Typesetting Math with OpTeX

Version 03, January 2021

Petr Olšák

This document is a brief summary of typesetting math. It describes TeX, Plain TeX and OpTeX features concerned to math. The first two types of features are documented in TeXbook in chapters 16, 17, and 18, but it is summarized here in short again in order to give a complete guide about math typesetting for OpTeX users.

◀ The OpTeX features which differs from standard TeX or Plain TeX are documented with the red triangle at the margin (like in this paragraph). Reader can simply distinguish between “standard” features (given by TeX or Plain TeX) and new OpTeX features.

There are more types of extensions: eTeX, luaTeX, Unicode math and OpTeX macros. The appropriate label (eTeX, LuaTeX, Unicode, OpTeX) is appended to the red triangle to inform you about the extension type. Nevertheless, OpTeX user doesn't have to worry about it, all extensions are available if Unicode Math font is loaded (e.g., by the command `\fonfam[lmfonts]`). See section 1.3.3 in OpTeX documentation about loading Unicode math fonts.

1 Basics structure of math formulas

1.1 General rules and terminology

The in-line math (in the paragraph) is created by `$`⟨*math list*⟩`$`. The display math (a standalone line between paragraphs) is created by `$$`⟨*math list*⟩`$$`. More than one line can be here if an appropriate macro is used. In-line math is processed in a TeX group in *in-line math mode*. The display math is processed in a TeX group in *display math mode*. Spaces are ignored in math modes, so `$x+y$` and `$x + y$` gives the same result: $x + y$.

The ⟨*math list*⟩ is a sequence of *math atoms* and *other material*. The math atoms are *single math objects* or composed math atoms.

- The single math object is a single character to be printed in math mode like `x`, `+`, `\int`.
- The math atom is constructed in general by `{`⟨*math list 1*⟩`}^{`⟨*math list 2*⟩`}_{`⟨*math list 3*⟩`}`. It consists from *nucleus* ⟨*math list 1*⟩, exponent ⟨*math list 2*⟩ and subscript ⟨*math list 3*⟩. Each part of the atom should be empty. If ⟨*math list 2*⟩ or ⟨*math list 3*⟩ is empty, we need not to write brackets and the prefix `^` or `_`. If the ⟨*math list 1*⟩ or ⟨*math list 2*⟩ or ⟨*math list 3*⟩ consist only from a single math object then we need not use brackets. For example `x^2` is a math atom with `x` in the nucleus, `2` in the exponent, and with empty subscript. Or `a_{i,j}` is a math atom with `a` in the nucleus, empty exponent, and `i,j` in the subscript.[1] The constructors for exponent `^` and subscript `_` can be used in arbitrary order after the nucleus, for example, `z_1^{x+y}` is the same math atom as `z^{x+y}_1`. The single math objects not followed by `^` nor `_` are considered as math atoms with this object in the nucleus and with empty exponent and subscript (this is a very common case). TeX assigns the *class* for each math atom, see section 1.2.
- Other material can be TeX box or glue (space) or `\kern` or `\vrule` etc.

Example: The `Z = \int_\Omega x^{2y} + z\, dx` generates $Z = \int_\Omega x^{2y} + z\,dx$ and it is ⟨*math list*⟩ which consists from:

- `Z` is math atom with empty exponent and subscript, class: Ord,
- `=` is math atom with empty exponent and subscript, class: Rel,
- `\int_\Omega` is math atom with empty exponent and with subscript `\Omega`, class: Op,
- `x^{2y}` is math atom with exponent `2y` and empty subscript, class: Ord,
  - `2` is math atom with empty exponent and subscript, class: Ord,
  - `y` is math atom with empty exponent and subscript, class: Ord,
- `+` is math atom with empty exponent and subscript, class: Bin,
- `z` is math atom with empty exponent and subscript, class: Ord,
- `\,` is another material, the glue (space) in this case,
- `d` is math atom with empty exponent and subscript, class: Ord,
- `x` is math atom with empty exponent and subscript, class: Ord.

1.2 Classes of math atoms

TeX assigns *a class* for each math atom.[2] This data type is used when TeX decides about horizontal spaces between atoms in the output. (Note that spaces in the input are ignored.) For example, `$xy$` prints two atoms without space between them but `$x+y$` is printed with small spaces around the `+` binary operator. Compare: $xy$ and $x + y$.

The class is assigned depending on the nucleus of the atom. If the nucleus is not a single math object, i.e. it is constructed by `{`⟨*math list*⟩`}` with braces then the atom has its class Ord. If the nucleus is a single math object constructed without braces then the class of the atom

[1] In OpTeX, the character `_` can be interpreted as a part of the control sequence name, not as the subscript constructor. But in common cases, constructions of math atoms are interpreted exactly as in plain TeX. See sections 2.2.2 and 2.15 of OpTeX documentation for more details. If you want to be sure that `_` is just a subscript constructor in OpTeX then you can set ``\catcode`\_=8`` but after this, you cannot use control sequences with `_` character.

[2] Using terminology of TeXbook, each single math object has its *class* but the math atom has its *kind* derived from this class. I use only one word for both meanings in this document.

depends on this single math object. Each single math object must be declared in TEX with its default class. The following table lists the classes with typical examples. The full set of all math objects used in math typesetting is listed in section 1.10 with their default classes.

	Class	Meaning	Example
0	Ord	ordinary object	variables, digits, $x, \mathbb{R}, \Gamma, 0, 1$
1	Op	big opertator	$\sum, \int, \bigcup$
2	Bin	binary operator	$+, \times, -, \pm, \cup$
3	Rel	reations	$=, \neq, \leq, \supseteq, \succsim$
4	Open	opening bracket	$\{, (, [, \langle$
5	Close	closing bracket	$\},),], \rangle$
6	Punct	punctuation	comma
	Inner	left-right	`\left...\right` outputs, see section 1.5

There are three space types used by the algorithm for horizontal spacing in the math formulas.

- Thin space: `\thinmuskip` primitive register, `\,` macro. Used around Op atoms.
- Medium space: `\medmuskip` primitive register, `\>` macro. Used around Bin atoms.
- Thick space: `\thickmusip` primitive register, `\;` macro. Used around Rel atoms.

		Right atom							
		Ord	Op	Bin	Rel	Open	Close	Punct	Inner
Left atom	Ord	0	1	2	3	0	0	0	1
	Op	1	1		3	0	0	0	1
	Bin	2	2			2			2
	Rel	3	3		0	3	0	0	3
	Open	0	0		0	0	0	0	0
	Close	0	1	2	3	0	0	0	1
	Punct	1	1		1	1	1	1	1
	Inner	1	1	2	3	1	0	1	1

Ord atoms are printed without spaces between them. The spaces are not cumulated, so the rule about spaces mentioned above is only a rough idea. The exact rule for horizontal spaces is given for each pairs of atoms in the table here. The symbol 0 means no space, 1 thin space, 2 medium space, and 3 means thick space.[3]

The Bin atom is automatically transformed to the Ord atom if no atom precedes or if Op, Bin, Rel, Open, or Punct atom precedes. And it is transformed to the Ord atom if Rel, Close or Punct atom follows. This corresponds to the empty cells in the table. Why such behavior? Compare "$0-3$" and "-3". The Bin atom in the second case behaves like Ord atom because it is *unary minus*. There is no space between the unary minus and the following object.

All medium spaces and thick spaces and some thin spaces from this table are omitted if the ⟨*math list*⟩ is processed in script or scriptscript styles (smaller size). See section 1.3 about math styles.

You can overwrite the default class derived from the nucleus of the atom by TEX primitives `\mathord`, `\mathop`, `\mathbin`, `\mathrel`, `\mathopen`, `\mathclose`, `\mathpunct` and `\mathinner`. They can precede a nucleus of the atom and they set the class of the atom. For example, `x \mathrel+ y` behaves like `x = y` from a spacing point of view but + is printed. Another example: `\mathop{\rm lim} z` creates the atom `lim` in roman font of class Op. So, the thin space is inserted between lim and z.

There are more special kinds of math atoms: fractions, math accents, radicals. They are constructed in a special way (see next sections) but they behave like Ord atom in the horizontal spacing algorithm.

1.3 Math styles

When a formula (or a sub-formula) is processed by TEX then one from four styles is active: display style (D), text style (T), script style (S) or scriptscript style (SS).

[3] The table presented here is built into TEX. On the other hand, LuaTEX allows to set arbitrary spaces between arbitrary pairs, see section 7.5.2 in the LuaTEX manual.

The T style is started in in-line math mode `$...$` and the D style is started in display math mode `$$...$$`. The first level of exponents or subscripts is processed in S style and the second and more levels of exponents or indexes are processed in SS style. There are special rules for math styles when fractions are constructed, see section 1.4.

The D and T styles use basic font size, S uses smaller font size (typically 70 %) and SS style uses more smaller font size (typically 50 %). Next levels of "more smaller fonts" are not used due to classical typographic rules.

The nucleus of Op atoms (big operators, $\sum$, $\int$, etc.) have typically bigger versions of the character shape for D style than for T style. So, there are four sizes for such math objects: one size for each math style. All other math objects (with non Op class) are printed only in three sizes: The sizes for T and D styles are equal.

The Op atom puts its exponent and subscript above and below the nucleus in D style but right to the nucleus in other styles:

`\sum_{i=1}^\infty` gives $\displaystyle\sum_{i=1}^\infty$ in D style and $\textstyle\sum_{i=1}^\infty$ in T style.

This default behavior of the Op atom can be modified by placing `\limits` or `\nolimits` or `\displaylimits` TeX primitive just after its nucleus before the constructors of exponent and/or index. The `\nolimits` puts exponent and subscript right to the nucleus (regardless of the current style) and `\limits` puts these objects above and below the nucleus (regardless of the current style). There can be more such primitives in a queue (due to a macro expansion, for instance). Then the last primitive in the queue wins. If the last primitive is `\displaylimits` then the default behavior is processed regardless there are `\limits` or `\nolimits` before it.

`\sum\nolimits_{i=1}^\infty` gives $\displaystyle\sum\nolimits_{i=1}^\infty$ in D style and $\textstyle\sum\nolimits_{i=1}^\infty$ in T style.

Atoms of all other classes have their exponents and/or subscripts only right to their nucleus without any exception.

The primitives `\displaystyle`, `\textstyle`, `\scriptstyle` and `\scriptscriptstyle` set the given style regardless the default rules. For example, you can create a formula in in-line math mode and in D style by `$\displaystyle` ⟨*formula*⟩`$` or a formula in display mode and T style can be printed by `$$\textstyle` ⟨*formula*⟩`$$`.

If a subformula is placed below something (below a line from root symbol, below a fraction line), then the processed style D, T, S or SS is *cramped*. The exponents are positioned slightly lower than in non-cramped style. The selectors `\displaystyle`...`\scriptscriptstyle` mentioned above select non-cramped style. The non-cramped style is selected when math mode starts too. You can select a cramped style by the macro `\cramped` at the start of the math ◀ OpTeX
formula or after the math-style selectors: `\scriptstyle\cramped` for example.

Several macros need to know what math style is currently processed (for example they need to draw something in an appropriate size). But it not possible simply due to the syntax of fractions (section 1.4). This syntax requires to process all math lists in two steps: the first step expands all macros and creates structured data of processed math list. The second step reads the output of the first step, switches between math styles and creates definitive output. So, macros (working in the first step) cannot know the current math style because it is set only in the second step. TeX supports the primitive `\matchchioce{`⟨*D*⟩`}{`⟨*T*⟩`}{`⟨*S*⟩`}{`⟨*SS*⟩`}` which prepares four math lists in the first step and only one of these four lists are used in the second step. We can put different macros into each of the four parameters of `\mathchoice`. Plain TeX supports the macro `\mathpalette` which gives a more comfortable interface of `\mathchoice` to the macro programmer. The cramped/non-cramped variants of the current style are kept when `\mathchioce` is used.

We describe another interface for creating macros depending on the current style. You can use ◀ OpTeX
`\mathstyles{`⟨*math list*⟩`}`. It behaves like `{`⟨*math list*⟩`}`, moreover, you can use the following commands inside such ⟨*math list*⟩:

- The macro `\currstyle`. It expands to `\displaystyle`, `\textstyle`, `\scriptstyle` or `\scriptscriptstyle` depending on the current math style when the `\mathstyles` was opened.
- The `\dobystyle{`⟨*D*⟩`}{`⟨*T*⟩`}{`⟨*S*⟩`}{`⟨*SS*⟩`}` is expandable macro. It expands its parameter ⟨*D*⟩, ⟨*T*⟩, ⟨*S*⟩ or ⟨*SS*⟩ depending on the current math style when `\mathstyles` was opened.
- The value of the `\stylenum` register is 0, 1, 2 or 3 depending on the current math style when `\mathstyles` was opened.

Example of usage of `\mathstyles`:

```
\def\mysymbol{\mathbin\mathstyles
   {\kern1pt\vrule height\mysymbolA width\mysymbolA\kern1pt}}
\def\mysymbolA{\dobystyle{5pt}{5pt}{3.5pt}{2.5pt}}
Test: $a\mysymbol b_{c \mysymbol d}$ or $a\mysymbol b\over c$.
```

This example gives Test: $a \blacksquare b_{c \blacksquare d}$ or $\frac{a \blacksquare b}{c}$.

The `\mathstyles` macro mentioned above uses TEX primitive `\mathchoice`, so it creates four math lists and only one is used. It may take more computer time in special cases. LuaTEX supports the `\mathstyle` primitive (no "`s`" at the end of this control sequence) which expands to values 0 to 7 depending on the current style: $D, D', T, T', S, S', SS, SS'$ (where X' means cramped variant of the style). This primitive does not use `\mathchoice` but it simply ignores the fraction syntax, so `$a\mysymbol b\over c$` cannot work if `\mysymbol` is defined using the `\mathstyle` primitive. See section 7.3.1 of LuaTEX documentation for more information.

◀ LuaTEX

1.4 Fractions

The fraction can be constructed by `{`⟨*numerator*⟩`\over`⟨*denominator*⟩`}`. If the fraction is only a single object in the whole math mode (between dollars), you need not use the outer braces, so you can write `$1\over2$` to get $\frac{1}{2}$.

The ⟨*numerator*⟩ and ⟨*denominator*⟩ are printed in "smaller" math style than current math style. More exactly the following schema is used. D: $\frac{T}{T}$.., T: $\frac{S}{S}$, S: $\frac{SS}{SS}$, SS: $\frac{SS}{SS}$. For example

`{a+b \over c}` is printed as $\dfrac{a+b}{c}$ in D style and as $\tfrac{a+b}{c}$ in T style.

The LATEX macro `\frac{`⟨*numerator*⟩`}{`⟨*denominator*⟩`}` is not supported in Plain TEX nor in OpTEX but you can define such macro if you want.

The syntax with `\over` is more preferred because it is more human-readable notation. You can write the fraction in the same manner as you can read it. You can compare: `$1\over2$` (one over two) with `$\frac12$` (frac twelve).

Besides the `\over` primitive, there are analogical TEX primitives which create "generalized" fractions. The result is similar to `{`⟨*above*⟩`\over `⟨*below*⟩`}` but there is something extra:

- `{`⟨*above*⟩`\atop`⟨*below*⟩`}` does `{`⟨*above*⟩`\over`⟨*below*⟩`}` but without the fractional rule.
- `{`⟨*above*⟩`\above`⟨*dimen*⟩⟨*below*⟩`}` creates fractional rule with ⟨*dimen*⟩ thickness.
- `{`⟨*above*⟩`\overwithdelims`⟨*delim-l*⟩⟨*delim-r*⟩⟨*below*⟩`}` adds the ⟨*delim-l*⟩ left to the fraction and the ⟨*delim-r*⟩ right to the fraction.
- `{`⟨*above*⟩`\atopwithdelims`⟨*delim-l*⟩⟨*delim-r*⟩⟨*below*⟩`}` is analogical to `\overwithdelims` but without fractional rule.
- `{`⟨*above*⟩`\abovewithdelims`⟨*delim-l*⟩⟨*delim-r*⟩⟨*dimen*⟩⟨*below*⟩`}` behaves as `\overwithdelims` but the fractional rule has ⟨*dimen*⟩ thickness.

The `\...withdelims` variants read ⟨*delim-l*⟩ and ⟨*delim-r*⟩, they must be declared as *math delimiter* in TEX. They are vertically scalable math objects, typically brackets. See section 1.5 for more information about math delimiters. Example:

`{n \atopwithdelims() k}` creates $\dbinom{n}{k}$ in D style and $\tbinom{n}{k}$ in T style.

The `\choose` macro is defined by `\def\choose{\atopwithdelims()}`, so the user can write `{n\choose k}` in order to get binomial coefficients.

1.5 Vertically scalable objects: math delimiters

The vertically scalable objects are called *delimiters*. For example, all types of brackets are declared as delimiters. This means that you can use a bracket in arbitrary vertical size.[4]

The following objects are declared as delimiters (i.e. vertically scalable):

<table>
<tr><td>source:</td><td><code>(</code></td><td><code>)</code></td><td><code>[</code></td><td><code>]</code></td><td><code>\{</code></td><td><code>\}</code></td><td><code>/</code></td><td><code>\backslash</code></td><td><code>\langle</code></td><td><code>\rangle</code></td><td><code>|</code></td><td><code>\|</code></td></tr>
<tr><td>output:</td><td>(</td><td>)</td><td>[</td><td>]</td><td>{</td><td>}</td><td>/</td><td>\</td><td>⟨</td><td>⟩</td><td>|</td><td>‖</td></tr>
</table>

source:	`\lfloor`	`\rfloor`	`\lceil`	`\rceil`
output:	⌊	⌋	⌈	⌉

source:	`\uparrow`	`\Uparrow`	`\dowarrow`	`\Downarrow`	`\updownarrow`	`\Updownarrow`
output:	↑	⇑	↓	⇓	↕	⇕

◀ Unicode

If you can produce the characters ⟨, ⟩,[5] ⌊, ⌋, ... ↕, ⇕ directly in your text editor then you can use these Unicode characters in your source instead of control sequences `\langle`, `\rangle`, `\lfloor`, `\rfloor` ... `\updownarrow`, `\Updownarrow`. For many users (including me), there is more simple to type `\lfloor` than to find how to create the ⌊ character in my text editor. Note that there exist text editors (Emacs, for example) enabling you to type `\lfloor` and this control sequence is immediately converted to the ⌊ Unicode character. Your source text looks pretty and you can use classical TEX sequences.

◀ Unicode

There are more delimiters, but it heavily depends on loaded Unicode Math font. For example, this document is printed in `latinmodern-math` font and there are six more delimiters `\lBrack` ⟦, `\rBrack` ⟧, `\lAngle` ⟪, `\rAngle` ⟫, `\lgroup` ⟮, `\rgroup` ⟯. See section 1.10 for table of all Unicode symbols for math typesetting.

Arbitrary tall formula can be surrounded by a pair of delimiters using `\left` and `\right` TEX primitives: `\left`⟨*delim*⟩ ⟨*formula*⟩ `\right`⟨*delim*⟩. The delimiters are scaled to the height and depth of the ⟨*formula*⟩ and vertically centered to the *math axis*.[6] Example:

`+ \left\{ \sum _{i=1}^\infty x_i \right)` gives $$+\left\{\sum_{i=1}^{\infty} x_i\right).$$

The pair `\left`⟨*delim*⟩ ⟨*formula*⟩ `\right`⟨*delim*⟩ creates the formula in a TEX group. Such group can be nested with another groups. Each `\left` must have its `\right` counterpart at the same group level. If you don't want to create visible delimiter, use dot instead ⟨*delim*⟩. Example:

`\left. \int_0^t e^{x^2}\,dx\, \right|_{t=42}` gives $$\left.\int_0^t e^{x^2}\,dx\,\right|_{t=42}$$

◀ eTEX

You can use `\middle`⟨*delim*⟩ inside the ⟨*formula*⟩ which is surrounded by `\left...\right`. Then the given ⟨*delim*⟩ is scaled to the same size like their `\left` and `\right` counterparts.

