New sections can be inserted using Insert ▸ Section or ⌥1. There are different levels of sections. At the top level, books are first subdivided into parts and chapters. Articles and chapters of books are next subdivided into sections, subsections and subsubsections. Smaller portions of text can finally be put into paragraphs and subparagraphs. The level of a section can be changed *a posteriori* using the variant system. For instance, you may use ^⇥ in order to change a section into subsection or a subsubsection into a paragraph.

Sections are numbered by default. You may toggle the numbering of individual sections using ^# or the IV icon on the focus toolbar. Paragraphs and subparagraphs form an exception: in most of the standard document styles they are not numbered , but this default behavior can be overridden via Focus ▸ Preferences ▸ Paragraph display numbers.

TeXmacs also provides an appendix tag for appendices at the end of a paper or book. Appendices are numbered A, B, C, … instead of 1, 2, 3, … In a similar vein, books may include a prologue (Insert ▸ Chapter ▸ Prologue) or epilogue (Insert ▸ Chapter ▸ Epilogue).

There are a few special types of unnumbered sections (or chapters) for which the content can be generated automatically: tables of contents, bibliographies, indexes, glossaries, lists of figures, etc. The mechanisms for automatic content generation will be explained in section 6.4.

The subdivision of your document into sections provides a skeleton that is useful for several purposes. We already mentioned the generation of tables of contents. We also recall that the focus bar displays the current section title, whenever the focus is on the entire document. By clicking on this title on the focus bar, TeXmacs opens a menu that allows you to easily jump to any of the other sections. When exporting a document to PDF, a suitable table of contents is also automatically included.

3.5 Quotations and prominent statements

There are many occasions on which a fragment of text needs to catch the eye. Mathematicians like to state their theorems prominently, whereas literary writings may use a special layout for quotations or poetry. High school teachers emphasize the most crucial parts of their courses inside frames. Being a structured editor, TeXmacs encourages you to use special markup in each of these situations. This in particular guarantees a uniform layout.

The markup described in this section is mainly intended for text fragments of medium length. In LaTeX, such markup elements are called *environments*. People with HTML [41] or XML [3] backgrounds prefer the terminology *block markup*. In section 3.7 below, we focus on *inline markup* for shorter pieces of text, such as emphasized definitions of words or monospaced function names inside computer programs.

3.5.1 Theorems

The backbone of mathematical discourse consists of theorems, proofs, and other auxiliary propositions and lemmas. Numerous documents in other areas are structured likewise, along hypotheses, principles, remarks, warnings, etc. In TeXmacs, such structures are called *enunciations*, and you may insert them using Insert ▸ Enunciation.

The most important "theorem-like" enunciations are emphasized, whereas the more casual "remark-like" enunciations are not. TeXmacs also includes several "exercise-like" enunciations for educational purposes, which may use a slightly different layout. Here follows an example of each of these three main types of enunciations:

THEOREM 3.1. *On a Turing machine with sufficiently many tapes, two n-digit numbers can be multiplied in time* $O(n \log n)$.

Note 3.2. The history of multiplication algorithms is a fascinating one, which goes back to ancient Babylon. The above theorem from [23] corresponds to the best currently known algorithm on a sequential computer.

Exercise 3.1. Show that the Euclidean division of a $2n$-digit integer by an n-digit one can also be computed in time $O(n \log n)$.

It is nice to give credits in the very statement of a theorem. From within an enunciation, this can be done using Focus ▸ Due to or the Due to button on the focus bar.

By default, enunciations are numbered. Assuming that your cursor is inside an enunciation, you may remove its number using the toggle Focus ▸ Numbered or the icon IV on the focus toolbar.

Americans use the convention that theorems, lemmas, propositions, remarks, etc. all share the same *counter*. This explains why the above Note is numbered 3.2, even though it is the first Note in this chapter. Some people prefer the "European numbering style", where each kind of enunciation carries its own individual counter. You may select this alternative numbering style using Focus ▸ Preferences ▸ European numbering style.

In books, TeXmacs prefixes the numbers of enunciations by the chapter number. For long articles or books with long chapters, you may prefer to use the section number as the prefix. This can be done using Focus ▸ Preferences ▸ Prefix by section number.

The Focus ▸ Preferences menu of an enunciation contains a few other interesting items that you may wish to try in order to customize the rendering.

3.5.2 Quotations and poetry

Three environments for quotations and poetry have been adapted from LaTeX to TeXmacs. The first one, quote (Insert ▸ Prominent ▸ Quote), should be used for one or more short, one-line, quotations.

Eureka! — *Archimedes*

$E = m c^2$ — *Einstein*

For longer quotations of more than one paragraph, you should rather use quotation (Insert ▸ Prominent ▸ Quotation).

For poetry, you may use the verse tag (Insert ▸ Prominent ▸ Verse). The layout is similar to quotations, but differs when it comes to line breaking. This is best seen when making the paragraph artificially narrow:

Het was in eenen tsinxen
daghe
Dat beede bosch ende haghe
Met groenen loveren waren
bevaen.
Nobel, die coninc, hadde ghe-
daen
Sijn hof crayeren over al
Dat hi waende, hadde hijs
gheval,
Houden ten wel groeten love.
Doe quamen tes sconinx hove
Alle die diere, groet ende
cleene,
Sonder vos Reynaert alleene.
Hi hadde te hove so vele mes-
daen
⋮

Van den vos Reynaerde

3.5.3 Other prominent text

The Insert ▸ Prominent menu also contains various items to single out portions of text in a visually oriented way. First of all, you may change the alignment of a block of text using Insert ▸ Prominent ▸ Left aligned, Insert ▸ Prominent ▸ Centered and Insert ▸ Prominent ▸ Right aligned:

Left aligned

Centered

Right aligned

TeXmacs next provides a series of tags that allow you to surround a block of text by vertical padding, horizontal lines, or a frame. These are illustrated below:

> Use Insert ▸ Prominent ▸ Padded to insert vertical padding around text.

> You may also put horizontal lines around it using Insert ▸ Prominent ▸ Lines around. If you only want a line above or below, then you should use Insert ▸ Prominent ▸ Overlined or Insert ▸ Prominent ▸ Underlined.

> Really important statements can be framed using Insert ▸ Prominent ▸ Framed.

These tags are often used in conjunction with other ones. For instance, we used quotations for each of the above examples in order to make clear where they start and end.

The markup for padding, overlines, underlines, and frames comes with several style options. The precise amount of padding can be changed via Focus ▸ Above and Focus ▸ Below or directly on the focus toolbar. The default amount of padding for all similar padded structures in your document can be specified using Focus ▸ Preferences ▸ Padding above and Focus ▸ Preferences ▸ Padding below. For overlined and underlined structures, you may also change the amount of space that separates the text from the lines. Framed boxes allow you to specify a similar horizontal separation space, as well as a background color.

Ornaments (Insert ▸ Prominent ▸ Ornament) are a more general kind of frames with even more bells and whistles. They are most convenient for laptop presentations, where you may use them for obtaining various graphical effects:

> THEOREM 3.3. *This is a theorem to remember.*

3.6 Item lists

TeXmacs implements three main types of lists, depending on whether the items are numbered or not and whether they contain a brief description.

Using Insert ▸ Itemize you may start a list with unnumbered items. You may optionally select a specific item tag like • (bullets), – (dashes), or → (arrows), to be put in front if the items in the list. Lists may be *nested* inside other lists, as follows:

- First item.
- Now comes the sublist:
  - A subitem.
  - Another one.
- A final item.

The rendering of the default item tag depends on the level of nesting: the • item tag for the outermost level, the ◦ item tag at the second level, and so on.

Numbered lists can be inserted using Insert ▸ Enumerate. Again, you may optionally choose a specific numbering style. Here is an example of an enumeration that was started using Insert ▸ Enumerate ▸ Roman:

I. A first item.

II. A second one.

III. And a last one.

The default numbering style again depends on the nesting level: at the outermost level, most styles use ordinary numbers. At the second level, it is common to use lowercase letters, and so on.

Descriptive lists are started using Insert ▸ Description and they allow you to describe a series of concepts:

Gnu. A hairy but gentle beast.

Gnat. Only lives in a zoo.

Yet once more, $\mathrm{T_EX_{MACS}}$ proposes several presentation styles, some of which are only appropriate if you know beforehand that the keys Gnu, Gnat, etc. are sufficiently short.

Assuming that you are inside an item list, pressing ↩ automatically starts a new item. If you need items that are several paragraphs long, then you can use ⇧↩ in order to start a new paragraph. $\mathrm{T_EX_{MACS}}$ also allows you to turn an unnumbered item list into a numbered one and *vice versa*, using the toggle Focus ▸ Numbered or the icon on the focus toolbar. The default layout of lists is rather spaced out. You may select a more compact rendering style using Focus ▸ Preferences ▸ Compact Lists. The low-level style parameters that control the spacing of icons can also be customized via the style preferences menu Focus ▸ Preferences.

Menu	Tag	Example	Purpose
Strong	strong	this is **important**	Indicate an important region of text
Emphasize	em	the *real* thing	Emphasize a region of text
Definition	dfn	A *gnu* is a horny beast	Definition of some concept
Sample	samp	the ae ligature æ	A sequence of literal characters
Name	name	the LINUX system	The name of a particular thing
Person	person	NIELS, il touche !	The name of a person
Cite	cite*	Melville's *Moby Dick*	A bibliographic citation
Abbreviation	abbr	I work at the C.N.R.S.	An abbreviation
Acronym	acronym	the HTML format	An acronym
Verbatim	verbatim	the program said `hello`	Verbatim text like program output
Keyboard	kbd	Please type `return`	Text to be entered on a keyboard
Code	code*	`cout << 1+1;` yields 2	Code of a computer program
Variable	var	`cp` *`srcfile destfile`*	Variables in a computer program
Deleted	deleted	Remove the ~~ti~~po	Mark text that has been deleted
Fill out	fill-out	My name is Sylvie	Text to be filled out
Marked	marked	Remind feeding the cat	A crucial passage

Table 3.7. Common content-based tags and how to obtain them via Insert ▸ Content tag.

3.7 Special textual markup

We have already encountered various *block markup* elements for prominent text of one or more paragraphs (see section 3.5). Similar *inline markup* elements for shorter pieces of text can be found in the submenu Insert ▸ Content tag.

For example, using Insert ▸ Content tag ▸ Strong, you can mark important words or concepts. The default rendering of **important text** uses a bold typeface, but certain document styles may opt for an alternative presentation. For this reason, you should *not* use the strong tag as a substitute for "bold rendering", but only for the precise purpose of singling out important fragments of text.

We have listed the most common inline content tags in Table 3.7. Due to name clashes, some tags do not carry the most intuitive names: the block tag definition is one of the standard TEXMACS enunciations (see section 3.5.1), the tag cite for citations based on bibliographic databases (see section 6.5), and code for multiple lines of code.

In addition to the above inline content markup, TEXMACS also provides several kinds of presentation markup. First of all, tags for several standard font sizes can be found in Insert ▸ Size tag. These sizes are relative to the size of the main document font. For instance, assuming that the main document is composed using a 10 point font, "small" text will use an 8 point font; if the base font size is 12 point, then "small" text rather corresponds to a 10 point font. See Table 3.8 for the available size tags. Notice that you may use ^⇥ and ^⇧⇥ to cycle among the various font sizes.

Size tag	Example
really-tiny	Really tiny
tiny	Tiny
very-small	Very small
small	Small
normal-size	Normal

Size tag	Example
large	Large
very-large	Very large
huge	Huge
really-huge	Really huge

Table 3.8. Common size tags available through Insert ▸ Size tag.

Other presentation markup can be found in Insert ▸ Presentation tag: you may use Underline, Overline, and Strike through for underlined, overlined, and ~~crossed out~~ text, respectively. (Although underlining is a common way to emphasize text in handwritten documents, we notice that this practice is considered bad taste for printed material.) Subscripts and superscripts as in 2^{nd} can be obtained using Subscript and Superscript or the keyboard shortcuts _ ⇥ and ^ ⇥. Using Pastel, Greyed, and Light, you may create translucent text with various degrees of transparency. This feature is typically used for laptop presentations in order to anticipate upcoming content. The graphical effect is best noticed when using colored text on background patterns.

Pastel red, green, and blue text
Greyed red, green, and blue text
Light red, green, and blue text
Normal red, green, and blue text

3.8 Fonts

As we have discussed at length by now (see sections 1.3 and 3.7), traditional typesetters tend to use special fonts only for specific, well-identified purposes. Users are recommended to use the appropriate markup for these purposes, rather than changing the font explicitly. Nevertheless, you may sometimes wish to select a particular font yourself, or search for a font with certain characteristics (e.g. a bold handwritten font).

3.8.1 Font management

Before you select a nominal font, beware of the fact that high quality fonts are really collections of fonts. It is quite common that a font comes with bold and italic variants. Rich font collections may implement even more variants: sans serif, `monospaced`, SMALL CAPITALS, etc. In addition, modern UNICODE fonts do not only provide glyphs for English text, but also for accented characters and other scripts (Greek, Cyrillic, Chinese, etc.). Similarly, scientific documents with mathematical formulas may require a UNICODE font with a large set of mathematical symbols [64, 2].

In comparison with other non-structured text editors, this idea of *font collections* is particularly important in TeXmacs. Indeed, assume that you have selected one of your favorite fonts. Which font is TeXmacs supposed to use when you start a mathematical formula, when a monospaced variant is needed for typesetting a computer program, or when part of your document is written in an exotic language? There are only a very limited number of fonts that allow you to do all this. Examples include the Computer Modern [37] and Stix fonts [54], as well as the TeX Gyre fonts Bonum, Pagella, Schola, Termes, and DejaVu [32].

The above discussion actually raises three main questions. First of all, how to find those font collections that support a certain number of features (e.g. all Cyrillic fonts with bold and italic variants)? Secondly, what to do if a certain feature is not available for the user's favorite font (e.g. typeset a computer program using a font with no monospaced variant)? Finally, how to specify separate fonts for specific tasks (one font for normal text, another one for bold section titles, and yet another one for mathematical formulas)?

