

The flexisym package

Author: Morten Høgholm
Inactively maintained by Will Robertson
Feedback: <https://github.com/wspr/breqn/issues>

2008/08/08 v0.97a

User's guide

For now, the user's guide is in breqn.

Implementation

1 flexisym

```
1 <*package>
2 \RequirePackage{expl3}[2009/08/05]
3 \ProvidesExplPackage{flexisym}{2013/03/16}{0.97c}{Make math characters macros}
4
5 \edef\do{%
6   \noexpand\AtEndOfPackage{%
7     \catcode\number'\="\number\catcode'\ "
8     \relax
9   }%
10 }
11 \do \let\do\relax
12 \catcode'\="=12
13 \let\@sym\@gobble
14 \DeclareOption{robust}{%
15   \def\@sym#1{%
16     \ifx\protect\@typeset@protect \else\protect#1\exp_after:wN\use_none:n\nn\fi
17   }%
18 }
```

The math groups (mg) here relate to `\textfontn`.

```
19 \def\mg@bin{2}% binary operators
20 \def\mg@rel{2}% relations
21 %%\def\mg@nre{B}% negated relations
22 \def\mg@del{3}% delimiters
```

```

23 %%\def\mg@arr{B}% arrows
24 \def\mg@acc{0}% accents
25 \def\mg@cop{3}% cumulative operators (sum, int)
26 \def\mg@latin{1}% (Latin) letters
27 \def\mg@greek{1}% (lowercase) Greek
28 \def\mg@GreeK{0}% (capital) Greek
29 %%\def\mg@bflatin{4}% bold upright Latin letters ?
30 %%\def\mg@Bbb{B}% blackboard bold
31 \def\mg@cal{2}% script/calligraphic
32 %%\def\mg@frak{5}% Fraktur letters
33 \def\mg@digit{0}% decimal digits % 1 = oldstyle, 0 = capital

```

This is how we insert mathchars. The command has three arguments: class, fam and slot position and so it is always given as hexadecimal. This way of separating things should make it easier to get this to work with XeTeX et al. which have many more slot positions

```

34 \cs_set_protected:Nn \math_char:Nnn {
35   \tex_mathchar:D \__int_eval:w " #1#2#3 \__int_eval_end:
36 }

```

Delimiters and radicals are similar except here we have both small and large variant. Radicals have no class.

```

37 \cs_set_protected:Nn \math_delimiter:NnnNn {
38   \tex_delimiter:D \__int_eval:w " #1#2#3#4#5 \__int_eval_end:
39 }
40 \cs_set_protected:Nn \math_radical:NnnNn {
41   \tex_radical:D \__int_eval:w " #1#2#3#4 \__int_eval_end:
42 }
43 \cs_set_protected:Nn \math_accent:Nnnnn {
44   \tex_mathaccent:D \__int_eval:w " #1 #2 #3 \__int_eval_end: {#4}
45 }
46
47 \let\sumlimits\displaylimits
48 \let\intlimits\nolimits
49 \let\name\limits\displaylimits

```

TeX defines eight types of atoms.

0. Ordinary
1. Operators
2. Binary
3. Relation
4. Open
5. Close
6. Punctuation
7. Inner

TeX defines eight math classes.

0. Ordinary
1. Operators
2. Binary
3. Relation
4. Open
5. Close
6. Punctuation
7. Variable family

flexisym/breqn extends this to types of classes.

0. Ordinary: (Ord), Bidirectional delimiters (DeB), Radicals (Rad), Accented items (Acc)
1. Operators: Cumulative Operators sum-like (COs), Cumulative Operators integral-like (COi)
2. Binary: (Bin)
3. Relation: (Rel), Arrow delimiters (DeA)
4. Open: (DeL)
5. Close (DeR)
6. Punctuation: (Pun)
7. Variable family: (Var)

Here's an overview of what we are about to do. Math chars of each type as defined by us need a basic operation for inserting it. We will call that function `\math_bsym_⟨type⟩:Nn`. Next there are compund symbols for each type which we name `\math_bcsym_⟨type⟩:Nn`. Also, there is inline mode and display mode which are different. We will call them for `\math_ism_⟨type⟩:Nn` `\math_icsym_⟨type⟩:Nn` for inline mode and `\math_dsym_⟨type⟩:Nn` and `\math_dcsym_⟨type⟩:Nn`. The code uses the terms `\math_sym_⟨type⟩:Nn` and `\math_csym_⟨type⟩:Nn` for the current meaning of things. First up the basic definitions. #1 is the math group it is from and #2 is the slot position.

```
50 \cs_new:Npn \math_bsym_Ord:Nn {\math_char:NNn 0 }% \m@Ord
51 \cs_new:Npn \math_bsym_Var:Nn {\math_char:NNn 7 }% \m@Var
52 \cs_new:Npn \math_bsym_Bin:Nn {\math_char:NNn 2 }% \m@Bin
53 \cs_new:Npn \math_bsym_Rel:Nn {\math_char:NNn 3 }% \m@Bin
54 \cs_new:Npn \math_bsym_Pun:Nn {\math_char:NNn 6 }% \m@Pun
```

```

55 \cs_new:Nn \math_bsym_COs:Nn { \math_char:NNn 1 #1 {#2} \sumlimits }% \m@COs
56 \cs_new:Nn \math_bsym_COi:Nn { \math_char:NNn 1 #1 {#2} \intlimits }% \m@COi
57 \cs_new:Nn \math_bsym_DeL:Nn { \math_sd_del_aux:Nnn 4 #1{#2} }% \m@DeL
58 \cs_new:Nn \math_bsym_DeR:Nn { \math_sd_del_aux:Nnn 5 #1{#2} }% \m@DeR
59 \cs_new:Nn \math_bsym_DeB:Nn { \math_sd_del_aux:Nnn 0 #1{#2} }% \m@DeB
60 \cs_new:Nn \math_bsym_DeA:Nn { \math_sd_del_aux:Nnn 3 #1{#2} }% \m@DeA
61 \cs_new:Nn \math_bsym_Rad:Nn { \math_sd_rad_aux:Nn #1{#2} }% \m@Rad
62 \cs_new:Npn \math_bsym_Acc:Nn #1#2#3#4 { \math_accent:NNnn #1#2{#3}{#4}}% name is wrong
63

```

Next is somewhat complicated internally. The way it is done is that delimiters and radicals need information about the smallest version of the symbol. If this smallest delimiter (SD) is defined, then use it. We have these functions to help us return the number. Extract the numbers to use and stick a function in front of it.

Code changed because now we require the smallest delimiter to be defined (it may be the same, no problem in that). So the two arguments present in `\math_bsym_DeL:Nn` are the location of extensible version (where the font will do the rest for us automatically). For each delimiter, a pointer is defined using the extensible characters family and slot as name and value equal to family and position of the smallest version. For \langle in standard L^AT_EX this is `{del}{00}` and `{OT1}{28}` respectively. Hence, `\math_bsym_DeL:Nn \mg@del {00}` must expand to `\math_delimiter:NNnNn 4 \mg@OT1 {28}\mg@del{00}`. So first expand away to get to the smallest version. Then call next function which shuffles the arguments around.

```

64 \cs_set:Npn \math_sd_del_aux:Nnn #1#2#3{
65   \exp_args:Nf \math_sd_del_auxi:nN {\use:c{sd@#2#3}} #1 #2{#3}
66 }
67 \cs_set:Npn \math_sd_del_auxi:nN #1#2{ \math_delimiter:NNnNn #2 #1 }

```

Same for radicals.

```

68 \cs_set:Npn \math_sd_rad_aux:Nn #1#2{
69   \exp_args:Nf \math_sd_rad_auxi:n {\use:c{sd@#1#2}} #1 {#2}
70 }
71 \cs_set:Npn \math_sd_rad_auxi:n #1{ \math_radical:NnNn #1 }
72
73

```

```

74 % \cs_set:Npn \math_sd_aux:nn #1#2 {
75 %   %\exp_args:Nnf \use:nn { #1} { \math_sd_auxi:Nn #2 }
76 %   \exp_args:Nnf \use:nn { #1} { \use:c{sd\use:nn#2} }
77 % }
78 % \cs_set:Npn \math_sd_auxi:Nn #1#2 {
79 %   \cs_if_free:cTF {sd@#1#2}
80 %   { #1{#2} }
81 %   { \use:c{sd@#1#2} }
82 % }

```

compound symbols here

```

83 \cs_set_protected:Npn \math_bcsym_Ord:Nn #1#2 { \@syntype \mathord { \OrdSymbol {#2} } }%
84 \cs_set_protected:Npn \math_bcsym_Var:Nn #1#2 { \@syntype \mathord { \OrdSymbol {#2} } }%
85 \cs_set_protected:Npn \math_bcsym_Bin:Nn #1#2 { \@syntype \mathbin { \OrdSymbol {#2} } }%

```

```

86 \cs_set_protected:Npn \math_bcsym_Rel:Nn #1#2 { \@syntype \mathrel { \OrdSymbol {#2} } }%
87 \cs_set_protected:Npn \math_bcsym_Pun:Nn #1#2 { \@syntype \mathpunct { \OrdSymbol {#2} } }%
88 \cs_set_protected:Npn \math_bcsym_COi:Nn #1#2 { \@syntype \mathop { \OrdSymbol {#2} \intl
89 \cs_set_protected:Npn \math_bcsym_COs:Nn #1#2 { \@syntype \mathop { \OrdSymbol {#2} \suml
90 \cs_set_protected:Npn \math_bcsym_DeL:Nn #1#2 { \@syntype \mathopen { \OrdSymbol {#2} } }%
91 \cs_set_protected:Npn \math_bcsym_DeR:Nn #1#2 { \@syntype \mathclose { \OrdSymbol {#2} } }%
92 \cs_set_protected:Npn \math_bcsym_DeB:Nn #1#2 { \@syntype \mathord { \OrdSymbol {#2} } }%
93 \cs_set_protected:Npn \math_bcsym_DeA:Nn #1#2 { \@syntype \mathrel { \OrdSymbol {#2} } }%
94 \cs_set_protected:Npn \math_bcsym_Acc:Nn { \@sym}%\@symAcc FIX!
95 % These three?
96 \cs_set_protected:Npn \math_bcsym_Ope:Nn #1#2{\@syntype\mathopen{\OrdSymbol{#2}}}\@symVar
97 \cs_set_protected:Npn \math_bcsym_Clo:Nn #1#2{\@syntype\mathclose{\OrdSymbol{#2}}}\@symVar
98 \cs_set_protected:Npn \math_bcsym_Inn:Nn #1#2{\@syntype\mathinner{\OrdSymbol{#2}}}\@symVar
99
100 \let\@syntype\@firstofone
101 \let\sym@global\global

The inline variants, using the basic operations. Currently we do not do any-
thing to inline math.

102 \cs_new:Npn \math_isym_Ord:Nn { \math_bsym_Ord:Nn }% \m@Ord
103 \cs_new:Npn \math_isym_Var:Nn { \math_bsym_Var:Nn }% \m@Var
104 \cs_new:Npn \math_isym_Bin:Nn { \math_bsym_Bin:Nn }% \m@Bin
105 \cs_new:Npn \math_isym_Rel:Nn { \math_bsym_Rel:Nn }% \m@Bin
106 \cs_new:Npn \math_isym_Pun:Nn { \math_bsym_Pun:Nn }% \m@Pun
107 \cs_new:Npn \math_isym_COs:Nn { \math_bsym_COs:Nn }% \m@COs
108 \cs_new:Npn \math_isym_COi:Nn { \math_bsym_COi:Nn }% \m@COi
109 \cs_new:Npn \math_isym_DeL:Nn { \math_bsym_DeL:Nn }% \m@DeL
110 \cs_new:Npn \math_isym_DeR:Nn { \math_bsym_DeR:Nn }% \m@DeR
111 \cs_new:Npn \math_isym_DeB:Nn { \math_bsym_DeB:Nn }% \m@DeB
112 \cs_new:Npn \math_isym_DeA:Nn { \math_bsym_DeA:Nn }% \m@DeA
113 \cs_new:Npn \math_isym_Rad:Nn { \math_bsym_Rad:Nn }% \m@Rad
114 \cs_new:Npn \math_isym_Acc:Nn { \math_bsym_DeL:Nn }% name is wrong
115 % inline compound
116 \cs_set_protected:Npn \math_icsym_Ord:Nn { \math_bcsym_Ord:Nn }
117 \cs_set_protected:Npn \math_icsym_Var:Nn { \math_bcsym_Var:Nn }
118 \cs_set_protected:Npn \math_icsym_Bin:Nn { \math_bcsym_Bin:Nn }
119 \cs_set_protected:Npn \math_icsym_Rel:Nn { \math_bcsym_Rel:Nn }
120 \cs_set_protected:Npn \math_icsym_Pun:Nn { \math_bcsym_Pun:Nn }
121 \cs_set_protected:Npn \math_icsym_COi:Nn { \math_bcsym_COi:Nn }
122 \cs_set_protected:Npn \math_icsym_COs:Nn { \math_bcsym_COs:Nn }
123 \cs_set_protected:Npn \math_icsym_DeL:Nn { \math_bcsym_DeL:Nn }
124 \cs_set_protected:Npn \math_icsym_DeR:Nn { \math_bcsym_DeR:Nn }
125 \cs_set_protected:Npn \math_icsym_DeB:Nn { \math_bcsym_DeB:Nn }
126 \cs_set_protected:Npn \math_icsym_DeA:Nn { \math_bcsym_DeA:Nn }
127 \cs_set_protected:Npn \math_icsym_Acc:Nn { \math_bcsym_Acc:Nn }
128 \cs_set_protected:Npn \math_icsym_Ope:Nn { \math_bcsym_Ope:Nn }
129 \cs_set_protected:Npn \math_icsym_Clo:Nn { \math_bcsym_Clo:Nn }
130 \cs_set_protected:Npn \math_icsym_Inn:Nn { \math_bcsym_Inn:Nn }