When a delimiter is used without `\left` nor `\right` prefix, then it is the Open, Close, Ord or Bin atom by its natural meaning: $(, [, \{, \ldots, \lfloor, \lceil$ are Open atoms, $),], \}, \ldots, \rfloor, \rceil$ are Close atoms, $/, \backslash, |, \|$ are Ord atoms and $\uparrow, \Uparrow, \ldots, \Updownarrow$ are Bin atoms. You can overwrite this default setting, for example `\mathclose(`. If delimiters are used with `\left` and `\right` prefixes then `\left`⟨*delim*⟩ behaves like Open atom, `\right`⟨*delim*⟩ behaves like Close atom and

[4] This is not exactly true, because traditional typography says that they cannot be scaled continuously but by visible steps. This means that there is a sequence of increasing brackets in the font, the reader must see a difference between every two sizes of brackets.

[5] Do not confuse $<, >$ and $\langle, \rangle$. The first pair are Rel atoms with meaning "less than" or "greater than", but the second pair are special types of brackets. They are not directly available at computer keyboards without using a keyboard macro.

[6] Math axis is a horizontal line passing through the center of symbols $+$ and $-$. All vertically scalable objects are vertically centered with respect to this axis.

the math list `\left`⟨*delim*⟩⟨*formula*⟩`\right`⟨*delim*⟩ is encapsulated as a single Inner atom. The `\middle`⟨*delim*⟩ behaves like Open atom at its left side and like Close atom at its right side.

The sequence of increasing delimiters can be printed by the following macros:

`(` $\to ($, `\big(` $\to \big($, `\bbig(` $\to \Big($[7], `\Big(` $\to \bigg($, `\bigg(` $\to \Bigg($, `\Bigg(` $\to \Bigg($.

The `\Bigg`⟨*delim*⟩ is not the maximal size of the bracket. Try `\left(\vbox to5cm{}\right.`, for example. You can see that the font "cheats" from certain sizes, because there are not all infinity number of sizes of brackets drawn in the font, of course.

The `\big`⟨*delim*⟩ creates Ord atom. We need to create Open atom for opening bracket and Close atom for closing bracket more often. Then we can use macros `\bigl`⟨*delim*⟩, `\bbigl`⟨*delim*⟩, `\Bigl`⟨*delim*⟩, `\biggl`⟨*delim*⟩, `\Biggl`⟨*delim*⟩ for creating Open atoms and `\bigr`⟨*delim*⟩, `\bbigr`⟨*delim*⟩, `\Bigr`⟨*delim*⟩, `\biggr`⟨*delim*⟩, `\Biggr`⟨*delim*⟩ for creating Close atoms. Unfortunately, the source is not too attractive when more sizes of brackets are used, but typographic traditions say that we have to distinguish brackets by the size in math mode if they are in equal types:

`\bbigl( f\bigl( 2(x+y) + z\bigr) \bbigr)'` gives $\Bigl(f\bigl(2(x+y)+z\bigr)\Bigr)'$.

The `math.opm` package from OpTEX provides doing this more comfortable.

1.6 Horizontally scalable objects: math accents

Arbitrary wide formula can be covered by *scalable math accent*. Example:

`\overrightarrow {a+b+c+d+e+f}` gives $\overrightarrow{a+b+c+d+e+f}$.

The usage is: control sequence of selected math accent followed by `{`⟨*math list*⟩`}`.

Standard scalable math accents are: `\overline` $\overline{abc}$, `\overbrace` $\overbrace{abc}$, `\overrightarrow` $\overrightarrow{abc}$, `\overleftarrow` $\overleftarrow{abc}$, `\underline` $\underline{abc}$, `\underbrace` $\underbrace{abc}$.

An Op atom is created. The exponents and subscripts are centered above and below the nucleus of this atom (regardless of the current style). Example:

`\overbrace {b\cdot b\cdot b \cdots b}^{k\times}` gives $\overbrace{b\cdot b\cdot b\cdots b}^{k\times}$

There are scalable accents with a limited maximum width: `\widehat` $\widehat{abc}$ and `\widetilde` $\widetilde{abc}$. If the formula is wider than the font can cover then the widest variant from the font is used and it is horizontally centered.

There are more scalable accents in Unicode math fonts: `\overparen` $\overparen{abc}$, `\underparen` $\underparen{abc}$, ◀ Unicode
`\overbracket` $\overbracket{abc}$, `\underbracket` $\underbracket{abc}$, `\overleftrightarrow` $\overleftrightarrow{abc}$, `\overleftharpoon` $\overleftharpoon{abc}$, `\overrightharpoon` $\overrightharpoon{abc}$, `\wideoverbar` $\wideoverbar{abc}$, `\widebreve` $\widebreve{abc}$, `\widecheck` $\widecheck{abc}$, `\wideutilde` $\wideutilde{abc}$, `\mathunderbar` $\mathunderbar{abc}$, `\underleftrightarrow` $\underleftrightarrow{abc}$, `\widebridgeabove` $\widebridgeabove{abc}$, `\underrightharpoondown` $\underrightharpoondown{abc}$, `\underleftharpoondown` $\underleftharpoondown{abc}$, `\underleftarrow` $\underleftarrow{abc}$, `\underrightarrow` $\underrightarrow{abc}$.

1.7 Fixed math accents

Fixed math accents can be applied to single math object or to the `{`⟨*math list*⟩`}`. The accent is centered (with respect of slanting axis) and the result is a nucleus of Ord atom. For example `\dot x` gives $\dot x$. The list of fixed math accents follows: `\acute x` $\acute x$, `\bar x` $\bar x$, `\breve x` $\breve x$, `\check x` $\check x$, `\dot x` $\dot x$, `\ddot x` $\ddot x$, `\grave x` $\grave x$, `\hat x` $\hat x$, `\vec x` $\vec x$, `\tilde x` $\tilde x$.

The additional fixed accents depends on used Unicode math font. The `latinmodern-math` ◀ Unicode
supports: `\ovhook x` $\ovhook{x}$, `\ocirc x` $\ocirc{x}$, `\leftharpoonaccent x` $\leftharpoonaccent{x}$, `\rightharpoonaccent x` $\rightharpoonaccent{x}$, `\dddot x` $\dddot x$, `\ddddot x` $\ddddot x$, `\widebridgeabove x` $\widebridgeabove{x}$, `\asteraccent x` $\asteraccent{x}$.

There exist one special math accent `'` (single quote, ASCII 39) which can be appended after a symbol like this: `f'` and it creates f' (typical meaning is the derivation of the given function). You can put more such accents, for example `g'''` gives g'''.

[7] Provided only in OpTEX.

1.8 Roots

There is a macro `\sqrt{`⟨*math list*⟩`}` to create square root. For example:

$$\texttt{\textbackslash sqrt\{\textbackslash sqrt\{\textbackslash sqrt\{x+1\}+1\}+1\}} \quad \text{gives} \quad \sqrt{\sqrt{\sqrt{x+1}+1}+1}$$

The n-th root is created by the macro `\root` ⟨n⟩`\of{`⟨*math list*⟩`}`. For example `\root k+1\of x` gives $\root k+1 \of x$.

1.9 Math alphabets

Letters $a \dots z$, $A \dots Z$ and $\alpha \dots \omega$ are printed in italic in math mode. This follows the traditional typographic rule. All other math symbols, digits, and uppercase Greek letters must be upright.[8] These rules are independent of the current variant of surrounding text font.

If we want to use the letters or digits in another than this default shape, then we can use *math alphabet selectors*: `\mit`, `\rm`, `\it`, `\bf`, `\cal`. OpTEX supports more such selectors `\script`, `\frak`, `\bbchar`, `\bi`, see section 1.3.3 in the OpTEX documentation. The math selectors have local validity in the group. ◀ OpTEX

The control sequences `\rm`, `\it`, `\bf`, and `\bi` act as variant selectors of fonts in non-math mode (text mode) and they act as math alphabet selectors in math mode. This "overlaying" concept is given by Plain TEX. Example: math operators lim, sin, cos, log, etc. must be printed unslanted. We are using `\lim`, `\sin`, `\cos`, `\log` etc. in math mode in order to comply with this typographic convention. For example `\sin` is defined as:

```
\def\sin {\mathop{\rm sin}\nolimits}
```

The `\rm` is used here as math alphabet selector, no variant selector of text fonts.

The list of all predefined `\rm`-like math operators follows: `\arccos`, `\arcsin`, `\arctan`, `\arg`, `\cos`, `\cosh`, `\cot`, `\coth`, `\deg`. `\det`, `\dim`, `\exp`, `\gcd`, `\hom`, `\inf`, `\ker`, `\lg`, `\lim`, `\liminf`, `\limsup`, `\ln`, `\log`, `\max`, `\min`, `\Pr`, `\scs`, `\sin`, `\sinh`, `\sup`, `\tan`, `\tanh`, You can define another such operator analogically.

Unicode font can include the following math alphabets: ◀ Unicode

```
\_rmvariables      % upright letters A-Z, a-z
\_bfvariables      % bold letters A-Z, a-z
\_itvariables      % italic letters A-Z, a-z
\_bivariables      % bold italic letters A-Z, a-z
\_calvariables     % calligraphic letters A-Z, a-z
\_bcalvariables    % calligraphic letters A-Z, a-z
\_frakvariables    % fraktur A-Z, a-z
\_bfrakvariables   % bold fraktur A-Z, a-z
\_sansvaraibales   % sans serif letters A-Z, a-z
\_bsansvaraibales  % bold sans serif letters A-Z, a-z
\_isansvaraibales  % slanted sans serif letters A-Z, a-z
\_bisansvaraibales % bold slanted sans serif letters A-Z, a-z
\_ttvariables      % monospace, typewriter letters A-Z, a-z
\_bbvariables      % double struck A-Z, a-z
\_rmdigits         % upright digits 0..9
\_bfdigits         % bold digits 0..9
\_sansdigits       % sans serif digits 0..9
\_bsansdigits      % bold sans serif digits 0..9
\_ttdigits         % monospace typewriter digits 0..9
\_bbdigits         % double-struck digits 0..9
\_rmgreek          % upright Greek letters \alpha-\omega
\_itgreek          % slanted Greek letters \alpha-\omega
\_bfgreek          % bold Greek letters \alpha-\omega
\_bigreek          % bold italic Greek letters \alpha-\omega
```

[8] French typographic convention says that uppercase Greek letters have to be in italic too. Use `\_itGreek` declaration in this case.

```
\_bsansgreek       % bold sans serif Greek letters \alpha-\omega
\_bisansgreek      % bold slanted snas serif Greek letters \alpha-\omega
\_itGreek          % slanted Greek letters \Alpha-\Omega
\_bfGreek          % bold Greek letters \Alpha-\Omega
\_biGreek          % bold italic Greek letters \Alpha-\Omega
\_bsansGreek       % bold sans serif Greek letters \Alpha-\Omega
\_bisansGreek      % bold slanted snas serif Greek letters \Alpha-\Omega
```

Not all Unicode math fonts include all math alphabets listed here. Typically, the lowercase letters of calligraphic shape and all letters of bold calligraphic shape are missing.

OpTEX defines internal math alphabet selectors as mentioned in the previous listing of math alphabets and sets as default:

◀ OpTEX

```
\_itvariables \_rmdigits \_itgreek \_rmGreek
```

Moreover, it defines the alphabet selectors at user level (see section 1.3.3 of the OpTEX manual). For example

```
\_def\_marm {\_inmath{\_rmavariables \_rmdigits}} % \_marm is used in \rm
```

`\_inmath` runs its parameter only in math mode and sets the math alphabets. You can see the file `unimath-codes.opm` where all these selectors are defined. You can redefine them. For example, OpTEX defines `\bf` as a math alphabet selector that selects sans serif bold in math. This is the common notation for vectors, tensors, and matrices. If you dislike this (maybe because plain TEX has different defaults), then you can define:

```
\_def\_mabf {\_inmath{\_bfvariables\_bfdigits\_bfgreek\_bfGreek}} % used in \bf
```

1.10 List of single math objects

All single math objects are listed in the `unimath-table.opm` or `unicode-math-table.tex` file. You can look into this file. The codes, TEX sequences, classes, and comments for all possible math codes are here. Maybe, your Unicode math font which is loaded does not support all these codes. You can try all codes of the currently loaded font by

◀ Unicode

◀ OpTEX

```
\input print-unimath.opm
```

The `unimath-table` is printed with characters available in the loaded font. If the character is unsupported by the font then the slot is empty and only TEX sequence and the class of the code is printed in the table. For example, this document loads `latimodern-math.otf` font. And the result from `\input print-unimath.opm` looks like the following ten pages.

◀ OpTEX

Unsupported characters can be replaced by characters from other Unicode math font, see section 5.3.

Codes U+00000 … U+10000

!	Close	\mathexclam
#	Ord	\mathoctothorpe
$	Ord	\mathdollar
%	Ord	\mathpercent
&	Ord	\mathampersand
(	Open	\lparen
)	Close	\rparen
+	Bin	\mathplus
,	Punct	\mathcomma
.	Ord	\mathperiod
/	Ord	\mathslash
:	Punct	\mathcolon
;	Punct	\mathsemicolon
<	Rel	\less
=	Rel	\equal
>	Rel	\greater
?	Close	\mathquestion
@	Ord	\mathatsign
[	Open	\lbrack
\	Ord	\backslash
]	Close	\rbrack
{	Open	\lbrace
|	Fence	\vert
}	Close	\rbrace
£	Ord	\mathsterling
¥	Ord	\mathyen
§	Ord	\mathsection
¬	Ord	\neg=\lnot
±	Bin	\pm
¶	Ord	\mathparagraph
·	Bin	\cdotp
×	Bin	\times
ð	Alph	\matheth=\eth
÷	Bin	\div
	Ord	\Zbar
`	Acc	\grave
´	Acc	\acute
ˆ	Acc	\hat
ˆ	Accw	\widehat
˜	Acc	\tilde
˜	Accw	\widetilde
¯	Acc	\bar
¯	Acc	\overbar
¯	Accw	\wideoverbar
˘	Acc	\breve
˘	Accw	\widebreve
˙	Acc	\dot
¨	Acc	\ddot
̉	Acc	\ovhook
˚	Acc	\ocirc=\mathring
ˇ	Acc	\check
ˇ	Accw	\widecheck
	Acc	\candra
	Acc	\oturnedcomma
	Acc	\ocommatopright
	Acc	\droang
˜	AccBw	\wideutilde
_	AccBw	\mathunderbar
/	AccO	\notaccent
↔	AccBw	\underleftrightarrow

Α	Alph	\mupAlpha
Β	Alph	\mupBeta
Γ	Alph	\mupGamma
Δ	Alph	\mupDelta
Ε	Alph	\mupEpsilon
Ζ	Alph	\mupZeta
Η	Alph	\mupEta
Θ	Alph	\mupTheta
Ι	Alph	\mupIota
Κ	Alph	\mupKappa
Λ	Alph	\mupLambda
Μ	Alph	\mupMu
Ν	Alph	\mupNu
Ξ	Alph	\mupXi
Ο	Alph	\mupOmicron
Π	Alph	\mupPi
Ρ	Alph	\mupRho
Σ	Alph	\mupSigma
Τ	Alph	\mupTau
Υ	Alph	\mupUpsilon
Φ	Alph	\mupPhi
Χ	Alph	\mupChi
Ψ	Alph	\mupPsi
Ω	Alph	\mupOmega
α	Alph	\mupalpha
β	Alph	\mupbeta
γ	Alph	\mupgamma
δ	Alph	\mupdelta
ε	Alph	\mupvarepsilon
ζ	Alph	\mupzeta
η	Alph	\mupeta
θ	Alph	\muptheta
ι	Alph	\mupiota
κ	Alph	\mupkappa
λ	Alph	\muplambda
μ	Alph	\mupmu
ν	Alph	\mupnu
ξ	Alph	\mupxi
ο	Alph	\mupomicron
π	Alph	\muppi
ρ	Alph	\muprho
ς	Alph	\mupvarsigma
σ	Alph	\mupsigma
τ	Alph	\muptau
υ	Alph	\mupupsilon
φ	Alph	\mupvarphi
χ	Alph	\mupchi
ψ	Alph	\muppsi
ω	Alph	\mupomega
ϑ	Alph	\mupvartheta
ϕ	Alph	\mupphi
ϖ	Alph	\mupvarpi
	Alph	\upDigamma
	Alph	\updigamma
ϰ	Alph	\mupvarkappa
ϱ	Alph	\mupvarrho
ϴ	Alph	\mupvarTheta
ϵ	Alph	\mupepsilon
	Ord	\upbackepsilon=\backepsilon
‐	Alph	\mathhyphen
―	Ord	\horizbar
‖	Fence	\Vert
‗	Ord	\twolowline
†	Bin	\dagger
‡	Bin	\ddagger
•	Bin	\smblkcircle=\bullet
	Ord	\enleadertwodots
…	Ord	\unicodeellipsis
′	Ord	\prime
″	Ord	\dprime
‴	Ord	\trprime
‵	Ord	\backprime
‶	Ord	\backdprime
‷	Ord	\backtrprime
	Ord	\caretinsert
	Ord	\Exclam
	Bin	\tieconcat

	Ord	\hyphenbullet
⁄	Bin	\fracslash
	Ord	\Question
	Rel	\closure
⁗	Ord	\qprime
€	Ord	\euro
⃐	Acc	\leftharpoonaccent
⃐	Accw	\overleftharpoon
⃑	Acc	\rightharpoonaccent
⃑	Accw	\overrightharpoon
⃒	Acc	\vertoverlay
⃖	Accw	\overleftarrow
⃗	Accw	\overrightarrow
⃗	Acc	\vec
⃛	Acc	\dddot
⃜	Acc	\ddddot
⃝	Ord	\enclosecircle
⃞	Ord	\enclosesquare
⃟	Ord	\enclosediamond
⃡	Accw	\overleftrightarrow
⃤	Ord	\enclosetriangle
	Acc	\annuity
⃨	AccB	\threeunderdot
⃩	Accw	\widebridgeabove
⃬	AccBw	\underrightharpoondown
⃭	AccBw	\underleftharpoondown
⃮	AccBw	\underleftarrow
⃯	AccBw	\underrightarrow
⃰	Acc	\asteraccent
ℂ	Alph	\BbbC
ℇ	Ord	\Eulerconst
	Alph	\mscrg
ℋ	Alph	\mscrH
ℌ	Alph	\mfrakH
ℍ	Alph	\BbbH
ℎ	Ord	\Planckconst
ℏ	Alph	\hslash=\hbar
ℐ	Alph	\mscrI
ℑ	Alph	\Im
ℒ	Alph	\mscrL
ℓ	Alph	\ell
ℕ	Alph	\BbbN
℘	Alph	\wp
ℙ	Alph	\BbbP
ℚ	Alph	\BbbQ
ℛ	Alph	\mscrR
ℜ	Alph	\Re
ℝ	Alph	\BbbR
ℤ	Alph	\BbbZ
℧	Ord	\mho
ℨ	Alph	\mfrakZ
	Alph	\turnediota
Å	Alph	\Angstrom
ℬ	Alph	\mscrB
ℭ	Alph	\mfrakC
	Alph	\mscre
ℰ	Alph	\mscrE
ℱ	Alph	\mscrF
	Ord	\Finv
ℳ	Alph	\mscrM
	Alph	\mscro
ℵ	Alph	\aleph
ℶ	Alph	\beth
ℷ	Alph	\gimel
ℸ	Alph	\daleth
ℼ	Ord	\Bbbpi
ℽ	Alph	\Bbbgamma
ℾ	Alph	\BbbGamma
ℿ	Alph	\BbbPi
⅀	Op	\Bbbsum
	Ord	\Game
	Ord	\sansLturned
	Ord	\sansLmirrored
	Ord	\Yup
ⅅ	Ord	\mitBbbD
ⅆ	Ord	\mitBbbd

