, 1484, 1507, 1511, 1513
\AskInc@AddList	75, 100, 529, 830, 843, 848
\AskInc@AddMatch	528, <u>535</u> ,
	541, 847, 1480, 1485, 1509, 1514
\AskInc@Answer	130, 141, 142, 143, 238,
	240, 242, 273, 277, 284, 318,
	397, 407, 412, 423, 436, 546,
	588, 589, 591, 594, 793, 861,

886, 911, 936, 961, 986, 991, 996, 1013, 1018, 1021, 1024, 1041, 1046, 1049, 1052, 1069, 1074, 1095, 1100, 1121, 1126, 1143, 1153, 1174, 1183, 1204, 1213, 1234, 1243, 1248, 1253, 1266, 1275, 1280, 1283, 1286, 1299, 1308, 1329, 1338, 1359, 1368, 1398, 1423, 1448, 1474, 1503	1300, 1309, 1330, 1339, 1360, 1369, 1399, 1424, 1449, 1475, 1504
\AskInc@AskFile 118, 122, 409	\AskInc@MatchOption 186, 190, 201, 248, 250, 253, 338, 380, 383, 387, 393, 420, 421
\AskInc@AskQuestion 586, 597	\AskInc@Next 378, 399, 403, 610, 613, 616, 640, 644
\AskInc@AskResult 587, 606, 608, 609, 612, 615, 618, 620, 623, 625, 628, 630, 634	\AskInc@OrgInclude 116, 124
\AskInc@AtBeginDocument 60, 65	\AskInc@OrgIncludeOnly 167, 410, 418, 422, 427
\AskInc@AtEnd 29, 30, 51, 165, 647	\AskInc@Pattern 511, 512, 515, 518
\AskInc@CharAll 211, 618	\AskInc@PdfMatchAvailabletrue . . 183
\AskInc@CharAsterisk 204, 238, 273, 413, 550, 557	\AskInc@PdfMatchInit 451, 466
\AskInc@CharDefault 213, 628	\AskInc@PreviousAnswer 242, 549, 557, 561, 568
\AskInc@CharExclam 208, 483	\AskInc@PreviousMatches 268, 295, 296, 299
\AskInc@CharHash 72, 76, 78, 80, 87, 89	\AskInc@PreviousMatchMethod 246, 552, 563
\AskInc@CharMinus 205, 276, 277, 416, 561	\AskInc@QuestionHeader . . 260, 351, 585
\AskInc@CharNo 210, 560, 576, 580, 615, 625	\AskInc@SanitizeAnswer . . 214, 240, 376
\AskInc@CharNone 212, 623	\AskInc@SaveAnswer 544, 593
\AskInc@CharPlus 206, 274, 288, 294	\AskInc@StringMakeMatch 177, 186, 326, 382, 383, 448
\AskInc@CharQuestion 207, 406	\AskInc@StringNoRegExp 176, 379
\AskInc@CharYes 209, 558, 572, 577, 587, 612, 620	\AskInc@StringPdfMatch 178, 190, 250, 330, 385, 387
\AskInc@CheckMatch 471, 478	\AskInc@Temp 284, 285, 287, 458, 459, 461, 462, 482, 483, 500, 502, 504, 512, 568, 569, 571, 578, 579, 598, 607
\AskInc@Default 558, 560, 572, 576, 577, 580, 603, 609, 630	\AskInc@TimeoutFiles 267, 352
\AskInc@DoAskFile 409, 555	\AskInc@TimeoutMainQuestion 342, 355
\AskInc@DoInclude 432	\AskInc@TimeoutMatchMethod 321, 354
\AskInc@DoMatch 439, 457, 468	\AskInc@TimeoutMethod 323, 326, 330, 336
\AskInc@File 272, 287, 293, 296, 300, 556, 571, 579, 602	\AskInc@TimeoutPreviousAnswer 308, 353
\AskInc@Files 100, 115, 269, 272, 578, 843	\AskInc@WriteAddFile 109, 121
\AskInc@FilterPattern 508, 510	\AskInc@WriteAddMatch 440, 538
\AskInc@Foundfalse 469, 486	\AskInc@WriteAux 64, 74, 98, 111, 129, 134, 160, 527, 540, 545
\AskInc@Foundtrue 493	\AskInc@WriteDefAddFile 96, 110
\AskInc@Include 432	\AskInc@WriteDefAddList . . 73, 97, 526
\AskInc@InvalidPattern . . 488, 495, 499	\AskInc@WriteDefAddMatch . . 525, 539
\AskInc@MainAnswer 399, 405	\AtBeginDocument 60, 63, 106, 127, 236, 535, 646, 718, 780, 1381
\AskInc@MainQuestion 350, 646	\AtBeginDocumentMode 665, 717, 1380
\AskInc@MakeMatchInit 449, 455	\AtEndDocument 761, 781
\AskInc@Mark 274, 276, 288, 293, 294	\AtEndOfPackage 126
\AskInc@MatchFile 119, 123, 438	
\AskInc@MatchFiles 161, 257, 285, 529, 569, 848	
\AskInc@MatchInit 426, 434	
\AskInc@MatchList 437, 461, 471, 505, 508, 514, 515, 517	
\AskInc@MatchMethod 135, 145, 158, 244, 246, 248, 279, 311, 314, 408, 414, 417, 420, 448, 803, 862, 887, 912, 937, 962, 987, 1014, 1042, 1070, 1096, 1122, 1144, 1154, 1175, 1184, 1205, 1214, 1235, 1244, 1267, 1276,	