```

The display variants, using the basic operations. Currently we do not do anything to inline math.

```

131 \cs_new:Npn \math_dsym_Ord:Nn { \math_bsym_Ord:Nn }
132 \cs_new:Npn \math_dsym_Var:Nn { \math_bsym_Var:Nn }
133 \cs_new:Npn \math_dsym_Bin:Nn { \math_bsym_Bin:Nn }
134 \cs_new:Npn \math_dsym_Rel:Nn { \math_bsym_Rel:Nn }
135 \cs_new:Npn \math_dsym_Pun:Nn { \math_bsym_Pun:Nn }
136 \cs_new:Npn \math_dsym_COs:Nn { \math_bsym_COs:Nn }
137 \cs_new:Npn \math_dsym_COi:Nn { \math_bsym_COi:Nn }
138 \cs_new:Npn \math_dsym_DeL:Nn { \math_bsym_DeL:Nn }
139 \cs_new:Npn \math_dsym_DeR:Nn { \math_bsym_DeR:Nn }
140 \cs_new:Npn \math_dsym_DeB:Nn { \math_bsym_DeB:Nn }
141 \cs_new:Npn \math_dsym_DeA:Nn { \math_bsym_DeA:Nn }
142 \cs_new:Npn \math_dsym_Rad:Nn { \math_bsym_Rad:Nn }
143 \cs_new:Npn \math_dsym_Acc:Nn { \math_bsym_DeL:Nn }
144 % inline compound
145 \cs_set_protected:Npn \math_dcsym_Ord:Nn { \math_bcsym_Ord:Nn }
146 \cs_set_protected:Npn \math_dcsym_Var:Nn { \math_bcsym_Var:Nn }
147 \cs_set_protected:Npn \math_dcsym_Bin:Nn { \math_bcsym_Bin:Nn }
148 \cs_set_protected:Npn \math_dcsym_Rel:Nn { \math_bcsym_Rel:Nn }
149 \cs_set_protected:Npn \math_dcsym_Pun:Nn { \math_bcsym_Pun:Nn }
150 \cs_set_protected:Npn \math_dcsym_COi:Nn { \math_bcsym_COi:Nn }
151 \cs_set_protected:Npn \math_dcsym_COs:Nn { \math_bcsym_COs:Nn }
152 \cs_set_protected:Npn \math_dcsym_DeL:Nn { \math_bcsym_DeL:Nn }
153 \cs_set_protected:Npn \math_dcsym_DeR:Nn { \math_bcsym_DeR:Nn }
154 \cs_set_protected:Npn \math_dcsym_DeB:Nn { \math_bcsym_DeB:Nn }
155 \cs_set_protected:Npn \math_dcsym_DeA:Nn { \math_bcsym_DeA:Nn }
156 \cs_set_protected:Npn \math_dcsym_Acc:Nn { \math_bcsym_Acc:Nn }
157 \cs_set_protected:Npn \math_dcsym_Ope:Nn { \math_bcsym_Ope:Nn }
158 \cs_set_protected:Npn \math_dcsym_Clo:Nn { \math_bcsym_Clo:Nn }
159 \cs_set_protected:Npn \math_dcsym_Inn:Nn { \math_bcsym_Inn:Nn }

```

Almost ready now! Now just need two commands to initialize these settings.

```

160 \cs_set:Npn \math_setup_inline_symbols:
161 {
162   \cs_set_eq:NN \math_sym_Ord:Nn \math_ism_Ord:Nn
163   \cs_set_eq:NN \math_sym_Var:Nn \math_ism_Var:Nn
164   \cs_set_eq:NN \math_sym_Bin:Nn \math_ism_Bin:Nn
165   \cs_set_eq:NN \math_sym_Rel:Nn \math_ism_Rel:Nn
166   \cs_set_eq:NN \math_sym_Pun:Nn \math_ism_Pun:Nn
167   \cs_set_eq:NN \math_sym_COs:Nn \math_ism_COs:Nn
168   \cs_set_eq:NN \math_sym_COi:Nn \math_ism_COi:Nn
169   \cs_set_eq:NN \math_sym_DeL:Nn \math_ism_DeL:Nn
170   \cs_set_eq:NN \math_sym_DeR:Nn \math_ism_DeR:Nn
171   \cs_set_eq:NN \math_sym_DeB:Nn \math_ism_DeL:Nn
172   \cs_set_eq:NN \math_sym_DeA:Nn \math_ism_DeA:Nn
173   \cs_set_eq:NN \math_sym_Rad:Nn \math_ism_Rad:Nn
174   \cs_set_eq:NN \math_sym_Acc:Nn \math_ism_DeL:Nn
175   \cs_set_eq:NN \math_csym_Ord:Nn \math_icsym_Ord:Nn
176   \cs_set_eq:NN \math_csym_Var:Nn \math_icsym_Var:Nn

```

```

177 \cs_set_eq:NN \math_csym_Bin:Nn \math_icsym_Bin:Nn
178 \cs_set_eq:NN \math_csym_Rel:Nn \math_icsym_Rel:Nn
179 \cs_set_eq:NN \math_csym_Pun:Nn \math_icsym_Pun:Nn
180 \cs_set_eq:NN \math_csym_COi:Nn \math_icsym_COi:Nn
181 \cs_set_eq:NN \math_csym_COs:Nn \math_icsym_COs:Nn
182 \cs_set_eq:NN \math_csym_DeL:Nn \math_icsym_DeL:Nn
183 \cs_set_eq:NN \math_csym_DeR:Nn \math_icsym_DeR:Nn
184 \cs_set_eq:NN \math_csym_DeB:Nn \math_icsym_DeB:Nn
185 \cs_set_eq:NN \math_csym_DeA:Nn \math_icsym_DeA:Nn
186 \cs_set_eq:NN \math_csym_Acc:Nn \math_icsym_Acc:Nn
187 \cs_set_eq:NN \math_csym_Ope:Nn \math_icsym_Ope:Nn
188 \cs_set_eq:NN \math_csym_Clo:Nn \math_icsym_Clo:Nn
189 \cs_set_eq:NN \math_csym_Inn:Nn \math_icsym_Inn:Nn
190 }
191
192 \cs_set:Npn \math_setup_display_symbols:
193 {
194 \cs_set_eq:NN \math_sym_Ord:Nn \math_dsym_Ord:Nn
195 \cs_set_eq:NN \math_sym_Var:Nn \math_dsym_Var:Nn
196 \cs_set_eq:NN \math_sym_Bin:Nn \math_dsym_Bin:Nn
197 \cs_set_eq:NN \math_sym_Rel:Nn \math_dsym_Rel:Nn
198 \cs_set_eq:NN \math_sym_Pun:Nn \math_dsym_Pun:Nn
199 \cs_set_eq:NN \math_sym_COs:Nn \math_dsym_COs:Nn
200 \cs_set_eq:NN \math_sym_COi:Nn \math_dsym_COi:Nn
201 \cs_set_eq:NN \math_sym_DeL:Nn \math_dsym_DeL:Nn
202 \cs_set_eq:NN \math_sym_DeR:Nn \math_dsym_DeR:Nn
203 \cs_set_eq:NN \math_sym_DeB:Nn \math_dsym_DeL:Nn
204 \cs_set_eq:NN \math_sym_DeA:Nn \math_dsym_DeA:Nn
205 \cs_set_eq:NN \math_sym_Rad:Nn \math_dsym_Rad:Nn
206 \cs_set_eq:NN \math_sym_Acc:Nn \math_dsym_DeL:Nn
207 \cs_set_eq:NN \math_csym_Ord:Nn \math_dcsym_Ord:Nn
208 \cs_set_eq:NN \math_csym_Var:Nn \math_dcsym_Var:Nn
209 \cs_set_eq:NN \math_csym_Bin:Nn \math_dcsym_Bin:Nn
210 \cs_set_eq:NN \math_csym_Rel:Nn \math_dcsym_Rel:Nn
211 \cs_set_eq:NN \math_csym_Pun:Nn \math_dcsym_Pun:Nn
212 \cs_set_eq:NN \math_csym_COi:Nn \math_dcsym_COi:Nn
213 \cs_set_eq:NN \math_csym_COs:Nn \math_dcsym_COs:Nn
214 \cs_set_eq:NN \math_csym_DeL:Nn \math_dcsym_DeL:Nn
215 \cs_set_eq:NN \math_csym_DeR:Nn \math_dcsym_DeR:Nn
216 \cs_set_eq:NN \math_csym_DeB:Nn \math_dcsym_DeB:Nn
217 \cs_set_eq:NN \math_csym_DeA:Nn \math_dcsym_DeA:Nn
218 \cs_set_eq:NN \math_csym_Acc:Nn \math_dcsym_Acc:Nn
219 \cs_set_eq:NN \math_csym_Ope:Nn \math_dcsym_Ope:Nn
220 \cs_set_eq:NN \math_csym_Clo:Nn \math_dcsym_Clo:Nn
221 \cs_set_eq:NN \math_csym_Inn:Nn \math_dcsym_Inn:Nn
222 }

```

Phew, that was it.

Well, almost. We need to set them up for use properly. Should they be added to `\everymath`? Probably, for math within displays. However, this is a lot of

extra processing which we could tackle in the display setup.

```

223 \math_setup_inline_symbols:
    Need an active character for a second. Don't rely on ~ being active!
224 \edef\tmp{\catcode\z@=\the\catcode\z@}
225 \catcode\z@=\active
226 \def\DeclareFlexSymbol#1#2#3#4{%
227   \begingroup
228   \cs_set_protected:Npx \@tempb{
229     \exp_not:N \@sym \exp_not:N #1 \exp_not:c {math_sym_#2:Nn}
230     \exp_not:c {mg@#3}{#4}
231   }
232   \ifcat\exp_not:N #1 \relax
233     \sym@global \let #1 \@tempb
234   \else
235     \sym@global \mathcode #1="8000 \relax
236     \lccode\z@=' #1 \relax
237     \lowercase{\sym@global \let ^^@ \@tempb}% zero char
238   \fi
239   \endgroup
240 }
241 \tmp % restore catcode
242 \cs_set:Npn \DeclareFlexDelimiter #1#2#3#4#5#6{
243   \DeclareFlexSymbol{#1}{#2}{#3}{#4}
244   \cs_gset:cpx{sd@ \use:c{mg@#3}{#4}{\exp_not:c{mg@#5}{#6}}
245 }
246
  \DeclareFlexCompoundSymbol{\cdots}{Inn}{\cdotp\cdotp\cdotp} \def\@symInn#1#2{\@symtyp
  \@symtype \mathinner{\OrdSymbol{\cdotp\cdotp\cdotp}}}
247 \def\DeclareFlexCompoundSymbol#1#2#3{%
248   \exp_args:NNo \DeclareRobustCommand#1{\csname math_csym_#2:Nn\endcsname#1{#3}}%
249   \sym@global \let #1 #1 \relax
250 }
251 \DeclareRobustCommand\textchar{\text@char\textfont}
252 \DeclareRobustCommand\scriptchar{\text@char\scriptfont}%
  Simplified the next bit because now the slot is read as one argument so no afteras-
  signment and what have you. Just drop the char directly.
253 \def\text@char@sym#1#2#3#4{% #3=fam, #4=slot
254   \begingroup
255     \cs_set_eq:NN \@sym \prg_do_nothing: % defense against infinite loops
    the next line will result in \scriptfont<num>, where #3 provides the <num>.
256     \the\text@script@char#3%
257     \char"#4\endgroup
258 }
259 \edef\tmp{\catcode\z@=\the\catcode\z@}
260 \catcode\z@=\active
261 \def\text@char#1#2{\begingroup
262   \check@mathfonts

```



```

263 \cs_set_eq:NN \text@script@char #1
264 \cs_set_eq:NN \@sym \text@char@sym
265 \cs_set_eq:NN \@syntype \use_ii:nn
266 \cs_set_eq:NN \OrdSymbol \use:n
267 \cs_set_eq:NN \ifmmode \iftrue
268 \everymath{ $\use_none:n$ }%
269 \def\mkern{\muskip\z@}
270 \cs_set_eq:NN\mskip\mkern
271 \ifcat\relax\noexpand#2% true if #2 is a cs.
272 #2%
273 \else
274 \lccode\z@=\expandafter'\string#2\relax
275 \lowercase{^^@}%
276 \fi
277 \endgroup
278 }
279 \tmp % restore catcode
280 \providecommand\textprime{}
281 \DeclareRobustCommand\textprime{\leavevmode
282 \raise.8ex\hbox{\text@char\scriptfont\prime}%
283 }
284 \ifundefined{resetMathstrut@}{-}{%
285 \def\resetMathstrut@{%
286 \setbox\z@\hbox{\textchar\vert}%
287 \ht\Mathstrutbox@ht\z@ \dp\Mathstrutbox@dp\z@
288 }%
289 }

```

Arrow fills. changed to 7mu as in amsmath

```

290 \ifundefined{rightarrowfill@}{-}{%
291 \def\rightarrowfill@#1{\m@th\setboxz@h{ $\relbar$ }\ht\z@\z@
292 $#1\copy\z@\mkern-7mu\cleaders
293 \hbox{ $\mkern-2mu\box\z@\mkern-2mu$ }\hfill
294 \mkern-6mu\OrdSymbol{\rightarrow}$}
295 \def\leftarrowfill@#1{\m@th\setboxz@h{ $\relbar$ }\ht\z@\z@
296 $#1\OrdSymbol{\leftarrow}\mkern-6mu\cleaders
297 \hbox{ $\mkern-2mu\copy\z@\mkern-2mu$ }\hfill
298 \mkern-7mu\box\z@}
299 \def\leftrightarrowfill@#1{\m@th\setboxz@h{ $\relbar$ }\ht\z@\z@
300 $#1\OrdSymbol{\leftarrow}\mkern-6mu\cleaders
301 \hbox{ $\mkern-2mu\box\z@\mkern-2mu$ }\hfill
302 \mkern-6mu\OrdSymbol{\rightarrow}$}
303 }