ⅇ	Ord	\mitBbbe
ⅈ	Ord	\mitBbbi
ⅉ	Ord	\mitBbbj
	Ord	\PropertyLine
	Bin	\upand
←	Rel	\leftarrow=\gets
↑	Rel	\uparrow
→	Rel	\rightarrow=\to
↓	Rel	\downarrow
↔	Rel	\leftrightarrow
↕	Rel	\updownarrow
↖	Rel	\nwarrow
↗	Rel	\nearrow
↘	Rel	\searrow
↙	Rel	\swarrow
↚	Rel	\nleftarrow
↛	Rel	\nrightarrow
	Rel	\leftwavearrow
	Rel	\rightwavearrow
↞	Rel	\twoheadleftarrow
↟	Rel	\twoheaduparrow
↠	Rel	\twoheadrightarrow
↡	Rel	\twoheaddownarrow
↢	Rel	\leftarrowtail
↣	Rel	\rightarrowtail
↤	Rel	\mapsfrom
↥	Rel	\mapsup
↦	Rel	\mapsto
↧	Rel	\mapsdown
	Ord	\updownarrowbar
↩	Rel	\hookleftarrow
↪	Rel	\hookrightarrow
↫	Rel	\looparrowleft
↬	Rel	\looparrowright
↭	Rel	\leftrightsquigarrow
↮	Rel	\nleftrightarrow
	Rel	\downzigzagarrow
↰	Rel	\Lsh
↱	Rel	\Rsh
↲	Rel	\Ldsh
↳	Rel	\Rdsh
↴	Ord	\linefeed
↵	Ord	\carriagereturn
↶	Rel	\curvearrowleft
↷	Rel	\curvearrowright
	Ord	\barovernorthwestarrow
	Ord	\barleftarrowrightarrowbar
↺	Ord	\acwopencirclearrow
↻	Ord	\cwopencirclearrow
↼	Rel	\leftharpoonup
↽	Rel	\leftharpoondown
↾	Rel	\upharpoonright
↿	Rel	\upharpoonleft
⇀	Rel	\rightharpoonup
⇁	Rel	\rightharpoondown
⇂	Rel	\downharpoonright
⇃	Rel	\downharpoonleft
⇄	Rel	\rightleftarrows
⇅	Rel	\updownarrows
⇆	Rel	\leftrightarrows
⇇	Rel	\leftleftarrows
⇈	Rel	\upuparrows
⇉	Rel	\rightrightarrows
⇊	Rel	\downdownarrows
⇋	Rel	\leftrightharpoons
⇌	Rel	\rightleftharpoons
⇍	Rel	\nLeftarrow
⇎	Rel	\nLeftrightarrow
⇏	Rel	\nRightarrow
⇐	Rel	\Leftarrow
⇑	Rel	\Uparrow
⇒	Rel	\Rightarrow
⇓	Rel	\Downarrow
⇔	Rel	\Leftrightarrow
⇕	Rel	\Updownarrow
⇖	Rel	\Nwarrow
⇗	Rel	\Nearrow

⇘	Rel	\Searrow
⇙	Rel	\Swarrow
⇚	Rel	\Lleftarrow
⇛	Rel	\Rrightarrow
⇜	Rel	\leftsquigarrow
⇝	Rel	\rightsquigarrow
	Ord	\nHuparrow
	Ord	\nHdownarrow
	Ord	\leftdasharrow
	Ord	\updasharrow
	Ord	\rightdasharrow
	Ord	\downdasharrow
	Rel	\barleftarrow
	Rel	\rightarrowbar
⇦	Ord	\leftwhitearrow
⇧	Ord	\upwhitearrow
⇨	Ord	\rightwhitearrow
⇩	Ord	\downwhitearrow
	Ord	\whitearrowupfrombar
	Rel	\circleonrightarrow
⇵	Rel	\downuparrows
⇶	Rel	\rightthreearrows
	Rel	\nvleftarrow
	Rel	\nvrightarrow
	Rel	\nvleftrightarrow
	Rel	\nVleftarrow
	Rel	\nVrightarrow
	Rel	\nVleftrightarrow
	Rel	\leftarrowtriangle
	Rel	\rightarrowtriangle
	Rel	\leftrightarrowtriangle
∀	Ord	\forall
∁	Ord	\complement
∂	Alph	\partial
∃	Ord	\exists
∄	Ord	\nexists
∅	Ord	\varnothing=\emptyset
∆	Ord	\increment
∇	Alph	\nabla
∈	Rel	\in
∉	Rel	\notin
∊	Rel	\smallin
∋	Rel	\ni=\owns
∌	Rel	\nni
∍	Rel	\smallni
∎	Ord	\QED
∏	Op	\prod
∐	Op	\coprod
∑	Op	\sum
−	Bin	\minus
∓	Bin	\mp
∔	Bin	\dotplus
∕	Bin	\divslash
∖	Bin	\smallsetminus
∗	Bin	\ast
∘	Bin	\vysmwhtcircle=\circ
∙	Bin	\vysmblkcircle
√	Open	\sqrt
√	Ord	\surd
	Open	\cuberoot
	Open	\fourthroot
∝	Rel	\propto
∞	Ord	\infty
∟	Ord	\rightangle
∠	Ord	\angle
∡	Ord	\measuredangle
∢	Ord	\sphericalangle
∣	Rel	\mid
∤	Rel	\nmid
∥	Rel	\parallel
∦	Rel	\nparallel
∧	Bin	\wedge=\land
∨	Bin	\vee=\lor
∩	Bin	\cap
∪	Bin	\cup
∫	Op	\int
∬	Op	\iint
∭	Op	\iiint
∮	Op	\oint
∯	Op	\oiint
∰	Op	\oiiint
∱	Op	\intclockwise
∲	Op	\varointclockwise
∳	Op	\ointctrclockwise
∴	Ord	\therefore
∵	Ord	\because
∶	Rel	\mathratio
∷	Rel	\Colon
∸	Bin	\dotminus
∹	Rel	\dashcolon
∺	Rel	\dotsminusdots
∻	Rel	\kernelcontraction
∼	Rel	\sim
∽	Rel	\backsim
∾	Bin	\invlazys
∿	Ord	\sinewave
≀	Bin	\wr
≁	Rel	\nsim
≂	Rel	\eqsim
≃	Rel	\simeq
≄	Rel	\nsime
≃	Rel	\sime
≄	Rel	\nsimeq
≅	Rel	\cong
≆	Rel	\simneqq
≇	Rel	\ncong
≈	Rel	\approx
≉	Rel	\napprox
≊	Rel	\approxeq
≋	Rel	\approxident
≌	Rel	\backcong
≍	Rel	\asymp
≎	Rel	\Bumpeq
≏	Rel	\bumpeq
≐	Rel	\doteq
≑	Rel	\Doteq
≒	Rel	\fallingdotseq
≓	Rel	\risingdotseq
≔	Rel	\coloneq
≕	Rel	\eqcolon
≖	Rel	\eqcirc
≗	Rel	\circeq
≘	Rel	\arceq
≙	Rel	\wedgeq
≚	Rel	\veeeq
≛	Rel	\stareq
≜	Rel	\triangleq
≝	Rel	\eqdef
≞	Rel	\measeq
≟	Rel	\questeq
≠	Rel	\ne=\neq
≡	Rel	\equiv
≢	Rel	\nequiv
≣	Rel	\Equiv
≤	Rel	\leq=\le
≥	Rel	\geq=\ge
≦	Rel	\leqq
≧	Rel	\geqq
≨	Rel	\lneqq
≩	Rel	\gneqq
≪	Rel	\ll
≫	Rel	\gg
≬	Rel	\between
≭	Rel	\nasymp
≮	Rel	\nless
≯	Rel	\ngtr
≰	Rel	\nleq
≱	Rel	\ngeq
≲	Rel	\lesssim
≳	Rel	\gtrsim
≴	Rel	\nlesssim
≵	Rel	\ngtrsim
≶	Rel	\lessgtr
≷	Rel	\gtrless
≸	Rel	\nlessgtr
≹	Rel	\ngtrless
≺	Rel	\prec
≻	Rel	\succ
≼	Rel	\preccurlyeq
≽	Rel	\succcurlyeq
≾	Rel	\precsim
≿	Rel	\succsim
⊀	Rel	\nprec
⊁	Rel	\nsucc
⊂	Rel	\subset
⊃	Rel	\supset
⊄	Rel	\nsubset
⊅	Rel	\nsupset
⊆	Rel	\subseteq
⊇	Rel	\supseteq
⊈	Rel	\nsubseteq
⊉	Rel	\nsupseteq
⊊	Rel	\subsetneq
⊋	Rel	\supsetneq
⊌	Bin	\cupleftarrow
⊍	Bin	\cupdot
⊎	Bin	\uplus
⊏	Rel	\sqsubset
⊐	Rel	\sqsupset
⊑	Rel	\sqsubseteq
⊒	Rel	\sqsupseteq
⊓	Bin	\sqcap
⊔	Bin	\sqcup
⊕	Bin	\oplus
⊖	Bin	\ominus
⊗	Bin	\otimes
⊘	Bin	\oslash
⊙	Bin	\odot
⊚	Bin	\circledcirc
⊛	Bin	\circledast
⊜	Bin	\circledequal
⊝	Bin	\circleddash
⊞	Bin	\boxplus
⊟	Bin	\boxminus
⊠	Bin	\boxtimes
⊡	Bin	\boxdot
⊢	Rel	\vdash
⊣	Rel	\dashv
⊤	Ord	\top
⊥	Ord	\bot
⊦	Rel	\assert
⊧	Rel	\models
⊨	Rel	\vDash
⊩	Rel	\Vdash
⊪	Rel	\Vvdash
⊫	Rel	\VDash
⊬	Rel	\nvdash
⊭	Rel	\nvDash
⊮	Rel	\nVdash
⊯	Rel	\nVDash
	Rel	\prurel
	Rel	\scurel
⊲	Rel	\vartriangleleft
⊳	Rel	\vartriangleright
⊴	Rel	\trianglelefteq
⊵	Rel	\trianglerighteq
⊶	Rel	\origof
⊷	Rel	\imageof
⊸	Rel	\multimap
⊹	Ord	\hermitmatrix
⊺	Bin	\intercal
⊻	Bin	\veebar
⊼	Bin	\barwedge
⊽	Bin	\barvee
⊾	Ord	\measuredrightangle
⊿	Ord	\varlrtriangle
⋀	Op	\bigwedge
⋁	Op	\bigvee
⋂	Op	\bigcap
⋃	Op	\bigcup
⋄	Bin	\smwhtdiamond=\diamond

⋅	Bin	\cdot
⋆	Bin	\star
⋇	Bin	\divideontimes
⋈	Rel	\bowtie
⋉	Bin	\ltimes
⋊	Bin	\rtimes
⋋	Bin	\leftthreetimes
⋌	Bin	\rightthreetimes
⋍	Rel	\backsimeq
⋎	Bin	\curlyvee
⋏	Bin	\curlywedge
⋐	Rel	\Subset
⋑	Rel	\Supset
⋒	Bin	\Cap
⋓	Bin	\Cup
	Rel	\pitchfork
⋕	Rel	\equalparallel
⋖	Rel	\lessdot
⋗	Rel	\gtrdot
⋘	Rel	\lll
⋙	Rel	\ggg
⋚	Rel	\lesseqgtr
⋛	Rel	\gtreqless
⋜	Rel	\eqless
⋝	Rel	\eqgtr
⋞	Rel	\curlyeqprec
⋟	Rel	\curlyeqsucc
⋠	Rel	\npreccurlyeq
⋡	Rel	\nsucccurlyeq
⋢	Rel	\nsqsubseteq
⋣	Rel	\nsqsupseteq
⋤	Rel	\sqsubsetneq
⋥	Rel	\sqsupsetneq
⋦	Rel	\lnsim
⋧	Rel	\gnsim
⋨	Rel	\precnsim
⋩	Rel	\succnsim
⋪	Rel	\nvartriangleleft
⋫	Rel	\nvartriangleright
⋬	Rel	\ntrianglelefteq
⋭	Rel	\ntrianglerighteq
⋮	Rel	\unicodevdots
⋯	Ord	\unicodecdots
⋰	Rel	\unicodeadots
⋱	Rel	\unicodeddots
	Rel	\disin
	Rel	\varisins
	Rel	\isins
	Rel	\isindot
	Rel	\varisinobar
	Rel	\isinobar
	Rel	\isinvb
	Rel	\isinE
	Rel	\nisd
	Rel	\varnis
	Rel	\nis
	Rel	\varniobar
	Rel	\niobar
	Rel	\bagmember
⌀	Ord	\diameter
	Ord	\house
⌅	Bin	\varbarwedge
⌆	Bin	\vardoublebarwedge
⌈	Open	\lceil
⌉	Close	\rceil
⌊	Open	\lfloor
⌋	Close	\rfloor
⌐	Ord	\invnot
	Ord	\sqlozenge
	Ord	\profline
	Ord	\profsurf
	Ord	\viewdata
⌙	Ord	\turnednot
⌜	Open	\ulcorner
⌝	Close	\urcorner
⌞	Open	\llcorner
⌟	Close	\lrcorner
⌠	Ord	\inttop
⌡	Ord	\intbottom
⌢	Rel	\frown
⌣	Rel	\smile
	Ord	\varhexagonlrbonds
	Ord	\conictaper
	Ord	\topbot
	Bin	\obar
	Rel	\APLnotslash
	Ord	\APLnotbackslash
	Ord	\APLboxupcaret
	Ord	\APLboxquestion
	Ord	\rangledownzigzagarrow
	Ord	\hexagon
⎛	Ord	\lparenuend
⎜	Ord	\lparenextender
⎝	Ord	\lparenlend
⎞	Ord	\rparenuend
⎟	Ord	\rparenextender
⎠	Ord	\rparenlend
⎡	Ord	\lbrackuend
⎢	Ord	\lbrackextender
⎣	Ord	\lbracklend
⎤	Ord	\rbrackuend
⎥	Ord	\rbrackextender
⎦	Ord	\rbracklend
⎧	Ord	\lbraceuend
⎨	Ord	\lbracemid
⎩	Ord	\lbracelend
⎪	Ord	\vbraceextender
⎫	Ord	\rbraceuend
⎬	Ord	\rbracemid
⎭	Ord	\rbracelend
	Ord	\intextender
	Ord	\harrowextender
	Open	\lmoustache
	Close	\rmoustache
⎲	Ord	\sumtop
⎳	Ord	\sumbottom
⎴	Over	\overbracket
⎵	Under	\underbracket
	Ord	\bbrktbrk
⎷	Ord	\sqrtbottom
	Ord	\lvboxline
	Ord	\rvboxline
	Ord	\varcarriagereturn
⏜	Over	\overparen
⏝	Under	\underparen
⏞	Over	\overbrace
⏟	Under	\underbrace
⏠	Ord	\obrbrak
⏡	Ord	\ubrbrak
	Ord	\trapezium
	Ord	\benzenr
	Ord	\strns
	Ord	\fltns
	Ord	\accurrent
	Ord	\elinters
␢	Ord	\blanksymbol
␣	Ord	\mathvisiblespace
	Ord	\bdtriplevdash
	Ord	\blockuphalf
	Ord	\blocklowhalf
█	Ord	\blockfull
	Ord	\blocklefthalf
	Ord	\blockrighthalf
░	Ord	\blockqtrshaded
▒	Ord	\blockhalfshaded
▓	Ord	\blockthreeqtrshaded
■	Ord	\mdlgblksquare
□	Ord	\mdlgwhtsquare
	Ord	\squoval
	Ord	\blackinwhitesquare
	Ord	\squarehfill
	Ord	\squarevfill
	Ord	\squarehvfill
	Ord	\squarenwsefill
	Ord	\squareneswfill
	Ord	\squarecrossfill
▪	Ord	\smblksquare
▫	Ord	\smwhtsquare
▬	Ord	\hrectangleblack
▭	Ord	\hrectangle
	Ord	\vrectangleblack
	Ord	\vrectangle
	Ord	\parallelogramblack
	Ord	\parallelogram
▲	Ord	\bigblacktriangleup
△	Bin	\bigtriangleup
	Ord	\blacktriangle
	Rel	\vartriangle
▶	Ord	\blacktriangleright
▷	Bin	\triangleright
	Ord	\smallblacktriangleright
	Ord	\smalltriangleright
	Ord	\blackpointerright
	Ord	\whitepointerright
▼	Ord	\bigblacktriangledown
▽	Ord	\bigtriangledown
	Ord	\blacktriangledown
	Ord	\triangledown
◀	Ord	\blacktriangleleft
◁	Bin	\triangleleft
	Ord	\smallblacktriangleleft
	Ord	\smalltriangleleft
	Ord	\blackpointerleft
	Ord	\whitepointerleft
	Ord	\mdlgblkdiamond
	Ord	\mdlgwhtdiamond
	Ord	\blackinwhitediamond
	Ord	\fisheye
◊	Ord	\mdlgwhtlozenge
○	Bin	\mdlgwhtcircle=\bigcirc
	Ord	\dottedcircle
	Ord	\circlevertfill
	Ord	\bullseye
●	Ord	\mdlgblkcircle
	Ord	\circlelefthalfblack
	Ord	\circlerighthalfblack
	Ord	\circlebottomhalfblack
	Ord	\circletophalfblack
	Ord	\circleurquadblack
	Ord	\blackcircleulquadwhite
	Ord	\blacklefthalfcircle
	Ord	\blackrighthalfcircle
	Ord	\inversebullet
	Ord	\inversewhitecircle
	Ord	\invwhiteupperhalfcircle
	Ord	\invwhitelowerhalfcircle
	Ord	\ularc
	Ord	\urarc
	Ord	\lrarc
	Ord	\llarc
	Ord	\topsemicircle
	Ord	\botsemicircle
	Ord	\lrblacktriangle
	Ord	\llblacktriangle
	Ord	\ulblacktriangle
	Ord	\urblacktriangle
◦	Ord	\smwhtcircle
	Ord	\squareleftblack
	Ord	\squarerightblack
	Ord	\squareulblack
	Ord	\squarelrblack
	Bin	\boxbar
	Ord	\trianglecdot
	Ord	\triangleleftblack
	Ord	\trianglerightblack