TeXmacs partially addresses these problems by maintaining a detailed *font database*. This database contains more information about fonts than the fonts themselves. For instance, TeXmacs is aware which fonts are sans serif, handwritten, outlined, monospaced, etc. The database also contains more subtle technical information about the general shape of the font (its boldness, aspect ratio, slant, ascent, descent, etc.). In the case when a font with certain required properties does not exist, this allows TeXmacs to search for the best available proxy. The same mechanism applies when a particular character is missing in a font.

The downside to the above approach is that part of the information in the database needs to be entered manually. Although the database contains entries for most of the standard fonts available under GNU/LINUX, MACOS, and WINDOWS, it may not recognize any further fonts that you may have purchased or installed on your system.

Another thing to keep in mind when using less common fonts is that TeXmacs comes with only a limited number of preinstalled fonts, such as the Stix fonts, the TeX Gyre fonts, and several fonts prefixed by "TeXmacs". Documents that only use these fonts will be rendered in the same way on different systems (assuming the versions of TeXmacs are identical). Any other fonts may be replaced by closest available proxies according to the font database.

3.8.2 Selecting a font

In TeXmacs, the global document font can be specified using Document ▸ Font. It is also possible to locally use another font using Format ▸ Font. Both Document ▸ Font and Format ▸ Font open the TeXmacs font browser, which looks like the window in Figure 3.2. The standard way to indicate fonts is via their Family, Style and Size (in points). In addition, TeXmacs provides several filters that allow you to search for fonts that meet one or more criteria.

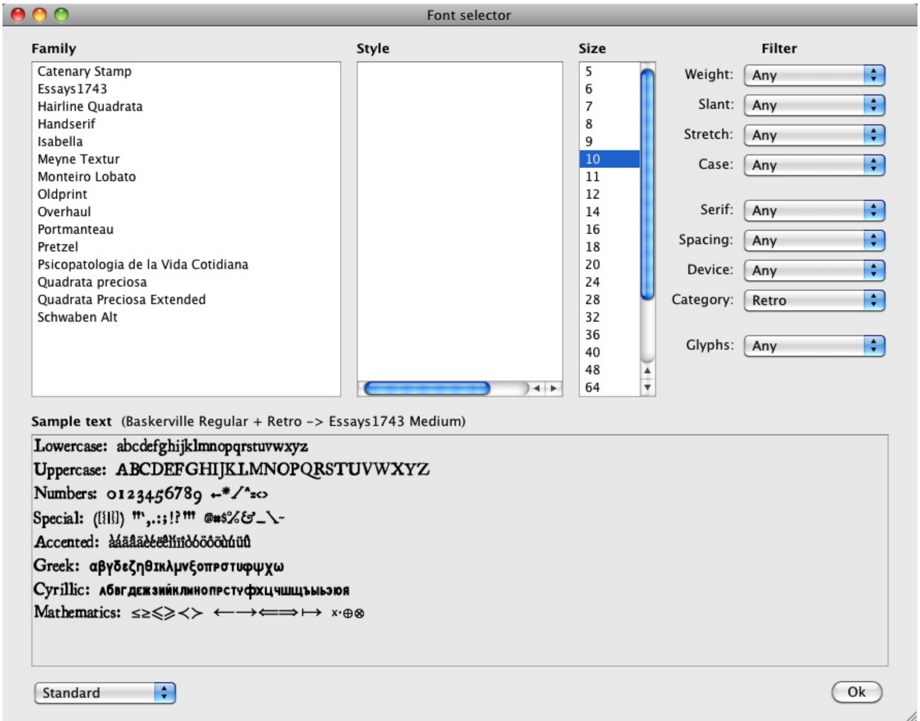


Figure 3.2. The TeXmacs font selector.

In the example, we first selected the "Baskerville Regular" font. Using the Category pop-up choice list on the right, we next searched for a font with a "Retro" look and feel that combines as well as possible with the Baskerville font. The header of the Sample text indicates that "Essays1743 Medium" is the best match that we could find on our system. The standard sample text shows the rendering of various characters in this font (other sample texts can be selected instead from the choice list that indicates Standard). The Family frame lists all "Retro" fonts that were found on our system. Since we did not select any of these font families yet, the Style frame does not contain any corresponding styles.

The easiest way to select a font is by clicking on a suitable family and then press Ok. If you rather select your font by picking a style or by using a filter (as in our example), then the semantics of font selection is slightly more subtle. For instance, when clicking on Ok in Figure 3.2, TeXmacs will insert a tag inside the document that asks for a "Retro" font. Since the surrounding text was typeset using the "Baskerville Regular" font, any text that you will enter next will be typeset using the "Essays1743 Medium" font, as indicated by the font selector; so far so good. However, whenever you copy and paste the newly entered text somewhere else, the appropriate "Retro" font for the surrounding text is not necessarily "Essays1743 Medium": TeXmacs might use another, more relevant font instead.

Although the font selector is easy to use, it is more limited when it comes to specifying a more precise fallback strategy in the case of missing fonts or characters. For example, it does not allow you to specify one particular font for text and another one for mathematics. TeXmacs provides a mechanism for building composite of this type, which is beyond the scope of this book, and currently only intended for expert users (if you are curious, then you may try Edit ▸ Preferences ▸ Other ▸ Advanced font customization before opening the font selector). Another way to force a particular font for, say, mathematics is to adapt the math macro so as to use this font.

That said, we notice that TeXmacs is good at emulating mathematical symbols for fonts that do not natively support them. For example, in case of the Essays1743 font, the arrow → is obtained by juxtaposing two minus signs - and the closing guillemet ›; we refer to [27] for more details.

3.8.3 Font characteristics

As explained above, fonts have three main characteristics:

Family. Fonts are grouped together into *families* with a similar design.

Shape. Inside the same font family, individual fonts have different *shapes*, such as bold, italic, small capitals, etc.

Size. The font *size* in points.

The actual font files on your hard disk usually correspond to the font family and the font shape; the font itself can automatically be scaled to any size.

The TeXmacs font database keeps track of several extra characteristics of fonts. The font selector allows you to filter on the following criteria:

Weight. The font *weight* corresponds to the "thickness" of the font:

Thin	Light	Medium	**Bold**	**Black**

Slant. The font *slant* determines the angle of the font:

Normal	*Italic*	*Oblique*

Stretch. This property determines the horizontal width for a fixed vertical height:

Condensed	Unextended	Wide

Case. This property determines how lowercase letters are capitalized:

Mixed	SMALL CAPITALS

Serif. This feature corresponds to decorations at the end of strokes that remind of letters carved into stone:

Serif	Sans Serif

Spacing. This feature corresponds to the horizontal spacing between characters:

Proportional	Monospaced

Device. This property can be used to imitate specific “writing devices”:

Print	Typewriter	Digital	Pen	Art pen	Chalk	Marker

Category. Various other font features:

Ancient	Attached	Calligraphic	Comic
Decorative	Distorted	Gothic	Handwritten
INITIALS	Medieval	Miscellaneous	Outline
Retro	SciFi	Title	

Each of the above properties really constitutes a *hint* on the kind of font that *should* be used. If no suitable font can be found on your particular system, then setting these properties may have no effect.

Let us finally recall that the left footer of the $\mathrm{T_EX_{MACS}}$ window indicates some of the most significant properties of the font being used at the current cursor position. For instance, for a generic newly created document, it displays “text roman 10”, which means that you are working in text mode using Knuth's 10 point Computer Modern “Roman” font [37].

3.9 Paragraph layout

Most of the time, the layout of paragraphs is determined automatically, as a function of the document style and structured markup inside your text. You may sometimes want to adjust this default layout. This can either be done globally, using Document ▸ Paragraph, or for one or more specific paragraphs, using Format ▸ Paragraph. We recall from section 3.5 that specific rendering effects can also be obtained using the block markup elements in Insert ▸ Prominent.

The alignment of paragraphs can be changed in the Format ▸ Paragraph ▸ Basic ▸ Alignment choice list. The four possible alignment styles are Left, Centered, Right and Justified. The default style is Justified, meaning that full lines are stretched as much as needed in order to align their right-hand sides.

There are two classical ways to make the beginnings of successive paragraphs easy to spot for the reader (see Figure 3.3): by indenting the first line of each paragraph or by inserting additional vertical whitespace between consecutive paragraphs. The extra indentation to be used for first lines of paragraphs can be specified in Format ▸ Paragraph ▸ Basic ▸ First indentation. The amount of whitespace between successive paragraphs can be changed in Format ▸ Paragraph ▸ Basic ▸ Interparagraph space.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

The quick brown fox again jumps over the lazy dog. The quick brown fox again jumps over the lazy dog.

The quick brown fox is getting tired, jumping over the lazy dog.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

The quick brown fox again jumps over the lazy dog. The quick brown fox again jumps over the lazy dog.

The quick brown fox is getting tired, jumping over the lazy dog.

Figure 3.3. Two common paragraph styles.

At this point it is useful to say a few words about *lengths* in TeXmacs. Most of the fields in the Format ▸ Paragraph dialogue window expect length values. For instance, in order to indent every first line of a paragraph by a quarter of an inch, you should enter `0.25in` as the value for Format ▸ Paragraph ▸ Basic ▸ First indentation. A TeXmacs length consists of a number (e.g. `0.25`) followed by a unit (e.g. `in`). Apart from the standard units `mm`, `cm`, and `in`, TeXmacs also provides several units of interest for typographers, such as `pt` (one point; approximately 1/72 inch), `fn` (the current font size; e.g. `10pt`), or `tab` (a special unit for indentations; by default, `1tab` equals `1.5fn` and the article style uses a first indentation of `1tab`); see Table 3.9 for an overview.

The dialogue window Format ▸ Paragraph allows you to specify a few other layout properties of paragraphs. The margins can be changed using Left margin and Right margin. For instance, inside quotations (see section 3.5.2), both the left and right margins are increased by `2tab`. The spacing between individual lines is controlled via Interline space. By default, TeXmacs uses a small interline space of `0.2fn`. A zero interline space makes ordinary text less readable, but can be acceptable for computer programs, while stuffing as much code

Unit	Purpose
`cm`	Centimeter
`mm`	Millimeter
`in`	Inch
`pt`	Typographic point
`dd`	Didot point
`pc`	Pica (`12pt`)
`cc`	Cicero (`12dd`)

Unit	Purpose
`fn`	Font size (quad space)
`tab`	Tab space
`spc`	Space in current font
`sep`	Separation space
`ex`	Height of x character
`ln`	Fraction bar width
`par`	Paragraph width
`pag`	Page height

Table 3.9. Available length units in TeXmacs.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

Figure 3.4. Illustration of three different interline spaces: `0fn`, `0.2fn`, and `1fn`.

as possible on the screen. An interline space of `1fn` can be useful for drafts in which readers are likely to make extensive annotations between the lines. These three different settings are illustrated in Figure 3.4.

TeXmacs finally allows you to typeset text in more than one column, while specifying the horizontal space between separate columns (using Number of columns and Column separation). The Format ▸ Paragraph ▸ Advanced tab contains yet more expert settings for the layout of paragraphs.

If your aim is not so much to force a precise layout, but rather to obtain certain graphical effects, then you may prefer to use the markup from Insert ▸ Prominent rather than the dialogue window Format ▸ Paragraph. Most of these tags were already discussed in section 3.5. You may use Insert ▸ Prominent ▸ Indent for adding a fixed amount of indentation to the left margin (this is often needed for computer programs). TeXmacs also provides the jump-in tag (Insert ▸ Prominent ▸ Indent) for typesetting text using a negative first indentation, with the effect of making paragraphs "jump in". A compact layout (Insert ▸ Prominent ▸ Compact) reduces the amount of interparagraph space; see Table 3.5.

The quick brown fox

Jumps over the lazy dog.

The quick fox

Jumps over dogs.

The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog.

The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog. The quick brown fox jumps over the lazy dog.

Figure 3.5. Effect of the markup elements indent, jump-in, and compact.

3.10 Page layout

The default page layout is determined by the document style. You can customize it using the dialogue window that is opened via Document ▸ Page. For a newly created document, you may also change some of the settings using the icon (or possibly , , , , or) on the focus toolbar (see section 2.7). In order to tweak the layout of a specific page, you should rather use Format ▸ Page.

The main characteristics of a page are its size (most of the standard page sizes—such as A4, A5, Letter, Legal, etc.—are supported by TeXmacs) and its orientation (portrait or landscape). These properties can be specified via Document ▸ Page ▸ Format ▸ Page type and Orientation.

Another major issue is how to display pages on your screen. It is crucial to understand that TeXmacs may render your document differently on screen and on paper[3.1]. TeXmacs supports four major rendering modes in Document ▸ Page ▸ Format ▸ Page rendering:

- The paper rendering reflects the actual way your document will be printed. Page breaks are made explicit by showing the pages separately, one below another.

- The papyrus rendering mode treats your document as if it were printed on a big roll of papyrus with the specified “page” width, but whose vertical height is as large as needed.

- The screen setting can be used if you do not intend to print your document. The page width and height are automatically adjusted to the size of your window. This is most useful when browsing web pages or using TeXmacs as an interface for some computer algebra system.

- The beamer mode is reserved for laptop presentations; see chapter 9.

The page breaking algorithm takes some time on large documents, which may hamper the reactivity of the editor while typing. For this reason, we recommend that you edit such documents in “papyrus” mode and only switch to “paper” mode when your document is almost finished (e.g. in order to inspect the page breaks).

3.1. For TeXmacs to be “truly wysiwyg”, you need to select the paper rendering mode and Document ▸ Page ▸ Margins ▸ Same screen margins as on paper. However, since screens and paper are different media, editing your document as if it were already printed is not always convenient. For instance, page breaks may heavily change while you are typing, and thereby cause unpleasant jumping around of the cursor. Sentences with page breaks may also become hard to read.