```

hey, this looks like a simple case switch...

```

304 \def\binrel@sym#1#2#3#4{%
305 \xdef\binrel@##1{%
306 \ifx\math_sym_Ord:Nn #2 \math_csym_Ord:Nn
307 \else\ifx\math_sym_Var:Nn#2 \math_csym_Var:Nn
308 \else\ifx\math_sym_COs:Nn#2 \math_csym_COs:Nn
309 \else\ifx\math_sym_COi:Nn#2 \math_csym_COi:Nn

```

```

310 \else\ifx\math_sym_Bin:Nn#2 \math_csym_Bin:Nn
311 \else\ifx\math_sym_Rel:Nn#2 \math_csym_Rel:Nn
312 \else\ifx\math_sym_Pun:Nn#2 \math_csym_Pun:Nn
313 \else\exp_not:N\@symErr \fi\fi\fi\fi\fi\fi\fi
314 ?{\exp_not:N\OrdSymbol{##1}}}%
315 }
316
317 \def\binrel@a{%
318 \def\math_sym_Ord:Nn##1##2{\gdef\binrel@#####1{\math_sym_Ord:Nn##1{\OrdSymbol{#####1}}}}%
319 \def\math_sym_Var:Nn##1##2{\gdef\binrel@#####1{\math_sym_Var:Nn##1{\OrdSymbol{#####1}}}}%
320 \def\math_sym_COs:Nn##1##2{\gdef\binrel@#####1{\math_sym_COs:Nn##1{\OrdSymbol{#####1}}}}%
321 \def\math_sym_COi:Nn##1##2{\gdef\binrel@#####1{\math_sym_COi:Nn##1{\OrdSymbol{#####1}}}}%
322 \def\math_sym_Bin:Nn##1##2{\gdef\binrel@#####1{\math_sym_Bin:Nn##1{\OrdSymbol{#####1}}}}%
323 \def\math_sym_Rel:Nn##1##2{\gdef\binrel@#####1{\math_sym_Rel:Nn##1{\OrdSymbol{#####1}}}}%
324 \def\math_sym_Pun:Nn##1##2{\gdef\binrel@#####1{\math_sym_Pun:Nn##1{\OrdSymbol{#####1}}}}%
325 }
326 \def\binrel@#1{%
327 \setbox\z@\hbox{${}
328 \let\mathchoice\@gobblethree
329 \let\@sym\binrel@sym \binrel@a
330 #1$}%
331 }
332 \def\@symextension{sym}
333 \newcommand\usesymbols[1]{%
334 \clist_map_variable:nNn{#1}\@tempb{%
335 \exp_args:No\@onefilewithoptions{\@tempb} [] []\@symextension
336 }%
337 }
338 % Need to introduce \ProvidesExplFile somehow
339 \newcommand\ProvidesSymbols[1]{\ProvidesFile{#1.sym}}
340 \DeclareRobustCommand{\not}[1]{\math_csym_Rel:Nn\not{\OrdSymbol{\notRel#1}}}
341 \DeclareRobustCommand{\OrdSymbol}[1]{%
342 \begingroup\mathchars@reset#1\endgroup
343 }
344 \def\mathchars@reset{\let\@sym\@sym@ord \let\@syntype\@syntype@ord
345 \let\OrdSymbol\relax}
346 \def\@syntype@ord#1#{}% a strange sort of \@gobble
347 \def\@sym@ord#1#2{\exp_after:wN\@sym@ord@a\string#2\@nil}%

```

Read delimited argument here. We want to find first character of DeA, Bin, etc. and the control sequence checked agains is $\math@DeL$, $\math@Pun$, etc. The lccode trick makes the . into an @ with catcode 12. This is what results when the code is called with `\string`. Beware of this when we change internal names for math groups! If a Delimiter is found, insert it with class 0 but use the smallest version available. Otherwise just insert math char of class 0. The code here is not pretty and it indicates it should be tackled differently!

```

348 \begingroup
349 \lccode'\.='\_ \lowercase{\endgroup
350 \def\@sym@ord@a#1.#2.}#3#4\@nil#5#6{%
351 \if D#3

```

```

352     %\math_ord_delim_aux:Nn #5{#6}
353     \math_sd_del_aux:Nnn 0 #5{#6}% check if this works!
354   \else
355     \math_char:NNn 0 #5{#6}
356   \fi
357 }
358 \cs_set:Nn \math_ord_delim_aux:Nn {
359   \math_sd_aux:nn { \math_char:NNn 0 } {#1{#2}}
360 }

```

Before declaring any math characters active, we have to take care of a small problem with `amsmath v2.x`, if it is loaded before `flexisym`. `\std@minus` and `\std@equal` are defined as

```

\mathchardef\std@minus\mathcode'\-\relax
\mathchardef\std@equal\mathcode'\=\relax

```

in `amsmath.sty` and again `\AtBeginDocument`. The latter is because

In case some alternative math fonts are loaded later. [`amsmath.dtx`]

The problem arises because `flexisym` sets the mathcode of all symbols to 32768 which is illegal for a `\mathchardef`.

We have to remove the assignments from the `\AtBeginDocument` hook as they will cause an error there.

```

361 \@ifpackageloaded{amsmath}{%
362   \begingroup

```

Split the contents of `\@begindocumenthook` by reading what we search for as a delimited argument and ensure these two assignments do not take place. It is questionable if anything reasonable can be done to them. In the case of a package such as `mathpazo` which defines

```

\DeclareMathSymbol{=}{\mathrel}{upright}{"3D}

```

the `\Relbar` will look wrong if we don't use the correct symbol. The way to solve this is define additional `.sym` files which contain the definition of `\relbar` and `\Relbar` needed. We need those additional files anyway for things like `\joinord`.

```

363   \long\def\next#1\mathchardef\std@minus\mathcode'\-\relax
364           \mathchardef\std@equal\mathcode'\=\relax#2\flexi@stop{%
365     \toks@{#1#2}%
366     \xdef\@begindocumenthook{\the\toks@}%
367   }%
368   \expandafter\next\@begindocumenthook\flexi@stop
369   \endgroup
370 }{}

```

There is problem when using `\DeclareMathOperator` as the operators defined call a command `\newmcodes@` which relies on the mathcode of `-` being less than

32768. We delay the definition `\AtBeginDocument` in case `amssymb` hasn't been loaded yet.

```

371 \AtBeginDocument{%
372 \def\newmcodes@{%
373 \mathcode '\ '39\space
374 \mathcode '\*42\space
375 \mathcode '\."613A\space
376 \ifnum\mathcode'\-45\space
377 \else

```

The extra check. Don't do anything if `-` is math active.

```

378 \ifnum\mathcode'\-32768\space
379 \else
380 \mathchardef \std@minus \mathcode '\-\relax
381 \fi
382 \fi
383 \mathcode '\-45\space
384 \mathcode '\ /47\space
385 \mathcode '\."603A\space\relax
386 }%
387 }

```

And we then continue with the options.

```

388 \DeclareOption{mathstyleoff}{%
389 \PassOptionsToPackage{noactivechars}{mathstyle}}
390 \DeclareOption{cmbase}{\usesymbols{cmbase}}
391 \DeclareOption{mathpazo}{\usesymbols{mathpazo}}
392 \DeclareOption{mathptmx}{\usesymbols{mathptmx}}
393 \ExecuteOptions{cmbase}
394 \ProcessOptions\relax
395 \renewcommand{\lnot}{\neg}
396 \renewcommand{\land}{\wedge}
397 \renewcommand{\lor}{\vee}
398 \renewcommand{\le}{\leq}
399 \renewcommand{\ge}{\geq}
400 \renewcommand{\ne}{\neq}
401 \renewcommand{\owns}{\ni}
402 \renewcommand{\gets}{\leftarrow}
403 \renewcommand{\to}{\rightarrow}
404 \renewcommand{\|}{\Vert}
405 \RequirePackage{mathstyle}
406 \endinput

```

2 cmbase, mathpazo, mathptmx

For each math font package we define a corresponding symbol file with extension `sym`. The Computer Modern base is called `cmbase` and `mathpazo` and `mathptmx` corresponds to the packages. The definitions are almost identical as they mostly concern the positions in the math font encodings. Look for differences in `\joinord`,

`\relbar` and `\Relbar`. If you inspect the source code, you'll see that the support for `mathptmx` didn't require any work but I thought it better to create a `sym` file to maintain a uniform interface.

Open question on ! and ?: maybe they should have type 'Pun' instead of 'DeR'. Need to search for uses in math in AMS archives. Or, maybe add a special 'Clo' type for them: non-extensible closing delimiter.

Default mathgroup setup.

```

407 <*cmbase | mathpazo | mathptmx>
408 <cmbase> \ProvidesSymbols{cmbase}[2007/12/19 v0.92]
409 <mathpazo> \ProvidesSymbols{mathpazo}[2010/07/11 v0.3]
410 <mathptmx> \ProvidesSymbols{mathptmx}[2010/07/11 v0.3]
411 \ExplSyntaxOn
412 \cs_gset:cpx {mg@OT1} {\hexnumber@\symoperators}
413 \cs_gset:cpx {mg@OML} {\hexnumber@\symletters}
414 \cs_gset:cpx {mg@OMS} {\hexnumber@\symsymbols}
415 \cs_gset:cpx {mg@OMX} {\hexnumber@\symlargesymbols}
416 \cs_gset:Npx \mg@bin {\mg@OMS}
417 \cs_gset:Npx \mg@del {\mg@OMX}
418 \cs_gset:Npx \mg@digit {\exp_not:c{mg@OT1}}
419 \cs_gset:Npn \mg@latin {\mg@OML}
420 \cs_gset_eq:NN \mg@Latin \mg@latin
421 \cs_gset_eq:NN \mg@greek \mg@latin
422 <cmbase | mathptmx> \cs_gset_eq:NN \mg@greek \mg@digit

```

Mathpazo takes the upper case greeks from the letter font if `slantedGreek` is in effect, but from *upright* if not. Mathptmx also takes the slanted greek from the letter font.

```

423 <mathpazo> \@ifpackagewith{mathpazo}{slantedGreek}{%
424 <mathpazo> \cs_gset_eq:NN \mg@greek \mg@latin
425 <mathpazo> }{%
426 <mathpazo> \cs_gset:cpx {mg@greek} {\hexnumber@\symupright}
427 <mathpazo> }
428 <mathptmx> \@ifpackagewith{mathptmx}{slantedGreek}{%
429 <mathptmx> \cs_gset_eq:NN \mg@greek \mg@latin
430 <mathptmx> }{}
431 \cs_gset_eq:NN \mg@rel \mg@bin
432 \cs_gset_eq:NN \mg@ord \mg@bin
433 \cs_gset_eq:NN \mg@cop \mg@del

```

Symbols from the 128-character `cmr` encoding. Paren and square bracket delimiters from this encoding are covered by the definitions in the `cmex` section, however.

```

434 \DeclareFlexSymbol{!} {\Pun}{OT1}{21}
435 \DeclareFlexSymbol{+} {\Bin}{OT1}{2B}
436 \DeclareFlexSymbol{:} {\Rel}{OT1}{3A}
437 \DeclareFlexSymbol{\colon} {\Pun}{OT1}{3A}
438 \DeclareFlexSymbol{;} {\Pun}{OT1}{3B}

```

```
439 \DeclareFlexSymbol{=}      {Rel}{OT1}{3D}
440 \DeclareFlexSymbol{?}     {Pun}{OT1}{3F}
```

$\mathcal{A}\mathcal{M}\mathcal{S}\mathcal{T}\mathcal{E}\mathcal{X}$, and therefore the `amsmath` package, make the uppercase Greek letters class 0 (nonvariable) instead of 7 (variable), to eliminate the glaring inconsistency with lowercase Greek. (In plain $\mathcal{T}\mathcal{E}\mathcal{X}$, `\bf\Delta` works, while `\bf\delta` doesn't.) Let us try to make them both variable (fonts permitting) instead of nonvariable.

```
441 \DeclareFlexSymbol{\Gamma} {Var}{Greek}{00}
442 \DeclareFlexSymbol{\Delta} {Var}{Greek}{01}
443 \DeclareFlexSymbol{\Theta} {Var}{Greek}{02}
444 \DeclareFlexSymbol{\Lambda} {Var}{Greek}{03}
445 \DeclareFlexSymbol{\Xi}     {Var}{Greek}{04}
446 \DeclareFlexSymbol{\Pi}     {Var}{Greek}{05}
447 \DeclareFlexSymbol{\Sigma} {Var}{Greek}{06}
448 \DeclareFlexSymbol{\Upsilon}{Var}{Greek}{07}
449 \DeclareFlexSymbol{\Phi}    {Var}{Greek}{08}
450 \DeclareFlexSymbol{\Psi}    {Var}{Greek}{09}
451 \DeclareFlexSymbol{\Omega}  {Var}{Greek}{0A}
```

Decimal digits.

```
452 \DeclareFlexSymbol{0}{Var}{digit}{30}
453 \DeclareFlexSymbol{1}{Var}{digit}{31}
454 \DeclareFlexSymbol{2}{Var}{digit}{32}
455 \DeclareFlexSymbol{3}{Var}{digit}{33}
456 \DeclareFlexSymbol{4}{Var}{digit}{34}
457 \DeclareFlexSymbol{5}{Var}{digit}{35}
458 \DeclareFlexSymbol{6}{Var}{digit}{36}
459 \DeclareFlexSymbol{7}{Var}{digit}{37}
460 \DeclareFlexSymbol{8}{Var}{digit}{38}
461 \DeclareFlexSymbol{9}{Var}{digit}{39}
```

Symbols from the 128-character `cmmi` encoding.

```
462 \DeclareFlexSymbol{,}{Pun}{OML}{3B}
463 \DeclareFlexSymbol{.}{Ord}{OML}{3A}
464 \DeclareFlexSymbol{/}{Ord}{OML}{3D}
465 \DeclareFlexSymbol{<}{Rel}{OML}{3C}
466 \DeclareFlexSymbol{>}{Rel}{OML}{3E}
```