◯	Ord	\lgwhtcircle
	Ord	\squareulquad
	Ord	\squarellquad
	Ord	\squarelrquad
	Ord	\squareurquad
	Ord	\circleulquad
	Ord	\circlellquad
	Ord	\circlelrquad
	Ord	\circleurquad
	Ord	\ultriangle
	Ord	\urtriangle
	Ord	\lltriangle
	Ord	\mdwhtsquare
	Ord	\mdblksquare
	Ord	\mdsmwhtsquare
	Ord	\mdsmblksquare
	Ord	\lrtriangle
	Ord	\bigstar
	Ord	\bigwhitestar
	Ord	\astrosun
	Ord	\danger
	Ord	\blacksmiley
	Ord	\sun
	Ord	\rightmoon
	Ord	\leftmoon
	Ord	\female
	Ord	\male
♠	Ord	\spadesuit
♡	Ord	\heartsuit
♢	Ord	\diamondsuit
♣	Ord	\clubsuit
♤	Ord	\varspadesuit
♥	Ord	\varheartsuit
♦	Ord	\vardiamondsuit
♧	Ord	\varclubsuit
	Ord	\quarternote
♪	Ord	\eighthnote
	Ord	\twonotes
♭	Ord	\flat
♮	Ord	\natural
♯	Ord	\sharp
	Ord	\acidfree
	Ord	\dicei
	Ord	\diceii
	Ord	\diceiii
	Ord	\diceiv
	Ord	\dicev
	Ord	\dicevi
	Ord	\circledrightdot
	Ord	\circledtwodots
	Ord	\blackcircledrightdot
	Ord	\blackcircledtwodots
	Ord	\Hermaphrodite
	Ord	\mdwhtcircle
	Ord	\mdblkcircle
	Ord	\mdsmwhtcircle
	Ord	\neuter
✓	Ord	\checkmark
✠	Ord	\maltese
	Ord	\circledstar
	Ord	\varstar
	Ord	\dingasterisk
	Open	\lbrbrak
	Close	\rbrbrak
	Ord	\draftingarrow
	Ord	\threedangle
	Ord	\whiteinwhitetriangle
⊥	Rel	\perp
	Ord	\subsetcirc
	Ord	\supsetcirc
	Open	\lbag
	Close	\rbag
	Bin	\veedot
	Rel	\bsolhsub
	Rel	\suphsol
	Open	\longdivision
	Ord	\diamondcdot
	Bin	\wedgedot
	Rel	\upin
	Rel	\pullback
	Rel	\pushout
	Op	\leftouterjoin
	Op	\rightouterjoin
	Op	\fullouterjoin
⟘	Op	\bigbot
⟙	Op	\bigtop
⟚	Rel	\DashVDash
⟛	Rel	\dashVdash
⟜	Rel	\multimapinv
⟝	Rel	\vlongdash
⟞	Rel	\longdashv
	Rel	\cirbot
⟠	Bin	\lozengeminus
⟡	Bin	\concavediamond
⟢	Bin	\concavediamondtickleft
⟣	Bin	\concavediamondtickright
	Bin	\whitesquaretickleft
	Bin	\whitesquaretickright
⟦	Open	\lBrack
⟧	Close	\rBrack
⟨	Open	\langle
⟩	Close	\rangle
⟪	Open	\lAngle
⟫	Close	\rAngle
	Open	\Lbrbrak
	Close	\Rbrbrak
⟮	Open	\lgroup
⟯	Close	\rgroup
	Rel	\UUparrow
	Rel	\DDownarrow
	Rel	\acwgapcirclearrow
	Rel	\cwgapcirclearrow
⟴	Rel	\rightarrowonoplus
⟵	Rel	\longleftarrow
⟶	Rel	\longrightarrow
⟷	Rel	\longleftrightarrow
⟸	Rel	\Longleftarrow
⟹	Rel	\Longrightarrow
⟺	Rel	\Longleftrightarrow
⟻	Rel	\longmapsfrom
⟼	Rel	\longmapsto
⟽	Rel	\Longmapsfrom
⟾	Rel	\Longmapsto
⟿	Rel	\longrightsquigarrow
	Rel	\nvtwoheadrightarrow
	Rel	\nVtwoheadrightarrow
	Rel	\nvLeftarrow
	Rel	\nvRightarrow
	Rel	\nvLeftrightarrow
	Rel	\twoheadmapsto
⤆	Rel	\Mapsfrom
⤇	Rel	\Mapsto
	Rel	\downarrowbarred
	Rel	\uparrowbarred
	Rel	\Uuparrow
	Rel	\Ddownarrow
	Rel	\leftbkarrow
	Rel	\rightbkarrow
	Rel	\leftdbkarrow
	Rel	\dbkarrow=\dbkarow
	Rel	\drbkarrow=\drbkarow
	Rel	\rightdotarrow
	Rel	\baruparrow
	Rel	\downarrowbar
	Rel	\nvrightarrowtail
	Rel	\nVrightarrowtail
	Rel	\twoheadrightarrowtail
	Rel	\nvtwoheadrightarrowtail
	Rel	\nVtwoheadrightarrowtail
	Rel	\lefttail
	Rel	\righttail
	Rel	\leftdbltail
	Rel	\rightdbltail
	Rel	\diamondleftarrow
	Rel	\rightarrowdiamond
	Rel	\diamondleftarrowbar
	Rel	\barrightarrowdiamond
	Rel	\nwsearrow
	Rel	\neswarrow
	Rel	\hknwarrow
	Rel	\hknearrow
	Rel	\hksearrow=\hksearow
	Rel	\hkswarrow=\hkswarow
	Rel	\tona
	Rel	\toea
	Rel	\tosa
	Rel	\towa
	Ord	\rdiagovfdiag
	Ord	\fdiagovrdiag
	Ord	\seovnearrow
	Ord	\neovsearrow
	Ord	\fdiagovnearrow
	Ord	\rdiagovsearrow
	Ord	\neovnwarrow
	Ord	\nwovnearrow
	Rel	\rightcurvedarrow
	Ord	\uprightcurvearrow
	Ord	\downrightcurvedarrow
	Rel	\leftdowncurvedarrow
	Rel	\rightdowncurvedarrow
	Rel	\cwrightarcarrow
	Rel	\acwleftarcarrow
	Rel	\acwoverarcarrow
	Rel	\acwunderarcarrow
	Rel	\curvearrowrightminus
	Rel	\curvearrowleftplus
	Rel	\cwundercurvearrow
	Rel	\ccwundercurvearrow
	Rel	\acwcirclearrow
	Rel	\cwcirclearrow
	Rel	\rightarrowshortleftarrow
	Rel	\leftarrowshortrightarrow
	Rel	\shortrightarrowleftarrow
	Rel	\rightarrowplus
	Rel	\leftarrowplus
	Rel	\rightarrowx
	Rel	\leftrightarrowcircle
	Rel	\twoheaduparrowcircle
	Rel	\leftrightharpoonupdown
	Rel	\leftrightharpoondownup
	Rel	\updownharpoonrightleft
	Rel	\updownharpoonleftright
	Rel	\leftrightharpoonupup
	Rel	\updownharpoonrightright
	Rel	\leftrightharpoondowndown
	Rel	\updownharpoonleftleft
	Rel	\barleftharpoonup
	Rel	\rightharpoonupbar
	Rel	\barupharpoonright
	Rel	\downharpoonrightbar
	Rel	\barleftharpoondown
	Rel	\rightharpoondownbar
	Rel	\barupharpoonleft
	Rel	\downharpoonleftbar
	Rel	\leftharpoonupbar
	Rel	\barrightharpoonup
	Rel	\upharpoonrightbar
	Rel	\bardownharpoonright
	Rel	\leftharpoondownbar
	Rel	\barrightharpoondown
	Rel	\upharpoonleftbar
	Rel	\bardownharpoonleft
	Rel	\leftharpoonsupdown
	Rel	\upharpoonsleftright
	Rel	\rightharpoonsupdown
	Rel	\downharpoonsleftright
	Rel	\leftrightharpoonsup
	Rel	\leftrightharpoonsdown
	Rel	\rightleftharpoonsup
	Rel	\rightleftharpoonsdown
	Rel	\leftharpoonupdash

	Rel	\dashleftharpoondown
	Rel	\rightharpoonupdash
	Rel	\dashrightharpoondown
	Rel	\updownharpoonsleftright
	Rel	\downupharpoonsleftright
	Rel	\rightimply
	Rel	\equalrightarrow
	Rel	\similarrightarrow
	Rel	\leftarrowsimilar
	Rel	\rightarrowsimilar
	Rel	\rightarrowapprox
	Rel	\ltlarr
	Rel	\leftarrowless
	Rel	\gtrarr
	Rel	\subrarr
	Rel	\leftarrowsubset
	Rel	\suplarr
	Rel	\leftfishtail
	Rel	\rightfishtail
	Rel	\upfishtail
	Rel	\downfishtail
	Fence	\Vvert
	Ord	\mdsmblkcircle
	Rel	\typecolon
	Open	\lBrace
	Close	\rBrace
	Open	\lParen
	Close	\rParen
	Open	\llparenthesis
	Close	\rrparenthesis
	Open	\llangle
	Close	\rrangle
	Open	\lbrackubar
	Close	\rbrackubar
	Open	\lbrackultick
	Close	\rbracklrtick
	Open	\lbracklltick
	Close	\rbrackurtick
	Open	\langledot
	Close	\rangledot
	Open	\lparenless
	Close	\rparengtr
	Open	\Lparengtr
	Close	\Rparenless
	Open	\lblkbrbrak
	Close	\rblkbrbrak
	Ord	\fourvdots
	Ord	\vzigzag
	Ord	\measuredangleleft
	Ord	\rightanglesqr
	Ord	\rightanglemdot
	Ord	\angles
	Ord	\angdnr
	Ord	\gtlpar
	Ord	\sphericalangleup
	Ord	\turnangle
	Ord	\revangle
	Ord	\angleubar
	Ord	\revangleubar
	Ord	\wideangledown
	Ord	\wideangleup
	Ord	\measanglerutone
	Ord	\measanglelutonw
	Ord	\measanglerdtose
	Ord	\measangleldtosw
	Ord	\measangleurtone
	Ord	\measangleultonw
	Ord	\measangledrtose
	Ord	\measangledltosw
	Ord	\revemptyset
	Ord	\emptysetobar
	Ord	\emptysetocirc
	Ord	\emptysetoarr
	Ord	\emptysetoarrl
	Bin	\circlehbar
	Bin	\circledvert
	Bin	\circledparallel
	Bin	\obslash
	Bin	\operp
	Ord	\obot
	Ord	\olcross
	Ord	\odotslashdot
	Ord	\uparrowoncircle
	Ord	\circledwhitebullet
	Ord	\circledbullet
	Bin	\olessthan
	Bin	\ogreaterthan
	Ord	\cirscir
	Ord	\cirE
	Bin	\boxdiag
	Bin	\boxbslash
	Bin	\boxast
	Bin	\boxcircle
	Bin	\boxbox
	Ord	\boxonbox
	Ord	\triangleodot
	Ord	\triangleubar
	Ord	\triangles
	Bin	\triangleserifs
	Rel	\rtriltri
	Rel	\ltrivb
	Rel	\vbrtri
	Rel	\lfbowtie
	Rel	\rfbowtie
	Rel	\fbowtie
	Rel	\lftimes
	Rel	\rftimes
	Bin	\hourglass
	Bin	\blackhourglass
	Open	\lvzigzag
	Close	\rvzigzag
	Open	\Lvzigzag
	Close	\Rvzigzag
	Ord	\iinfin
	Ord	\tieinfty
	Ord	\nvinfty
	Rel	\dualmap
	Ord	\laplac
	Rel	\lrtriangleeq
	Bin	\shuffle
	Rel	\eparsl
	Rel	\smeparsl
	Rel	\eqvparsl
	Rel	\gleichstark
	Ord	\thermod
	Ord	\downtriangleleftblack
	Ord	\downtrianglerightblack
	Ord	\blackdiamonddownarrow
	Bin	\mdlgblklozenge
	Ord	\circledownarrow
	Ord	\blackcircledownarrow
	Ord	\errbarsquare
	Ord	\errbarblacksquare
	Ord	\errbardiamond
	Ord	\errbarblackdiamond
	Ord	\errbarcircle
	Ord	\errbarblackcircle
	Rel	\ruledelayed
∖	Bin	\setminus
	Bin	\dsol
	Bin	\rsolbar
	Op	\xsol
	Op	\xbsol
	Bin	\doubleplus
	Bin	\tripleplus
	Open	\lcurvyangle
	Close	\rcurvyangle
	Bin	\tplus
	Bin	\tminus
⨀	Op	\bigodot
⨁	Op	\bigoplus
⨂	Op	\bigotimes
⨃	Op	\bigcupdot
⨄	Op	\biguplus
⨅	Op	\bigsqcap
⨆	Op	\bigsqcup
	Op	\conjquant
	Op	\disjquant
⨉	Op	\bigtimes
	Op	\modtwosum
	Op	\sumint
⨌	Op	\iiiint
	Op	\intbar
	Op	\intBar
	Op	\fint
	Op	\cirfnint
⨑	Op	\awint
	Op	\rppolint
	Op	\scpolint
	Op	\npolint
	Op	\pointint
	Op	\sqint
	Op	\intlarhk
	Op	\intx
	Op	\intcap
	Op	\intcup
	Op	\upint
	Op	\lowint
	Op	\Join
	Op	\bigtriangleleft
	Op	\zcmp
	Op	\zpipe
	Op	\zproject
	Bin	\ringplus
	Bin	\plushat
	Bin	\simplus
	Bin	\plusdot
	Bin	\plussim
	Bin	\plussubtwo
	Bin	\plustrif
	Bin	\commaminus
	Bin	\minusdot
	Bin	\minusfdots
	Bin	\minusrdots
	Bin	\opluslhrim
	Bin	\oplusrhrim
⨯	Bin	\vectimes
	Bin	\dottimes
	Bin	\timesbar
	Bin	\btimes
	Bin	\smashtimes
	Bin	\otimeslhrim
	Bin	\otimesrhrim
	Bin	\otimeshat
	Bin	\Otimes
	Bin	\odiv
	Bin	\triangleplus
	Bin	\triangleminus
	Bin	\triangletimes
	Bin	\intprod
	Bin	\intprodr
	Bin	\fcmp
⨿	Bin	\amalg
	Bin	\capdot
	Bin	\uminus
	Bin	\barcup
	Bin	\barcap
	Bin	\capwedge
	Bin	\cupvee
	Bin	\cupovercap
	Bin	\capovercup
	Bin	\cupbarcap
	Bin	\capbarcup
	Bin	\twocups
	Bin	\twocaps
	Bin	\closedvarcup
	Bin	\closedvarcap
	Bin	\Sqcap
	Bin	\Sqcup
	Bin	\closedvarcupsmashprod
	Bin	\wedgeodot

	Bin	\veeodot
	Bin	\Wedge
	Bin	\Vee
	Bin	\wedgeonwedge
	Bin	\veeonvee
	Bin	\bigslopedvee
	Bin	\bigslopedwedge
	Rel	\veeonwedge
	Bin	\wedgemidvert
	Bin	\veemidvert
	Bin	\midbarwedge
	Bin	\midbarvee
	Bin	\doublebarwedge
	Bin	\wedgebar
	Bin	\wedgedoublebar
	Bin	\varveebar
	Bin	\doublebarvee
	Bin	\veedoublebar
	Bin	\dsub
	Bin	\rsub
	Rel	\eqdot
	Rel	\dotequiv
	Rel	\equivVert
	Rel	\equivVvert
	Rel	\dotsim
	Rel	\simrdots
	Rel	\simminussim
	Rel	\congdot
	Rel	\asteq
	Rel	\hatapprox
	Rel	\approxeqq
	Bin	\eqqplus
	Bin	\pluseqq
	Rel	\eqqsim
	Rel	\Coloneq
	Rel	\eqeq
	Rel	\eqeqeq
	Rel	\ddotseq
	Rel	\equivDD
	Rel	\ltcir
	Rel	\gtcir
	Rel	\ltquest
	Rel	\gtquest
⩽	Rel	\leqslant
⩾	Rel	\geqslant
	Rel	\lesdot
	Rel	\gesdot
	Rel	\lesdoto
	Rel	\gesdoto
	Rel	\lesdotor
	Rel	\gesdotol
⪅	Rel	\lessapprox
⪆	Rel	\gtrapprox
⪇	Rel	\lneq
⪈	Rel	\gneq
⪉	Rel	\lnapprox
⪊	Rel	\gnapprox
⪋	Rel	\lesseqqgtr
⪌	Rel	\gtreqqless
	Rel	\lsime
	Rel	\gsime
	Rel	\lsimg
	Rel	\gsiml
	Rel	\lgE
	Rel	\glE
	Rel	\lesges
	Rel	\gesles
⪕	Rel	\eqslantless
⪖	Rel	\eqslantgtr
	Rel	\elsdot
	Rel	\egsdot
	Rel	\eqqless
	Rel	\eqqgtr
	Rel	\eqqslantless
	Rel	\eqqslantgtr
	Rel	\simless
	Rel	\simgtr
	Rel	\simlE
	Rel	\simgE
	Rel	\Lt
	Rel	\Gt
	Rel	\partialmeetcontraction
	Rel	\glj
	Rel	\gla
	Rel	\ltcc
	Rel	\gtcc
	Rel	\lescc
	Rel	\gescc
	Rel	\smt
	Rel	\lat
	Rel	\smte
	Rel	\late
	Rel	\bumpeqq
⪯	Rel	\preceq
⪰	Rel	\succeq
	Rel	\precneq
	Rel	\succneq
	Rel	\preceqq
	Rel	\succeqq
	Rel	\precneqq
	Rel	\succneqq
	Rel	\precapprox
	Rel	\succapprox
	Rel	\precnapprox
	Rel	\succnapprox
	Rel	\Prec
	Rel	\Succ
	Rel	\subsetdot
	Rel	\supsetdot
	Rel	\subsetplus
	Rel	\supsetplus
	Rel	\submult
	Rel	\supmult
	Rel	\subedot
	Rel	\supedot
	Rel	\subseteqq
	Rel	\supseteqq
	Rel	\subsim
	Rel	\supsim
	Rel	\subsetapprox
	Rel	\supsetapprox
	Rel	\subsetneqq
	Rel	\supsetneqq
	Rel	\lsqhook
	Rel	\rsqhook
	Rel	\csub
	Rel	\csup
	Rel	\csube
	Rel	\csupe
	Rel	\subsup
	Rel	\supsub
	Rel	\subsub
	Rel	\supsup
	Rel	\suphsub
	Rel	\supdsub
	Rel	\forkv
	Rel	\topfork
	Rel	\mlcp
	Rel	\forks
	Rel	\forksnot
	Rel	\shortlefttack
	Rel	\shortdowntack
	Rel	\shortuptack
	Ord	\perps
	Rel	\vDdash
	Rel	\dashV
	Rel	\Dashv
	Rel	\DashV
	Rel	\varVdash
	Rel	\Barv
	Rel	\vBar
	Rel	\vBarv
	Rel	\barV
	Rel	\Vbar
	Rel	\Not
	Rel	\bNot
	Rel	\revnmid
	Rel	\cirmid
	Rel	\midcir
	Ord	\topcir
	Rel	\nhpar
	Rel	\parsim
	Bin	\interleave
	Bin	\nhVvert
	Bin	\threedotcolon
	Rel	\lllnest
	Rel	\gggnest
	Rel	\leqqslant
	Rel	\geqqslant
	Bin	\trslash
	Op	\biginterleave
	Bin	\sslash
	Bin	\talloblong
	Op	\bigtalloblong
	Ord	\squaretopblack
	Ord	\squarebotblack
	Ord	\squareurblack
	Ord	\squarellblack
	Ord	\diamondleftblack
	Ord	\diamondrightblack
	Ord	\diamondtopblack
	Ord	\diamondbotblack
⬚	Ord	\dottedsquare
	Ord	\lgblksquare
	Ord	\lgwhtsquare
	Ord	\vysmblksquare
	Ord	\vysmwhtsquare
	Ord	\pentagonblack
	Ord	\pentagon
	Ord	\varhexagon
	Ord	\varhexagonblack
	Ord	\hexagonblack
	Ord	\lgblkcircle
	Ord	\mdblkdiamond
	Ord	\mdwhtdiamond
	Ord	\mdblklozenge
	Ord	\mdwhtlozenge
	Ord	\smblkdiamond
	Ord	\smblklozenge
	Ord	\smwhtlozenge
	Ord	\blkhorzoval
	Ord	\whthorzoval
	Ord	\blkvertoval
	Ord	\whtvertoval
	Rel	\circleonleftarrow
⬱	Rel	\leftthreearrows
	Rel	\leftarrowonoplus
⬳	Rel	\longleftsquigarrow
	Rel	\nvtwoheadleftarrow
	Rel	\nVtwoheadleftarrow
	Rel	\twoheadmapsfrom
	Rel	\twoheadleftdbkarrow
	Rel	\leftdotarrow
	Rel	\nvleftarrowtail
	Rel	\nVleftarrowtail
	Rel	\twoheadleftarrowtail
	Rel	\nvtwoheadleftarrowtail
	Rel	\nVtwoheadleftarrowtail
	Rel	\leftarrowx
	Rel	\leftcurvedarrow
	Rel	\equalleftarrow
	Rel	\bsimilarleftarrow
	Rel	\leftarrowbackapprox
	Rel	\rightarrowgtr
	Rel	\rightarrowsupset
	Rel	\LLeftarrow
	Rel	\RRightarrow
	Rel	\bsimilarrightarrow
	Rel	\rightarrowbackapprox
	Rel	\similarleftarrow