B

\begin 737, 762, 773, 792, 802, 808, 856, 859, 873, 881, 884, 898, 906, 909, 923, 931, 934, 948, 956, 959, 973, 981, 984, 1000, 1008, 1011, 1028, 1036, 1039, 1056, 1064, 1067, 1082, 1090, 1093, 1108, 1116, 1119, 1134, 1142, 1151, 1165, 1173, 1181, 1195, 1203, 1211, 1225, 1233, 1241, 1257, 1265, 1273, 1290, 1298, 1306, 1320,

1328, 1336, 1350, 1358, 1366, 1385, 1393, 1396, 1410, 1418, 1421, 1435, 1443, 1446, 1460, 1469, 1472, 1489, 1498, 1501, 1518	\BOTTOM 756, 759	\escapechar 216 \Expect ... 738, 763, 774, 793, 803, 814
C		F
\catcode 3, 4, 6, 7, 8, 12, 13, 14, 15, 16, 17, 18, 21, 22, 24, 25, 26, 27, 31, 33, 682, 690, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 821, 826	\chapter 650, 654, 658	\filea 651 \fileb 655 \filec 659 \filec@ntents 819, 820
\CheckAnswer 787, 855, 880, 905, 930, 955, 980, 1007, 1035, 1063, 1089, 1115, 1141, 1172, 1202, 1232, 1264, 1297, 1327, 1357, 1392, 1417, 1442, 1467, 1496	\CheckInclude 771, 874, 875, 876, 899, 900, 901, 924, 925, 926, 949, 950, 951, 974, 975, 976, 1001, 1002, 1003, 1029, 1030, 1031, 1057, 1058, 1059, 1083, 1084, 1085, 1109, 1110, 1111, 1135, 1136, 1137, 1166, 1167, 1168, 1196, 1197, 1198, 1226, 1227, 1228, 1258, 1259, 1260, 1291, 1292, 1293, 1321, 1322, 1323, 1351, 1352, 1353, 1386, 1387, 1388, 1411, 1412, 1413, 1436, 1437, 1438, 1461, 1462, 1463, 1490, 1491, 1492, 1519, 1520, 1521	G \g@addto@macro 89, 298, 444, 517, 591, 789, 799, 818, 837 \gdef 87, 130, 135, 141, 143, 161, 442, 546, 589, 619, 624, 629, 788, 798, 835, 861, 862, 886, 887, 911, 912, 936, 937, 961, 962, 986, 987, 991, 996, 1013, 1014, 1018, 1021, 1024, 1041, 1042, 1046, 1049, 1052, 1069, 1070, 1074, 1095, 1096, 1100, 1121, 1122, 1126, 1143, 1144, 1153, 1154, 1174, 1175, 1183, 1184, 1204, 1205, 1213, 1214, 1234, 1235, 1243, 1244, 1248, 1253, 1266, 1267, 1275, 1276, 1280, 1283, 1286, 1299, 1300, 1308, 1309, 1329, 1330, 1338, 1339, 1359, 1360, 1368, 1369, 1398, 1399, 1423, 1424, 1448, 1449, 1474, 1475, 1503, 1504 \GetFileContents 809, 814, 815
\CheckMatchMethod .. 797, 1468, 1497		I
\comma@list 437		\if@filesw 66, 782
\comma@normalize 436		\if@partsw 147
\comma@parse@normalized ... 470, 507		\ifAskInc@Found 467, 472
\CompatibilityMode 668		\ifAskInc@PdfMatchAvailable 179, 188, 251, 329, 386
\count@ 678, 682, 684, 685, 688, 690, 691, 692		\ifcase 76, 484, 491, 831
\csname 10, 181, 775		\ifnum 684, 691
		\ifx 54, 76, 78, 80, 142, 158, 181, 250, 269, 273, 277, 279, 287, 294, 295, 311, 338, 379, 382, 385, 396, 406, 413, 416, 421, 441, 448, 483, 512, 514, 557, 561, 563, 571, 576, 579, 587, 588, 608, 612, 615, 618, 623, 628, 668, 717, 736, 764, 775, 831, 832, 833, 1380
D		J
\DeclareOption 152, 153, 154, 185, 189, 193, 200		\jobname 814, 815, 856, 859, 881, 884, 906, 909, 931, 934, 956, 959, 981, 984, 1008, 1011, 1036, 1039, 1064, 1067, 1090, 1093, 1116, 1119, 1142, 1151, 1173, 1181, 1203, 1211, 1233, 1241, 1265, 1273,
\do 272, 286, 570, 578		
\documentclass 671		
\documentstyle 669		
E		
\end 739, 769, 777, 795, 805, 816, 858, 872, 883, 897, 908, 922, 933, 947, 958, 972, 983, 999, 1010, 1027, 1038, 1055, 1066, 1081, 1092, 1107, 1118, 1133, 1150, 1164, 1180, 1194, 1210, 1224, 1240, 1256, 1272, 1289, 1305, 1319, 1335, 1349, 1365, 1379, 1395, 1409, 1420, 1434, 1445, 1459, 1471, 1488, 1500, 1517, 1524		\immediate 67, 783 \include 117, 120, 271, 772 \includeonly 149, 168, 169, 662, 1381, 1383, 1531, 1539 \IncludeTests 731
\endcsname 10, 181, 775		
\endinput 51		
\endlinechar 5, 11, 23, 825		
\EndTest 695, 728		