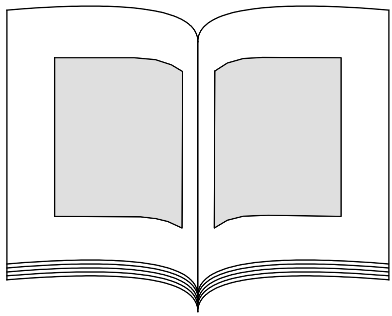


Figure 3.6. Schematic representation of the typical layout of a book: the left margins are larger than the right margins on the (even) left-hand pages, whereas they are smaller on the (odd) right-hand pages.

The second most important decision about page layout concerns the margins; these can be specified in the tab Document ▸ Page ▸ Margins. Remember that the lines of your text should not be made too long, thereby keeping them easy to read. On standard paper of A4 or Letter size, this can be achieved by using sufficiently large margins or a larger font. Another option is to use a two column layout. It is also interesting to know that printed books tend to use different margins for odd and even pages: see Figure 3.6. Since margins on actual paper can be quite large, you finally may wish to use different margins when editing documents on your screen. In Document ▸ Page ▸ Margins, you are therefore allowed to separately specify paper margins and screen margins (or force the same settings for both using Same screen margins as on paper).

The remaining global settings for page layout are intended for more expert users. In particular, it is recommended to *not* specify page headers and footers through Document ▸ Page ▸ Headers, except when you selected the `generic` document style. The reason is that title blocks and section titles often override these settings, since the main title, the authors and titles of sections are typically used inside page headers.

On the other hand, it is safe to customize the headers and footers of a particular page using Format ▸ Page ▸ This page header and This page footer. You may also change the number of a particular page (and all the subsequent ones) using Format ▸ Page ▸ This page number. You may finally change the way page numbers are rendered using Format ▸ Page ▸ Page number rendering.

Chapter 4
Mathematics

You enter *math mode* by starting a new formula or by placing the cursor inside an existing formula. In math mode, the menus, the icon toolbars, and the keyboard behavior are adapted so as to facilitate the insertion of mathematical symbols and markup. For instance, typing - > inserts the arrow →.

Recent versions of TeXmacs also include optional "semantic editing" facilities that will briefly be described at the end of this chapter (see [26, 29] for more details). When used appropriately, these allow you to write documents in which all formulas are at least correct from a syntactical point of view. A "syntax corrector" is included to assist you with this task. Syntactic correctness is for instance important when using formulas as inputs for a computer algebra system. Syntactically correct documents are also less likely to contain "typos". Further functionalities, such as semantic search and replace, should be developed in the future.

4.1 Incorporating mathematical formulas

TeXmacs provides three main ways to insert mathematical formulas into the main text:

Inline formulas. Short formulas—such as $a^2+b^2=c^2$—are usually embedded directly into the main text flow. Such *inline formulas* can be inserted using $ or Insert ▸ Mathematics ▸ Inline formula.

The typesetter attempts to squeeze inline formulas as much as possible, so that they do not disrupt the general presentation. For example, the presentation $\lim_{n\to\infty} \log n / n = 0$ is preferred over $\displaystyle\lim_{n\to\infty} \log n / n = 0$, and $x = \frac{1}{2}$ over $x = \dfrac{1}{2}$. Nevertheless, you can force the more voluminous renderings using Format ▸ Display style ▸ on.

Displayed formulas. Big or important formulas are usually displayed on separate lines:

$$x^n + y^n = z^n.$$

Such *displayed formulas* can be inserted using ⌥$ or Insert ▸ Mathematics ▸ Displayed formula and use a less compressed layout than inline formulas. You may turn an inline formula into a displayed one and *vice versa* using ^⇥. Displayed formulas can also be numbered using the toggle ^# or Focus ▸ Numbered.

Equation arrays. For the presentation of multiple equations, it is best to align them as in the following example:

$$x+0 \;=\; x$$

$$\begin{aligned} x+(-x) &= 0 \\ x+y &= y+x \\ (x+y)+z &= x+(y+z). \end{aligned}$$

A similar layout is often required for step by step computations

$$\begin{aligned} (\mathrm{e}^{\sin x}+\sin \mathrm{e}^x)' &= (\mathrm{e}^{\sin x})'+(\sin \mathrm{e}^x)' \\ &= (\sin x)'\,\mathrm{e}^{\sin x}+(\mathrm{e}^x)'\sin \mathrm{e}^x \\ &= \mathrm{e}^{\sin x}\cos x+\mathrm{e}^x\sin \mathrm{e}^x, \end{aligned}$$

in which case several left-hand members are simply left empty.

In order to create so-called *equation arrays* of this type, you may use ⌥& or Insert ▸ Mathematics ▸ Several equations. The typesetter uses a table with three columns for the rendering, where the first column is aligned to the right, the third one to the left, and the middle column is centered. New rows are created by pressing ↩. In fact, since equation arrays are special forms of tables, all table editing rules apply (see chapter 5). In a sense, they are also somewhat superfluous: you may obtain the same layout by creating an appropriate table inside a displayed formula. Nevertheless, equation arrays are so common in mathematical documents that it is nice to have special markup for them.

4.2 Mathematical symbols

Following the basic design principles of the user interface (see section 2.1), TeXmacs allows you to enter most mathematical symbols in at least four ways: using the menus, the icon toolbars, appropriate keyboard shortcuts, or LaTeX commands.

For example, the binary relation "$\leqslant$" can both be found in the Insert ▸ Symbol ▸ Binary relation menu and in the toolbar menu under the icon $\prec$. When hovering the mouse pointer for a while over the $\leqslant$ in either of these menus, the corresponding keyboard shortcut < = will appear in a help balloon. Users who are familiar with LaTeX, may also enter the symbol using \ l e q s l a n t ↩.

The keyboard shortcuts for mathematical symbols were designed according to a small number of simple rules, which make them easy to remember. Before we go into more details, it should also be noticed that symbols carry a precise syntactical semantics in TeXmacs. In particular, TeXmacs carefully distinguishes between so-called *homoglyphs*, which are distinct symbols that look the same. For more information, see sections 4.9 and 4.11 below.

For example, the vertical bar $|$ can be used as a separator for defining sets $R^> = \{x \in R \mid x > 0\}$, but also as the binary relation "divides" (e.g. $11 \mid 1001$), or for restricting the domain of a function: $f|_E$. Such homoglyphs have different binding forces and often come with a different spacing. The most annoying ambiguity is between invisible multiplication $x y$ and function application $\sin x$, which are respectively entered using the shortcuts * and ␣.

As a general rule, we also note that TeXmacs takes care of the spacing inside mathematical formulas. For instance, you enter $a+b$ by typing a + b, not

α	a ⇥	β	b ⇥	γ	g ⇥	δ	d ⇥	ε	e ⇥	ϵ	e ⇥ ⇥ ⇥
ζ	z ⇥	η	h ⇥	θ	j ⇥	ϑ	j ⇥ ⇥ ⇥	ι	i ⇥	κ	k ⇥
$\varkappa$	k ⇥ ⇥	λ	l ⇥	μ	m ⇥	ν	n ⇥	ξ	x ⇥	o	o ⇥
π	p ⇥	ϖ	p ⇥ ⇥ ⇥	ρ	r ⇥	ϱ	r ⇥ ⇥	σ	s ⇥	ς	s ⇥ ⇥
τ	t ⇥	υ	u ⇥	φ	f ⇥	ϕ	f ⇥ ⇥	ψ	y ⇥	χ	q ⇥
ω	w ⇥										
A	A ⇥	B	B ⇥	Γ	G ⇥	Δ	D ⇥	E	E ⇥	Z	Z ⇥
H	H ⇥	Θ	J ⇥	I	I ⇥	K	K ⇥	Λ	L ⇥	M	M ⇥
N	N ⇥	Ξ	X ⇥	O	O ⇥	Π	P ⇥	P	R ⇥	Σ	S ⇥
T	T ⇥	Υ	U ⇥	Φ	F ⇥	Ψ	Y ⇥	X	Q ⇥	Ω	W ⇥

Table 4.1. Keyboard shortcuts for Greek characters.

a ␣ + ␣ b. The space bar is reserved for function application: typing s i n ␣ x produces $\sin x$.

4.2.1 Letter-like symbols

Mathematicians like to use Greek characters as well as letters in several special fonts for particular purposes. For instance, the set of natural numbers is typically denoted by $\mathbb{N}$ and a maximal ideal by $\mathfrak{m}$.

Greek characters can be obtained using the Tab key ⇥, as variants of the usual Roman letters. For instance, you may enter β and Λ using b ⇥ and L ⇥. Table 4.1 shows keyboard shortcuts for the complete Greek alphabet. Notice that the Greek letters ε, θ, κ, π, ρ, σ, and φ admit alternative renderings ϵ, ϑ, $\varkappa$, ϖ, ϱ, ς, and ϕ.

TeXmacs reserves the function keys F5, F6, F7, and F8 for entering characters in special mathematical fonts:

F5. This is the keyboard prefix for upright mathematical symbols. For example, F5 S produces S instead of S, whereas F5 l ⇥ produces $\mathrm{\lambda}$ instead of λ.

F6. This keyboard prefix is used for producing bold letters such as $\boldsymbol{v}$ (F6 v) or $\boldsymbol{S}$ (F6 S). You may combine it with the other prefixes F5, F7, and F8. For example, F6 F5 v yields $\mathbf{v}$ and F6 F7 X yields $\boldsymbol{\mathcal{X}}$.

⇧F6. You may use this prefix for "blackboard bold" fonts. The classical sets $\mathbb{C}$, $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{R}$, and $\mathbb{Z}$ can for example be obtained using the shortcuts ⇧F6 C, ⇧F6 N, ⇧F6 Q, ⇧F6 R, and ⇧F6 Z. An even easier way to obtain these symbols is using C C, N N, Q Q, R R, and Z Z, i.e. by typing twice the same uppercase letter.

F7. This is the keyboard prefix for calligraphic symbols such as $\mathcal{A}$ (F7 A) and $\mathcal{P}$ (F7 P). Notice that not all fonts provide calligraphic variants $\mathcal{a}$, $\mathcal{b}$, $\mathcal{c}$, ... for lowercase letters. Whenever such variants are missing for a given font, then TeXmacs will use a system-dependent substitute font instead (see section 3.8.1).

d	`d ⇥ ⇥`	∂	`d ⇥ ⇥ ⇥`	e	`e ⇥ ⇥`	i	`i ⇥ ⇥`
γ	`g ⇥ ⇥`	λ	`l ⇥ ⇥`	π	`p ⇥ ⇥`	Γ	`G ⇥ ⇥ ⇥`
$\hbar$	`h ⇥ ⇥`	$\imath$	`i ⇥ ⇥ ⇥`	$\jmath$	`j ⇥ ⇥`	ℓ	`l ⇥ ⇥ ⇥`
D	`D ⇥ ⇥`	∇	`V ⇥ ⇥`	Δ	`D ⇥ ⇥ ⇥ ⇥`	∞	`@ @`
$\Re$	`R E`	$\Im$	`I M`	∁	`C ⇥ ⇥ ⇥`	⅁	`G ⇥ ⇥ ⇥ ⇥`
∀	`A ⇥ ⇥`	∃	`E ⇥ ⇥`	∃⃒	`E ⇥ ⇥ ⇥`	℘	`P ⇥ ⇥`
ℵ	`A ⇥ ⇥ ⇥`	ℶ	`B ⇥ ⇥ ⇥`	ℷ	`G ⇥ ⇥`	ℸ	`D ⇥ ⇥ ⇥`
϶	`e ⇥ ⇥ ⇥ ⇥`	ϖ	`w ⇥ ⇥`	Ǝ	`E ⇥ ⇥ ⇥ ⇥`	℧	`W ⇥ ⇥`

Table 4.2. Keyboard shortcuts for several letter-like symbols.

- `F8`. This keyboard prefix corresponds to the "Fraktur" font. For example, the shortcuts `F8 m` and `F8 J` produce $\mathfrak{m}$ and $\mathfrak{J}$.

In fact, we note that TEXMACS does not consider mathematical letter-like symbols like $\boldsymbol{v}$, ℕ, $\mathcal{A}$, and $\mathfrak{m}$ as being typeset in a separate font. Instead, such symbols are interpreted as special characters in an extended (UNICODE) alphabet. In particular, the font selector (see section 3.8.2) does not contain any entry for—say—the "blackboard bold" font.

There exist a few more letter-like symbols that cannot be obtained through the systematic mechanisms from above. First of all, the important mathematical constants i, e, and π are entered using `i ⇥ ⇥`, `e ⇥ ⇥`, and `p ⇥ ⇥`. Notice that TEXMACS uses an upright rendering for these constants, which allows you to distinguish them from the letters i, e, and π. The d and λ from differential- and lambda-calculus are entered similarly, using `d ⇥ ⇥` and `l ⇥ ⇥`. Table 4.2 shows the list of keyboard shortcuts for these and other letter-like mathematical symbols.

4.2.2 Other symbols

There are a few simple rules that allow you to enter most non-letter-like mathematical symbols using "natural" key-combinations. The most important rule is juxtaposition: `- >` yields $\rightarrow$, `- - >` yields $\longrightarrow$, and `> =` yields $\geqslant$. Similarly, `| ⇥ -` yields $\vdash$, `| - >` yields $\mapsto$, and `- > < -` yields $\rightleftarrows$. The other rules are all based on the use of a special key:

- `⇥`. This is the main key for obtaining *variants* (see section 2.3.2). Some symbols have many variants. For example, `<` yields $<$, `< ⇥` yields $\in$, `< ⇥ ⇥` yields $\subset$, `< ⇥ ⇥ ⇥` yields $\prec$, `< ⇥ ⇥ ⇥ ⇥` yields $\sqsubset$, `< ⇥ ⇥ ⇥ ⇥ ⇥` yields $\langle\rangle$, and `< ⇥ ⇥ ⇥ ⇥ ⇥ ⇥` returns to $<$. You may "cycle back" among the variants using `⇧⇥`, so that `< ⇥ ⇥ ⇧⇥` is equivalent to `< ⇥`. For symbols with many variants, the last variants are obtained most efficiently by cycling back. For example, quick shortcuts for $\langle\rangle$ and $\sqsubset$ are `< ⇧⇥` and `< ⇧⇥ ⇧⇥`.