	Rel	\leftarrowapprox
	Rel	\leftarrowbsimilar
	Rel	\rightarrowbsimilar
	Ord	\medwhitestar
	Ord	\medblackstar
	Ord	\smwhitestar
	Ord	\rightpentagonblack
	Ord	\rightpentagon
	Ord	\postalmark
	Ord	\hzigzag

Codes U+10001 … U+1EEF1

𝐀	Alph	\mbfA
𝐁	Alph	\mbfB
𝐂	Alph	\mbfC
𝐃	Alph	\mbfD
𝐄	Alph	\mbfE
𝐅	Alph	\mbfF
𝐆	Alph	\mbfG
𝐇	Alph	\mbfH
𝐈	Alph	\mbfI
𝐉	Alph	\mbfJ
𝐊	Alph	\mbfK
𝐋	Alph	\mbfL
𝐌	Alph	\mbfM
𝐍	Alph	\mbfN
𝐎	Alph	\mbfO
𝐏	Alph	\mbfP
𝐐	Alph	\mbfQ
𝐑	Alph	\mbfR
𝐒	Alph	\mbfS
𝐓	Alph	\mbfT
𝐔	Alph	\mbfU
𝐕	Alph	\mbfV
𝐖	Alph	\mbfW
𝐗	Alph	\mbfX
𝐘	Alph	\mbfY
𝐙	Alph	\mbfZ
𝐚	Alph	\mbfa
𝐛	Alph	\mbfb
𝐜	Alph	\mbfc
𝐝	Alph	\mbfd
𝐞	Alph	\mbfe
𝐟	Alph	\mbff
𝐠	Alph	\mbfg
𝐡	Alph	\mbfh
𝐢	Alph	\mbfi
𝐣	Alph	\mbfj
𝐤	Alph	\mbfk
𝐥	Alph	\mbfl
𝐦	Alph	\mbfm
𝐧	Alph	\mbfn
𝐨	Alph	\mbfo
𝐩	Alph	\mbfp
𝐪	Alph	\mbfq
𝐫	Alph	\mbfr
𝐬	Alph	\mbfs
𝐭	Alph	\mbft
𝐮	Alph	\mbfu
𝐯	Alph	\mbfv
𝐰	Alph	\mbfw
𝐱	Alph	\mbfx
𝐲	Alph	\mbfy
𝐳	Alph	\mbfz
𝐴	Alph	\mitA
𝐵	Alph	\mitB
𝐶	Alph	\mitC
𝐷	Alph	\mitD
𝐸	Alph	\mitE
𝐹	Alph	\mitF
𝐺	Alph	\mitG
𝐻	Alph	\mitH
𝐼	Alph	\mitI
𝐽	Alph	\mitJ
𝐾	Alph	\mitK
𝐿	Alph	\mitL
𝑀	Alph	\mitM
𝑁	Alph	\mitN
𝑂	Alph	\mitO
𝑃	Alph	\mitP
𝑄	Alph	\mitQ
𝑅	Alph	\mitR
𝑆	Alph	\mitS
𝑇	Alph	\mitT
𝑈	Alph	\mitU
𝑉	Alph	\mitV
𝑊	Alph	\mitW
𝑋	Alph	\mitX
𝑌	Alph	\mitY
𝑍	Alph	\mitZ
𝑎	Alph	\mita
𝑏	Alph	\mitb
𝑐	Alph	\mitc
𝑑	Alph	\mitd
𝑒	Alph	\mite
𝑓	Alph	\mitf
𝑔	Alph	\mitg
𝑖	Alph	\miti
𝑗	Alph	\mitj
𝑘	Alph	\mitk
𝑙	Alph	\mitl
𝑚	Alph	\mitm
𝑛	Alph	\mitn
𝑜	Alph	\mito
𝑝	Alph	\mitp
𝑞	Alph	\mitq
𝑟	Alph	\mitr
𝑠	Alph	\mits
𝑡	Alph	\mitt
𝑢	Alph	\mitu
𝑣	Alph	\mitv
𝑤	Alph	\mitw
𝑥	Alph	\mitx
𝑦	Alph	\mity
𝑧	Alph	\mitz
𝑨	Alph	\mbfitA
𝑩	Alph	\mbfitB
𝑪	Alph	\mbfitC
𝑫	Alph	\mbfitD
𝑬	Alph	\mbfitE
𝑭	Alph	\mbfitF
𝑮	Alph	\mbfitG
𝑯	Alph	\mbfitH
𝑰	Alph	\mbfitI
𝑱	Alph	\mbfitJ
𝑲	Alph	\mbfitK
𝑳	Alph	\mbfitL
𝑴	Alph	\mbfitM
𝑵	Alph	\mbfitN
𝑶	Alph	\mbfitO
𝑷	Alph	\mbfitP
𝑸	Alph	\mbfitQ
𝑹	Alph	\mbfitR
𝑺	Alph	\mbfitS
𝑻	Alph	\mbfitT
𝑼	Alph	\mbfitU
𝑽	Alph	\mbfitV
𝑾	Alph	\mbfitW
𝑿	Alph	\mbfitX
𝒀	Alph	\mbfitY
𝒁	Alph	\mbfitZ
𝒂	Alph	\mbfita
𝒃	Alph	\mbfitb
𝒄	Alph	\mbfitc
𝒅	Alph	\mbfitd
𝒆	Alph	\mbfite
𝒇	Alph	\mbfitf
𝒈	Alph	\mbfitg
𝒉	Alph	\mbfith
𝒊	Alph	\mbfiti
𝒋	Alph	\mbfitj
𝒌	Alph	\mbfitk
𝒍	Alph	\mbfitl
𝒎	Alph	\mbfitm
𝒏	Alph	\mbfitn
𝒐	Alph	\mbfito
𝒑	Alph	\mbfitp
𝒒	Alph	\mbfitq
𝒓	Alph	\mbfitr
𝒔	Alph	\mbfits
𝒕	Alph	\mbfitt
𝒖	Alph	\mbfitu
𝒗	Alph	\mbfitv
𝒘	Alph	\mbfitw
𝒙	Alph	\mbfitx
𝒚	Alph	\mbfity
𝒛	Alph	\mbfitz
𝒜	Alph	\mscrA
𝒞	Alph	\mscrC
𝒟	Alph	\mscrD
𝒢	Alph	\mscrG
𝒥	Alph	\mscrJ
𝒦	Alph	\mscrK
𝒩	Alph	\mscrN
𝒪	Alph	\mscrO
𝒫	Alph	\mscrP
𝒬	Alph	\mscrQ
𝒮	Alph	\mscrS
𝒯	Alph	\mscrT
𝒰	Alph	\mscrU
𝒱	Alph	\mscrV
𝒲	Alph	\mscrW
𝒳	Alph	\mscrX
𝒴	Alph	\mscrY
𝒵	Alph	\mscrZ
	Alph	\mscra
	Alph	\mscrb
	Alph	\mscrc
	Alph	\mscrd
	Alph	\mscrf
	Alph	\mscrh
	Alph	\mscri
	Alph	\mscrj
	Alph	\mscrk
	Alph	\mscrl
	Alph	\mscrm
	Alph	\mscrn
	Alph	\mscrp
	Alph	\mscrq
	Alph	\mscrr
	Alph	\mscrs
	Alph	\mscrt
	Alph	\mscru
	Alph	\mscrv
	Alph	\mscrw
	Alph	\mscrx
	Alph	\mscry
	Alph	\mscrz
𝓐	Alph	\mbfscrA
𝓑	Alph	\mbfscrB
𝓒	Alph	\mbfscrC
𝓓	Alph	\mbfscrD
𝓔	Alph	\mbfscrE
𝓕	Alph	\mbfscrF
𝓖	Alph	\mbfscrG
𝓗	Alph	\mbfscrH
𝓘	Alph	\mbfscrI
𝓙	Alph	\mbfscrJ
𝓚	Alph	\mbfscrK
𝓛	Alph	\mbfscrL
𝓜	Alph	\mbfscrM
𝓝	Alph	\mbfscrN
𝓞	Alph	\mbfscrO
𝓟	Alph	\mbfscrP
𝓠	Alph	\mbfscrQ
𝓡	Alph	\mbfscrR
𝓢	Alph	\mbfscrS
𝓣	Alph	\mbfscrT
𝓤	Alph	\mbfscrU
𝓥	Alph	\mbfscrV
𝓦	Alph	\mbfscrW
𝓧	Alph	\mbfscrX
𝓨	Alph	\mbfscrY
𝓩	Alph	\mbfscrZ
	Alph	\mbfscra
	Alph	\mbfscrb
	Alph	\mbfscrc
	Alph	\mbfscrd
	Alph	\mbfscre
	Alph	\mbfscrf
	Alph	\mbfscrg
	Alph	\mbfscrh
	Alph	\mbfscri
	Alph	\mbfscrj
	Alph	\mbfscrk
	Alph	\mbfscrl
	Alph	\mbfscrm
	Alph	\mbfscrn
	Alph	\mbfscro
	Alph	\mbfscrp
	Alph	\mbfscrq
	Alph	\mbfscrr
	Alph	\mbfscrs
	Alph	\mbfscrt
	Alph	\mbfscru
	Alph	\mbfscrv
	Alph	\mbfscrw
	Alph	\mbfscrx
	Alph	\mbfscry
	Alph	\mbfscrz
𝔄	Alph	\mfrakA
𝔅	Alph	\mfrakB
𝔇	Alph	\mfrakD
𝔈	Alph	\mfrakE
𝔉	Alph	\mfrakF
𝔊	Alph	\mfrakG
𝔍	Alph	\mfrakJ
𝔎	Alph	\mfrakK
𝔏	Alph	\mfrakL
𝔐	Alph	\mfrakM
𝔑	Alph	\mfrakN
𝔒	Alph	\mfrakO
𝔓	Alph	\mfrakP
𝔔	Alph	\mfrakQ
𝔖	Alph	\mfrakS
𝔗	Alph	\mfrakT
𝔘	Alph	\mfrakU
𝔙	Alph	\mfrakV
𝔚	Alph	\mfrakW
𝔛	Alph	\mfrakX
𝔜	Alph	\mfrakY
𝔞	Alph	\mfraka
𝔟	Alph	\mfrakb
𝔠	Alph	\mfrakc
𝔡	Alph	\mfrakd
𝔢	Alph	\mfrake
𝔣	Alph	\mfrakf
𝔤	Alph	\mfrakg
𝔥	Alph	\mfrakh
𝔦	Alph	\mfraki
𝔧	Alph	\mfrakj
𝔨	Alph	\mfrakk

𝔩	Alph	\mfrakl
𝔪	Alph	\mfrakm
𝔫	Alph	\mfrakn
𝔬	Alph	\mfrako
𝔭	Alph	\mfrakp
𝔮	Alph	\mfrakq
𝔯	Alph	\mfrakr
𝔰	Alph	\mfraks
𝔱	Alph	\mfrakt
𝔲	Alph	\mfraku
𝔳	Alph	\mfrakv
𝔴	Alph	\mfrakw
𝔵	Alph	\mfrakx
𝔶	Alph	\mfraky
𝔷	Alph	\mfrakz
𝔸	Alph	\BbbA
𝔹	Alph	\BbbB
𝔻	Alph	\BbbD
𝔼	Alph	\BbbE
𝔽	Alph	\BbbF
𝔾	Alph	\BbbG
𝕀	Alph	\BbbI
𝕁	Alph	\BbbJ
𝕂	Alph	\BbbK
𝕃	Alph	\BbbL
𝕄	Alph	\BbbM
𝕆	Alph	\BbbO
𝕊	Alph	\BbbS
𝕋	Alph	\BbbT
𝕌	Alph	\BbbU
𝕍	Alph	\BbbV
𝕎	Alph	\BbbW
𝕏	Alph	\BbbX
𝕐	Alph	\BbbY
𝕒	Alph	\Bbba
𝕓	Alph	\Bbbb
𝕔	Alph	\Bbbc
𝕕	Alph	\Bbbd
𝕖	Alph	\Bbbe
𝕗	Alph	\Bbbf
𝕘	Alph	\Bbbg
𝕙	Alph	\Bbbh
𝕚	Alph	\Bbbi
𝕛	Alph	\Bbbj
𝕜	Alph	\Bbbk
𝕝	Alph	\Bbbl
𝕞	Alph	\Bbbm
𝕟	Alph	\Bbbn
𝕠	Alph	\Bbbo
𝕡	Alph	\Bbbp
𝕢	Alph	\Bbbq
𝕣	Alph	\Bbbr
𝕤	Alph	\Bbbs
𝕥	Alph	\Bbbt
𝕦	Alph	\Bbbu
𝕧	Alph	\Bbbv
𝕨	Alph	\Bbbw
𝕩	Alph	\Bbbx
𝕪	Alph	\Bbby
𝕫	Alph	\Bbbz
𝕬	Alph	\mbffrakA
𝕭	Alph	\mbffrakB
𝕮	Alph	\mbffrakC
𝕯	Alph	\mbffrakD
𝕰	Alph	\mbffrakE
𝕱	Alph	\mbffrakF
𝕲	Alph	\mbffrakG
𝕳	Alph	\mbffrakH
𝕴	Alph	\mbffrakI
𝕵	Alph	\mbffrakJ
𝕶	Alph	\mbffrakK
𝕷	Alph	\mbffrakL
𝕸	Alph	\mbffrakM
𝕹	Alph	\mbffrakN
𝕺	Alph	\mbffrakO
𝕻	Alph	\mbffrakP
𝕼	Alph	\mbffrakQ
𝕽	Alph	\mbffrakR
𝕾	Alph	\mbffrakS
𝕿	Alph	\mbffrakT
𝖀	Alph	\mbffrakU
𝖁	Alph	\mbffrakV
𝖂	Alph	\mbffrakW
𝖃	Alph	\mbffrakX
𝖄	Alph	\mbffrakY
𝖅	Alph	\mbffrakZ
𝖆	Alph	\mbffraka
𝖇	Alph	\mbffrakb
𝖈	Alph	\mbffrakc
𝖉	Alph	\mbffrakd
𝖊	Alph	\mbffrake
𝖋	Alph	\mbffrakf
𝖌	Alph	\mbffrakg
𝖍	Alph	\mbffrakh
𝖎	Alph	\mbffraki
𝖏	Alph	\mbffrakj
𝖐	Alph	\mbffrakk
𝖑	Alph	\mbffrakl
𝖒	Alph	\mbffrakm
𝖓	Alph	\mbffrakn
𝖔	Alph	\mbffrako
𝖕	Alph	\mbffrakp
𝖖	Alph	\mbffrakq
𝖗	Alph	\mbffrakr
𝖘	Alph	\mbffraks
𝖙	Alph	\mbffrakt
𝖚	Alph	\mbffraku
𝖛	Alph	\mbffrakv
𝖜	Alph	\mbffrakw
𝖝	Alph	\mbffrakx
𝖞	Alph	\mbffraky
𝖟	Alph	\mbffrakz
𝖠	Alph	\msansA
𝖡	Alph	\msansB
𝖢	Alph	\msansC
𝖣	Alph	\msansD
𝖤	Alph	\msansE
𝖥	Alph	\msansF
𝖦	Alph	\msansG
𝖧	Alph	\msansH
𝖨	Alph	\msansI
𝖩	Alph	\msansJ
𝖪	Alph	\msansK
𝖫	Alph	\msansL
𝖬	Alph	\msansM
𝖭	Alph	\msansN
𝖮	Alph	\msansO
𝖯	Alph	\msansP
𝖰	Alph	\msansQ
𝖱	Alph	\msansR
𝖲	Alph	\msansS
𝖳	Alph	\msansT
𝖴	Alph	\msansU
𝖵	Alph	\msansV
𝖶	Alph	\msansW
𝖷	Alph	\msansX
𝖸	Alph	\msansY
𝖹	Alph	\msansZ
𝖺	Alph	\msansa
𝖻	Alph	\msansb
𝖼	Alph	\msansc
𝖽	Alph	\msansd
𝖾	Alph	\msanse
𝖿	Alph	\msansf
𝗀	Alph	\msansg
𝗁	Alph	\msansh
𝗂	Alph	\msansi
𝗃	Alph	\msansj
𝗄	Alph	\msansk
𝗅	Alph	\msansl
𝗆	Alph	\msansm
𝗇	Alph	\msansn
𝗈	Alph	\msanso
𝗉	Alph	\msansp
𝗊	Alph	\msansq
𝗋	Alph	\msansr
𝗌	Alph	\msanss
𝗍	Alph	\msanst
𝗎	Alph	\msansu
𝗏	Alph	\msansv
𝗐	Alph	\msansw
𝗑	Alph	\msansx
𝗒	Alph	\msansy
𝗓	Alph	\msansz
𝗔	Alph	\mbfsansA
𝗕	Alph	\mbfsansB
𝗖	Alph	\mbfsansC
𝗗	Alph	\mbfsansD
𝗘	Alph	\mbfsansE
𝗙	Alph	\mbfsansF
𝗚	Alph	\mbfsansG
𝗛	Alph	\mbfsansH
𝗜	Alph	\mbfsansI
𝗝	Alph	\mbfsansJ
𝗞	Alph	\mbfsansK
𝗟	Alph	\mbfsansL
𝗠	Alph	\mbfsansM
𝗡	Alph	\mbfsansN
𝗢	Alph	\mbfsansO
𝗣	Alph	\mbfsansP
𝗤	Alph	\mbfsansQ
𝗥	Alph	\mbfsansR
𝗦	Alph	\mbfsansS
𝗧	Alph	\mbfsansT
𝗨	Alph	\mbfsansU
𝗩	Alph	\mbfsansV
𝗪	Alph	\mbfsansW
𝗫	Alph	\mbfsansX
𝗬	Alph	\mbfsansY
𝗭	Alph	\mbfsansZ
𝗮	Alph	\mbfsansa
𝗯	Alph	\mbfsansb
𝗰	Alph	\mbfsansc
𝗱	Alph	\mbfsansd
𝗲	Alph	\mbfsanse
𝗳	Alph	\mbfsansf
𝗴	Alph	\mbfsansg
𝗵	Alph	\mbfsansh
𝗶	Alph	\mbfsansi
𝗷	Alph	\mbfsansj
𝗸	Alph	\mbfsansk
𝗹	Alph	\mbfsansl
𝗺	Alph	\mbfsansm
𝗻	Alph	\mbfsansn
𝗼	Alph	\mbfsanso
𝗽	Alph	\mbfsansp
𝗾	Alph	\mbfsansq
𝗿	Alph	\mbfsansr
𝘀	Alph	\mbfsanss
𝘁	Alph	\mbfsanst
𝘂	Alph	\mbfsansu
𝘃	Alph	\mbfsansv
𝘄	Alph	\mbfsansw
𝘅	Alph	\mbfsansx
𝘆	Alph	\mbfsansy
𝘇	Alph	\mbfsansz
𝘈	Alph	\mitsansA
𝘉	Alph	\mitsansB
𝘊	Alph	\mitsansC
𝘋	Alph	\mitsansD
𝘌	Alph	\mitsansE
𝘍	Alph	\mitsansF
𝘎	Alph	\mitsansG
𝘏	Alph	\mitsansH
𝘐	Alph	\mitsansI
𝘑	Alph	\mitsansJ
𝘒	Alph	\mitsansK
𝘓	Alph	\mitsansL
𝘔	Alph	\mitsansM
𝘕	Alph	\mitsansN
𝘖	Alph	\mitsansO
𝘗	Alph	\mitsansP
𝘘	Alph	\mitsansQ
𝘙	Alph	\mitsansR
𝘚	Alph	\mitsansS
𝘛	Alph	\mitsansT
𝘜	Alph	\mitsansU
𝘝	Alph	\mitsansV
𝘞	Alph	\mitsansW
𝘟	Alph	\mitsansX
𝘠	Alph	\mitsansY
𝘡	Alph	\mitsansZ
𝘢	Alph	\mitsansa
𝘣	Alph	\mitsansb
𝘤	Alph	\mitsansc
𝘥	Alph	\mitsansd
𝘦	Alph	\mitsanse
𝘧	Alph	\mitsansf
𝘨	Alph	\mitsansg
𝘩	Alph	\mitsansh
𝘪	Alph	\mitsansi
𝘫	Alph	\mitsansj
𝘬	Alph	\mitsansk
𝘭	Alph	\mitsansl
𝘮	Alph	\mitsansm
𝘯	Alph	\mitsansn
𝘰	Alph	\mitsanso
𝘱	Alph	\mitsansp
𝘲	Alph	\mitsansq
𝘳	Alph	\mitsansr
𝘴	Alph	\mitsanss
𝘵	Alph	\mitsanst
𝘶	Alph	\mitsansu
𝘷	Alph	\mitsansv
𝘸	Alph	\mitsansw
𝘹	Alph	\mitsansx
𝘺	Alph	\mitsansy
𝘻	Alph	\mitsansz
𝘼	Alph	\mbfitsansA
𝘽	Alph	\mbfitsansB
𝘾	Alph	\mbfitsansC
𝘿	Alph	\mbfitsansD
𝙀	Alph	\mbfitsansE
𝙁	Alph	\mbfitsansF
𝙂	Alph	\mbfitsansG
𝙃	Alph	\mbfitsansH
𝙄	Alph	\mbfitsansI
𝙅	Alph	\mbfitsansJ
𝙆	Alph	\mbfitsansK
𝙇	Alph	\mbfitsansL
𝙈	Alph	\mbfitsansM
𝙉	Alph	\mbfitsansN
𝙊	Alph	\mbfitsansO
𝙋	Alph	\mbfitsansP
𝙌	Alph	\mbfitsansQ
𝙍	Alph	\mbfitsansR
𝙎	Alph	\mbfitsansS
𝙏	Alph	\mbfitsansT
𝙐	Alph	\mbfitsansU
𝙑	Alph	\mbfitsansV
𝙒	Alph	\mbfitsansW
𝙓	Alph	\mbfitsansX
𝙔	Alph	\mbfitsansY
𝙕	Alph	\mbfitsansZ
𝙖	Alph	\mbfitsansa
𝙗	Alph	\mbfitsansb
𝙘	Alph	\mbfitsansc
𝙙	Alph	\mbfitsansd
𝙚	Alph	\mbfitsanse
𝙛	Alph	\mbfitsansf
𝙜	Alph	\mbfitsansg
𝙝	Alph	\mbfitsansh
𝙞	Alph	\mbfitsansi
𝙟	Alph	\mbfitsansj
𝙠	Alph	\mbfitsansk
𝙡	Alph	\mbfitsansl
𝙢	Alph	\mbfitsansm
𝙣	Alph	\mbfitsansn