To do: make the `Var` property of `lc Greek` work properly.

```
467 \DeclareFlexSymbol{\alpha}  {Var}{greek}{0B}
468 \DeclareFlexSymbol{\beta}   {Var}{greek}{0C}
469 \DeclareFlexSymbol{\gamma}  {Var}{greek}{0D}
470 \DeclareFlexSymbol{\delta} {Var}{greek}{0E}
471 \DeclareFlexSymbol{\epsilon}{Var}{greek}{0F}
472 \DeclareFlexSymbol{\zeta}   {Var}{greek}{10}
473 \DeclareFlexSymbol{\eta}    {Var}{greek}{11}
474 \DeclareFlexSymbol{\theta}  {Var}{greek}{12}
475 \DeclareFlexSymbol{\iota}   {Var}{greek}{13}
476 \DeclareFlexSymbol{\kappa}  {Var}{greek}{14}
477 \DeclareFlexSymbol{\lambda} {Var}{greek}{15}
```

```

478 \DeclareFlexSymbol{\mu}           {Var}{greek}{16}
479 \DeclareFlexSymbol{\nu}           {Var}{greek}{17}
480 \DeclareFlexSymbol{\xi}           {Var}{greek}{18}
481 \DeclareFlexSymbol{\pi}           {Var}{greek}{19}
482 \DeclareFlexSymbol{\rho}          {Var}{greek}{1A}
483 \DeclareFlexSymbol{\sigma}        {Var}{greek}{1B}
484 \DeclareFlexSymbol{\tau}          {Var}{greek}{1C}
485 \DeclareFlexSymbol{\upsilon}      {Var}{greek}{1D}
486 \DeclareFlexSymbol{\phi}          {Var}{greek}{1E}
487 \DeclareFlexSymbol{\chi}          {Var}{greek}{1F}
488 \DeclareFlexSymbol{\psi}          {Var}{greek}{20}
489 \DeclareFlexSymbol{\omega}        {Var}{greek}{21}
490 \DeclareFlexSymbol{\varepsilon}   {Var}{greek}{22}
491 \DeclareFlexSymbol{\vartheta}     {Var}{greek}{23}
492 \DeclareFlexSymbol{\varpi}        {Var}{greek}{24}
493 \DeclareFlexSymbol{\varrho}       {Var}{greek}{25}
494 \DeclareFlexSymbol{\varsigma}     {Var}{greek}{26}
495 \DeclareFlexSymbol{\varphi}       {Var}{greek}{27}

```

Note that in plain \TeX `\imath` and `\jmath` are not variable-font. But if a `j` changes font to, let's say, sans serif or calligraphic, a dotless `j` in the same context should change font in the same way.

```

496 \DeclareFlexSymbol{\imath}        {Var}{OML}{7B}
497 \DeclareFlexSymbol{\jmath}        {Var}{OML}{7C}
498 \DeclareFlexSymbol{\ell}          {Ord}{OML}{60}
499 \DeclareFlexSymbol{\wp}           {Ord}{OML}{7D}
500 \DeclareFlexSymbol{\partial}       {Ord}{OML}{40}
501 \DeclareFlexSymbol{\flat}         {Ord}{OML}{5B}
502 \DeclareFlexSymbol{\natural}       {Ord}{OML}{5C}
503 \DeclareFlexSymbol{\sharp}        {Ord}{OML}{5D}
504 \DeclareFlexSymbol{\triangleleft} {Bin}{OML}{2F}
505 \DeclareFlexSymbol{\triangleright} {Bin}{OML}{2E}
506 \DeclareFlexSymbol{\star}         {Bin}{OML}{3F}
507 \DeclareFlexSymbol{\smile}        {Rel}{OML}{5E}
508 \DeclareFlexSymbol{\frown}        {Rel}{OML}{5F}
509 \DeclareFlexSymbol{\leftharpoonup} {Rel}{OML}{28}
510 \DeclareFlexSymbol{\leftharpoondown} {Rel}{OML}{29}
511 \DeclareFlexSymbol{\rightharpoonup} {Rel}{OML}{2A}
512 \DeclareFlexSymbol{\rightharpoondown} {Rel}{OML}{2B}

```

Latin

```

513 \DeclareFlexSymbol{a}{Var}{latin}{61}
514 \DeclareFlexSymbol{b}{Var}{latin}{62}
515 \DeclareFlexSymbol{c}{Var}{latin}{63}
516 \DeclareFlexSymbol{d}{Var}{latin}{64}
517 \DeclareFlexSymbol{e}{Var}{latin}{65}
518 \DeclareFlexSymbol{f}{Var}{latin}{66}
519 \DeclareFlexSymbol{g}{Var}{latin}{67}
520 \DeclareFlexSymbol{h}{Var}{latin}{68}
521 \DeclareFlexSymbol{i}{Var}{latin}{69}

```

```

522 \DeclareFlexSymbol{j}{Var}{latin}{6A}
523 \DeclareFlexSymbol{k}{Var}{latin}{6B}
524 \DeclareFlexSymbol{l}{Var}{latin}{6C}
525 \DeclareFlexSymbol{m}{Var}{latin}{6D}
526 \DeclareFlexSymbol{n}{Var}{latin}{6E}
527 \DeclareFlexSymbol{o}{Var}{latin}{6F}
528 \DeclareFlexSymbol{p}{Var}{latin}{70}
529 \DeclareFlexSymbol{q}{Var}{latin}{71}
530 \DeclareFlexSymbol{r}{Var}{latin}{72}
531 \DeclareFlexSymbol{s}{Var}{latin}{73}
532 \DeclareFlexSymbol{t}{Var}{latin}{74}
533 \DeclareFlexSymbol{u}{Var}{latin}{75}
534 \DeclareFlexSymbol{v}{Var}{latin}{76}
535 \DeclareFlexSymbol{w}{Var}{latin}{77}
536 \DeclareFlexSymbol{x}{Var}{latin}{78}
537 \DeclareFlexSymbol{y}{Var}{latin}{79}
538 \DeclareFlexSymbol{z}{Var}{latin}{7A}
539 \DeclareFlexSymbol{A}{Var}{Latin}{41}
540 \DeclareFlexSymbol{B}{Var}{Latin}{42}
541 \DeclareFlexSymbol{C}{Var}{Latin}{43}
542 \DeclareFlexSymbol{D}{Var}{Latin}{44}
543 \DeclareFlexSymbol{E}{Var}{Latin}{45}
544 \DeclareFlexSymbol{F}{Var}{Latin}{46}
545 \DeclareFlexSymbol{G}{Var}{Latin}{47}
546 \DeclareFlexSymbol{H}{Var}{Latin}{48}
547 \DeclareFlexSymbol{I}{Var}{Latin}{49}
548 \DeclareFlexSymbol{J}{Var}{Latin}{4A}
549 \DeclareFlexSymbol{K}{Var}{Latin}{4B}
550 \DeclareFlexSymbol{L}{Var}{Latin}{4C}
551 \DeclareFlexSymbol{M}{Var}{Latin}{4D}
552 \DeclareFlexSymbol{N}{Var}{Latin}{4E}
553 \DeclareFlexSymbol{O}{Var}{Latin}{4F}
554 \DeclareFlexSymbol{P}{Var}{Latin}{50}
555 \DeclareFlexSymbol{Q}{Var}{Latin}{51}
556 \DeclareFlexSymbol{R}{Var}{Latin}{52}
557 \DeclareFlexSymbol{S}{Var}{Latin}{53}
558 \DeclareFlexSymbol{T}{Var}{Latin}{54}
559 \DeclareFlexSymbol{U}{Var}{Latin}{55}
560 \DeclareFlexSymbol{V}{Var}{Latin}{56}
561 \DeclareFlexSymbol{W}{Var}{Latin}{57}
562 \DeclareFlexSymbol{X}{Var}{Latin}{58}
563 \DeclareFlexSymbol{Y}{Var}{Latin}{59}
564 \DeclareFlexSymbol{Z}{Var}{Latin}{5A}

```

The `\ldotPun` glyph is used in constructing the `\ldots` symbol. It is just a period with a different math symbol class. `\lhookRel` and `\rhookRel` are used in a similar way for building hooked arrow symbols.

```

565 \DeclareFlexSymbol{\ldotPun}{Pun}{OML}{3A}
566 \def\ldotp{\ldotPun}
567 \DeclareFlexSymbol{\lhookRel}{Rel}{OML}{2C}

```


568 `\DeclareFlexSymbol{\rhookRel}{Rel}{OML}{2D}`

Symbols from the 128-character cmsy encoding.

569 `\DeclareFlexSymbol{*}` `{Bin}{bin}{03} % \ast`
570 `\DeclareFlexSymbol{-}` `{Bin}{bin}{00}`
571 `\DeclareFlexSymbol{|}` `{Ord}{OMS}{6A}`
572 `\DeclareFlexSymbol{\aleph}` `{Ord}{ord}{40}`
573 `\DeclareFlexSymbol{\Re}` `{Ord}{ord}{3C}`
574 `\DeclareFlexSymbol{\Im}` `{Ord}{ord}{3D}`
575 `\DeclareFlexSymbol{\infty}` `{Ord}{ord}{31}`
576 `\DeclareFlexSymbol{\prime}` `{Ord}{ord}{30}`
577 `\DeclareFlexSymbol{\emptyset}` `{Ord}{ord}{3B}`
578 `\DeclareFlexSymbol{\nabla}` `{Ord}{ord}{72}`
579 `\DeclareFlexSymbol{\top}` `{Ord}{ord}{3E}`
580 `\DeclareFlexSymbol{\bot}` `{Ord}{ord}{3F}`
581 `\DeclareFlexSymbol{\triangle}` `{Ord}{ord}{34}`
582 `\DeclareFlexSymbol{\forall}` `{Ord}{ord}{38}`
583 `\DeclareFlexSymbol{\exists}` `{Ord}{ord}{39}`
584 `\DeclareFlexSymbol{\neg}` `{Ord}{ord}{3A}`
585 `\DeclareFlexSymbol{\clubsuit}` `{Ord}{ord}{7C}`
586 `\DeclareFlexSymbol{\diamondsuit}` `{Ord}{ord}{7D}`
587 `\DeclareFlexSymbol{\heartsuit}` `{Ord}{ord}{7E}`
588 `\DeclareFlexSymbol{\spadesuit}` `{Ord}{ord}{7F}`
589 `\DeclareFlexSymbol{\smallint}` `{COs}{OMS}{73}`

Binary operators.

590 `\DeclareFlexSymbol{\bigtriangleup}` `{Bin}{bin}{34}`
591 `\DeclareFlexSymbol{\bigtriangledown}` `{Bin}{bin}{35}`
592 `\DeclareFlexSymbol{\wedge}` `{Bin}{bin}{5E}`
593 `\DeclareFlexSymbol{\vee}` `{Bin}{bin}{5F}`
594 `\DeclareFlexSymbol{\cap}` `{Bin}{bin}{5C}`
595 `\DeclareFlexSymbol{\cup}` `{Bin}{bin}{5B}`
596 `\DeclareFlexSymbol{\ddagger}` `{Bin}{bin}{7A}`
597 `\DeclareFlexSymbol{\dagger}` `{Bin}{bin}{79}`
598 `\DeclareFlexSymbol{\sqcap}` `{Bin}{bin}{75}`
599 `\DeclareFlexSymbol{\sqcup}` `{Bin}{bin}{74}`
600 `\DeclareFlexSymbol{\uplus}` `{Bin}{bin}{5D}`
601 `\DeclareFlexSymbol{\amalg}` `{Bin}{bin}{71}`
602 `\DeclareFlexSymbol{\diamond}` `{Bin}{bin}{05}`
603 `\DeclareFlexSymbol{\bullet}` `{Bin}{bin}{0F}`
604 `\DeclareFlexSymbol{\wr}` `{Bin}{bin}{6F}`
605 `\DeclareFlexSymbol{\div}` `{Bin}{bin}{04}`
606 `\DeclareFlexSymbol{\odot}` `{Bin}{bin}{0C}`
607 `\DeclareFlexSymbol{\oslash}` `{Bin}{bin}{0B}`
608 `\DeclareFlexSymbol{\otimes}` `{Bin}{bin}{0A}`
609 `\DeclareFlexSymbol{\ominus}` `{Bin}{bin}{09}`
610 `\DeclareFlexSymbol{\oplus}` `{Bin}{bin}{08}`
611 `\DeclareFlexSymbol{\mp}` `{Bin}{bin}{07}`
612 `\DeclareFlexSymbol{\pm}` `{Bin}{bin}{06}`
613 `\DeclareFlexSymbol{\circ}` `{Bin}{bin}{0E}`
614 `\DeclareFlexSymbol{\bigcirc}` `{Bin}{bin}{0D}`

```

615 \DeclareFlexSymbol{\setminusminus} {Bin}{bin}{6E}
616 \DeclareFlexSymbol{\cdot} {Bin}{bin}{01}
617 \DeclareFlexSymbol{\ast} {Bin}{bin}{03}
618 \DeclareFlexSymbol{\times} {Bin}{bin}{02}