  The variant mechanism is particularly powerful when used in conjunction with juxtaposition. For example, `< =`, `< = ⇥`, `< = ⇥ ⇥`, and `< = ⇥ ⇥ ⇥` respectively yield $\leqslant$, $\leq$, $\leqq$, and $\Leftarrow$. Notice that the juxtapositions are horizontal for the first three shortcuts, but vertical for the last one. Similarly, the shortcuts `< = =`, `< ⇥ ⇥ =`, and `< < < =` produce $\Longleftarrow$, $\subseteq$, and $\lll$.

- @. This key is used for putting symbols into circles or boxes. For example, @ + yields $\oplus$ and @ x yields $\otimes$. Similarly, @ ⇥ + yields $\boxplus$.

- /. Slashes are used for negations. For example, = / yields $\neq$ and < = / yields $\nleqslant$. Notice that < = ⇥ ⇥ / yields $\nleqq$, whereas < = ⇥ ⇥ / ⇥ yields $\lneqq$.

- !. After entering an arrow, this key forces scripts to be placed above or below the arrow. For example, - - > ^ x yields $\longrightarrow^x$, but - - > ! ^ x yields $\xrightarrow{x}$.

- s i n. Special operators such as "sin" are simply obtained by typing s i n. More precisely, any sequence of two or more unaccented Latin letters is regarded as an operator by TeXmacs. In order to enter the formula $x\,y$ (x times y), you have to explicitly insert a multiplication using *. Indeed, typing x y yields xy, whereas x * y produces the desired result. We also recall that TeXmacs makes an explicit distinction between multiplication and function application. This means that you should type L ␣ x in order to produce $L\,x$ (L applied to x), and s i n ␣ s i n ␣ x for getting $\sin \sin x$.

Apart from these general rules, there are a few other shortcuts and facts that are worth knowing:

- *. By default, this key is used for "invisible" multiplications, as in $x y = x \cdot y$. The "visible" multiplication symbols $\times$, $\ast$, $\cdot$, and $\wedge$ (wedge product) are obtained as variants, using * ⇥, * ⇥ ⇥, etc.

  Notice that the presence of an invisible multiplication is indicated by a small space. TeXmacs provides a few other invisible operators, such as invisible function application ($\sin x$), invisible addition ($5\,{}^3/_4$), etc.; see section 4.11. You may use Document ▸ Informative flags ▸ Detailed in order to display additional visual hints that will not appear in print, but that allow you to distinguish between different types of invisible operators while editing.

- ␣. The space symbol is reserved for function application, as in $\exp x$. By default, TeXmacs tries to avoid the insertion of any spurious space. Typing a ␣ + ␣ b is therefore equivalent to typing a + b. If you want to force TeXmacs to insert all spaces you type, then you should select Edit ▸ Preferences ▸ Space bar in math mode ▸ Allow spurious spaces.

  Inserting a space of a specified length is also possible using Format ▸ Whitespace ▸ Horizontal space. From a semantic point of view, TeXmacs ignores all whitespace of this kind. Large spaces that should be considered as function applications or commas can be entered as variants of spaces and commas. For instance, the space in the function application $\varphi\ x$ can be entered using ␣ ⇥. Notice that this space is larger than the standard function application space in φx.

$\oplus$	`@` `+`	$\ominus$	`@` `-`	$\otimes$	`@` `*`	$\oslash$	`@` `/` `⇥`
$\odot$	`@` `.`	$\circledcirc$	`@` `@` `⇥`	$\circledast$	`@` `*` `⇥`	⦶	`@` `|`
$\boxplus$	`@` `⇥` `+`	$\boxminus$	`@` `⇥` `-`	$\boxtimes$	`@` `⇥` `*`	⧄	`@` `⇥` `/`
$\boxdot$	`@` `⇥` `.`	⧈	`@` `⇥` `@`	⧆	`@` `⇥` `*` `⇥`	◫	`@` `⇥` `|`
$\pm$	`+` `-`	$\mp$	`-` `+`	$\times$	`*` `⇥`	$\div$	`/` `⇥` `⇥`
$\ast$	`*` `⇥` `⇥`	$\star$	`*` `⇥` `⇥` `⇥` `⇥` `⇥`	$\circ$	`@`	$\bullet$	`@` `⇥` `⇥`
$\cap$	`&` `⇥`	$\cup$	`%` `⇥`	$\uplus$	`%` `⇥` `+`	$\cdot$	`*` `⇥` `⇥` `⇥`
$\sqcap$	`&` `⇥` `⇥`	$\sqcup$	`%` `⇥` `⇥`	$\vee$	`%`	$\wedge$	`&`
$\curlyvee$	`%` `⇥` `⇥` `⇥` `⇥`	$\curlywedge$	`&` `⇥` `⇥` `⇥` `⇥`	$\veebar$	`%` `-`	$\barwedge$	`-` `&`
$\ltimes$	`|` `⇥` `*`	$\rtimes$	`*` `⇥` `⇥` `|`	$\leftthreetimes$	`T` `⇥` `⇥` `⇥` `⇥`	$\rightthreetimes$	`T` `⇥` `⇥` `⇥` `⇥` `⇥`

Table 4.3. Binary operators (see also Insert ▸ Symbol ▸ Binary operator).

- `.`. The default dot stands for the decimal dot, as in 3.14159. The dot symbol has several variants: the "point" operator from λ-calculus (`.` `⇥`) as in $\lambda x.x^2$, the "dummy" dot $\cdot$ (`.` `⇥` `⇥`) as in $g = f(x, \cdot)$, and the invisible dummy dot (`.` `⇥` `⇥` `⇥`).

  Various kinds of ellipses $\ldots$, $\cdots$, $^{\cdots}$, $\vdots$, $\ddots$, and $\iddots$ can be obtained using the shortcuts `.` `.`, `.` `.` `⇥`, etc. The "lower" ellipses should be used in expressions such as $x_1, \ldots, x_n$ and $K[[x_1; \ldots; x_n]]$, whereas the centered ellipses should be preferred in $x_1 + \cdots + x_n$ and $x_1 \cdots x_n$. Notice that TeXmacs provides the additional shortcuts `,` `,`, `+` `+`, etc. for entering "$, \ldots,$", "$+ \cdots +$", etc.

- `,`. The comma most commonly serves as a separator, as in $f(x, y, z)$ and $x_1,\ldots,x_n$. Variants are the "decimal comma" (like in 3,14159), the invisible comma (as in the matrix (M_{ij})), and various "wide invisible commas", as in the list of formulas

$$a_1 = b_1 \qquad a_2 = b_1 + b_2 \qquad a_3 = b_1 + b_2 + b_3.$$

- `&`, `%`. The logical connectors $\wedge$ and $\vee$ are obtained using `&` and `%`. The conjunction $\wedge$ has many variants (namely $\cap$, $\sqcap$, $\prod$, $\curlywedge$, $\bigcurlywedge$, $\bigcurlywedge$, and &), and so does the conjunction $\vee$ (namely $\cup$, $\sqcup$, $\coprod$, $\curlyvee$, $\bigcurlyvee$, $\bigcurlyvee$, ‰, and %). The literal symbols & and % are obtained most efficiently through `F5` `&` and `F5` `%`. Alternative shortcuts for $\wedge$ and $\vee$ are `/` `\` and `\` `/`, with approximately the same variants as above.

- `#`. This key is used for the "number" prefix #, with variants $\sharp$ and $\natural$. The "flat" symbol $\flat$ can be obtained using `b` `⇥` `⇥`.

4.2.3 Keyboard shortcuts for common symbols

For the reader's convenience, we have included Tables 4.3–4.9 with keyboard shortcuts for common symbols. The shortcuts are easily remembered when applying the rules from the previous section. We also note that some of the keyboard shortcuts may change from one version of TeXmacs to another, e.g. due to the appearance of a new symbol.

∼	~	≃	~ -	≈	~ ~	≅	~ =
≁	~ /	≄	~ - /	≉	~ ~ /	≇	~ = /
≍	= ⇥	≡	= ⇥ ⇥	≅	~ =	≋	= ⇥ ⇥ ⇥ ⇥
≭	= ⇥ /	≢	= ⇥ ⇥ /	≇	~ = /	≋̸	= ⇥ ⇥ ⇥ ⇥ /
≏	~ - ⇥ ⇥	≎	~ ~ ⇥	≗	@ =	≖	= @
∽	~ ⇥	⋍	~ - ⇥	**∼**	F6 ~	**≈**	F6 ~ ~
≠	= /	∝	@ @ ⇥ ⇥	≊	~ ~ =	≜	@ = ⇥

Table 4.4. Equalities, similarities, and their negations (see also Insert ▸ Symbol ▸ Binary relation).

<	<	⩽	< =	≤	< = ⇥	≦	< = ⇥ ⇥
>	>	⩾	> =	≥	> = ⇥	≧	> = ⇥ ⇥
≪	< <	≪̲	< < =	⋘	< < <	⋘̲	< < < =
≫	> >	≫̲	> > =	⋙	> > >	⋙̲	> > > =
≺	< ⇥ ⇥ ⇥	≼	< ⇥ ⇥ ⇥ =	⪯	< ⇥ ⇥ ⇥ = ⇥	≾	< ⇥ ⇥ ⇥ ~
≻	> ⇥ ⇥ ⇥	≽	> ⇥ ⇥ ⇥ =	⪰	> ⇥ ⇥ ⇥ = ⇥	≿	> ⇥ ⇥ ⇥ ~
⪻	< < ⇥	⪻̲	< < ⇥ =	⋘	< < < ⇥	⋘̲	< < < ⇥ =
⪼	> > ⇥	⪼̲	> > ⇥ =	⋙	> > > ⇥	⋙̲	> > > ⇥ =
⊂	< ⇥ ⇥	⊆	< ⇥ ⇥ =	⫅	< ⇥ ⇥ = ⇥	∈	< ⇥
⊃	> ⇥ ⇥	⊇	> ⇥ ⇥ =	⫆	> ⇥ ⇥ = ⇥	∋	> ⇥
⊏	< ⇥ ⇥ ⇥ ⇥	⊑	< ⇥ ⇥ ⇥ ⇥ =	⋐	< < ⇥ ⇥	⋹	< ⇥ ⇥ +
⊐	> ⇥ ⇥ ⇥ ⇥	⊒	> ⇥ ⇥ ⇥ ⇥ =	⋑	> > ⇥ ⇥	⋺	> ⇥ ⇥ +
⊲	< | ⇥	⊴	< | ⇥ =	⊴	< | ⇥ = ⇥	◀	< | ⇥ ⇥
⊳	| ⇥ >	⊵	| ⇥ > =	⊵	| ⇥ > = ⇥	▶	| ⇥ > ⇥
≲	< ~	⪅	< ~ ~	≾	< ⇥ ⇥ ⇥ ~	⪷	< ⇥ ⇥ ⇥ ~ ~
≳	> ⇥ ~	⪆	> ⇥ ~ ~	≳	> ⇥ ~	⪆	> ⇥ ~ ~

Table 4.5. Comparison relations (see also Insert ▸ Symbol ▸ Binary relation).

∉	< ⇥ /	∌	> ⇥ /				
≮	< /	≰	< = /	≰	< = ⇥ /	≨	< = ⇥ / ⇥
≯	> /	≱	> = /	≱	> = ⇥ /	≩	> = ⇥ / ⇥
≨	< = ⇥ ⇥ / ⇥	⪇	< = ⇥ ⇥ / ⇥ ⇥	⋦	< ~ / ⇥	⪉	< ~ ~ / ⇥
≩	> = ⇥ ⇥ / ⇥	⪈	> = ⇥ ⇥ / ⇥ ⇥	⋧	> ~ / ⇥	⪊	> ~ ~ / ⇥
⊊	< ⇥ ⇥ = / ⇥	⊈	< ⇥ ⇥ = ⇥ / ⇥	⊊	< ⇥ ⇥ = / ⇥ ⇥	⫋	< ⇥ ⇥ = ⇥ / ⇥ ⇥
⊋	> ⇥ ⇥ = / ⇥	⊉	> ⇥ ⇥ = ⇥ / ⇥	⊋	> ⇥ ⇥ = / ⇥ ⇥	⫌	> ⇥ ⇥ = ⇥ / ⇥ ⇥
⊀	< ⇥ ⇥ ⇥ /	⋠	< ⇥ ⇥ ⇥ = /	⪱	< ⇥ ⇥ ⇥ = ⇥ /	⪵	< ⇥ ⇥ ⇥ = ⇥ ⇥ / ⇥
⊁	> ⇥ ⇥ ⇥ /	⋡	> ⇥ ⇥ ⇥ = /	⪲	> ⇥ ⇥ ⇥ = ⇥ /	⪶	> ⇥ ⇥ ⇥ = ⇥ ⇥ / ⇥
⋨	< ⇥ ⇥ ⇥ ~ / ⇥	⪹	< ⇥ ⇥ ⇥ ~ ~ / ⇥	⋢	< ⇥ ⇥ ⇥ ⇥ /	⋤	< ⇥ ⇥ ⇥ ⇥ = /
⋩	> ⇥ ⇥ ⇥ ~ / ⇥	⪺	> ⇥ ⇥ ⇥ ~ ~ / ⇥	⋣	> ⇥ ⇥ ⇥ ⇥ /	⋥	> ⇥ ⇥ ⇥ ⇥ = /

Table 4.6. Negations of comparison relations (see also Insert ▸ Symbol ▸ Negation).