1298, 1306, 1328, 1336, 1358, 1366, 1393, 1396, 1418, 1421, 1443, 1446, 1469, 1472, 1498, 1501	
L	
\LogTests	732
\loop	679, 689
M	
\makeatletter	676, 733
\makeatother	712
\MakeMatcher	459
\MatchMethodText	798, 804
\MessageBreak	149, 172, 195, 390, 634
N	
\NeedsTeXFormat	2
\newcommand	116, 167, 758, 771, 787, 797
\newif	179, 467
\numexpr	55
O	
\org@filecontents	819, 822
\org@typein	734, 740
\outputmode	58
P	
\PackageWarning	170, 389
\PackageWarningNoLine	148, 194, 501, 633
\pdf@filedump	811
\pdf@filesize	811
\pdf@unescapehex	810
\pdfextension	54, 56, 57
\pdffeedback	55
\pdfifastobj	55
\pdfmatch	196, 331, 391, 484, 491, 502
\pdfnames	56
\pdfobj	57
\pdfoutput	58
\ProcessOptions	155, 203
\protected	55, 56, 57
\providecommand	63, 75, 99, 115, 140, 528, 830, 842, 847
\ProvidesPackage	52
R	
\RangeCatcodeInvalid	687, 696, 697, 698, 699
\renewcommand	120, 169
\repeat	686, 693
\RequirePackage	435, 456
\RestoreCatcodes	677, 680, 681, 711
S	
\space	149, 171, 232, 324, 331, 338, 344, 374
\Stack	758, 854, 879, 904, 929, 954, 979, 1006, 1034, 1062, 1088, 1114, 1140, 1171, 1201, 1231, 1263, 1296, 1326, 1356, 1391, 1416, 1441, 1466, 1495
T	
\Test	695, 714
\the	11, 12, 13, 14, 15, 16, 17, 18, 31, 682
\TMP@EnsureCode	28, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50
\typein	373, 601, 734, 735
\typeout	261, 262, 263, 264, 265, 271, 293, 305, 309, 318, 319, 322, 334, 337, 343, 344, 346, 347, 348, 375, 605, 752, 753
U	
\unexpanded	743, 748, 793, 794, 803, 804
\usepackage	672, 673, 674, 719, 723
W	
\write	67, 783
X	
\x	9, 21
Z	
\zap@space	231