𝙤	Alph	\mbfitsanso
𝙥	Alph	\mbfitsansp
𝙦	Alph	\mbfitsansq
𝙧	Alph	\mbfitsansr
𝙨	Alph	\mbfitsanss
𝙩	Alph	\mbfitsanst
𝙪	Alph	\mbfitsansu
𝙫	Alph	\mbfitsansv
𝙬	Alph	\mbfitsansw
𝙭	Alph	\mbfitsansx
𝙮	Alph	\mbfitsansy
𝙯	Alph	\mbfitsansz
𝙰	Alph	\mttA
𝙱	Alph	\mttB
𝙲	Alph	\mttC
𝙳	Alph	\mttD
𝙴	Alph	\mttE
𝙵	Alph	\mttF
𝙶	Alph	\mttG
𝙷	Alph	\mttH
𝙸	Alph	\mttI
𝙹	Alph	\mttJ
𝙺	Alph	\mttK
𝙻	Alph	\mttL
𝙼	Alph	\mttM
𝙽	Alph	\mttN
𝙾	Alph	\mttO
𝙿	Alph	\mttP
𝚀	Alph	\mttQ
𝚁	Alph	\mttR
𝚂	Alph	\mttS
𝚃	Alph	\mttT
𝚄	Alph	\mttU
𝚅	Alph	\mttV
𝚆	Alph	\mttW
𝚇	Alph	\mttX
𝚈	Alph	\mttY
𝚉	Alph	\mttZ
𝚊	Alph	\mtta
𝚋	Alph	\mttb
𝚌	Alph	\mttc
𝚍	Alph	\mttd
𝚎	Alph	\mtte
𝚏	Alph	\mttf
𝚐	Alph	\mttg
𝚑	Alph	\mtth
𝚒	Alph	\mtti
𝚓	Alph	\mttj
𝚔	Alph	\mttk
𝚕	Alph	\mttl
𝚖	Alph	\mttm
𝚗	Alph	\mttn
𝚘	Alph	\mtto
𝚙	Alph	\mttp
𝚚	Alph	\mttq
𝚛	Alph	\mttr
𝚜	Alph	\mtts
𝚝	Alph	\mttt
𝚞	Alph	\mttu
𝚟	Alph	\mttv
𝚠	Alph	\mttw
𝚡	Alph	\mttx
𝚢	Alph	\mtty
𝚣	Alph	\mttz
𝚤	Alph	\imath
𝚥	Alph	\jmath
𝚨	Alph	\mbfAlpha
𝚩	Alph	\mbfBeta
𝚪	Alph	\mbfGamma
𝚫	Alph	\mbfDelta
𝚬	Alph	\mbfEpsilon
𝚭	Alph	\mbfZeta
𝚮	Alph	\mbfEta
𝚯	Alph	\mbfTheta
𝚰	Alph	\mbfIota
𝚱	Alph	\mbfKappa
𝚲	Alph	\mbfLambda
𝚳	Alph	\mbfMu
𝚴	Alph	\mbfNu
𝚵	Alph	\mbfXi
𝚶	Alph	\mbfOmicron
𝚷	Alph	\mbfPi
𝚸	Alph	\mbfRho
𝚹	Alph	\mbfvarTheta
𝚺	Alph	\mbfSigma
𝚻	Alph	\mbfTau
𝚼	Alph	\mbfUpsilon
𝚽	Alph	\mbfPhi
𝚾	Alph	\mbfChi
𝚿	Alph	\mbfPsi
𝛀	Alph	\mbfOmega
𝛁	Alph	\mbfnabla
𝛂	Alph	\mbfalpha
𝛃	Alph	\mbfbeta
𝛄	Alph	\mbfgamma
𝛅	Alph	\mbfdelta
𝛆	Alph	\mbfvarepsilon
𝛇	Alph	\mbfzeta
𝛈	Alph	\mbfeta
𝛉	Alph	\mbftheta
𝛊	Alph	\mbfiota
𝛋	Alph	\mbfkappa
𝛌	Alph	\mbflambda
𝛍	Alph	\mbfmu
𝛎	Alph	\mbfnu
𝛏	Alph	\mbfxi
𝛐	Alph	\mbfomicron
𝛑	Alph	\mbfpi
𝛒	Alph	\mbfrho
𝛓	Alph	\mbfvarsigma
𝛔	Alph	\mbfsigma
𝛕	Alph	\mbftau
𝛖	Alph	\mbfupsilon
𝛗	Alph	\mbfvarphi
𝛘	Alph	\mbfchi
𝛙	Alph	\mbfpsi
𝛚	Alph	\mbfomega
𝛛	Alph	\mbfpartial
𝛜	Alph	\mbfepsilon
𝛝	Alph	\mbfvartheta
𝛞	Alph	\mbfvarkappa
𝛟	Alph	\mbfphi
𝛠	Alph	\mbfvarrho
𝛡	Alph	\mbfvarpi
𝛢	Alph	\mitAlpha
𝛣	Alph	\mitBeta
𝛤	Alph	\mitGamma
𝛥	Alph	\mitDelta
𝛦	Alph	\mitEpsilon
𝛧	Alph	\mitZeta
𝛨	Alph	\mitEta
𝛩	Alph	\mitTheta
𝛪	Alph	\mitIota
𝛫	Alph	\mitKappa
𝛬	Alph	\mitLambda
𝛭	Alph	\mitMu
𝛮	Alph	\mitNu
𝛯	Alph	\mitXi
𝛰	Alph	\mitOmicron
𝛱	Alph	\mitPi
𝛲	Alph	\mitRho
𝛳	Alph	\mitvarTheta
𝛴	Alph	\mitSigma
𝛵	Alph	\mitTau
𝛶	Alph	\mitUpsilon
𝛷	Alph	\mitPhi
𝛸	Alph	\mitChi
𝛹	Alph	\mitPsi
𝛺	Alph	\mitOmega
𝛻	Alph	\mitnabla
𝛼	Alph	\mitalpha
𝛽	Alph	\mitbeta
𝛾	Alph	\mitgamma
𝛿	Alph	\mitdelta
𝜀	Alph	\mitvarepsilon
𝜁	Alph	\mitzeta
𝜂	Alph	\miteta
𝜃	Alph	\mittheta
𝜄	Alph	\mitiota
𝜅	Alph	\mitkappa
𝜆	Alph	\mitlambda
𝜇	Alph	\mitmu
𝜈	Alph	\mitnu
𝜉	Alph	\mitxi
𝜊	Alph	\mitomicron
𝜋	Alph	\mitpi
𝜌	Alph	\mitrho
𝜍	Alph	\mitvarsigma
𝜎	Alph	\mitsigma
𝜏	Alph	\mittau
𝜐	Alph	\mitupsilon
𝜑	Alph	\mitvarphi
𝜒	Alph	\mitchi
𝜓	Alph	\mitpsi
𝜔	Alph	\mitomega
𝜕	Alph	\mitpartial
𝜖	Alph	\mitepsilon
𝜗	Alph	\mitvartheta
𝜘	Alph	\mitvarkappa
𝜙	Alph	\mitphi
𝜚	Alph	\mitvarrho
𝜛	Alph	\mitvarpi
𝜜	Alph	\mbfitAlpha
𝜝	Alph	\mbfitBeta
𝜞	Alph	\mbfitGamma
𝜟	Alph	\mbfitDelta
𝜠	Alph	\mbfitEpsilon
𝜡	Alph	\mbfitZeta
𝜢	Alph	\mbfitEta
𝜣	Alph	\mbfitTheta
𝜤	Alph	\mbfitIota
𝜥	Alph	\mbfitKappa
𝜦	Alph	\mbfitLambda
𝜧	Alph	\mbfitMu
𝜨	Alph	\mbfitNu
𝜩	Alph	\mbfitXi
𝜪	Alph	\mbfitOmicron
𝜫	Alph	\mbfitPi
𝜬	Alph	\mbfitRho
𝜭	Alph	\mbfitvarTheta
𝜮	Alph	\mbfitSigma
𝜯	Alph	\mbfitTau
𝜰	Alph	\mbfitUpsilon
𝜱	Alph	\mbfitPhi
𝜲	Alph	\mbfitChi
𝜳	Alph	\mbfitPsi
𝜴	Alph	\mbfitOmega
𝜵	Alph	\mbfitnabla
𝜶	Alph	\mbfitalpha
𝜷	Alph	\mbfitbeta
𝜸	Alph	\mbfitgamma
𝜹	Alph	\mbfitdelta
𝜺	Alph	\mbfitvarepsilon
𝜻	Alph	\mbfitzeta
𝜼	Alph	\mbfiteta
𝜽	Alph	\mbfittheta
𝜾	Alph	\mbfitiota
𝜿	Alph	\mbfitkappa
𝝀	Alph	\mbfitlambda
𝝁	Alph	\mbfitmu
𝝂	Alph	\mbfitnu
𝝃	Alph	\mbfitxi
𝝄	Alph	\mbfitomicron
𝝅	Alph	\mbfitpi
𝝆	Alph	\mbfitrho
𝝇	Alph	\mbfitvarsigma
𝝈	Alph	\mbfitsigma
𝝉	Alph	\mbfittau
𝝊	Alph	\mbfitupsilon
𝝋	Alph	\mbfitvarphi
𝝌	Alph	\mbfitchi
𝝍	Alph	\mbfitpsi
𝝎	Alph	\mbfitomega
𝝏	Alph	\mbfitpartial
𝝐	Alph	\mbfitepsilon
𝝑	Alph	\mbfitvartheta
𝝒	Alph	\mbfitvarkappa
𝝓	Alph	\mbfitphi
𝝔	Alph	\mbfitvarrho
𝝕	Alph	\mbfitvarpi
𝝖	Alph	\mbfsansAlpha
𝝗	Alph	\mbfsansBeta
𝝘	Alph	\mbfsansGamma
𝝙	Alph	\mbfsansDelta
𝝚	Alph	\mbfsansEpsilon
𝝛	Alph	\mbfsansZeta
𝝜	Alph	\mbfsansEta
𝝝	Alph	\mbfsansTheta
𝝞	Alph	\mbfsansIota
𝝟	Alph	\mbfsansKappa
𝝠	Alph	\mbfsansLambda
𝝡	Alph	\mbfsansMu
𝝢	Alph	\mbfsansNu
𝝣	Alph	\mbfsansXi
𝝤	Alph	\mbfsansOmicron
𝝥	Alph	\mbfsansPi
𝝦	Alph	\mbfsansRho
𝝧	Alph	\mbfsansvarTheta
𝝨	Alph	\mbfsansSigma
𝝩	Alph	\mbfsansTau
𝝪	Alph	\mbfsansUpsilon
𝝫	Alph	\mbfsansPhi
𝝬	Alph	\mbfsansChi
𝝭	Alph	\mbfsansPsi
𝝮	Alph	\mbfsansOmega
𝝯	Alph	\mbfsansnabla
𝝰	Alph	\mbfsansalpha
𝝱	Alph	\mbfsansbeta
𝝲	Alph	\mbfsansgamma
𝝳	Alph	\mbfsansdelta
𝝴	Alph	\mbfsansvarepsilon
𝝵	Alph	\mbfsanszeta
𝝶	Alph	\mbfsanseta
𝝷	Alph	\mbfsanstheta
𝝸	Alph	\mbfsansiota
𝝹	Alph	\mbfsanskappa
𝝺	Alph	\mbfsanslambda
𝝻	Alph	\mbfsansmu
𝝼	Alph	\mbfsansnu
𝝽	Alph	\mbfsansxi
𝝾	Alph	\mbfsansomicron
𝝿	Alph	\mbfsanspi
𝞀	Alph	\mbfsansrho
𝞁	Alph	\mbfsansvarsigma
𝞂	Alph	\mbfsanssigma
𝞃	Alph	\mbfsanstau
𝞄	Alph	\mbfsansupsilon
𝞅	Alph	\mbfsansvarphi
𝞆	Alph	\mbfsanschi
𝞇	Alph	\mbfsanspsi
𝞈	Alph	\mbfsansomega
𝞉	Alph	\mbfsanspartial
𝞊	Alph	\mbfsansepsilon
𝞋	Alph	\mbfsansvartheta
𝞌	Alph	\mbfsansvarkappa
𝞍	Alph	\mbfsansphi
𝞎	Alph	\mbfsansvarrho
𝞏	Alph	\mbfsansvarpi
𝞐	Alph	\mbfitsansAlpha
𝞑	Alph	\mbfitsansBeta
𝞒	Alph	\mbfitsansGamma
𝞓	Alph	\mbfitsansDelta
𝞔	Alph	\mbfitsansEpsilon
𝞕	Alph	\mbfitsansZeta
𝞖	Alph	\mbfitsansEta
𝞗	Alph	\mbfitsansTheta
𝞘	Alph	\mbfitsansIota
𝞙	Alph	\mbfitsansKappa

𝞚	Alph	\mbfitsansLambda	𝞴	Alph	\mbfitsanslambda	𝟐	Ord	\mbftwo	𝟪	Ord	\msanseight
𝞛	Alph	\mbfitsansMu	𝞵	Alph	\mbfitsansmu	𝟑	Ord	\mbfthree	𝟫	Ord	\msansnine
𝞜	Alph	\mbfitsansNu	𝞶	Alph	\mbfitsansnu	𝟒	Ord	\mbffour	𝟬	Ord	\mbfsanszero
𝞝	Alph	\mbfitsansXi	𝞷	Alph	\mbfitsansxi	𝟓	Ord	\mbffive	𝟭	Ord	\mbfsansone
𝞞	Alph	\mbfitsansOmicron	𝞸	Alph	\mbfitsansomicron	𝟔	Ord	\mbfsix	𝟮	Ord	\mbfsanstwo
𝞟	Alph	\mbfitsansPi	𝞹	Alph	\mbfitsanspi	𝟕	Ord	\mbfseven	𝟯	Ord	\mbfsansthree
𝞠	Alph	\mbfitsansRho	𝞺	Alph	\mbfitsansrho	𝟖	Ord	\mbfeight	𝟰	Ord	\mbfsansfour
𝞡	Alph	\mbfitsansvarTheta	𝞻	Alph	\mbfitsansvarsigma	𝟗	Ord	\mbfnine	𝟱	Ord	\mbfsansfive
𝞢	Alph	\mbfitsansSigma	𝞼	Alph	\mbfitsanssigma	𝟘	Ord	\Bbbzero	𝟲	Ord	\mbfsanssix
𝞣	Alph	\mbfitsansTau	𝞽	Alph	\mbfitsanstau	𝟙	Ord	\Bbbone	𝟳	Ord	\mbfsansseven
𝞤	Alph	\mbfitsansUpsilon	𝞾	Alph	\mbfitsansupsilon	𝟚	Ord	\Bbbtwo	𝟴	Ord	\mbfsanseight
𝞥	Alph	\mbfitsansPhi	𝞿	Alph	\mbfitsansvarphi	𝟛	Ord	\Bbbthree	𝟵	Ord	\mbfsansnine
𝞦	Alph	\mbfitsansChi	𝟀	Alph	\mbfitsanschi	𝟜	Ord	\Bbbfour	𝟶	Ord	\mttzero
𝞧	Alph	\mbfitsansPsi	𝟁	Alph	\mbfitsanspsi	𝟝	Ord	\Bbbfive	𝟷	Ord	\mttone
𝞨	Alph	\mbfitsansOmega	𝟂	Alph	\mbfitsansomega	𝟞	Ord	\Bbbsix	𝟸	Ord	\mtttwo
𝞩	Alph	\mbfitsansnabla	𝟃	Alph	\mbfitsanspartial	𝟟	Ord	\Bbbseven	𝟹	Ord	\mttthree
𝞪	Alph	\mbfitsansalpha	𝟄	Alph	\mbfitsansepsilon	𝟠	Ord	\Bbbeight	𝟺	Ord	\mttfour
𝞫	Alph	\mbfitsansbeta	𝟅	Alph	\mbfitsansvartheta	𝟡	Ord	\Bbbnine	𝟻	Ord	\mttfive
𝞬	Alph	\mbfitsansgamma	𝟆	Alph	\mbfitsansvarkappa	𝟢	Ord	\msanszero	𝟼	Ord	\mttsix
𝞭	Alph	\mbfitsansdelta	𝟇	Alph	\mbfitsansphi	𝟣	Ord	\msansone	𝟽	Ord	\mttseven
𝞮	Alph	\mbfitsansvarepsilon	𝟈	Alph	\mbfitsansvarrho	𝟤	Ord	\msanstwo	𝟾	Ord	\mtteight
𝞯	Alph	\mbfitsanszeta	𝟉	Alph	\mbfitsansvarpi	𝟥	Ord	\msansthree	𝟿	Ord	\mttnine
𝞰	Alph	\mbfitsanseta		Alph	\mbfDigamma	𝟦	Ord	\msansfour		Op	\arabicmaj
𝞱	Alph	\mbfitsanstheta		Alph	\mbfdigamma	𝟧	Ord	\msansfive		Op	\arabichad
𝞲	Alph	\mbfitsansiota	𝟎	Ord	\mbfzero	𝟨	Ord	\msanssix			
𝞳	Alph	\mbfitsanskappa	𝟏	Ord	\mbfone	𝟩	Ord	\msansseven			

It isn't very comfortable to find something in the previous table if you know the shape. You can try the online web tool Detexify. You can draw the symbol here and the suggestion of TEX sequence is printed.