←	`<` `-`	⇐	`<` `=` `⇥` `⇥` `⇥`	↼	`/` `-` `⇥`	↽	`/` `-` `⇥` `⇥`
→	`-` `>`	⇒	`=` `>`	⇀	`-` `/` `⇥`	⇁	`-` `/` `⇥` `⇥`
↔	`<` `-` `>`	⇔	`<` `=` `>`	⇋	`<` `-` `-` `>` `⇥` `⇥` `⇥`	⇌	`-` `>` `<` `-` `⇥`
⇇	`<` `-` `<` `-`	↢	`<` `-` `<`	↩	`<` `-` `|` `⇥`	↞	`<` `<` `-`
⇉	`-` `>` `-` `>`	↣	`>` `-` `>`	↪	`|` `-` `>` `⇥`	↠	`-` `>` `>`
⇆	`<` `-` `-` `>` `⇥` `⇥`	⇄	`-` `>` `<` `-`	↦	`|` `-` `>`		
↫	`<` `-` `@`	↬	`@` `-` `>`	⇝	`~` `>`	↭	`<` `~` `>`
↑	`<` `-` `⇥`	↕	`<` `-` `>` `⇥`	⇑	`=` `>` `⇥` `⇥`	⇈	`<` `-` `<` `-` `⇥`
↓	`<` `-` `⇥` `⇥`	⇕	`<` `=` `>` `⇥`	⇓	`=` `>` `⇥`	⇊	`<` `-` `<` `-` `⇥` `⇥`
↿	`/` `-` `⇥` `⇥` `⇥` `⇥`	↾	`/` `-` `⇥` `⇥` `⇥`	↖	`<` `-` `⇥` `⇥` `⇥`	↗	`-` `>` `⇥` `⇥` `⇥` `⇥`
⇃	`/` `-` `⇥` `⇥` `⇥` `⇥` `⇥`	⇂	`/` `-` `⇥` `⇥` `⇥` `⇥` `⇥` `⇥`	↙	`<` `-` `⇥` `⇥` `⇥` `⇥`	↘	`-` `>` `⇥` `⇥` `⇥`

Table 4.7. Arrows (see also Insert ▸ Symbol ▸ Arrow).

⟵	`<` `-` `-`	⟸	`<` `=` `=`	⟻	`<` `-` `-` `|`	↩ (long)	`<` `-` `-` `|` `⇥`
⟶	`-` `-` `>`	⟹	`=` `=` `>`	⟼	`|` `-` `-` `>`	↪ (long)	`|` `-` `-` `>` `⇥`
⟷	`<` `-` `-` `>`	⟺	`<` `=` `=` `>`	—	`-` `-`	═	`=` `=`
↑ (long)	`<` `-` `-` `⇥`	↓ (long)	`<` `-` `-` `⇥` `⇥`	⇑ (long)	`<` `=` `=` `⇥`	⇓ (long)	`<` `=` `=` `⇥` `⇥`
↥ (long)	`<` `-` `-` `|` `⇥` `⇥`	↧ (long)	`<` `-` `-` `|` `⇥` `⇥` `⇥` `⇥`	↑ hook (long)	`<` `-` `-` `|` `⇥` `⇥` `⇥`	↓ hook (long)	`<` `-` `-` `|` `⇥` `⇥` `⇥` `⇥` `⇥`

Table 4.8. Long arrows (see also Insert ▸ Symbol ▸ Arrow).

¬	`!` `⇥`	∅	`@` `/`	△	`@` `⇥` `⇥` `⇥` `⇥`	▲	`@` `⇥` `⇥` `⇥` `⇥` `⇥`
∘	`@`	□	`@` `⇥`	●	`@` `⇥` `⇥`	■	`@` `⇥` `⇥` `⇥`
⊤	`T` `⇥` `⇥`	⊥	`T` `⇥` `⇥` `⇥`	⊢	`|` `⇥` `-`	⊣	`-` `|` `⇥`
⊩	`|` `|` `⇥` `-`	⊪	`|` `|` `|` `⇥` `-`	⊨	`|` `⇥` `=`		
♭	`b` `⇥` `⇥`	♯	`#` `⇥`	♮	`#` `⇥` `⇥`		
♦	`<` `>` `⇥`	♥	`<` `>` `⇥` `⇥`	♠	`<` `>` `⇥` `⇥` `⇥`	♣	`<` `>` `⇥` `⇥` `⇥` `⇥`
…	`.` `.`	⋯	`.` `.` `⇥`	˙˙˙	`.` `.` `⇥` `⇥`		
⋮	`.` `.` `⇥` `⇥` `⇥`	⋱	`.` `.` `⇥` `⇥` `⇥` `⇥`	⋰	`.` `.` `⇥` `⇥` `⇥` `⇥` `⇥`		

Table 4.9. Miscellaneous symbols (see also Insert ▸ Symbol ▸ Miscellaneous).

4.2.4 When a symbol is hard to find

For certain symbols like ✠, it may not be clear how to obtain it using the systematic mechanisms described in this section. Since the symbol vaguely reminds one of addition, you may take your chances and try all variants of $+$. If you are unlucky, or if the symbol is so exotic (e.g. ♫) that it reminds you of nothing at all, then you have several fall-back options:

TeXmacs menus. You may try to look it up in the submenus of Insert ▸ Symbol and especially in Insert ▸ Symbol ▸ Miscellaneous.

System tools. Most operating systems integrate tools for visualizing UNICODE characters and copy and paste them to other applications. For example, MACOS integrates "Character Viewer" for this purpose.

LATEX names. If you know the LATEX name of the symbol, then you may enter it using that name. For example, ♫ corresponds to the LATEX command `\twonotes`, so you may obtain this symbol via `\` `t` `w` `o` `n` `o` `t` `e` `s` `↩`.

Internal TEXMACS names. Most mathematical symbols are internally known under the same name as under LATEX. Symbols can directly be entered using their TEXMACS name via the keyboard shortcut `^q`. Hence `^q` `t` `w` `o` `n` `o` `t` `e` `s` `↩` also produces ♫.

UNICODE numbers. You may also look up the UNICODE number of your symbol inside UNICODE tables that you can find on the web. The hexadecimal UNICODE number prefixed by # can also be used as an internal TEXMACS name for the symbol. For example, the symbol ♫ corresponds to the UNICODE character 266B, so you may enter it by typing `^q` `#` `2` `6` `6` `B` `↩`.

4.3 Two-dimensional mathematical markup

Subscripts and superscripts. Subscripts and superscripts are created using the special keys `_` and `^`. For example, you can produce a_n by typing `a` `_` `n` and a^{a^x} via `a` `^` `a` `^` `x`. Notice that you have to explicitly move the cursor out of the script whenever needed, usually by pushing `→` once. In other words, $a_n + b_n$ is obtained through `a` `_` `n` `→` `+` `b` `_` `n`, whereas `a` `_` `n` `+` `p` yields a_{n+p}.

Subformulas may simultaneously carry a subscript and a superscript. For instance, a_n^2 is obtained using `a` `_` `n` `→` `^` `2`. The order in which a subscript and a superscript are entered does not affect the rendering, but is important for the semantics: if we entered the above formula a_n^2 using `a` `^` `2` `→` `_` `n`, then it would mean $(a^2)_n$ instead of $(a_n)^2$.

TEXMACS also allows you to put subscripts and superscripts at the left-hand side of an expression, using the shortcuts `⌥l` `_` and `⌥l` `^`. For example, you may enter the formula ${}^{\pi}x$ by typing `⌥l` `^` `p` `⇥` `→` `x`. A formula may be surrounded by as many as four subscripts and superscripts, as in ${}^{*}_{*}\mathrm{He}^{*}_{*}$.

In some cases, subscripts and superscripts are put below or above a given expression, rather than to its right. This is typically so for big and limit-like operators in displayed equations:

$$\lim_{n\to\infty} \sum_{k=0}^{n} \frac{1}{k^2} = \frac{\pi^2}{6}.$$

It is also possible to explicitly typeset a script above or below another expression using the shortcuts `⌥a` and `⌥b`:

$$f^{\circ n} = f \circ \overset{n\times}{\cdots} \circ f \qquad\qquad x = \underset{n\to\infty}{\mathrm{statlim}}\, x_n.$$

Shortcut	Purpose	Inline	Displayed
⌥f	Ordinary fractions	$\frac{a}{b+c}$	$\dfrac{a}{b+c}$
⌥f ⇥	Inline fractions	$\frac{a}{b+c}$	$\frac{a}{b+c}$
⌥f ⇥ ⇥	Displayed fractions	$\dfrac{a}{b+c}$	$\dfrac{a}{b+c}$
⌥f ⇥ ⇥ ⇥	Slashed fractions	$^1/_2$	$^1/_2$
⌥f ⇥ ⇥ ⇥ ⇥	Continued fractions	$\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{1+\ddots}}}$	$\cfrac{1}{1+\cfrac{1}{1+\cfrac{1}{1+\ddots}}}$

Table 4.10. Various blends of fractions.

Primes. Single and double primes can be entered using the keys ' and ". Primes behave much in the same way as superscripts, but it is slightly more efficient to insert or remove them (compare entering $a'+b$ and a^2+b using a ' + b *versus* a ^ 2 → + b). Using `, you may also insert a "left backprime", as in $`a$.

There are various alternative types of primes, which can be obtained as variants: the primes in $a^{`}$, a^*, $a^\dagger$, $a^\ddagger$, and $a^{✱}$ are entered using the shortcuts ' ⇥, ' ⇥ ⇥, etc. You may for example write V^* for the dual of V and $f^\dagger$ for the logarithmic derivative f'/f of a function f. Backprimes and double primes also have several variants. For instance, you may enter V^{**} and ${}^{✱}a$ using V " ⇥ ⇥ and ` ⇧⇥ a.

Fractions. TeXmacs uses ⌥ as the principal modifier key for the insertion of mathematical markup. You may for example insert a fraction using ⌥f (or Insert ▸ Fraction). There are several kinds of fractions, which are listed in Table 4.10. Alternatively, fractions may use a one-dimensional layout as in a/b, $15:5$, and $15\div 3$. The symbols /, :, and ÷ are obtained through /, / ⇥, and / ⇥ ⇥.

Square roots and n-th roots. You may enter a square root using ⌥s (or Insert square root) and an n-th root using ⌥s ⇥ (or Insert ▸ N-th root). Hence, ⌥s x yields $\sqrt{x}$ and ⌥s ⇥ ← ← 3 yields $\sqrt[3]{x}$.

The degree n of an n-th root can also be added *a posteriori* using ⌥→. For example, assume that your cursor is at the indicated position in $\sqrt{x|}$. Then typing ⌥→ leads you to the state $\sqrt[|]{x}$. Conversely, pressing ⌫ in the latter situation will remove the degree and bring you back to the original state.

Notice that the sqrt tag for square roots and n-th roots is an example of a tag with a variable number of arguments (either one or two in this case). The above keyboard shortcuts ⌥→ and ⌫ can be used more generally in such situations for the insertion and removal of arguments: see section 10.5.

Ordinary text. Ordinary text can be inserted into a formula using ⌥$ or Insert ▸ Text. This is what we did for the text " is sufficiently large" below:

$$L = \{x \mid x \text{ is sufficiently large}\}.$$

Negations. Some of the TeXmacs relation symbols do not admit any negated versions. You may manually negate a symbol or a formula using ⌥n or Insert ▸ Negation:

$$a \not\mathcal{R} b \iff \neg(a \mathcal{R} b).$$

4.4 Big operators

Big operators—such as integration $\int$ and n-ary summation $\sum$—are entered using the keyboard prefix ⇧F5 or from the Insert ▸ Symbol ▸ Big operator menu. Table 4.11 lists the available big operators together with the corresponding keyboard shortcuts. The rendering of big operators is quite different for inline and displayed formulas: compare $\sum_{k=1}^{\infty} 1/k^2$ and $\int_0^1 f(x)\,\mathrm{d}x$ with

$$\sum_{k=1}^{\infty} \frac{1}{k^2} \qquad \int_0^1 f(x)\,\mathrm{d}x.$$

The big integral signs has two variants, depending on where you want to place subscripts and superscripts in display mode. By default, the scripts are placed to the right:

$$\int_0^{\infty} \frac{\mathrm{d}x}{1+x^2}.$$

The alternative rendering "with limits"

$$\int\limits_0^{\infty} \frac{\mathrm{d}x}{1+x^2}.$$

can be obtained using the ! postfix, by typing ⇧F5 I !. Some people also prefer integrals to be rendered using an upright font. This gives rise to two further variants. For instance, you may obtain the triple integral in

$$\iiint\limits_V f(x,y,z)\,\mathrm{d}x \wedge \mathrm{d}y \wedge \mathrm{d}z$$

using the shortcut ⇧F5 I I I ⇥ ⇥ ⇥. We recall that it is good practice to enter the d's of the differentials using d ⇥ ⇥, so as to distinguish them from the letter d in $a+b+c+d$. Similarly, the exterior product was entered using * &.

Shortcut	Result	Shortcut	Result	Shortcut	Result
⇧F5 I	$\int$	⇧F5 I I	$\iint$	⇧F5 I I I	$\iiint$
⇧F5 O	$\oint$	⇧F5 O O	$\oiint$		
⇧F5 A	$\coprod$	⇧F5 P	$\prod$	⇧F5 S	$\sum$
⇧F5 @ +	$\bigoplus$	⇧F5 @ *	$\bigotimes$	⇧F5 @ .	$\bigodot$
⇧F5 U	$\bigcup$	⇧F5 N	$\bigcap$	⇧F5 U +	$\biguplus$
⇧F5 U ⇥	$\bigsqcup$	⇧F5 N ⇥	$\bigsqcap$	⇧F5 U ⇥ ⇥	$\Box$
⇧F5 V	$\bigvee$	⇧F5 V ⇥	$\curlyvee$	⇧F5 V ⇥ ⇥	$\bigtriangledown$
⇧F5 W	$\bigwedge$	⇧F5 W ⇥	$\curlywedge$	⇧F5 W ⇥ ⇥	$\bigtriangleup$
⇧F5 B	$\Vert$	⇧F5 B ⇥	$\vert\vert\vert$		

Table 4.11. Big mathematical operators.