```

Relation symbols.

```

619 \DeclareFlexSymbol{\propto} {Rel}{rel}{2F}
620 \DeclareFlexSymbol{\sqsubseteq} {Rel}{rel}{76}
621 \DeclareFlexSymbol{\sqsupseteq} {Rel}{rel}{77}
622 \DeclareFlexSymbol{\parallel} {Rel}{rel}{6B}
623 \DeclareFlexSymbol{\mid} {Rel}{rel}{6A}
624 \DeclareFlexSymbol{\dashv} {Rel}{rel}{61}
625 \DeclareFlexSymbol{\vdash} {Rel}{rel}{60}
626 \DeclareFlexSymbol{\nearrow} {Rel}{rel}{25}
627 \DeclareFlexSymbol{\searrow} {Rel}{rel}{26}
628 \DeclareFlexSymbol{\nwarrow} {Rel}{rel}{2D}
629 \DeclareFlexSymbol{\swarrow} {Rel}{rel}{2E}
630 \DeclareFlexSymbol{\Leftrightarrow} {Rel}{rel}{2C}
631 \DeclareFlexSymbol{\Leftarrow} {Rel}{rel}{28}
632 \DeclareFlexSymbol{\Rightarrow} {Rel}{rel}{29}
633 \DeclareFlexSymbol{\leq} {Rel}{rel}{14}
634 \DeclareFlexSymbol{\geq} {Rel}{rel}{15}
635 \DeclareFlexSymbol{\succ} {Rel}{rel}{1F}
636 \DeclareFlexSymbol{\prec} {Rel}{rel}{1E}
637 \DeclareFlexSymbol{\approx} {Rel}{rel}{19}
638 \DeclareFlexSymbol{\succeq} {Rel}{rel}{17}
639 \DeclareFlexSymbol{\preceq} {Rel}{rel}{16}
640 \DeclareFlexSymbol{\supset} {Rel}{rel}{1B}
641 \DeclareFlexSymbol{\subset} {Rel}{rel}{1A}
642 \DeclareFlexSymbol{\supseteq} {Rel}{rel}{13}
643 \DeclareFlexSymbol{\subseteq} {Rel}{rel}{12}
644 \DeclareFlexSymbol{\in} {Rel}{rel}{32}
645 \DeclareFlexSymbol{\ni} {Rel}{rel}{33}
646 \DeclareFlexSymbol{\gg} {Rel}{rel}{1D}
647 \DeclareFlexSymbol{\ll} {Rel}{rel}{1C}
648 \DeclareFlexSymbol{\leftrightharpoonup} {Rel}{rel}{24}
649 \DeclareFlexSymbol{\leftarrow} {Rel}{rel}{20}
650 \DeclareFlexSymbol{\rightarrow} {Rel}{rel}{21}
651 \DeclareFlexSymbol{\sim} {Rel}{rel}{18}
652 \DeclareFlexSymbol{\simeq} {Rel}{rel}{27}
653 \DeclareFlexSymbol{\perp} {Rel}{rel}{3F}
654 \DeclareFlexSymbol{\equiv} {Rel}{rel}{11}
655 \DeclareFlexSymbol{\asymp} {Rel}{rel}{10}

```

The `\notRel` glyph is a special zero-width glyph intended only for use in constructing negated symbols. `\mapstoRel` and `\cdotPun` have similar but more restricted applications.

```

656 \DeclareFlexSymbol{\notRel} {Rel}{rel}{36}
657 \DeclareFlexSymbol{\mapstoOrd} {Ord}{OMS}{37}
658 \DeclareFlexSymbol{\cdotOrd} {Ord}{OMS}{01}
659 \cs_set:Npn\cdotp{\mathpunct{\cdotOrd}}

```

Symbols from the 128-character cmex encoding. COs stands for ‘cumulative operator (sum-like)’. COi stands for ‘cumulative operator (integral-like)’. These typically differ only in the default placement of limits. cop stands for ‘cumulative operator math group’.

```

660 \DeclareFlexSymbol{\coprod} {COs}{cop}{60}
661 \DeclareFlexSymbol{\bigvee} {COs}{cop}{57}
662 \DeclareFlexSymbol{\bigwedge} {COs}{cop}{56}
663 \DeclareFlexSymbol{\biguplus} {COs}{cop}{55}
664 \DeclareFlexSymbol{\bigcap} {COs}{cop}{54}
665 \DeclareFlexSymbol{\bigcup} {COs}{cop}{53}
666 \DeclareFlexSymbol{\int} {COi}{cop}{52}
667 \DeclareFlexSymbol{\prod} {COs}{cop}{51}
668 \DeclareFlexSymbol{\sum} {COs}{cop}{50}
669 \DeclareFlexSymbol{\bigotimes} {COs}{cop}{4E}
670 \DeclareFlexSymbol{\bigoplus} {COs}{cop}{4C}
671 \DeclareFlexSymbol{\bigodot} {COs}{cop}{4A}
672 \DeclareFlexSymbol{\oint} {COi}{cop}{48}
673 \DeclareFlexSymbol{\bigsqcup} {COs}{cop}{46}

```

Delimiter symbols. DeL stands for ‘delimiter (left)’. DeR stands for ‘delimiter (right)’. DeB stands for ‘delimiter (bidirectional)’. The principal encoding point for an extensible delimiter is the first link in the list of linked sizes as specified in the font metric information. For a math encoding such as OT1/OML/OMS/OMX where not all sizes of a given delimiter reside in a given font, the extra encoding point for the smallest delimiter must be supplied by defining

```
\sd@GXX
```

where G is the mathgroup and XX is the hexadecimal glyph position. `\DeclareFlexDelimiter` does that for us.

```

674 \DeclareFlexDelimiter{\rangle}{DeR}{del}{0B}{OMS}{69}
675 \DeclareFlexDelimiter{\langle}{DeL}{del}{0A}{OMS}{68}
676 \DeclareFlexDelimiter{\rbrace}{DeR}{del}{09}{OMS}{67}
677 \DeclareFlexDelimiter{\lbrace}{DeL}{del}{08}{OMS}{66}
678 \DeclareFlexDelimiter{\rceil} {DeR}{del}{07}{OMS}{65}
679 \DeclareFlexDelimiter{\lceil} {DeL}{del}{06}{OMS}{64}
680 \DeclareFlexDelimiter{\rfloor}{DeR}{del}{05}{OMS}{63}
681 \DeclareFlexDelimiter{\lfloor}{DeL}{del}{04}{OMS}{62}
682 \DeclareFlexDelimiter{()} {DeL}{del}{00}{OT1}{28}
683 \DeclareFlexDelimiter{)} {DeR}{del}{01}{OT1}{29}
684 \DeclareFlexDelimiter{[]} {DeL}{del}{02}{OT1}{5B}
685 \DeclareFlexDelimiter{]} {DeR}{del}{03}{OT1}{5D}
686 \DeclareFlexDelimiter{\lVert} {DeL}{del}{0D}{OMS}{6B}
687 \DeclareFlexDelimiter{\rVert} {DeR}{del}{0D}{OMS}{6B}
688 \DeclareFlexDelimiter{\lvert} {DeL}{del}{0C}{OMS}{6A}
689 \DeclareFlexDelimiter{\rvert} {DeR}{del}{0C}{OMS}{6A}
690 \DeclareFlexDelimiter{\Vert} {DeB}{del}{0D}{OMS}{6B}
691 \DeclareFlexDelimiter{\vert} {DeB}{del}{0C}{OMS}{6A}

```

Maybe make the vert bars mathord instead of delimiter, to discourage poor usage.

```
692 \DeclareFlexDelimiter{|}{DeB}{del}{OC}{OMS}{6A}
693 \DeclareFlexDelimiter{/}{DeB}{del}{OE}{OML}{3D}
```

These wacky delimiters need to be supported I guess for compatibility reasons. The DeA delimiter type is a special case used only for these arrows.

```
694 \DeclareFlexDelimiter{\lmoustache} {DeL}{del}{40}{del}{7A}
695 \DeclareFlexDelimiter{\rmoustache} {DeR}{del}{41}{del}{7B}
696 \DeclareFlexDelimiter{\lgroup}      {DeL}{del}{3A}{del}{3A}
697 \DeclareFlexDelimiter{\rgroup}     {DeR}{del}{3B}{del}{3B}
698 \DeclareFlexDelimiter{\bracevert}  {DeB}{del}{3E}{del}{3E}
699 \DeclareFlexDelimiter{\arrowvert}  {DeB}{del}{3C}{OMS}{6A}
700 \DeclareFlexDelimiter{\Arrowvert}  {DeB}{del}{3D}{OMS}{6B}
701 \DeclareFlexDelimiter{\uparrow}     {DeA}{del}{78}{OMS}{22}
702 \DeclareFlexDelimiter{\downarrow}   {DeA}{del}{79}{OMS}{23}
703 \DeclareFlexDelimiter{\updownarrow}{DeA}{del}{3F}{OMS}{6C}
704 \DeclareFlexDelimiter{\Uparrow}    {DeA}{del}{7E}{OMS}{2A}
705 \DeclareFlexDelimiter{\Downarrow}  {DeA}{del}{7F}{OMS}{2B}
706 \DeclareFlexDelimiter{\Updownarrow}{DeA}{del}{77}{OMS}{6D}
707 \DeclareFlexDelimiter{\backslash}   {DeB}{del}{OF}{OMS}{6E}
```

3 Some compound symbols

The following symbols are not robust in standard L^AT_EX because they use # or \mathpalette (which is not robust and contains a # in its expansion): \angle, \cong, \notin, \rightleftharpoons.

In this definition of \hbar, the symbol is cobbled together from a math italic h and the cmr overbar accent glyph.

```
708 \DeclareFlexSymbol{\hbarOrd}{Ord}{OT1}{16}
709 \DeclareFlexCompoundSymbol{\hbar}{Ord}{\hbarOrd\mkern-9mu h}
```

For \surd, the interior symbol gets math class 1 (cumulative operator) to make the glyph vertically centered on the math axis, but the desired horizontal spacing is the spacing for a mathord. (Couldn't it just be class mathopen, though?)

```
710 \DeclareFlexSymbol{\surdOrd}{Ord}{OMS}{70}
711 \DeclareFlexCompoundSymbol{\surd}{Ord}{\mathop{\surdOrd}}
```

As shown in this definition of \angle, rule dimens are not allowed to use math-units, unfortunately.

```
712 \DeclareFlexCompoundSymbol{\angle}{Ord}{%
713   \vbox{\ialign{%
714     $m@th\scriptstyle##$\cr
715     \notRel\mathrel{\mkern14mu}\cr
716     \noalign{\nointerlineskip}%
717     \mkern2.5mu\leaders\hrule \@height.34pt\hfill\mkern2.5mu\cr
718   }}}%
719 }
```

The `\not` function, which is defined in the `flexisym` package, requires a suitably defined `\notRel` symbol.

```
720 \DeclareFlexCompoundSymbol{\neq}{Rel}{\not{=}}
.
721 \DeclareFlexCompoundSymbol{\mapsto}{Rel}{\mapstoOrd\rightarrow}
The \@vereq function ends by centering the whole construction on the math axis,
unlike \buildrel where the base symbol remains at its normal altitude. Fur-
thermore, \@vereq leaves the math style of the top symbol as given instead of
downsizing to scriptstyle.
722 \DeclareFlexCompoundSymbol{\cong}{Rel}{\mathpalette\@vereq\sim}
The \m@th in the fontmath.ltx definition of \notin is superfluous unless \c@ncel
doesn't include it (which was perhaps true in an older version of plain.tex?).
723 \providecommand*\joinord{}
724 \langle cmbase | mathptmx \rangle \renewcommand*\joinord{\mkern-3mu }
725 \langle mathpazo \rangle \renewcommand*\joinord{\mkern-3.45mu }
726 \DeclareFlexCompoundSymbol{\notin}{Rel}{\mathpalette\c@ncel\in}
727 \DeclareFlexCompoundSymbol{\rightleftharpoons}{Rel}{\mathpalette\rlh@{}}
728 \DeclareFlexCompoundSymbol{\doteq}{Rel}{\buildrel\textstyle.\over=}
729 \DeclareFlexCompoundSymbol{\hookrightarrow}{Rel}{\lhookRel\joinord\rightarrow}
730 \DeclareFlexCompoundSymbol{\hookleftarrow}{Rel}{\leftarrow\joinord\rhookRel}
731 \DeclareFlexCompoundSymbol{\bowtie}{Rel}{\triangleright\joinord\triangleleft}
732 \DeclareFlexCompoundSymbol{\models}{Rel}{\vert\joinord=}
733 \DeclareFlexCompoundSymbol{\Longrightarrow}{Rel}{\Relbar\joinord\Rightarrow}
734 \DeclareFlexCompoundSymbol{\longrightarrow}{Rel}{\relbar\joinord\rightarrow}
735 \DeclareFlexCompoundSymbol{\Longleftarrow}{Rel}{\Leftarrow\joinord\Relbar}
736 \DeclareFlexCompoundSymbol{\longleftarrow}{Rel}{\leftarrow\joinord\relbar}
737 \DeclareFlexCompoundSymbol{\longmapsto}{Rel}{\mapstochar\longrightarrow}
738 \DeclareFlexCompoundSymbol{\longleftarrowrightarrow}{Rel}{\leftarrow\joinord\rightarrow}
739 \DeclareFlexCompoundSymbol{\Longleftarrowrightarrow}{Rel}{\Leftarrow\joinord\Rightarrow}
```

Here is what you get from the old definition of `\iff`.

```
\glue 2.77771 plus 2.77771
\glue(\thickmuskip) 2.77771 plus 2.77771
\OMS/cmsy/m/n/10 (
\hbox(0.0+0.0)x-1.66663
.kern -1.66663
\OMS/cmsy/m/n/10 )
\penalty 500
\glue 2.77771 plus 2.77771
\glue(\thickmuskip) 2.77771 plus 2.77771
```

Looks like it could be simplified slightly. But it's not so easy as it looks to do it without screwing up the line breaking possibilities.

```
740 \renewcommand*\iff{%
741 \mskip\thickmuskip\Longleftarrow\mskip\thickmuskip
742 }
```

Some dotly symbols.

```

743 \DeclareFlexCompoundSymbol{\cdots}{Inn}{\cdotp\cdotp\cdotp}%
744 \DeclareFlexCompoundSymbol{\vdots}{Ord}{%
745   \vbox{\baselineskip4\p@ \lineskiplimit\z@
746     \kern6\p@\hbox{.}\hbox{.}\hbox{.}}}
747 \DeclareFlexCompoundSymbol{\ddots}{Inn}{%
748   \mkern1mu\raise7\p@
749   \vbox{\kern7\p@\hbox{.}}\mkern2mu%
750   \raise4\p@\hbox{.}\mkern2mu\raise\p@\hbox{.}\mkern1mu%
751 }