The generic TEX sequences for the Greek letters can be used: `\alpha` α, `\beta` β, `\gamma` γ, `\delta` δ, `\varepsilon` ε, `\zeta` ζ, `\eta` η, `\theta` θ, `\iota` ι, `\kappa` κ, `\lambda` λ, `\mu` μ, `\nu` ν, `\xi` ξ, `\omicron` o, `\pi` π, `\rho` ρ, `\varsigma` ς, `\sigma` σ, `\tau` τ, `\upsilon` υ, `\varphi` φ, `\chi` χ, `\psi` ψ, `\omega` ω, `\vardelta` δ, `\epsilon` ϵ, `\vartheta` ϑ, `\varkappa` $\varkappa$, `\phi` ϕ, `\varrho` ϱ, `\varpi` ϖ and `\Alpha` A, `\Beta` B, `\Gamma` Γ, `\Delta` Δ, `\Epsilon` E, `\Zeta` Z, `\Eta` H, `\Theta` Θ, `\Iota` I, `\Kappa` K, `\Lambda` Λ, `\Mu` M, `\Nu` N, `\Xi` Ξ, `\Omicron` O, `\Pi` Π, `\Rho` P, `\Sigma` Σ, `\Tau` T, `\Upsilon` Υ, `\Phi` Φ, `\Chi` X, `\Psi` Ψ, `\Omega` Ω.

The variant of Greek letters in the output (upright, italic, bold, bold sans serif, etc.) written by the sequences `\alpha`, `\beta` etc. depends on the math alphabet selected by `\_itgreek`, `\_rmgreek`, etc. selectors. The user-level selectors `\bf` and `\bi` set `\_bsansgreek` and `\_bisansgreek`, so `{\bi\delta}` produces 𝞭. ◀ OpTEX

All characters available in the math font can be accessed by TEX control sequence or by directly using the Unicode character in the document source. Example: ◀ Unicode

```
$$
   \sum_{k=0}^\infty e^{(\alpha+i\beta_k)} =
   e^\alpha \sum_{k=0}^\infty e^{i\beta_k} =
   e^\alpha \sum_{k=0}^\infty (\cos\beta_k + i\sin\beta_k).
$$
```

or

```
$$
   ∑_{k=0}^∞ e^{(α + iβ_k)} = e^α ∑_{k=0}^∞ e^{iβ_k}
                            = e^α ∑_{k=0}^∞ (\cos β_k + i\sin β_k).
$$
```

both gives the same result:

$$\sum_{k=0}^\infty e^{(\alpha+i\beta_k)} = e^\alpha \sum_{k=0}^\infty e^{i\beta_k} = e^\alpha \sum_{k=0}^\infty (\cos\beta_k + i\sin\beta_k).$$

2 Other specialities

2.1 The `\not` prefix

You can apply `\not` before a following math object. The slash / is overprinted such math object, for example `$a \not= b$` gives $a \not= b$.

If there exists a direct Unicode character for the negation of a relation symbol (for example ◀ OpTEX `\ne` creates $\ne$ directly as a character U+2260) then `\not`⟨*char*⟩ expands to appropriate Unicode character. For example `\not=` expands to `\ne` or `\not\in` expands to `\notin`. If such character does not exist then the centered / is overprinted over the next character.

2.2 The `\buildrel` macro: text over the relation

The macro `\buildrel` ⟨*text*⟩`\over` ⟨*relation*⟩ creates a new atom Rel with the ⟨*relation*⟩ and with the smaller ⟨*text*⟩ above this ⟨*relation*⟩. Example: `$M \buildrel\rm def\over= X\cup Y$` gives $M \buildrel\rm def\over= X\cup Y$.

2.3 Spaces

Spaces between atoms are created automatically as were mentioned in section 1.2. But sometimes you have to help TEX to create appropriate space. You can use following macros:

- `\,` is *thin space* used around Op atoms, after comma, etc.: $|\,|$,
- `\!` is negative thin space,
- `\>` is *medium space* used around Bin atoms: $|\>|$,
- `\;` is *thick space* used around Rel atoms: $|\;|$,
- `\quad` is *em space*: $|\quad|$,
- `\qquad` is *double em space*: $|\qquad|$.

Of course, you can use *direct space* `\␣` which is TEX primitive and gives interword space: $|\ |$ or you can use `\hskip` ⟨*value*⟩ to put arbitrary space.

The space size of `\,`, `\!` resp. `\>`, resp. `\;` is given by `\thinmuskip`, resp. `\medmuskip`, resp. `\thickmuskip` values. You can see in the `plain.tex` file that these default values differ very little in their basic size but there is no stretchability/shrinkability in the `\,` space, there is small stretchability in the `\>` space, and more stretchability in the `\;` space.

The registers `\thinmuskip`, `\medmuskip`, and `\thickmuskip` store so-called *mu values* given by math unit `mu`. It is 1/18 em and this unit depends on the current font size used in the math formula (S or SS styles use smaller font size, the `mu` unit is smaller here). You can use `\mskip` instead `\hskip` or `\mkern` instead `\kern` if you want to use this special `mu` unit. It is allowed only in math mode.

If `\nonscript` precedes `\hskip`, `\mskip`, `\kern`, or `\mkern` then this space is applied only in D or T style.

The `\quad` and `\qquad` spaces have fixed width and they can be used in text mode too. (OpTEX allows to use `\,` in text mode too). Use `\quad` or `\qquad` if you want to separate more formulas created in single math mode. Examples of typical usage of spaces:

```
$$ \alpha\,(x+y), \qquad \int_a^b \!\! f(x)\,{\rm d}x, \qquad \Gamma_{\!i}. $$
```

$$\alpha\,(x+y), \qquad \int_a^b \!\! f(x)\,{\rm d}x, \qquad \Gamma_{\!i}.$$

2.4 Texts in math mode

If you write `$Hello world!$` (i.e. Hello world in math mode), then you get $Hello world!$. It is interpreted as the product of variables H, and e, and l^2, and o, etc., followed by the symbol ! used for factorial. The non-ASCII letters (with accents) don't work at all because they are never used as symbols for variables. Spaces are ignored.

If you want to write a short text in the math mode, then you can use `\hbox{`⟨*text*⟩`}`. The `\hbox` primitive initializes text mode regardless of the "outer mode". Example:

```
$$ \sum_{n=0}^\infty (-1)^n a_n \hbox{ converges, if $a_n\searrow 0$.} $$
```

$$\sum_{n=0}^\infty (-1)^n a_n \hbox{ converges, if } a_n\searrow 0.$$

Note the space before the word “converges”. The space before `\hbox` is irrelevant. Second notice: the example shows the text mode inside math mode and the in-line math mode inside this text mode. The same result can be produced by:

```
$$ \sum_{n=0}^\infty (-1)^n a_n \hbox{ converges, if } a_n\searrow 0. $$
```

The difference can be visible if the formula $a_n \searrow 0$ includes a fraction, for example $\frac12 a_n \searrow 0$. The first example prints the fraction in the text style and the second example prints it in the display style.

The disadvantage of `\hbox` is that it starts in the text mode independently of the current style, but we want to use smaller font in S or SS styles. You can use `\mathbox{`⟨*text*⟩`}` in such situations. This macro behaves like `\hbox` but the text is appropriately smaller in S and SS styles. Example: ◀ OpTEX

`{\mathbox{cena}\over\mathbox{výkon}}` gives $\displaystyle{\hbox{cena}\over\hbox{výkon}}$ in D style and $\textstyle{\hbox{cena}\over\hbox{výkon}}$ in T style.

Note that $\frac{\hbox{cena}}{\hbox{výkon}}$ means $\frac{\rm price}{\rm performance}$ and you can write `$\rm price\over performance$` when you are using only words without spaces and accented letters. But phrases with spaces or accented letters should be printed in text mode using `\hbox` or `\mathbox`.

2.5 `\vcenter`

The `\vcenter` primitive behaves like `\vbox`, but it can be used only in math mode and its result is vertically centered to the math axis. For example, matrices, are created by tables in `\vcenter`.

All big objects in math formula are centered on the math axis and the baseline is ignored. In the following example, we create a new big math operator by `\vcenter`:

```
$$
  \def\myop#1{\mathop{\vcenter{\frame{\vbox to2em{\vss\hbox{ $#1$ }\vss}}}}}
  F(x) = \myop{x(i)}_{i=1}^\infty
$$
```

$$F(x) = \mathop{\boxed{x(i)}}_{i=1}^\infty$$

2.6 Three dots

You can write `$1,2,\dots,n$` to get $1,2,\dots,n$. The `\dots` macro puts thin space between dots and after the last dot, so the five object: comma, dots, comma are exactly equidistant.

Typographic conventions say that you have to use the repeating symbol before and after three dots (comma in the previous example) and the three dots should be at baseline if the repeating symbol is at baseline. Or they should be at the math axis if the repeating symbol is at the math axis. We have to use `\cdots` instead `\dots` in the second case. Example:

`a_1, a_2, \dots, a_n, \quad a_1 + a_2 + \cdots + a_n` $a_1, a_2, \dots, a_n, \quad a_1 + a_2 + \cdots + a_n$

There are `\vdots` $\vdots$, `\ddots` $\ddots$ and `\adots` ⋰ which can be used in matrices.

Three dots like the output of the `\dots` macro are present as a single character in fonts too. This character is called ellipsis. Font designers typically suggest this character with smaller spaces between dots than we need in math mode. So the rule about equidistant “comma, three dots, comma” is not met when this character is used. You can try `$1,2,\unicodeellipsis, n$` and `$1+2+\unicodecdots + n$`. You get 1, 2, …, n and 1 + 2 + ⋯ + n. If you feel that this is better, then you can set: `\let\dots=\unicodeellipsis \let\cdots=\unicodecdots`. ◀ Unicode

The Unicode fonts include compact variants `\unicodevdots` ⋮, `\unicodeddots` ⋱ and `\unicodeadots` ⋰ too.

2.7 Phantoms and \smash

The `\phantom{`⟨*math list*⟩`}` macro creates an invisible subformula equal to the formula generated by ⟨*math list*⟩. It has its size, so it can interfere with surrounding visible subformulas. This macro is very useful for aligning with special requirements. Examples are shown in sections 3.1 and 4.4.

The `\vphantom{`⟨*math list*⟩`}` has only its vertical size (i.e. its height and depth), the width is zero. The `\hphantom{`⟨*math list*⟩`}` has only its width.

The `\smash{`⟨*math list*⟩`}` is the opposite of `\vphantom`. It creates visible subformula but it has only its width. Its height and depth are zero.

The result of these macros is the nucleus of an Ord atom. You can use `\mathop`, `\mathbin`, etc. primitives to change this class. For example `$a\mathrel{\phantom{=}}b$` creates the same formula as $a = b$ with the same distance from a to b but without the equal sign: $a \mathrel{} b$.

These macros work in the text mode too. Then their argument is the ⟨*horizontal list*⟩.

3 Structured objects

3.1 Matrices

The macro `\matrix{`⟨*data*⟩`}` creates a vertically centered table of items. The ⟨*data*⟩ includes ⟨*items*⟩ separated by `&` and rows are separated by `\cr`. The number of columns and rows are unlimited. Columns are printed centered and separated by the `\quad` space. The vertically scalable brackets around the table are not printed. You can use `\pmatrix{`⟨*data*⟩`}` instead `\matrix{`⟨*data*⟩`}`: the vertically scalable parentheses () are inserted around the table. Examples:

```
$$
  {\bf A} = \pmatrix{a_{1,1} & a_{1,2} & \cdots & a_{1,n} \cr
                     a_{2,1} & a_{2,2} & \cdots & a_{2,n} \cr
                     \vdots  & \vdots  & \ddots & \vdots  \cr
                     a_{m,1} & a_{m,2} & \cdots & a_{m,n} },
$$
```

$$
\mathbf{A} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix},
$$

or:

```
\def\qmatrix[#1]{\left[\matrix{#1}\right]}
$$
  \qmatrix[a&b&c\cr d&e&f\cr g&h&i] \cdot \qmatrix[x_1\cr x_2\cr x_3]
  = \qmatrix[b_1\cr b_2\cr b_3].
$$
```

$$
\begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}.
$$

If you need to align the columns by another way than to center, then you can use the phantom. Compare:

```
$$
  \pmatrix{ 1 & -1 & 0  \cr
            0 &  2 & 13 \cr
           -3 &  0 & 5 }    \quad \hbox{or} \quad
  \def\0{\phantom0} \def\+{\phantom+}
  \pmatrix { \+1 &  -1 & \00 \cr
             \+0 & \+2 &  13 \cr
              -3 & \+0 & \05 }
$$
```

$$\begin{pmatrix} 1 & -1 & 0 \\ 0 & 2 & 13 \\ -3 & 0 & 5 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 1 & -1 & 0 \\ 0 & 2 & 13 \\ -3 & 0 & 5 \end{pmatrix}$$

◀ OpTEX

Another option to set the right aligned matrix is setting the `\lmfil`: Its value is used on the left side in each `\matrix` item. The right side is set directly to `\hfil`.

```
$$
  \lmfil={\hfill}  % left matrix filler = \hfill
  \pmatrix{ 1 & -1 & 0 \cr 0 & 2 & 13 \cr -3 & 0 & 5 }
$$
```

◀ OpTEX

If you want to draw a vertical line inside the matrix, you can use `\adef|` as in the following example:

```
$$
  \adef|{\kern-.2em&\strut\vrule&\kern-.2em}
  \def\+{\phantom+}
  \pmatrix{1 & 2 & 3 | 0 \cr 4 & 5 & 6 | 1 \cr 7 & 8 & 9 | 2 } \sim
  \pmatrix{1 & \+2 & \+3 | 0 \cr 0 & -3 & -6 | 1 \cr 0 & -6 & -12 | 2} \sim
  \pmatrix{1 & 2 & 3 | 0 \cr 0 & 1 & 2 | -1/3 \cr 0 & 0 & 0 | 1 }
$$
```

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 0 \\ 4 & 5 & 6 & 1 \\ 7 & 8 & 9 & 2 \end{array}\right) \sim \left(\begin{array}{ccc|c} 1 & 2 & 3 & 0 \\ 0 & -3 & -6 & 1 \\ 0 & -6 & -12 & 2 \end{array}\right) \sim \left(\begin{array}{ccc|c} 1 & 2 & 3 & 0 \\ 0 & 1 & 2 & -1/3 \\ 0 & 0 & 0 & 1 \end{array}\right)$$

If you want to put something before the opening bracket in the matrix, you can use another `\matrix`. Example:

```
$$
  \adef|{\kern-.2em&\strut\vrule&\kern-.2em}
  \def\+{\phantom+}
  \def\r{{\bf r}}
  \pmatrix{1 & 2 & 3 | 0 \cr 4 & 5 & 6 | 1 \cr 7 & 8 & 9 | 2 } \ \sim \
  \matrix{\cr 2.\r - 4\cdot 1.\r: \cr 3.\r -7\cdot 1.\r: }
  \pmatrix{1 & \+2 & \+3 | 0 \cr 0 & -3 & -6 | 1 \cr 0 & -6 & -12 | 2}
$$
```

$$\left(\begin{array}{ccc|c} 1 & 2 & 3 & 0 \\ 4 & 5 & 6 & 1 \\ 7 & 8 & 9 & 2 \end{array}\right) \ \sim \ \begin{matrix} \\ 2.\mathbf{r} - 4\cdot 1.\mathbf{r}: \\ 3.\mathbf{r} - 7\cdot 1.\mathbf{r}: \end{matrix} \left(\begin{array}{ccc|c} 1 & 2 & 3 & 0 \\ 0 & -3 & -6 & 1 \\ 0 & -6 & -12 & 2 \end{array}\right)$$

Plain TEX defines the `\bordermatrix` macro which allows you to create a top row above the brackets and left column before the opening bracket. TEXbook shows the following example:

```
$$
  M = \bordermatrix{ &C&I&C'\cr
                  C&1&0&0 \cr I&1-b&0 \cr C'&0&a&1-a }
$$
```

$$M = \begin{array}{c} \\ C \\ I \\ C' \end{array} \begin{array}{c} \begin{array}{ccc} C & I & C' \end{array} \\ \begin{pmatrix} 1 & 0 & 0 \\ 1-b & 0 & \\ 0 & a & 1-a \end{pmatrix} \end{array}$$

◀ OpTEX

The `\matrix` macro used in D style creates all its items in T style. If you are using the `\matrix` macro in T style or S style (but not in fractions nor another matrices) then the resulting table is appropriately smaller and all its items are processed in S style. If you are using `\matrix` in

SS style then the items are in SS style too. The following example shows one-column matrix in script style:

`\sum_{\matrix{i\in M\cr j\in N\cr k\in P}} x_{i,j,k}` creates: $\sum_{\matrix{i\in M\cr j\in N\cr k\in P}} x_{i,j,k}$

3.2 Cases

The `\cases` macro can be used as in the following example:

```
$$
  f(x) = \cases{        1       & for $x\le 0$, \cr
                 100-x\over 100 & when $0 < x \le 100$, \cr
                        0       & in other cases. }
$$
```

$$f(x) = \begin{cases} 1 & \text{for } x \le 0, \\ \frac{100-x}{100} & \text{when } 0 < x \le 100, \\ 0 & \text{in other cases.} \end{cases}$$

The `\cases` macro behaves like a special `\matrix` with two left-aligned columns and with left vertically scaled brace {. The first column is processed in math mode and T style, the second column is processed in text mode. We have to use `$...$` in the second column if there is math material.