4.5 Large delimiters

Brackets inside mathematical formulas should always match. As soon as you open a bracket "(", TeXmacs therefore automatically inserts the corresponding closing bracket ")". Occasionally, it may occur that you do not want the closing bracket, or that you want to replace it by another kind of bracket. This can be done as follows:

- If your cursor is just before the closing bracket inside $(a,b|)$, then pressing] will turn the expression into $(a,b]|$. More generally, this mechanism can be used to turn any closing bracket into one of any other form.
- If your cursor is just behind the closing bracket inside $\{{a \atop b}\}|$, then pressing ⌫ will remove the closing bracket, yielding $\{{a \atop b}|$. Similarly, in order to remove an opening bracket, put the cursor just in front of it and press ⌦. Notice that pressing ⌫ or ⌦ in the example $f(|)$ will remove both brackets.

Notice that the automatic insertion of matching closing brackets can be disabled using Edit ▸ Preferences ▸ Keyboard ▸ Automatic brackets ▸ Disabled. It should also be noticed that the closing curly bracket was not really removed in the above formula $\{{a \atop b}|$: TeXmacs rather replaced it with an "invisible" closing bracket (see section 4.11), thereby ensuring that all brackets remain matching. Selection of the preference Automatic brackets ▸ Disabled has a similar effect of replacing visible closing brackets by invisible ones.

For some delimiters, such as |, the opening and closing versions coincide. For instance, entering a vertical bar | will produce an absolute value. The vertical bar is often used as a separator as well. You may insert a small bar-separator |

using | ⇥ or F5 |. Large bar-separators are entered using ⌥m | or | ⇥ ⇥; they are used for producing the vertical bars in formulas like

$$\left\langle \frac{a}{b+c} \middle| \frac{p}{q+r} \middle| \frac{a}{b+c} \right\rangle.$$

We notice that such large separators do not exist in (LA)TEX. There may be as many middle delimiters between a left and a right delimiter as needed and they are all scaled to the same appropriate size. Notice that not all vertical bars are delimiters. For example, the binary relation "divides" is entered using | ⇥ ⇥ ⇥ ⇥ or F5 | ⇥ ⇥; we will see more examples in section 4.11.

In TEXMACS, large delimiters may either be "left delimiters", "right delimiters" or "middle delimiters". By default, (, [, {, and ⟨ are left delimiters, whereas),], }, and ⟩ are right delimiters. But their status can be changed using the ⌥l, ⌥r, and ⌥m prefixes. For example, ⌥l) produces), considered as a large left delimiter, together with the corresponding "closing bracket" (. Open French-style intervals $]a,b[$ can therefore be entered using ⌥l].

By default, the sizes of the brackets are adjusted to the expression between the brackets. Small delimiters, created using the ⌥-prefix, keep their sizes independently of the enclosed expression. Thus, the brackets in the left-hand and right-hand expressions below were entered respectively using (and ⌥(:

$$f\left(\sqrt{\mathrm{e}^{\mathrm{e}^{\mathrm{e}^x}}+1}\right) \qquad f(\sqrt{\mathrm{e}^{\mathrm{e}^{\mathrm{e}^x}}+1}).$$

You may also use ^* in order to toggle between large and small delimiters.

Sometimes you may want large delimiters of a particular size, instead of the self-adjusting ones. This can be achieved in two main ways. The best strategy is usually to tweak the size of the expression between the brackets using the Format ▸ Adjust menu. For example, if you think that the brackets in the formula

$$f\left(\frac{x}{y}\right)$$

are slightly too large, then you may reduce the size of the fraction using Format ▸ Adjust ▸ Reduce, which yields

$$f\left(\tfrac{x}{y}\right).$$

TEXMACS also allows you to directly increase or decrease the size of the brackets themselves using the keyboard shortcuts ⌘↑ and ⌘↓. In order to reset to the initial size, you may use ⌘⌫.

We finally notice that it is possible to insert a pair of invisible brackets using ⌥l .. This is for instance useful in computational contexts, when formulas should have a precise, not merely visual semantics. Alternatively, you may use ⌘= to confine a selected formula inside a "horizontal group". This additionally prevents the formula from being hyphenated.

Shortcut	Example	Wide example	Shortcut	Example	Wide example
⌥~	$\tilde{x}$	$\widetilde{x+y}$	⌥B	$\bar{x}$	$\overline{x+y}$
⌥^	$\hat{x}$	$\widehat{x+y}$	⌥C	$\check{x}$	$\widecheck{x+y}$
⌥A	$\overset{\frown}{x}$	$\overset{\frown}{x+y}$	⌥U	$\breve{x}$	$\overset{\smile}{x+y}$
⌥V	$\vec{x}$	$\overrightarrow{AB}$	⌥o < >	$\overleftrightarrow{x}$	$\overleftrightarrow{ABCD}$
⌥o <	$\overleftarrow{x}$	$\overleftarrow{ABCD}$	⌥o >	$\overrightarrow{x}$	$\overrightarrow{ABCD}$

Table 4.12. Keyboard shortcuts for wide mathematical accents.

4.6 Wide accents and extensible arrows

Mathematical accents are created using the modifier key ⌥ or the prefix ⌥o (as in Over). For example, $\hat{\varphi}$ is entered by typing ⌥^ f ⇥, and $\overrightarrow{ABCD}$ using ⌥o > A B C D. Many of these accents stretch with the formulas below them, as in $\widetilde{\varphi+\psi}$. The list of all such *wide accents* is given in Table 4.12. Other accents never stretch with the formulas below them: see Table 4.13.

The same accents may be inserted below the expressions using the ⌥u prefix (Under). Hence $\underline{x+y}$ is entered using ⌥u B x + y and $\underleftrightarrow{ABCD}$ is obtained by typing ⌥u < > A B C D.

Shortcut	Example	Shortcut	Result
⌥.	$\dot{x}$	⌥'	$\acute{x}$
⌥"	$\ddot{x}$	⌥`	$\grave{x}$
⌥" ⇥	$\dddot{x}$	⌥@	$\mathring{x}$
⌥" ⇥ ⇥	$\ddddot{x}$		

Table 4.13. Keyboard shortcuts for non-stretchable accents.

So-called *overbraces* and *underbraces* are entered in a similar way as wide accents. They have the additional property that superscripts and subscripts are placed above and below them. In Table 4.14, we have listed the available overbraces. The expression below can therefore be obtained by typing ⌥o { a + b + c → ^ S ⇥:

$$\overbrace{a+b+c}^{\Sigma}.$$

Shortcut	Example	Shortcut	Example
⌥o {	$\overbrace{x+y+z}$	⌥o }	$\overset{\underbrace{\hphantom{x+y+z}}}{x+y+z}$
⌥o (	$\overparen{x+y+z}$	⌥o)	$\overset{\underparen{\hphantom{x+y+z}}}{x+y+z}$
⌥o [	$\overbracket{x+y+z}$	⌥o]	$\overset{\underbracket{\hphantom{x+y+z}}}{x+y+z}$

Table 4.14. Wide overbraces.

$\overline{a+b+c}$	$\xrightarrow{a+b+c}$	$\xleftarrow{a+b+c}$	$\xleftrightarrow{a+b+c}$	$\xmapsto{a+b+c}$
$\overset{a+b+c}{=\!=}$	$\xRightarrow{a+b+c}$	$\xLeftarrow{a+b+c}$	$\xLeftrightarrow{a+b+c}$	

Table 4.15. Available extensible arrows.

Besides delimiters and accents, some of the arrows are also extensible in TeXmacs. This is typically needed whenever they come with a superscript above them (or a subscript below them). For instance, the following formula was entered by typing E - > ^ f @ * g → F:

$$E \xrightarrow{f \otimes g} F.$$

Here - > ^ was used as a keyboard shortcut for creating an extensible arrow with a superscript above it. Whenever you need a non-extensible arrow with an ordinary superscript (in the upper-right corner), the trick is to "break" the keyboard shortcut; e.g. E - > ← → ^ + → F yields

$$E \rightarrow^{+} F.$$

Table 4.15 lists the available extensible arrows in TeXmacs.

4.7 Matrices and mathematical tables

New 1×1 matrices can be entered using Insert ▸ Table ▸ Matrix, ⌘t N m (Table New Matrix), (⌘t N T (Table New Table), or \ m a t r i x ↩. Matrices are special kinds of tables, which means that the full range of table editing facilities is available (see chapter 5). For now, it suffices to know that new rows and columns can be inserted using the shortcuts ↩, ⌥→, ⌥←, ⌥↑, and ⌥↓. For instance, the matrix

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

can be entered using ⌘t N m a ⌥→ b ↩ c → d. Empty rows and columns can be removed simply by pressing ⌫.

There are a few other kinds of table-like structures that are useful in mathematics, starting with the following variants of matrices: determinants (⌘t N d or | ⌘t N T), matrices with angular brackets (⌘t N M or [⌘t N T), and ordinary centered tables (⌘t N T), which correspond to matrices with no brackets at all:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \begin{vmatrix} a & b \\ c & d \end{vmatrix} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \begin{matrix} a & b \\ c & d \end{matrix}.$$

Note that you may cycle among the first three variants using ^⇥. Secondly, you may create so-called "choice lists" using ⌘t N c or { ⌘t N T, as in

$$f(x) = \begin{cases} 0 & \text{if } x \leqslant 0 \\ e^{-\frac{1}{x^2}} & \text{otherwise.} \end{cases}$$

Finally, $\mathrm{T_EX_{MACS}}$ provides the stack tag for tables in which the spacing is reduced to the strict minimum (⌘t N s). This is useful when tables are used in subscripts or superscripts:

$$S = \sum_{\substack{1 \leqslant i \leqslant n \\ 1 \leqslant j \leqslant m}} \varphi(x_i, y_j).$$

4.8 Commutative diagrams

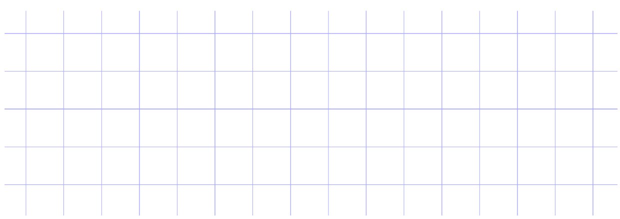
The most appropriate way to create commutative diagrams depends on the required level of complexity. The simplest kind of diagrams can be constructed as centered tables:

$$\begin{tikzcd} A \amalg B & A \arrow[l, "\iota_A"'] \arrow[d, "\alpha"] \\ B \arrow[u, "\iota_B"] \arrow[r, "\beta"'] & X \end{tikzcd} . \tag{4.1}$$

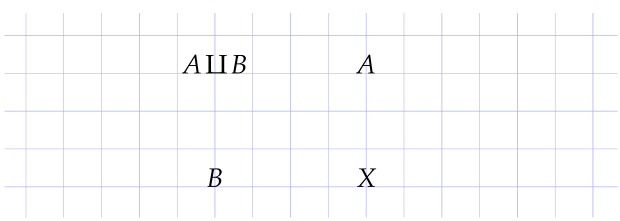
Here we recommend the use of long horizontal and vertical arrows. The small scripts ι_B and α beside the long vertical arrows were created using the markup elements left-script and right-script.

Centered tables may quickly become problematic when you need diagonal or curved arrows. The most general approach to commutative diagrams is to consider them as pictures. For general information about how to draw technical pictures, we refer to chapter 8. In this section, we will describe those features that are useful for creating the same commutative diagram (4.1) as above.

Inside a mathematical formula, we start with the creation of an empty commutative diagram using Insert ▸ Image ▸ Commutative diagram. This inserts a small grid of the following type:



In order to ensure a uniform design, the main nodes $A \amalg B$, A, B, and X should be placed at points on the grid, using the default alignment (namely: horizontally centered and vertically aligned at the baseline):



At the next stage, we want to add the arrows. $\mathrm{T_EX_{MACS}}$ provides a special mode for adding arrows surrounded by text: Insert ▸ Arrow with text. Furthermore, when moving the mouse around the four formulas, you will observe that there are eight special positions around each formula that "attract" the pointer. These special points should be used as the starting and end points of your arrows:

$$\begin{tikzcd} A \amalg B & A \arrow[l, "x"'] \arrow[d, "x"] \\ B \arrow[u, "x"] \arrow[r, "x"] & X \arrow[from=1-1, "x"] \end{tikzcd}$$

For the diagonal arrow, a dashed line style would actually have been more appropriate. Such alternative line styles can be selected using Focus ▸ Line dashes. At the next stage, you should select Insert ▸ Mathematics and edit the formulas next to the arrows. You may use ^⇥ to put the text at the other side of an arrow.

$$\begin{tikzcd} A \amalg B & A \arrow[l, "\iota_A"'] \arrow[d, "\alpha"] \\ B \arrow[u, "\iota_B"] \arrow[r, "\beta"'] & X \arrow[from=1-1, dashed] \end{tikzcd}$$

In order to finalize the commutative diagram, it now suffices to remove the grid and "crop" the diagram using Insert ▸ Grid ▸ Default and Insert ▸ Geometry ▸ Crop ▸ Crop:

$$\begin{tikzcd} A \amalg B & A \arrow[l, "\iota_A"'] \arrow[d, "\alpha"] \\ B \arrow[u, "\iota_B"] \arrow[r, "\beta"'] & X \arrow[from=1-1, dashed] \end{tikzcd} . \tag{4.2}$$

If the diagram needs further editing, then you may restore the grid using Insert ▸ Grid ▸ Notebook.

A few further remarks are in order. First of all, $\mathrm{T_EX_{MACS}}$ does *not* automatically adjust the endpoints of the arrows if you edit one of the formulas A, B, X, or $A \amalg B$. On the other hand, the position of the formula next to the arrow is automatically updated when modifying the endpoints. In documents with several commutative diagrams, we also recommend to consistently opt for either one of the two styles exemplified in (4.1) and (4.2).