```

.

```

752 \def\relbar{\begingroup \def\smash@{tb}% in case amsmath is loaded
753   \mathpalette\mathsm@sh{\mathchar"200 }\endgroup}

```

For `\Relbar` we take an equal sign of class 0 (Ord) from the operator family. For `cmr` and `mathptmx` we know this is family 0.

```

754 <cmbase | mathptmx> \def\Relbar{\mathchar"3D }

```

For the `mathpazo` setup we need to use the equal sign from `cmr` and so must insert class 0 and use the symbol from the upright symbols.

```

755 <mathpazo> \edef\Relbar{\mathchar\string"hexnumber@\symupright3D }

```

Done.

```

756 \ExplSyntaxOff
757 </cmbase | mathpazo | mathptmx>

```

Various synonyms such as `\le` for `\leq` and `\to` for `\rightarrow` are defined in `flexisym` with `\def` instead of `\let`, for slower execution speed but smaller chance of synchronization problems.

```

758 <*msabm>
759 \ProvidesSymbols{msabm}[2001/09/08 v0.91]
760 \ExplSyntaxOn

```

```

761 \RequirePackage{amsfonts}\relax

```

```

762 \cs_gset:cpx{mg@MSA}{\hexnumber@\symAMSa}%
763 \cs_gset:cpx{mg@MSB}{\hexnumber@\symAMSb}%

764 \DeclareFlexSymbol{\boxdot}           {Bin}{MSA}{00}
765 \DeclareFlexSymbol{\boxplus}         {Bin}{MSA}{01}
766 \DeclareFlexSymbol{\boxtimes}        {Bin}{MSA}{02}
767 \DeclareFlexSymbol{\square}          {Ord}{MSA}{03}
768 \DeclareFlexSymbol{\blacksquare}     {Ord}{MSA}{04}
769 \DeclareFlexSymbol{\centerdot}       {Bin}{MSA}{05}
770 \DeclareFlexSymbol{\lozenge}         {Ord}{MSA}{06}
771 \DeclareFlexSymbol{\blacklozenge}    {Ord}{MSA}{07}
772 \DeclareFlexSymbol{\circlearrowright}{Rel}{MSA}{08}
773 \DeclareFlexSymbol{\circlearrowleft}{Rel}{MSA}{09}

```

In `amsfonts.sty`:

```

774 %\DeclareFlexSymbol{\rightleftharpoons}{Rel}{MSA}{0A}
775 \DeclareFlexSymbol{\leftrightharpoons}{Rel}{MSA}{0B}

```

776	<code>\DeclareFlexSymbol{\boxminus}</code>	<code>{Bin}{MSA}{0C}</code>
777	<code>\DeclareFlexSymbol{\Vdash}</code>	<code>{Rel}{MSA}{0D}</code>
778	<code>\DeclareFlexSymbol{\Vvdash}</code>	<code>{Rel}{MSA}{0E}</code>
779	<code>\DeclareFlexSymbol{\vDash}</code>	<code>{Rel}{MSA}{0F}</code>
780	<code>\DeclareFlexSymbol{\twoheadrightarrow}</code>	<code>{Rel}{MSA}{10}</code>
781	<code>\DeclareFlexSymbol{\twoheadleftarrow}</code>	<code>{Rel}{MSA}{11}</code>
782	<code>\DeclareFlexSymbol{\leftleftarrows}</code>	<code>{Rel}{MSA}{12}</code>
783	<code>\DeclareFlexSymbol{\rightrightarrows}</code>	<code>{Rel}{MSA}{13}</code>
784	<code>\DeclareFlexSymbol{\upuparrows}</code>	<code>{Rel}{MSA}{14}</code>
785	<code>\DeclareFlexSymbol{\downdownarrows}</code>	<code>{Rel}{MSA}{15}</code>
786	<code>\DeclareFlexSymbol{\upharpoonright}</code>	<code>{Rel}{MSA}{16}</code>
787	<code>\let\restriction\upharpoonright</code>	
788	<code>\DeclareFlexSymbol{\downharpoonright}</code>	<code>{Rel}{MSA}{17}</code>
789	<code>\DeclareFlexSymbol{\upharpoonleft}</code>	<code>{Rel}{MSA}{18}</code>
790	<code>\DeclareFlexSymbol{\downharpoonleft}</code>	<code>{Rel}{MSA}{19}</code>
791	<code>\DeclareFlexSymbol{\rightarrowtail}</code>	<code>{Rel}{MSA}{1A}</code>
792	<code>\DeclareFlexSymbol{\leftarrowtail}</code>	<code>{Rel}{MSA}{1B}</code>
793	<code>\DeclareFlexSymbol{\leftrightarrows}</code>	<code>{Rel}{MSA}{1C}</code>
794	<code>\DeclareFlexSymbol{\rightleftarrows}</code>	<code>{Rel}{MSA}{1D}</code>
795	<code>\DeclareFlexSymbol{\Lsh}</code>	<code>{Rel}{MSA}{1E}</code>
796	<code>\DeclareFlexSymbol{\Rsh}</code>	<code>{Rel}{MSA}{1F}</code>
797	<code>\DeclareFlexSymbol{\rightsquigarrow}</code>	<code>{Rel}{MSA}{20}</code>
798	<code>\DeclareFlexSymbol{\leftrightsquigarrow}</code>	<code>{Rel}{MSA}{21}</code>
799	<code>\DeclareFlexSymbol{\looparrowleft}</code>	<code>{Rel}{MSA}{22}</code>
800	<code>\DeclareFlexSymbol{\looparrowright}</code>	<code>{Rel}{MSA}{23}</code>
801	<code>\DeclareFlexSymbol{\circeq}</code>	<code>{Rel}{MSA}{24}</code>
802	<code>\DeclareFlexSymbol{\succsim}</code>	<code>{Rel}{MSA}{25}</code>
803	<code>\DeclareFlexSymbol{\gtrsim}</code>	<code>{Rel}{MSA}{26}</code>
804	<code>\DeclareFlexSymbol{\gtrapprox}</code>	<code>{Rel}{MSA}{27}</code>
805	<code>\DeclareFlexSymbol{\multimap}</code>	<code>{Rel}{MSA}{28}</code>
806	<code>\DeclareFlexSymbol{\therefore}</code>	<code>{Rel}{MSA}{29}</code>
807	<code>\DeclareFlexSymbol{\because}</code>	<code>{Rel}{MSA}{2A}</code>
808	<code>\DeclareFlexSymbol{\doteqdot}</code>	<code>{Rel}{MSA}{2B}</code>
809	<code>\let\Doteq\doteqdot</code>	
810	<code>\DeclareFlexSymbol{\triangleq}</code>	<code>{Rel}{MSA}{2C}</code>
811	<code>\DeclareFlexSymbol{\precsim}</code>	<code>{Rel}{MSA}{2D}</code>
812	<code>\DeclareFlexSymbol{\lessim}</code>	<code>{Rel}{MSA}{2E}</code>
813	<code>\DeclareFlexSymbol{\lessapprox}</code>	<code>{Rel}{MSA}{2F}</code>
814	<code>\DeclareFlexSymbol{\eqslantless}</code>	<code>{Rel}{MSA}{30}</code>
815	<code>\DeclareFlexSymbol{\eqslantgtr}</code>	<code>{Rel}{MSA}{31}</code>
816	<code>\DeclareFlexSymbol{\curlyeqprec}</code>	<code>{Rel}{MSA}{32}</code>
817	<code>\DeclareFlexSymbol{\curlyeqsucc}</code>	<code>{Rel}{MSA}{33}</code>
818	<code>\DeclareFlexSymbol{\preccurlyeq}</code>	<code>{Rel}{MSA}{34}</code>
819	<code>\DeclareFlexSymbol{\leqq}</code>	<code>{Rel}{MSA}{35}</code>
820	<code>\DeclareFlexSymbol{\leqslant}</code>	<code>{Rel}{MSA}{36}</code>
821	<code>\DeclareFlexSymbol{\lessgtr}</code>	<code>{Rel}{MSA}{37}</code>
822	<code>\DeclareFlexSymbol{\backprime}</code>	<code>{Ord}{MSA}{38}</code>
823	<code>\DeclareFlexSymbol{\risingdotseq}</code>	<code>{Rel}{MSA}{3A}</code>
824	<code>\DeclareFlexSymbol{\fallingdotseq}</code>	<code>{Rel}{MSA}{3B}</code>
825	<code>\DeclareFlexSymbol{\succcurlyeq}</code>	<code>{Rel}{MSA}{3C}</code>

```

826 \DeclareFlexSymbol{\geqq}           {Rel}{MSA}{3D}
827 \DeclareFlexSymbol{\geqslant}      {Rel}{MSA}{3E}
828 \DeclareFlexSymbol{\gtrless}       {Rel}{MSA}{3F}

in amsfonts.sty

829 %% \DeclareFlexSymbol{\sqsubset}    {Rel}{MSA}{40}
830 %% \DeclareFlexSymbol{\sqsupset}    {Rel}{MSA}{41}
831 \DeclareFlexSymbol{\vartriangleright} {Rel}{MSA}{42}
832 \DeclareFlexSymbol{\vartriangleleft} {Rel}{MSA}{43}
833 \DeclareFlexSymbol{\trianglerighteq} {Rel}{MSA}{44}
834 \DeclareFlexSymbol{\trianglelefteq} {Rel}{MSA}{45}
835 \DeclareFlexSymbol{\bigstar}        {Ord}{MSA}{46}
836 \DeclareFlexSymbol{\between}        {Rel}{MSA}{47}
837 \DeclareFlexSymbol{\blacktriangledown} {Ord}{MSA}{48}
838 \DeclareFlexSymbol{\blacktriangleright} {Rel}{MSA}{49}
839 \DeclareFlexSymbol{\blacktriangleleft} {Rel}{MSA}{4A}
840 \DeclareFlexSymbol{\vartriangle}     {Rel}{MSA}{4D}
841 \DeclareFlexSymbol{\blacktriangle}   {Ord}{MSA}{4E}
842 \DeclareFlexSymbol{\triangledown}   {Ord}{MSA}{4F}
843 \DeclareFlexSymbol{\eqcirc}         {Rel}{MSA}{50}
844 \DeclareFlexSymbol{\lesseqgtr}     {Rel}{MSA}{51}
845 \DeclareFlexSymbol{\gtreqless}     {Rel}{MSA}{52}
846 \DeclareFlexSymbol{\lesseqqgtr}    {Rel}{MSA}{53}
847 \DeclareFlexSymbol{\gtreqqless}    {Rel}{MSA}{54}
848 \DeclareFlexSymbol{\Rrightarrow}    {Rel}{MSA}{56}
849 \DeclareFlexSymbol{\Lleftarrow}    {Rel}{MSA}{57}
850 \DeclareFlexSymbol{\veebar}         {Bin}{MSA}{59}
851 \DeclareFlexSymbol{\barwedge}       {Bin}{MSA}{5A}
852 \DeclareFlexSymbol{\doublebarwedge} {Bin}{MSA}{5B}

In amsfonts.sty

853 %%\DeclareFlexSymbol{\angle}        {Ord}{MSA}{5C}
854 \DeclareFlexSymbol{\measuredangle}  {Ord}{MSA}{5D}
855 \DeclareFlexSymbol{\sphericalangle} {Ord}{MSA}{5E}
856 \DeclareFlexSymbol{\varpropto}     {Rel}{MSA}{5F}
857 \DeclareFlexSymbol{\smallsmile}    {Rel}{MSA}{60}
858 \DeclareFlexSymbol{\smallfrown}    {Rel}{MSA}{61}
859 \DeclareFlexSymbol{\Subset}        {Rel}{MSA}{62}
860 \DeclareFlexSymbol{\Supset}        {Rel}{MSA}{63}
861 \DeclareFlexSymbol{\Cup}           {Bin}{MSA}{64}
862 \let\doublecup\Cup
863 \DeclareFlexSymbol{\Cap}           {Bin}{MSA}{65}
864 \let\doublecap\Cap
865 \DeclareFlexSymbol{\curlywedge}    {Bin}{MSA}{66}
866 \DeclareFlexSymbol{\curlyvee}     {Bin}{MSA}{67}
867 \DeclareFlexSymbol{\leftthreetimes} {Bin}{MSA}{68}
868 \DeclareFlexSymbol{\rightthreetimes} {Bin}{MSA}{69}
869 \DeclareFlexSymbol{\subseteqq}     {Rel}{MSA}{6A}
870 \DeclareFlexSymbol{\supseteqq}     {Rel}{MSA}{6B}
871 \DeclareFlexSymbol{\bumpeq}       {Rel}{MSA}{6C}
872 \DeclareFlexSymbol{\Bumpeq}       {Rel}{MSA}{6D}