4 Lines in display mode

4.1 General principles

The `$$`⟨*formula*⟩`$$` finalizes previous paragraph, prints centered ⟨*formula*⟩ on single line with a vertical space above and below and opens next paragraph with no indentation.

From TeX's point of view, the text above `$$`⟨*formula*⟩`$$` plus text below is a single paragraph interrupted by display ⟨*formula*⟩. If there is no text above (i.e. the opening `$$` are in vertical mode), then the internal `\noindent` is processed first and the empty line above ⟨*formula*⟩ is created. Thus, it is definitely a bad idea to open display mode in vertical mode: never put an empty line before `$$`⟨*formula*⟩`$$`. On the other hand, the empty line just after `$$`⟨*formula*⟩`$$` says that the paragraph is finalized by the ⟨*formula*⟩ and the next text (after the empty line) opens the next paragraph with indentation. Summary:

- “Text above `$$`⟨*formula*⟩`$$` text below”: the ⟨*formula*⟩ interrupts the paragraph with “Text above” and “text below”. The “text below” is without indentation.
- “Text above `$$`⟨*formula*⟩`$$` empty line Text below”: the “Text below” opens new paragraph with indentation.
- “Empty line `$$`⟨*formula*⟩`$$`” is bad practice.

In contrast with the paragraph breaking, there is no built-in algorithm for breaking display ⟨*formula*⟩ to more lines. If the ⟨*formula*⟩ is too wide then overfull `\hbox` occurs and human must decide about splitting the ⟨*formula*⟩ to more lines. The macros `\displaylines` and `\eqalign` are intended to such task, see sections 4.3 and 4.4.

On the other hand, the in-line math ⟨*formula*⟩, i.e. the `$`⟨*formula*⟩`$` in a paragraph, can be broken after a Bin atom (with penalty `\binoppenalty`) or after a Rel atom (with penalty `\relpenalty`). If you don't want to break such a formula at a specific place then use `\nobreak`, for example `$a+\nobreak b$`. If you want never to break such formulas then you can set `\binoppenalty=10000`, `\relpenalty=10000`. (Default values are 700 and 500.)

4.2 References to display lines

The `$$`⟨*formula*⟩`\eqno`⟨*mark*⟩`$$` prints centered ⟨*formula*⟩ and the ⟨*mark*⟩ at right margin. The `$$`⟨*formula*⟩`\leqno`⟨*mark*⟩`$$` prints centered ⟨*formula*⟩ and the ⟨*mark*⟩ at left margin. Examples:

```
$$ a^2 + b^2 = c^2  \eqno (1) $$
```

$$a^2 + b^2 = c^2 \tag{1}$$

```
$$ a^2 + b^2 = c^2  \leqno \rm Py $$
```

Py $$a^2 + b^2 = c^2$$

The ⟨*mark*⟩ is processed in math mode and T style. If you want to put a text here then you have to use `\rm` ⟨*text*⟩ or `\hbox{`⟨*text*⟩`}`.

◀ OpTeX

The auto-generated ⟨*mark*⟩ can be created by `$$`⟨*formula*⟩`\eqmark$$`. The ⟨*label*⟩ can be used previously declared by `\label[`⟨*label*⟩`]`. Then it is associated to such auto-generated ⟨*mark*⟩ and you can write a reference in the form `see equation~\ref[`⟨*label*⟩`]`. You can write `\eqmark[`⟨*label*⟩`]` instead of `\label[`⟨*label*⟩`]...\eqmark`. See the OpTeX manual, section 1.4.3.

4.3 The `\displaylines` macro

This macro creates more horizontally centered formulas in one display mode. The syntax is

```
$$ \displaylines{
   ⟨formula⟩ \cr
   ⟨formula⟩ \cr
   ... \cr
   ⟨formula⟩
} $$
```

Usage of `\displaylines` is more prefered than doing more display modes just behind each other. Example:

```
$$ \displaylines{
   (3x^3 + 4x^2 + 5x + 6) \cdot (x^6 + x^2 + 5) = \cr
   = 3x^9 + 4x^8 + 5x^7 + 6x^6 + 3x^5 + 4x^4 + 20x^3 + 26x^2 + 25x + 30.
} $$
```

$$\displaylines{(3x^3 + 4x^2 + 5x + 6) \cdot (x^6 + x^2 + 5) = \\ = 3x^9 + 4x^8 + 5x^7 + 6x^6 + 3x^5 + 4x^4 + 20x^3 + 26x^2 + 25x + 30.}$$

The output from `\displaylines` macro is breakable to more pages because the lines are not encapsulated in one box. The macro uses a special feature of `\halign` primitive. The disadvantage is that you cannot use `\eqno` nor `\leqno` nor `\eqmark`.

◀ OpTeX

OpTeX provides alternative form: `\displaylines to`⟨*dimen*⟩ `{`⟨*data*⟩`}`. Then the centered formulas are encapsulated in a `\vcenter` box of width ⟨*dimen*⟩ and usage of `\eqno` or `\leqno` or `\eqmark` is allowed. The individual lines have the form `\hbox to`⟨*dimen*⟩ and formulas are centered using `\hss` from both sides. This means that you can set arbitrary ⟨*dimen*⟩ without visual change of the formulas. Use smaller ⟨*dimen*⟩ value than `\hsize` (or `\displaywidth`[9]) if you want to center formulas with `\eqno` appended. The internal TeX rule says: the formula with `\eqno` (or `\leqno`) is centered if its width is less or equal `\displaywidth-4`⟨*mark-width*⟩. So

```
$$ \displaylines to\hsize-8em {⟨formula⟩\cr⟨formula⟩}\eqno (1) $$
```

[9] `\displaywith` = `\hsize` in most cases but it is real display width when `\parshape` or `\hangindent` is used.

does global centering, because size of `(1)` is less than 2 em. You can do more experiments with this example, for example `\diplaylines to\hsize{...}` puts the `\eqno` mark to the next line in the display environment. Read TEXbook or TEXbook naruby, where the precise explanation about such positioning is.

You can use "`\displaylines to`⟨*dimen*⟩" for more applications. For example, you can put more "`\displaylines to`⟨*dimen*⟩" in single display mode, one next to second in order to creating more centered columns with formulas; the width of such columns are controlled by the ⟨*dimen*⟩ parameter. Or, you can give an exception for several lines: ◀ OpTEX

```
$$\displaylines to 10cm {⟨formula⟩\hfill \cr
                         ⟨formula⟩\cr
                         ⟨formula⟩\cr
                         \hfill ⟨formula⟩}
$$
```

The example above gives similar result as the LATEX `multline` environment:

$$\begin{multline*}(3x^3+4x^2+5x+6)\cdot(x^6+x^2+5) = \\ = 3x^9+3x^5+15x^3+4x^8+4x^4+20x^2+5x^7+5x^3+25x+6x^6+6x^2+30 = \\ = 3x^9+4x^8+5x^7+6x^6+3x^5+4x^4+20x^3+26x^2+25x+30.\end{multline*}$$

4.4 The `\eqalign` macro

The usage is:

```
$$ \eqalign { ⟨left-side⟩ & ⟨right-side⟩ \cr
              ⟨left-side⟩ & ⟨right-side⟩ \cr
              ... \cr
              ⟨left-side⟩ & ⟨right-side⟩
} $$
```

The `\vcenter` box is created with two columns, left column is right aligned and right column is left aligned. Example:

```
$$ \eqalign{
    x + 2y + 3z &= 600 \cr
   12x + y - 3z &= 7 \cr
    4x - y + 5z &= -5 \cr
} $$
```

$$\begin{aligned} x+2y+3z &= 600 \\ 12x+y-3z &= 7 \\ 4x-y+5z &= -5 \end{aligned}$$

The tab "`&`" should be used just before a relation, i.e. `&=` is right, `=&` is wrong. All lines are aligned to the used tab.

Maybe you want more precise alignment in the example above. You can use `\phantom`:

```
$$ \def\1{\phantom1} \def\+{\phantom+}
   \eqalign{
     x +  2y + 3z &=  600 \cr
   12x + \1y - 3z &= \+7 \cr
    4x - \1y + 5z &=  -5 \cr
} $$
```

$$\begin{aligned} x+2y+3z &= 600 \\ 12x+y-3z &= 7 \\ 4x-y+5z &= -5 \end{aligned}$$

Another typical usage of the `\eqalign` macro:

```
$$ \eqalign{
  p(x)\,q(x) &= (3x^3 + 4x^2 + 5x + 6) \cdot (x^6 + x^2 + 5) = \cr
             &= 3x^9 + 4x^8 + 5x^7 + 6x^6 + 3x^5 + 4x^4 + 20x^3 + 26x^2 + 25x + 30.
} $$
```

$$\begin{aligned} p(x)\,q(x) &= (3x^3+4x^2+5x+6)\cdot(x^6+x^2+5) = \\ &= 3x^9+4x^8+5x^7+6x^6+3x^5+4x^4+20x^3+26x^2+25x+30. \end{aligned}$$

In OpTeX, the `\eqalign` macro is more flexible. You can set the `\baselineskip` value by the `\eqlines` parameter and math style by the `\eqstyle` parameter. For example, you need to put the system of “equations” as a subscript of a sum operator:

```
$$
  \sum_{\eqlines{\baselineskip=.7\baselineskip}\eqstyle{\scriptstyle}\eqalign{
                i &\in A  \cr
                j &\in B\cup C  \cr
                m &\in C }}
   i + j + m
$$
```

$$\sum_{\substack{i\in A\\ j\in B\cup C\\ m\in C}} i+j+m$$

You can write more equation systems one next second:

```
$$\eqalign{
     x + y + z &= 1      &&  a + b + c &= -1 \cr
         u + v &= 20     &&      f + g &= -20 \cr
             i &= j
}$$
```

You can use the third column for centered equations without aligning point. For example:

```
$$ \left( \eqspace=0pt \eqalign{&& ⟨first equation⟩\cr
                                && ⟨second equation⟩} \right) $$
```

The `\eqspace` is additional space used in the third column to separate equation systems one next second.

OpTeX extensions summary:

- `\eqlines` and `\eqstyle` set baselineskip and math style of the formulas.
- `\eqalign` allows more than two columns: The first column is right-aligned (no space). The second is left-aligned (no space). The third column (if used) is centered with `\eqspace/2` at the left and right boundary of the column. The fourth is the same as the first. The fifth is the same as second etc. The number of columns that can be used in `\eqalign` is unlimited.

4.5 The `\eqalign` macro with references

You can give common mark to whole equation system by `$$\eqalign{`⟨*data*⟩`}\eqno` ⟨*mark*⟩`$$`. If you want to give marks to individual lines of the equation system, then you can use another macro: `\eqalignno`. The usage is similar as `\eqalign` but the third column (if used) is intended to the equation mark. Example:

```
$$ \eqalignno{
    x + 2y + 3z &= 600 & \rm(A) \cr
   12x + y - 3z &= 7   & \rm(B) \cr
    4x - y + 5z &= -5  & \rm(C) \cr
} $$
```

$$\begin{aligned} x+2y+3z &= 600 && \text{(A)} \\ 12x+y-3z &= 7 && \text{(B)} \\ 4x-y+5z &= -5 && \text{(C)} \end{aligned}$$

The `\leqalignno` macro is similar to `\eqalignno` but the marks are at the left margin. The OpTeX extensions of `\eqalign` are not available in `\eqalignno` nor `\leqalignno` macros.

◀ OpTeX

You can use auto-generated marks by `\eqmark` macro: `$$\eqalign{`⟨*data*⟩`}\eqmark$$` or:

```
$$ \eqalignno{
     x + 2y + 3z &= 600 & \eqmark[A] \cr
   12x + y - 3z &= 7   & \eqmark[B] \cr
    4x - y + 5z &= -5  & \eqmark[C] \cr
} $$
```

More examples of alignments in display math mode (including comparison with LaTeX code) are shown in the document op-mathalign.pdf.

5 Concept of loading math fonts

5.1 Math families

TeX can use more than one math font in math mode. This was a necessity in the old days when only 128-characters fonts existed. Each math font used in math mode has its *math family* represented by a number. Math family is a collection of three (almost) equal fonts in three sizes: first for `\textstyle` and `\displaystyle`, second for `\scriptstyle` and third for `\scriptscriptstyle`.

◀ Unicode

When Unicode math font is loaded then it includes all three optical sizes and all characters needed for typesetting math formula. Theoretically, we can use only one math family with this single font. But more math families (i.e. more fonts in math mode) is still possible. You can combine characters from more fonts (Unicode fonts and old TFM fonts together) in one math formula.

◀ OpTeX

OpTeX loads the main Unicode math font into math family 1. The math families 2 and 3 are reserved for specific TeXnical reasons, family 4 is used for `\script` font and families 5, 6, 7, etc. can be used by user for loading more fonts. The default macro for loading math fonts looks like:

```
\_def\_normalunimath{%
    \_loadumathfamily 1 {\_unimathfont}{} % Main Unicode math font
    \_loadmathfamily  4 rsfs              % \script (old TFM font)
    \_setunimathdimens   % set dimen parameters used in math formulas internally
}%
\let\_normalmath=\_normalunimath % this is done when Unicode math is initialized
```

Whenever OpTeX needs to resize math fonts (for example in footnotes or titles), it calls the `\_normalmath` macro to reload all math families to the desired size. If you want to add the next font, you can add `\_loadunimathfamily` ⟨*family*⟩ `{`⟨*Unicode-font*⟩`}{`⟨*features*⟩`}` or `\_loadmathfamily` ⟨*family*⟩ ⟨*TFM-font*⟩ into the `\_normalmath` macro. The example in section 5.3 shows how to do it.

5.2 Two variants of math fonts: normal and bold

All math formulas in the whole document need only one Unicode font (or only one collection of math fonts as mentioned in section 5.1). But this is not really true if titles are in bold font. If a math formula is present in such a bold title then all characters of this formula must be bolder. For example "normal" variables must be in bold italic in titles, symbols like + = must be bold and "normal bold" letters (e.g., indicating vectors in math formula) must be extra bold in titles. It means that all fonts from the collection of math fonts must be bolder. We need a second collection of math fonts with bolder shape. Unfortunately, it is not always available.

◀ OpTeX

If you have bold variant of used Unicode math font, then you can use `\loadboldmath` command, for example:

```
\loadmath     {[xitsmath-regular]}
\loadboldmath {[xitsmath-bold]} \to {[xitsmath-regular]}
```

OpTeX uses `\_normalmath` macro for loading collection of math fonts in “normal” cases and `\_boldmath` macro for bold titles. The font declared by `\loadboldmath` is used in second case. But if the bold variant of the font is not available (this is unfortunately more typical), then OpTeX uses faked bold of main Unicode math font (the `embolden` internal font feature is used). So, the default `\_boldmath` macro defined by OpTeX looks like:

```
\_def\_boldunimath{%
   \_ifx\_unimathboldfont \_undefined
      \_loadumathfamily 1 {\_unimathfont}{embolden=1.7;} % Main faked bold
   \_else
      \_loadumathfamily 1 {\_unimathboldfont}{} % Main real bold font
   \_fi
   \_loadmathfamily 4 rsfs  % rsfs in not in bold, unfortunately
   \_setunimathdimens
}%
\let\_boldmath=\_boldunimath % this is done when Unicode math is initialized
```

5.3 Using additional math fonts

◀ OpTeX

The main Unicode math font is loaded by `\loadmath` and `\loadboldmath` (typically processed when `\fontfam[`⟨*family-name*⟩`]` is declared). Moreover, you can load *additional Unicode math fonts* and you can combine characters from main math font and these additional fonts. This can be usable if the main math font doesn’t include all desired characters or you dislike some character shapes from main font. OpTeX enables to load additional math fonts by `\addUmathfont` and characters from additional fonts can be declared by `\mathchars`. See the section 2.16 of OpTeX documentation for more details about this.

The rest of this section includes an example which shows another task: how to combine main Unicode math font with old 8-bit math font `bbold10.tfm`. This font includes double stroked characters, for example, double stroked plus, double stroked Greek letters and digits. Try to run `pdftex testfont`, then answer the question about the name of the font: `bbold10` and then type command `\table\end`. The `testfont.pdf` is printed with the table of characters of this font. Most of these characters cannot be found in Unicode math fonts.

We can re-define the `\_normalmath` macro by:

```
\addto\_normalmath {\_loadmathfamily 5 bbold }

\_regtfm bbold 0 bbold5 6 bbold7 8.5 bbold10 * % using all bbold*.pfb
\_normalmath  % reload the math fonts collection
```

The string “`bbold`” is declared by `\_regtfm` as a collection of all `bbold*.tfm` fonts, the optical sizes are supported.

Finally, we must to declare new TeX sequences for accessing the characters from the new font, for example:

```
\Umathchardef \bbplus 2 5 "2B
\Umathchardef \bble   3 5 "3C
\Umathchardef \bbge   3 5 "3E
```

The `\Umathchardef` TeX primitive declares new TeX sequence used in math typesetting. The first parameter is a class number (2 means Bin, 3 means Rel, see the table in the section 1.2). The second parameter is a math family number. It is 5, see the redefinition of the `\_normalmath` macro above. The third parameter is a slot in the font. Now you can try to use these characters:

`a \bbplus b \bbge c` gives $a ⧺ b ⩾ c$.

Maybe, you want to declare a special math selector which can be used as `$\bball a+b>c$` in order to get $𝕒 ⧺ 𝕓 ⩾ 𝕔$. Then you can define:

```
\def\bball {\bbchar \Umathcodenum `+ \bbplus
                    \Umathcodenum `> \bbge
                    \Umathcodenum `< \bble }
```

If you want to add all double stroked Greek letters into `\bball` selector, then you can do something like this:

```
\def\setbbgreek #1 {\tmpnum=#1\xargs\setbbgreekA}
\def\setbbgreekA #1{\Umathcode \_ea`#1 0 5 \tmpnum \advance\tmpnum by1 }
\addto\bball {%
   \setbbgreek "0B \alpha \beta \gamma \delta \epsilon \zeta \eta
                   \theta \iota \kappa \lambda \mu \nu \xi \pi \rho \sigma
                   \tau \upsilon \phi \chi ;
   \setbbgreek "7F \omega ;
   \setbbgreek "00 \Gamma \Delta \Theta \Lambda \Xi \Pi \Sigma \Upsilon
                   \Phi \Psi \Omega ; }
```

The `\Umathcode` ⟨*input-code*⟩ ⟨*class*⟩ ⟨*family*⟩ ⟨*font-slot*⟩ primitive is used here. The control sequences `\alpha`, `\beta`, `\gamma` etc. are macros which expand to the Unicode character of appropriate (non-math) Greek letter. We set the Umathcode to such character, for example `` \Umathcode `α 0 5 "0B `` is processed.

The bold variant of the font `bbold*.tfm` is not available, unfortunately. We have to settle for normal version of the font in the `\_boldmath` macro:

```
   \addto \_boldmath {\_loadmathfamily 5 bbold }
```

6 Index

Control sequences listed at pages 9–19 are not mentioned here again.