4.9 Semantics of mathematical formulas

Recent versions of $\mathrm{T_EX_{MACS}}$ can help you to write documents in which all formulas are at least correct from a syntactical point of view. For example, in the formula $a+b$, the computer will understand that + is an operator that applies to the arguments a and b. It will also flag $a+$ as being incorrect. In fact, the "semantics" that we aim at does not go any further: $\mathrm{T_EX_{MACS}}$ is unaware of the mathematical nature of addition.

Semantic editing does require additional efforts from the user, or at least a little adaptation. For instance, it is the user's job to enter multiplications using the shortcut * and function applications using ␣. These operations cannot be distinguished from their appearance, since they are both printed as invisible whitespace. However, the semantics of these operations is clearly very different.

Although semantically correct documents are usually not very different from informal presentation-oriented documents as far as typesetting is concerned, the additional user effort may pay off for several reasons:

- Adequate semantics is a prerequisite when using formulas as input for a computer algebra system.
- Syntactically correct documents are less likely to contain "typos" or more intricate mathematical errors.
- For certain editing operations, such as cut and paste, one may directly select subformulas that are meaningful from the syntactical point of view.
- It reduces the risk of using non-standard notations that will be difficult to understand for potential readers of your work.

Furthermore, other semantic facilities might be integrated in the feature, such as semantic search and replace, or semantic search on the web.

In order to activate the semantic editing facilities, you must toggle Edit ▸ Preferences ▸ Mathematics ▸ Semantic editing. In the semantic editing mode, several of the structured editing features of $\mathrm{T_EX_{MACS}}$ apply to the syntactic structure of the formula, rather than the visual structure of the document. For instance, the *semantic focus* is usually a subformula of the current focus. Similarly, only syntactically meaningful subformulas can be selected when making a selection.

The semantic focus is useful for several reasons. First of all, it is displayed in green if the formula is syntactically correct and in red if you made an error. This allows to quickly notice any typos while entering a formula. Secondly, if you have any doubt on the precedence of a mathematical operator or relation, then the semantic focus tells you the default interpretation: by putting your cursor right next to your operator, the subexpression to which the operator applies will be highlighted. In the case of an addition, or a more general associative operator, all summands are highlighted.

4.10 Common errors and syntax corrections

By default, the semantic editing mode "understands" most classical mathematical notations. This is achieved through the use of a carefully designed grammar for mainstream mathematics. Obviously, the use of a fixed grammar may cause the following problems:

- Mathematical formulas frequently contain *ad hoc* notations. For instance, the formulas might contain some text or meaningful whitespace. Another example of an *ad hoc* notation is the sign sequence $++-+-+$. In such cases, the user should explicitly annotate the appropriate parts of the formula in order to make them semantically meaningful; see section 4.12.

- The TEXMACS grammar used for the interpretation of mathematical formulas may be incomplete or inadequate for certain situations. It is possible to customize or extend the grammar using the standard TEXMACS macro mechanism (see section 4.12). Notations for specific areas may be grouped together in dedicated style packages.

Besides these intrinsically hard-to-avoid problems, the following common mistakes are a further source of trouble for associating semantics to mathematical formulas:

- Since TEXMACS is a wysiwyg editor, some of the structure of the document is invisible to the user. For example, the presence of a mathematical formula $x+y$ is indicated through the use of an italic slant and special spacing. However, in the formula $f(x)$ it is easy to type the closing bracket outside the formula, with no visual difference.

- Various mathematical notations are visually ambiguous. For example, $a\,(b+c)$ would usually be understood as $a\cdot(b+c)$, whereas $f(x+y)$ typically corresponds to a function application. In the semantic editing mode, the user is expected to resolve this ambiguity by hand by entering multiplications using * and spaces using ␣. The multiply/apply ambiguity is one of the main sources of syntax errors, since many users do not pay attention to invisible differences. Similarly, the $\wedge$ glyph could be the "logical and" or the "wedge product". This "homoglyph" issue will be addressed in more detail in section 4.11 below.

- It could be that a text was originally written in LATEX or an old version of TEXMACS. In that case, the document contains no special indication on matching brackets or the scopes of big operators. For example, in the formula $[x,y[$, should we interpret the second bracket as a closing bracket? This is indeed the standard French notation for an interval with an open right end. More generally, all problems that we have mentioned so far tend to be present simultaneously when trying to associate semantics to existing documents.

After activation of the semantic editing mode, you may check whether a formula is correct by positioning your cursor inside it and looking at the color of the bounding box of the semantic focus: a green color corresponds to a correct formula and a red color indicates an error in the formula. Alternatively, assuming that the focus is on a mathematical formula, you may select Focus ▸ Preferences ▸ Highlight incorrect formulas, in which all incorrect formulas are highlighted inside red boxes.

For the second kind of common errors, TEXMACS includes an automatic syntax corrector. Assuming that your cursor is inside a formula, you may use Edit ▸ Correct ▸ Correct all for the correction of all formulas in your document, or the correction of the current selection. If the versioning tool is activated (see section 10.9), then you may use Edit ▸ Correct ▸ Correct manually to show the differences between the original and the corrected versions. You may then use the versioning tool to go through these differences and select the preferred versions.

The precise algorithms that are used for the correction may be enabled or disabled from Edit ▸ Preferences ▸ Mathematics ▸ Manual correction:

Remove superfluous invisible operators. This algorithm is used in order to remove any superfluous function applications or multiplications. For instance, users who are accustomed to editing ASCII files often type spaces around binary infixes such as addition. Such "function applications" will be removed by this algorithm.

Insert missing invisible operators. In LATEX, multiplications and function applications are never entered explicitly. When importing a LATEX document, it is therefore important to detect and insert missing multiplications and function applications.

Homoglyph substitutions. This algorithm may perform some other useful substitutions of symbols by visually similar, but semantically distinct symbols. For instance, whenever appropriate, the backslash symbol \ is replaced by the binary "set difference" infix (as in $X \setminus Y$).

From the Edit ▸ Preferences ▸ Mathematics ▸ Automatic correction, you may also select the correction algorithms that should be applied automatically whenever you open a file. These corrections are always carried out when importing a LATEX file.

After syntax correction, the remaining errors indicate genuine typos at worst, or non-standard or unsupported notations at best. We also note that "correct" formulas do not necessarily have the intended meaning. In order to check whether the operators indeed apply to the intended arguments, you should keep an eye on the current focus while typing your formulas.

4.11 Semantics of mathematical symbols

The mathematical symbols in TEXMACS all come with a certain number of properties that correspond to their intended meaning. For example, TEXMACS is aware that "+" is an infix operator, whereas "!" is rather a postfix, and "," a separator.

TEXMACS has special symbols $\mathrm{e} = 2.71828\cdots$, $\pi = 3.14159\cdots$, and $\mathrm{i} = \sqrt{-1}$ for important mathematical constants. These constants are displayed "upright" and they can be entered using the shortcuts `e ⇥ ⇥`, `p ⇥ ⇥`, and `i ⇥ ⇥`. Note that ordinary variables using the same letters are rendered in italics, as e, π, and i.

However, semantically distinct symbols may display in a similar way. For instance, the comma separator, as in $f(x,y)$, is different from the decimal comma, as in $3{,}14159\cdots$. Notice that the two symbols admit different spacing rules. Semantically distinct symbols that are rendered using the same glyph are called *homoglyphs*. Notice that our semantics is purely syntactic: for instance, the + infix is commonly used for addition, but sometimes also for the concatenation of strings. Nevertheless, these two uses do not differ from a syntactical point of view, since the + symbol remains a binary infix operator with the same precedence with respect to other operators.

The most confusing homoglyphs are the various invisible symbols supported by TEXMACS:

- The multiplication, entered by `*`. Example: $a b$.
- Function application, entered by `␣`. Example: $\sin x$.
- An invisible separator, entered by `, ⇥ ⇥`. Example: $A = (a_{ij})$.
- An invisible addition, entered by `+ ⇥ ⇥ ⇥ ⇥`. Example: $17\,{}^3/_8$.
- An invisible symbol, entered by `. ⇥ ⇥ ⇥`. Examples: the increment operator $+1$, the zeros in a diagonal matrix

  $$\begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix},$$

  or the omitted exponent one in $x_1^3 - x_1^2 + x_1 - 1$ (using an invisible exponent for the degree one term allowed us to vertically align all subscripts).
- An invisible bracket (mainly for internal use). A matching pair of invisible brackets is entered using `( ⇥`.

We recommend in particular that you make it a habit to systematically distinguishing between multiplication and function application: we already mentioned the ambiguity $a(b+c)$ *versus* $f(x+y)$), which cannot be resolved automatically.

Shortcut	Glyph	Example	Semantics
*		$a\,b$	Multiplication
␣		$\sin x$	Function application
, ⇥ ⇥		$a_{ij} = a_{ji}$	Invisible separator
+ ⇥ ⇥ ⇥ ⇥		$17\,{}^{3}\!/_{8}$	Invisible addition
. ⇥ ⇥ ⇥		$+1 : x \mapsto x+1$	Invisible symbol
(⇥		$\Phi \equiv \forall x, P(x)$	Invisible bracket(s)
|	$\vert$	$\vert -x \vert = \vert x \vert$	Absolute value
| ⇥	$\vert$	$\langle a \vert b \vert c \vert d \rangle$	Separating bar
| ⇥ ⇥	$\vert$	$\left\langle \frac{v+w}{2} \middle\vert \frac{u-\bar{u}}{2} \right\rangle$	Extensible middle bar
| ⇥ ⇥ ⇥ ⇥	$\vert$	$\{x \in \mathbb{R} \mid x > 0\}$	Such that bar
| ⇥ ⇥ ⇥ ⇥ ⇥	$\vert$	$11 \mid 1001$	Divides relation
,	,	$f(x,y)$	Comma separator
, ⇥	,	123,456	Decimal comma
.	.	123.456	Decimal point
. ⇥	.	$\lambda x . x^2$	Dot connector
* ⇥ ⇥ ⇥	$\cdot$	$v \cdot w$	Dot multiplication
. ⇥ ⇥	$\cdot$	$\cdot + 1$	Dummy wildcard
:	:	$a{:}b{:}c$	Separator
: ⇥	:	$x : \mathsf{Int}$	Type satisfaction
/ ⇥	:	$121 : 11 = 11$	Division
: ⇥ ⇥ ⇥	:	$\{x \in E : P(x)\}$	Such that colon
\ ⇥	\	$\backslash x$	Backslash
\ ⇥ ⇥	\	$\mathbb{N}^{>} = \mathbb{N} \setminus \{0\}$	Set minus
&	$\wedge$	$1 = 1 \wedge 2 = 2$	Logical and
* &	$\wedge$	$\mathrm{d}x \wedge \mathrm{d}y$	Wedge product

Table 4.16. Homoglyphs supported by TeXmacs.

TeXmacs mainly relies on the standard variant system for entering homoglyphs. See Table 4.16 for the complete list of supported homoglyphs.

4.12 Customizing the semantics

We have done our best to support most of the traditional mathematical notations in TeXmacs. Nevertheless, you may sometimes need notations with a non-standard semantics. Certain areas may also require special notations that are not supported by default.

TeXmacs provides a very simple syntax primitive that allows you to manually override the default syntactical semantics of a formula. Assuming that semantic editing was activated, you may insert a syntax tag using ⌥x x or Insert ▸ Semantics ▸ Other. The first argument contains the formula as it should be displayed, whereas the second argument contains the formula as it should be interpreted.

For example, if we enter $\mathcal{R}$ as the first argument and $<$ as the second one, then the $\mathcal{R}$ will be interpreted as a binary relation, exactly in the same way as $<$. Moreover, the spacing around $\mathcal{R}$ will be adapted, so as to mimic the spacing around $<$. In this particular example, we might have obtained the same result by using the math-relation tag, which is essentially a syntax tag $<$ as second argument. Most standard operator types are available from the Insert ▸ Semantics menu or using the keyboard prefix ⌥x. In particular, you may use ⌥x ␣ to simply ignore a formula and ⌥x o in order to make the formula behave as an ordinary symbol (such as the letter "x").

The syntax primitive is particularly powerful in combination with the TEXMACS macro language. For instance, consider the formula $C=1/2\pi\mathrm{i}\oint f(z)\,\mathrm{d}z$. It is likely that the intended interpretation of $1/2\pi\mathrm{i}$ is $1/(2\pi\mathrm{i})$ and not $(1/2)\pi\mathrm{i}$. Therefore, if we often use the constant $2\pi\mathrm{i}$, then it is convenient to define a macro twopii by

⟨assign|*twopii*|⟨macro|⟨syntax|$2\pi\mathrm{i}$|$(2\pi\mathrm{i})$⟩⟩⟩

You may group such macros together into style package with your favorite notations. Future versions of TEXMACS might also provide style packages with notations dedicated to specific areas.

Let us finally note that there are usually several ways for redefining the semantics of a formula. For example, an alternative way to define the macro twopii is using

⟨assign|*twopii*|⟨macro|$2\pi\mathrm{i}$⟩⟩

where we inserted a pair of invisible brackets around $2\pi\mathrm{i}$. Similarly, in the formula

$$\mathrm{e}^{\sqrt{x}+\mathrm{e}^{\sqrt{\log x}+\mathrm{e}^{\sqrt{\log\log x}+\cdot^{\cdot^{\cdot}}+\log\log\log x}+\log\log x}+\log x},$$

we may select the whole formula and give it the semantics of an ordinary symbol, by pressing ⌥x o. However, a nicer solution is to only select the subformula $\cdot^{\cdot^{\cdot}}$, and give it the semantics of an ordinary symbol. Yet another example is the sign sequence $++-+-+$ mentioned earlier. This sequence can be interpreted correctly by inserting invisible separators between the different signs using the , ␣ ␣ shortcut.

Chapter 5

Tables

Tables form one of the most versatile layout primitives of $\text{T}_\text{E}\text{X}_\text{MACS}$. You can create ordinary textual tables and specify the properties of individual cells in a very precise way. More generally, matrices, aligned lists of equations, and various other kinds of markup are also formatted as tables.