```


873	<code>\DeclareFlexSymbol{\l111}</code>	<code>{Rel}{MSA}{6E}</code>
874	<code>\let\l11less\l11</code>	
875	<code>\DeclareFlexSymbol{\ggg}</code>	<code>{Rel}{MSA}{6F}</code>
876	<code>\let\gggtr\ggg</code>	
877	<code>\DeclareFlexSymbol{\circledS}</code>	<code>{Ord}{MSA}{73}</code>
878	<code>\DeclareFlexSymbol{\pitchfork}</code>	<code>{Rel}{MSA}{74}</code>
879	<code>\DeclareFlexSymbol{\dotplus}</code>	<code>{Bin}{MSA}{75}</code>
880	<code>\DeclareFlexSymbol{\backsim}</code>	<code>{Rel}{MSA}{76}</code>
881	<code>\DeclareFlexSymbol{\backsimeq}</code>	<code>{Rel}{MSA}{77}</code>
882	<code>\DeclareFlexSymbol{\complement}</code>	<code>{Ord}{MSA}{7B}</code>
883	<code>\DeclareFlexSymbol{\intercal}</code>	<code>{Bin}{MSA}{7C}</code>
884	<code>\DeclareFlexSymbol{\circledcirc}</code>	<code>{Bin}{MSA}{7D}</code>
885	<code>\DeclareFlexSymbol{\circledast}</code>	<code>{Bin}{MSA}{7E}</code>
886	<code>\DeclareFlexSymbol{\circleddash}</code>	<code>{Bin}{MSA}{7F}</code>
	Begin AMSb declarations	
887	<code>\DeclareFlexSymbol{\lvertneqq}</code>	<code>{Rel}{MSB}{00}</code>
888	<code>\DeclareFlexSymbol{\gvertneqq}</code>	<code>{Rel}{MSB}{01}</code>
889	<code>\DeclareFlexSymbol{\nleq}</code>	<code>{Rel}{MSB}{02}</code>
890	<code>\DeclareFlexSymbol{\ngeq}</code>	<code>{Rel}{MSB}{03}</code>
891	<code>\DeclareFlexSymbol{\nless}</code>	<code>{Rel}{MSB}{04}</code>
892	<code>\DeclareFlexSymbol{\ngtr}</code>	<code>{Rel}{MSB}{05}</code>
893	<code>\DeclareFlexSymbol{\nprec}</code>	<code>{Rel}{MSB}{06}</code>
894	<code>\DeclareFlexSymbol{\nsucc}</code>	<code>{Rel}{MSB}{07}</code>
895	<code>\DeclareFlexSymbol{\lneqq}</code>	<code>{Rel}{MSB}{08}</code>
896	<code>\DeclareFlexSymbol{\gneqq}</code>	<code>{Rel}{MSB}{09}</code>
897	<code>\DeclareFlexSymbol{\nleqslant}</code>	<code>{Rel}{MSB}{0A}</code>
898	<code>\DeclareFlexSymbol{\ngeqslant}</code>	<code>{Rel}{MSB}{0B}</code>
899	<code>\DeclareFlexSymbol{\lneq}</code>	<code>{Rel}{MSB}{0C}</code>
900	<code>\DeclareFlexSymbol{\gneq}</code>	<code>{Rel}{MSB}{0D}</code>
901	<code>\DeclareFlexSymbol{\npreceq}</code>	<code>{Rel}{MSB}{0E}</code>
902	<code>\DeclareFlexSymbol{\nsucceq}</code>	<code>{Rel}{MSB}{0F}</code>
903	<code>\DeclareFlexSymbol{\precnsim}</code>	<code>{Rel}{MSB}{10}</code>
904	<code>\DeclareFlexSymbol{\succnsim}</code>	<code>{Rel}{MSB}{11}</code>
905	<code>\DeclareFlexSymbol{\lnsim}</code>	<code>{Rel}{MSB}{12}</code>
906	<code>\DeclareFlexSymbol{\gnsim}</code>	<code>{Rel}{MSB}{13}</code>
907	<code>\DeclareFlexSymbol{\nleqq}</code>	<code>{Rel}{MSB}{14}</code>
908	<code>\DeclareFlexSymbol{\ngeqq}</code>	<code>{Rel}{MSB}{15}</code>
909	<code>\DeclareFlexSymbol{\precneqq}</code>	<code>{Rel}{MSB}{16}</code>
910	<code>\DeclareFlexSymbol{\succneqq}</code>	<code>{Rel}{MSB}{17}</code>
911	<code>\DeclareFlexSymbol{\precnapprox}</code>	<code>{Rel}{MSB}{18}</code>
912	<code>\DeclareFlexSymbol{\succnapprox}</code>	<code>{Rel}{MSB}{19}</code>
913	<code>\DeclareFlexSymbol{\lnapprox}</code>	<code>{Rel}{MSB}{1A}</code>
914	<code>\DeclareFlexSymbol{\gnapprox}</code>	<code>{Rel}{MSB}{1B}</code>
915	<code>\DeclareFlexSymbol{\nsim}</code>	<code>{Rel}{MSB}{1C}</code>
916	<code>\DeclareFlexSymbol{\ncong}</code>	<code>{Rel}{MSB}{1D}</code>
917	<code>\DeclareFlexSymbol{\diagup}</code>	<code>{Ord}{MSB}{1E}</code>
918	<code>\DeclareFlexSymbol{\diagdown}</code>	<code>{Ord}{MSB}{1F}</code>
919	<code>\DeclareFlexSymbol{\varsubsetneq}</code>	<code>{Rel}{MSB}{20}</code>
920	<code>\DeclareFlexSymbol{\varsupsetneq}</code>	<code>{Rel}{MSB}{21}</code>

921	<code>\DeclareFlexSymbol{\nsubseteqq}</code>	<code>{Rel}{MSB}{22}</code>
922	<code>\DeclareFlexSymbol{\nsupseteqq}</code>	<code>{Rel}{MSB}{23}</code>
923	<code>\DeclareFlexSymbol{\subseteq}</code>	<code>{Rel}{MSB}{24}</code>
924	<code>\DeclareFlexSymbol{\supseteq}</code>	<code>{Rel}{MSB}{25}</code>
925	<code>\DeclareFlexSymbol{\varsubseteq}</code>	<code>{Rel}{MSB}{26}</code>
926	<code>\DeclareFlexSymbol{\varsupseteq}</code>	<code>{Rel}{MSB}{27}</code>
927	<code>\DeclareFlexSymbol{\subsetneq}</code>	<code>{Rel}{MSB}{28}</code>
928	<code>\DeclareFlexSymbol{\supsetneq}</code>	<code>{Rel}{MSB}{29}</code>
929	<code>\DeclareFlexSymbol{\nsubseteq}</code>	<code>{Rel}{MSB}{2A}</code>
930	<code>\DeclareFlexSymbol{\nsupseteq}</code>	<code>{Rel}{MSB}{2B}</code>
931	<code>\DeclareFlexSymbol{\nparallel}</code>	<code>{Rel}{MSB}{2C}</code>
932	<code>\DeclareFlexSymbol{\nmid}</code>	<code>{Rel}{MSB}{2D}</code>
933	<code>\DeclareFlexSymbol{\nshortmid}</code>	<code>{Rel}{MSB}{2E}</code>
934	<code>\DeclareFlexSymbol{\nshortparallel}</code>	<code>{Rel}{MSB}{2F}</code>
935	<code>\DeclareFlexSymbol{\nvdash}</code>	<code>{Rel}{MSB}{30}</code>
936	<code>\DeclareFlexSymbol{\nVdash}</code>	<code>{Rel}{MSB}{31}</code>
937	<code>\DeclareFlexSymbol{\nvDash}</code>	<code>{Rel}{MSB}{32}</code>
938	<code>\DeclareFlexSymbol{\nVDash}</code>	<code>{Rel}{MSB}{33}</code>
939	<code>\DeclareFlexSymbol{\ntrianglerighteq}</code>	<code>{Rel}{MSB}{34}</code>
940	<code>\DeclareFlexSymbol{\ntrianglelefteq}</code>	<code>{Rel}{MSB}{35}</code>
941	<code>\DeclareFlexSymbol{\ntriangleleft}</code>	<code>{Rel}{MSB}{36}</code>
942	<code>\DeclareFlexSymbol{\ntriangleright}</code>	<code>{Rel}{MSB}{37}</code>
943	<code>\DeclareFlexSymbol{\nleftarrow}</code>	<code>{Rel}{MSB}{38}</code>
944	<code>\DeclareFlexSymbol{\nrightarrow}</code>	<code>{Rel}{MSB}{39}</code>
945	<code>\DeclareFlexSymbol{\nLeftarrow}</code>	<code>{Rel}{MSB}{3A}</code>
946	<code>\DeclareFlexSymbol{\nRightarrow}</code>	<code>{Rel}{MSB}{3B}</code>
947	<code>\DeclareFlexSymbol{\nLeftrightarrow}</code>	<code>{Rel}{MSB}{3C}</code>
948	<code>\DeclareFlexSymbol{\nleqtrights}</code>	<code>{Rel}{MSB}{3D}</code>
949	<code>\DeclareFlexSymbol{\divideontimes}</code>	<code>{Bin}{MSB}{3E}</code>
950	<code>\DeclareFlexSymbol{\varnothing}</code>	<code>{Ord}{MSB}{3F}</code>
951	<code>\DeclareFlexSymbol{\nexists}</code>	<code>{Ord}{MSB}{40}</code>
952	<code>\DeclareFlexSymbol{\Finv}</code>	<code>{Ord}{MSB}{60}</code>
953	<code>\DeclareFlexSymbol{\Game}</code>	<code>{Ord}{MSB}{61}</code>

In amsfonts.sty:

954	<code>%%\DeclareFlexSymbol{\mho}</code>	<code>{Ord}{MSB}{66}</code>
955	<code>\DeclareFlexSymbol{\eth}</code>	<code>{Ord}{MSB}{67}</code>
956	<code>\DeclareFlexSymbol{\eqsim}</code>	<code>{Rel}{MSB}{68}</code>
957	<code>\DeclareFlexSymbol{\beth}</code>	<code>{Ord}{MSB}{69}</code>
958	<code>\DeclareFlexSymbol{\gimel}</code>	<code>{Ord}{MSB}{6A}</code>
959	<code>\DeclareFlexSymbol{\daleth}</code>	<code>{Ord}{MSB}{6B}</code>
960	<code>\DeclareFlexSymbol{\lessdot}</code>	<code>{Bin}{MSB}{6C}</code>
961	<code>\DeclareFlexSymbol{\gtrdot}</code>	<code>{Bin}{MSB}{6D}</code>
962	<code>\DeclareFlexSymbol{\ltimes}</code>	<code>{Bin}{MSB}{6E}</code>
963	<code>\DeclareFlexSymbol{\rtimes}</code>	<code>{Bin}{MSB}{6F}</code>
964	<code>\DeclareFlexSymbol{\shortmid}</code>	<code>{Rel}{MSB}{70}</code>
965	<code>\DeclareFlexSymbol{\shortparallel}</code>	<code>{Rel}{MSB}{71}</code>
966	<code>\DeclareFlexSymbol{\smallsetminus}</code>	<code>{Bin}{MSB}{72}</code>
967	<code>\DeclareFlexSymbol{\thicksim}</code>	<code>{Rel}{MSB}{73}</code>
968	<code>\DeclareFlexSymbol{\thickapprox}</code>	<code>{Rel}{MSB}{74}</code>

```

969 \DeclareFlexSymbol{\approxeq}      {Rel}{MSB}{75}
970 \DeclareFlexSymbol{\succapprox}    {Rel}{MSB}{76}
971 \DeclareFlexSymbol{\precapprox}    {Rel}{MSB}{77}
972 \DeclareFlexSymbol{\curvearrowleft}{Rel}{MSB}{78}
973 \DeclareFlexSymbol{\curvearrowright}{Rel}{MSB}{79}
974 \DeclareFlexSymbol{\digamma}       {Ord}{MSB}{7A}
975 \DeclareFlexSymbol{\varkappa}      {Ord}{MSB}{7B}
976 \DeclareFlexSymbol{\Bbbk}          {Ord}{MSB}{7C}
977 \DeclareFlexSymbol{\hslash}        {Ord}{MSB}{7D}

In amsfonts.sty:
978 %\DeclareFlexSymbol{\hbar}          {Ord}{MSB}{7E}
979 \DeclareFlexSymbol{\backepsilon}   {Rel}{MSB}{7F}
980 \ExplSyntaxOff
981 </msabm>

```

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; numbers in *roman* refer to the code lines where the entry is used.

Symbols	
\	\@symDeB 92
"	\@symDeL 90
'	\@symDeR 91
*	\@symErr 313
-	\@symOrd 83
.	\@symPun 87
/	\@symRel 86
:	\@symVar 84, 96–98
=	\@symextension 332, 335
\@begindocumenthook	\@syntype . 83–93, 96–98, 100, 265, 344
\@firstofone	\@syntype@ord 344, 346
\@gobble	\@tempb 228, 233, 237, 334, 335
\@gobblethree	\@typeset@protect 16
\@height	\@vereq 722
\@ifpackageloaded	_ 35, 38, 41, 44, 349
\@ifpackagewith	\ 404
\@ifundefined	
\@nil	A
\@onefilewithoptions	\active 225, 260
\@sym 13, 15, 94, 229, 255, 264, 329, 344	\aleph 572
\@sym@ord	\alpha 467
\@sym@ord@a	\amalg 601
\@symAcc	\angle 712, 853
\@symBin	\approx 637
\@symCOi	\approxeq 969
\@symCOs	\Arrowvert 700
\@symDeA	\arrowvert 699

<code>\ast</code>	569, 617	<code>\bracevert</code>	698
<code>\asymp</code>	655	<code>\buildrel</code>	728
<code>\AtBeginDocument</code>	371	<code>\bullet</code>	603
<code>\AtEndOfPackage</code>	6	<code>\Bumpeq</code>	872
		<code>\bumpeq</code>	871

B

<code>\backepsilon</code>	979
<code>\backprime</code>	822
<code>\backsim</code>	880
<code>\backsimeq</code>	881
<code>\backslash</code>	707
<code>\barwedge</code>	851
<code>\baselineskip</code>	745
<code>\Bbbk</code>	976
<code>\because</code>	807
<code>\begingroup</code>	227, 254, 261, 342, 348, 362, 752
<code>\beta</code>	468
<code>\beth</code>	957
<code>\between</code>	836
<code>\bigcap</code>	664
<code>\bigcirc</code>	614
<code>\bigcup</code>	665
<code>\bigodot</code>	671
<code>\bigoplus</code>	670
<code>\bigotimes</code>	669
<code>\bigsqcup</code>	673
<code>\bigstar</code>	835
<code>\bigtriangledown</code>	591
<code>\bigtriangleup</code>	590
<code>\biguplus</code>	663
<code>\bigvee</code>	661
<code>\bigwedge</code>	662
<code>\binrel@</code>	326
<code>\binrel@@</code>	305, 318–324
<code>\binrel@a</code>	317, 329
<code>\binrel@sym</code>	304, 329
<code>\blacklozenge</code>	771
<code>\blacksquare</code>	768
<code>\blacktriangle</code>	841
<code>\blacktriangledown</code>	837
<code>\blacktriangleleft</code>	839
<code>\blacktriangleright</code>	838
<code>\bot</code>	580
<code>\bowtie</code>	731
<code>\box</code>	293, 298, 301
<code>\boxdot</code>	764
<code>\boxminus</code>	776
<code>\boxplus</code>	765
<code>\boxtimes</code>	766

C

<code>\c@ncel</code>	726
<code>\Cap</code>	863, 864
<code>\cap</code>	594
<code>\catcode</code>	7, 12, 224, 225, 259, 260
<code>\cdot</code>	616
<code>\cdotOrd</code>	658, 659
<code>\cdottp</code>	659, 743
<code>\cdots</code>	743
<code>\centerdot</code>	769
<code>\char</code>	257
<code>\check@mathfonts</code>	262
<code>\chi</code>	487
<code>\circ</code>	613
<code>\circeq</code>	801
<code>\circlearrowleft</code>	773
<code>\circlearrowright</code>	772
<code>\circledast</code>	885
<code>\circledcirc</code>	884
<code>\circleddash</code>	886
<code>\circledS</code>	877
<code>\cleaders</code>	292, 296, 300
<code>\clist</code>	334
<code>\clubsuit</code>	585
<code>\colon</code>	437
<code>\complement</code>	882
<code>\cong</code>	722
<code>\coprod</code>	660
<code>\copy</code>	292, 297
<code>\crr</code>	714, 715, 717
<code>\cs</code>	34, 37, 40, 43, 50–62, 64, 67, 68, 71, 74, 78, 79, 83– 94, 96–98, 102–114, 116–143, 145–160, 162–189, 192, 194– 221, 228, 242, 244, 255, 263– 267, 270, 358, 412–422, 424, 426, 429, 431–433, 659, 762, 763
<code>\csname</code>	248
<code>\Cup</code>	861, 862
<code>\cup</code>	595
<code>\curlyeqprec</code>	816
<code>\curlyeqsucc</code>	817
<code>\curlyvee</code>	866
<code>\curlywedge</code>	865

<code>\curvearrowleft</code>	972		
<code>\curvearrowright</code>	973		
D			
<code>\dagger</code>	597		
<code>\daleth</code>	959		
<code>\dashv</code>	624		
<code>\ddagger</code>	596		
<code>\ddots</code>	747		
<code>\DeclareFlexCompoundSymbol</code>			
.....	247, 709, 711, 712,		
	720–722, 726–739, 743, 744, 747		
<code>\DeclareFlexDelimiter</code> ..	242, 674–707		
<code>\DeclareFlexSymbol</code> ..	226, 243, 434–		
	565, 567–658, 660–673, 708,		
	710, 764–786, 788–808, 810–		
	861, 863, 865–873, 875, 877–979		
<code>\DeclareOption</code>	14, 388, 390–392		
<code>\DeclareRobustCommand</code>			
.....	248, 251, 252, 281, 340, 341		
<code>\def</code> ..	15, 19–33, 226, 247, 253, 261,		
	269, 285, 291, 295, 299, 304,		
	317–324, 326, 332, 344, 346,		
	347, 350, 363, 372, 566, 752, 754		
<code>\Delta</code>	442		
<code>\delta</code>	470		
<code>\diagdown</code>	918		
<code>\diagup</code>	917		
<code>\diamond</code>	602		
<code>\diamondsuit</code>	586		
<code>\digamma</code>	974		
<code>\displaylimits</code>	47, 49		
<code>\div</code>	605		
<code>\divideontimes</code>	949		
<code>\do</code>	5, 11		
<code>\Doteq</code>	809		
<code>\doteq</code>	728		
<code>\doteqdot</code>	808, 809		
<code>\dotplus</code>	879		
<code>\doublebarwedge</code>	852		
<code>\doublecap</code>	864		
<code>\doublecup</code>	862		
<code>\Downarrow</code>	705		
<code>\downarrow</code>	702		
<code>\downdownarrows</code>	785		
<code>\downharpoonleft</code>	790		
<code>\downharpoonright</code>	788		
<code>\dp</code>	287		
		E	
<code>\edef</code>	5, 224, 259, 755		
<code>\ell</code>	498		
<code>\else</code> ..	16, 234, 273, 307–313, 354, 377, 379		
<code>\emptyset</code>	577		
<code>\endcsname</code>	248		
<code>\endgroup</code>			
	239, 257, 277, 342, 349, 369, 753		
<code>\endinput</code>	406		
<code>\epsilon</code>	471		
<code>\eqcirc</code>	843		
<code>\eqsim</code>	956		
<code>\eqslantgtr</code>	815		
<code>\eqslantless</code>	814		
<code>\equiv</code>	654		
<code>\eta</code>	473		
<code>\eth</code>	955		
<code>\everymath</code>	268		
<code>\ExecuteOptions</code>	393		
<code>\exists</code>	583		
<code>\exp</code> ..	16, 65, 69, 75, 76, 229, 230, 232,		
	244, 248, 313, 314, 335, 347, 418		
<code>\expandafter</code>	274, 368		
<code>\ExplSyntaxOff</code>	756, 980		
<code>\ExplSyntaxOn</code>	411, 760		
		F	
<code>\fallingdotseq</code>	824		
<code>\fi</code> ..	16, 238, 276, 313, 356, 381, 382		
<code>\Finv</code>	952		
<code>\flat</code>	501		
<code>\flexi@stop</code>	364, 368		
<code>\forall</code>	582		
<code>\frown</code>	508		
		G	
<code>\Game</code>	953		
<code>\Gamma</code>	441		
<code>\gamma</code>	469		
<code>\gdef</code>	318–324		
<code>\ge</code>	399		
<code>\geq</code>	399, 634		
<code>\geqq</code>	826		
<code>\geqslant</code>	827		
<code>\gets</code>	402		
<code>\gg</code>	646		
<code>\ggg</code>	875, 876		
<code>\gggtr</code>	876		
<code>\gimel</code>	958		
<code>\global</code>	101		

<code>\lmoustache</code>	694	<code>\mathchars@reset</code>	342, 344
<code>\lnapprox</code>	913	<code>\mathchoice</code>	328
<code>\lneq</code>	899	<code>\mathclose</code>	91, 97
<code>\lneqq</code>	895	<code>\mathcode</code>	235, 363, 364, 373–376, 378, 380, 383–385
<code>\lnot</code>	395	<code>\mathinner</code>	98
<code>\lnsim</code>	905	<code>\mathop</code>	88, 89, 711
<code>\long</code>	363	<code>\mathopen</code>	90, 96
<code>\Longleftarrow</code>	735	<code>\mathord</code>	83, 84, 92
<code>\longleftarrow</code>	736	<code>\mathpalette</code>	722, 726, 727, 753
<code>\Longleftrightarrow</code>	739, 741	<code>\mathpunct</code>	87, 659
<code>\longleftrightarrow</code>	738	<code>\mathrel</code>	86, 93, 715
<code>\longmapsto</code>	737	<code>\mathsm@sh</code>	753
<code>\Longrightarrow</code>	733	<code>\Mathstrutbox@</code>	287
<code>\longrightarrow</code>	734, 737	<code>\measuredangle</code>	854
<code>\looparrowleft</code>	799	<code>\mg@acc</code>	24
<code>\looparrowright</code>	800	<code>\mg@arr</code>	23
<code>\lor</code>	397	<code>\mg@Bbb</code>	30
<code>\lowercase</code>	237, 275, 349	<code>\mg@bflatin</code>	29
<code>\lozenge</code>	770	<code>\mg@bin</code>	19, 416, 431, 432
<code>\Lsh</code>	795	<code>\mg@cal</code>	31
<code>\ltimes</code>	962	<code>\mg@cop</code>	25, 433
<code>\lVert</code>	686	<code>\mg@del</code>	22, 417, 433
<code>\lvert</code>	688	<code>\mg@digit</code>	33, 418, 422
<code>\lvertneqq</code>	887	<code>\mg@frac</code>	32
M			
<code>\m@Bin</code>	52, 53, 104, 105	<code>\mg@Greek</code>	28, 422, 424, 429
<code>\m@COi</code>	56, 108	<code>\mg@greek</code>	27, 421
<code>\m@COs</code>	55, 107	<code>\mg@Latin</code>	420
<code>\m@DeA</code>	60, 112	<code>\mg@latin</code>	26, 419–421, 424, 429
<code>\m@DeB</code>	59, 111	<code>\mg@nre</code>	21
<code>\m@DeL</code>	57, 109	<code>\mg@OML</code>	419
<code>\m@DeR</code>	58, 110	<code>\mg@OMS</code>	416
<code>\m@Ord</code>	50, 102	<code>\mg@OMX</code>	417
<code>\m@Pun</code>	54, 106	<code>\mg@ord</code>	432
<code>\m@Rad</code>	61, 113	<code>\mg@rel</code>	20, 431
<code>\m@th</code>	291, 295, 299, 714	<code>\mho</code>	954
<code>\m@Var</code>	51, 103	<code>\mid</code>	623
<code>\mapsto</code>	721	<code>\mkern</code>	269, 270, 292– 294, 296–298, 300–302, 709, 715, 717, 724, 725, 748–750
<code>\mapstochar</code>	737	<code>\models</code>	732
<code>\mapstoOrd</code>	657, 721	<code>\mp</code>	611
<code>\math</code>	34, 37, 40, 43, 50–62, 64, 65, 67–69, 71, 74, 75, 78, 83–94, 96–98, 102–114, 116– 143, 145–160, 162–189, 192, 194–221, 223, 306–312, 318– 324, 340, 352, 353, 355, 358, 359	<code>\mskip</code>	270, 741
<code>\mathbin</code>	85	<code>\mu</code>	478
<code>\mathchar</code>	753–755	<code>\multimap</code>	805
<code>\mathchardef</code>	363, 364, 380	<code>\muskip</code>	269
N			
<code>\nabla</code>	578		
<code>\namelimits</code>	49		

<code>\ProvidesExplPackage</code>	3	<code>\searrow</code>	627
<code>\ProvidesFile</code>	339	<code>\setbox</code>	286, 327
<code>\ProvidesSymbols</code> ..	339, 408–410, 759	<code>\setboxz@h</code>	291, 295, 299
<code>\Psi</code>	450	<code>\setminus</code>	615
<code>\psi</code>	488	<code>\sharp</code>	503
R			
<code>\raise</code>	282, 748, 750	<code>\shortmid</code>	964
<code>\rangle</code>	674	<code>\shortparallel</code>	965
<code>\rbrace</code>	676	<code>\Sigma</code>	447
<code>\rceil</code>	678	<code>\sigma</code>	483
<code>\Re</code>	573	<code>\sim</code>	651, 722
<code>\relax</code>	8, 11, 232, 235, 236, 249, 271, 274, 345, 363, 364, 380, 385, 394, 761	<code>\simeq</code>	652
<code>\Relbar</code>	733, 735, 754, 755	<code>\smallfrown</code>	858
<code>\relbar</code> ..	291, 295, 299, 734, 736, 752	<code>\smallint</code>	589
<code>\renewcommand</code> ..	395–404, 724, 725, 740	<code>\smallsetminus</code>	966
<code>\RequirePackage</code>	2, 405, 761	<code>\smallsmile</code>	857
<code>\resetMathstrut@</code>	285	<code>\smash@</code>	752
<code>\restriction</code>	787	<code>\smile</code>	507
<code>\rfloor</code>	680	<code>\space</code>	373–376, 378, 383–385
<code>\rgroup</code>	697	<code>\spadesuit</code>	588
<code>\rho</code>	482	<code>\sphericalangle</code>	855
<code>\rhoRel</code>	568, 730	<code>\sqcap</code>	598
<code>\Rightarrow</code>	632, 733, 739	<code>\sqcup</code>	599
<code>\rightarrow</code>	294, 302, 403, 650, 721, 729, 734, 738	<code>\sqsubset</code>	829
<code>\rightarrowfill@</code>	291	<code>\sqsubseteq</code>	620
<code>\rightarrowtail</code>	791	<code>\sqsupset</code>	830
<code>\rightharpoondown</code>	512	<code>\sqsupseteq</code>	621
<code>\rightharpoonup</code>	511	<code>\square</code>	767
<code>\rightleftarrows</code>	794	<code>\star</code>	506
<code>\rightleftharpoons</code>	727, 774	<code>\std@equal</code>	364
<code>\rightrightarrow</code>	783	<code>\std@minus</code>	363, 380
<code>\rightsquigarrow</code>	797	<code>\string</code>	274, 347, 755
<code>\rightthreetimes</code>	868	<code>\Subset</code>	859
<code>\risingdotseq</code>	823	<code>\subset</code>	641
<code>\rlh@</code>	727	<code>\subseteq</code>	643
<code>\rmoustache</code>	695	<code>\subseteqq</code>	869
<code>\Rrightarrow</code>	848	<code>\subsetneq</code>	927
<code>\Rsh</code>	796	<code>\subsetneqq</code>	923
<code>\rtimes</code>	963	<code>\succ</code>	635
<code>\rVert</code>	687	<code>\succapprox</code>	970
<code>\rvert</code>	689	<code>\succcurlyeq</code>	825
S			
<code>\scriptchar</code>	252	<code>\succeq</code>	638
<code>\scriptfont</code>	252, 282	<code>\succnapprox</code>	912
<code>\scriptstyle</code>	714	<code>\succneqq</code>	910
		<code>\succnsim</code>	904
		<code>\succsim</code>	802
		<code>\sum</code>	668
		<code>\sumlimits</code>	47, 55, 89
		<code>\Supset</code>	860
		<code>\supset</code>	640
		<code>\supseteq</code>	642

	Z	
<code>\z@</code>	224, 225, 236, 259, 260, 269, 274, 286, 287, 291–	293, 295, 297–299, 301, 327, 745
		<code>\zeta</code> 472