5.1 Different flavors of tables

The main kinds of tables are listed in the Insert ▸ Table menu and also available through the keyboard prefix ⌘t N. Here is an example of a "wide block" that was created using Insert ▸ Table ▸ Wide block or ⌘t N W:

Isaac Newton	1642–1726	English physicist and mathematician. With his book *Philosophiæ Naturalis Principia Mathematica* ("Mathematical Principles of Natural Philosophy"), he is considered to be the father of classical mechanics. Newton is also one of the founders of modern calculus.
Leonhard Euler	1707–1783	Prolific Swiss mathematician and physicist, who made major discoveries in many branches of mathematics and physics, such as infinitesimal calculus, number theory, graph theory, partial differential equations, fluid mechanics, astronomy, etc.

The distinctive feature of a "block" is that the borders of cells are visible as lines. Furthermore, a "wide" table extends over the entire width of a paragraph and long lines inside individual cells are wrapped when necessary.

Remark 5.1. In other word processors, "wide tables" are usually the default. $\text{T}_\text{E}\text{X}_\text{MACS}$ provides many kinds of tables that are *not* wide, in which case text inside cells is usually *not* line-wrapped. This is indeed most natural for small inline tables like matrices inside a formula. However, it is the responsibility of the user to prevent such tables from getting too large and not fitting on a single line. Line wrapping can be enabled via Focus ▸ Cell ▸ Line wrapping ▸ Top or by toggling the ☰ icon on the focus bar. In that case, you should also indicate the desired width of the cell (see section 5.3.2).

Insert ▸ Table ▸ Big table creates a "big" paragraph-wide table with a number and a legend. If you rather wish to fit several numbered tables with accompanying legends on a single line, then you may use Insert ▸ Table ▸ Small table. Tables 5.1 and 5.2 show an example of two small juxtaposed tables.

boom	tree
hallo	hello
wiskunde	mathematics

Table 5.1. Plain tabular, ⌘t N t.

boom	tree
hallo	hello
wiskunde	mathematics

Table 5.2. Centered block, ⌘t N B.

Inside math mode, a few additional kinds of tables Matrix, Determinant, Choice, and Stack are available through the Insert ▸ Table menu (see Figures 5.1–5.4). The stack tag corresponds to tables that are compressed as much as possible. This makes them useful inside subscripts, superscripts, or other places where little space is available.

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}$$

Figure 5.1. Matrix

$$\begin{vmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{vmatrix}$$

Figure 5.2. Determinant

$$f(x) = \begin{cases} 0 & \text{if } x \leqslant 0 \\ \mathrm{e}^{-\frac{1}{x}} & \text{if } x > 0 \end{cases}$$

Figure 5.3. Choice

$$\sigma_n = \sum_{\substack{1 \leqslant i \leqslant n \\ i \leqslant j \leqslant 2i}} \varphi_{i,j}$$

Figure 5.4. Stack

There are several other table-like environments and new ones may be created by the user. For instance, using Insert ▸ Mathematics ▸ Equations or ⌥&, you may insert an eqnarray* environment. This allows mathematical users to align lists of equations that extend over entire lines. An example of such a list of equations is

$$\begin{aligned} [\sin(f(x)\,g(x))]' &= (f(x)\,g(x))'\cos(f(x)\,g(x)) \\ &= (f'(x)\,g(x) + f(x)\,g'(x))\cos(f(x)\,g(x)) \end{aligned}$$

5.2 Basic table editing

When starting a new table, its size is minimal (usually 1×1) and its cells are empty. New rows and columns can be inserted using the keyboard shortcuts ⌥←, ⌥→, ⌥↑, and ⌥↓. For instance, ⌥↓ creates a new row below the current cursor position, as illustrated in Table 5.3 below. Alternatively, you may start a new row by hitting the ↵ key. New rows and columns can also be inserted from the Focus ▸ Resize menu, or using the icons , , , and on the focus bar. Finally, it is possible to specify the exact number of rows and columns of a table using Focus ▸ Table ▸ Size ▸ Set number of rows and Focus ▸ Table ▸ Size ▸ Set number of columns.

	⌥→	⌥←	⌥↓	⌥↑	↩
$\begin{pmatrix} a & b\vert \\ c & d \end{pmatrix}$	$\begin{pmatrix} a & b & \vert \\ c & d & \end{pmatrix}$	$\begin{pmatrix} a & \vert & b \\ c & & d \end{pmatrix}$	$\begin{pmatrix} a & b \\ & \vert \\ c & d \end{pmatrix}$	$\begin{pmatrix} & \vert \\ a & b \\ c & d \end{pmatrix}$	$\begin{pmatrix} a & b \\ \vert & \\ c & d \end{pmatrix}$

Table 5.3. Inserting new rows and columns using the cursor keys. Starting from the initial 2×2 matrix at the left hand side, with the cursor behind the character b, the table indicates the results after pressing the keys ⌥→, ⌥←, ⌥↓, ⌥↑, and ↩.

	⌫	⌦
$\begin{pmatrix} a & \vert & b \\ c & & d \end{pmatrix}$	$\begin{pmatrix} a\vert & b \\ c & d \end{pmatrix}$	$\begin{pmatrix} a & \vert b \\ c & d \end{pmatrix}$
$\begin{pmatrix} a & b \\ \vert & \\ c & d \end{pmatrix}$	$\begin{pmatrix} a & b\vert \\ c & d \end{pmatrix}$	$\begin{pmatrix} a & b \\ \vert c & d \end{pmatrix}$

Table 5.4. Removing empty rows and columns using the keys ⌫ and ⌦.

Rows and columns that are entirely empty can be deleted using the keys ⌫ or ⌦, as indicated in Table 5.4. No matter whether they are empty or not, rows and columns can be deleted from the Focus ▸ Resize menu, or using the icons, , , and from the focus bar.

Rows, columns, and more general rectangular selections of cells can easily be cut, copied, and pasted. Selections of cells are indicated using a different color (magenta) from that of usual selections (which are red). One typical scenario for copying and pasting a rectangular selection of cells is shown in Figure 5.5.

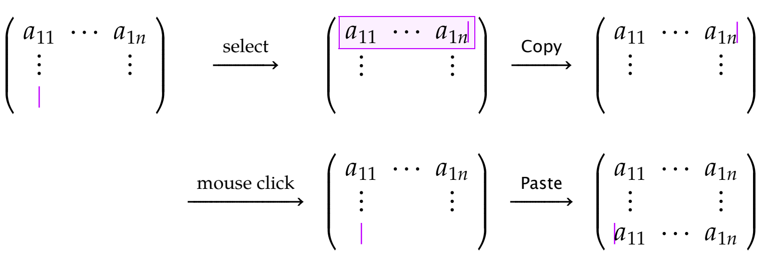


Figure 5.5. One typical scenario of copying and pasting a range of cells. Starting with the matrix at the left-hand side, we first select the top row using the mouse. We next copy the selection using Edit ▸ Copy or ⌘W and reposition the cursor inside the first cell of the third row. We finally paste the selection using Edit ▸ Paste or ^y.

5.3 Cell properties

5.3.1 Cell operation modes

The individual cells of tables have several properties that can be changed by the user: horizontal and vertical alignment, background color, width and height, border width and padding, etc. One first way to edit these properties is to open the cell property editor using Format ▸ Cell. Alternatively, changes can be made via the entries in the Focus ▸ Cell menu or the corresponding icons on the focus bar. Some properties can also be edited using the keyboard.

By default, cell operations are performed on the cell that contains the cursor. It is also possible to jointly modify the cell properties of all cells in the table, or of all cells in the row or column that contains the cursor. For this, it suffices to change the "cell operation mode", by selecting Entire table, Row or Column in the menu Focus ▸ Cell ▸ Operation mode. Similarly, if a selection of cells is active, then the properties are modified jointly for all cells in the selection.

Internally, cell properties are associated to rectangular regions of cells. Let us mention one useful implication of this fact. Assume that the first row of the following table was colored by selecting the first row and changing the background color to "pastel yellow" using Focus ▸ Cell ▸ Background color:

English	house	tree	beast
French	maison	arbre	bête
Dutch	huis	boom	beest

Then TEX_{MACS} associates this background color to the entire region corresponding to the first row. Now assume that we insert a new column between house and tree. This will yield

English	house		tree	beast
French	maison		arbre	bête
Dutch	huis		boom	beest

In other words, it is not necessary to specify the background color of the top cell again in the new column, since this cell is "naturally part of the region" for which we set the color to pastel yellow. This would not have been the case if we had separately specified the background color for each of the cells in the top row of the original table. In that case, the insertion of a new column between house and tree would have resulted in

English	house		tree	beast
French	maison		arbre	bête
Dutch	huis		boom	beest

5.3.2 Width and height

The widths of columns and heights of rows are computed automatically as a function of the content of the cells and constraints that can be specified by the user.

The width of a column is the maximum of the widths of its cells and similarly for the height of a row. The default width of a cell simply coincides with the width of its contents (except when you enable line wrapping, in which case

Nice	Dull
Even nicer	Even duller
Really very very nice	Extremely dull

Figure 5.6. Automatic width.

Nice	Dull
Even nicer	Even duller
Really very very nice	Extremely dull

Figure 5.7. Exact width of `1 in`.

Nice	Dull
Even nicer	Even duller
Really very very nice	Extremely dull

Figure 5.8. Minimal width of `1 in`.

Nice	Dull
Even nicer	Even duller
Really very very nice	Extremely dull

Figure 5.9. Maximal width of `1 in`.

the default width becomes zero). The default width can be overridden using Focus ▸ Cell ▸ Width, Format ▸ Cell ▸ Width, or the icon menu on the focus bar. The user may either specify an exact, minimal, or maximal width for the cell.

When specifying a minimal width, the corresponding column has a guaranteed minimal width; the actual width may be larger if one of the cells is even wider. When specifying a maximal width for all cells, it is guaranteed that the width of the corresponding column will never exceed the specified value. However, this also means that any contents that don't fit into this width may overlap with neighboring cells. The same problem can occur if you specify an exact width. The possible settings are illustrated in Figures 5.6–5.9.

If your table is too wide or if you manually specified its width, then it may happen that the available horizontal space is not completely used up by the columns. This occurs for instance when starting a new wide table using Insert ▸ Table ▸ Wide block. By default, the unused space is distributed evenly over all columns, but this setting can be overridden by specifying stretch factors using Focus ▸ Cell ▸ Width ▸ Stretch factor or Format ▸ Cell ▸ Width ▸ Stretch. In Figure 5.10, we have shown an example table in which the unused space has been distributed evenly among the two columns. In Figure 5.11, we have shown the same table with a stretch factor of 0 for the first column and 1 for the second one.

The heights of rows are determined following similar principles. In addition, TeXmacs adjusts the vertical limits of cells in order to ensure a uniform layout for ordinary text. Indeed, characters like “a”, “e”, and “x” are shorter than “b”, “d”, and “k”, and not as deep as “p”, “q”, and “g”. If the vertical limits of cells

17	A Fermat prime number
60	Babylonian number base
256	Number of byte values

Figure 5.10. Evenly distribute unused space.

17	A Fermat prime number
60	Babylonian number base
256	Number of byte values

Figure 5.11. Grant all unused space to the last column.

were not adjusted, then the formula

$$\begin{pmatrix} a & x \\ x & a \end{pmatrix} + \begin{pmatrix} f & x \\ x & f \end{pmatrix}$$

would rather be displayed as follows:

$$\left(\begin{smallmatrix} a & x \\ x & a \end{smallmatrix}\right) + \begin{pmatrix} f & x \\ x & f \end{pmatrix}$$

In other words, TeXmacs adjusts cells such that they vertically extend to the highest and deepest character in the current font. This feature can be turned off, or only applied to the top or the bottom, via the menu Focus ▸ Cell ▸ Height ▸ Adjust limits or Format ▸ Cell ▸ Text height correction.

5.3.3 Borders and padding

The user may specify the widths of the four borders of each cell using Format ▸ Cell ▸ Border. The actual width of a border is taken to be the maximum of the specified border widths of adjacent cells. For instance, in the table below, all cells have a top and left border width of `2ln` and a bottom and right border width of `0ln`

top left	top	top right
left	central	right
bottom left	bottom	bottom right

The actual border width between the “top left” cell and the “top” cell is `2ln`, which is indeed the maximum of `0ln` (the right border width of the “top left” cell) and `2ln` (the left border width of the “top” cell). The length unit “`ln`” corresponds to the “standard line width” that is appropriate for the current font. In particular, the width of a fraction bar is `1ln`.

The specification of each of the borders of every individual cell can be tedious. For this reason, a simplified interface has been made available through Focus ▸ Cell ▸ Border and the icon on the focus bar. Using this interface, all four borders of a cell or selection of cells can be changed simultaneously, which includes the corresponding changes in the border widths of the neighboring cells. For example, after positioning the cursor inside the “central” cell of the above table, and clicking on , we obtain

top left	top	top right
left	central	right
bottom left	bottom	bottom right

Similarly, after selecting the four “central”, “right”, “bottom”, and “bottom right” cells, and clicking on , we obtain

top left	top	top right
left	central	right
bottom left	bottom	bottom right

	0 sep	1 sep	1 spc	2 spc
0 sep	Test test	Test test	Test test	Test test
1 sep	Test test	Test test	Test test	Test test
1 spc	Test test	Test test	Test test	Test test
2 spc	Test test	Test test	Test test	Test test

Table 5.5. Typical settings for the horizontal and vertical cell padding.

Notice that the borders that are created in this way have a default width of 1 ln. For other border widths, you may use ▸ Pen width or the more detailed interface from Format ▸ Cell.

Besides the borders, it is also possible to customize the amount of padding between borders of cells and their actual contents. Again, it is possible to separately specify the amount of padding in all four directions via Format ▸ Cell ▸ Padding, Focus ▸ Cell ▸ Padding, or the icon . By default, $\mathrm{T_EX_{MACS}}$ uses a horizontal padding of 1 spc (the default width of a space character for the current font) and a vertical padding of 1 sep (standard separation space for the current font). In Table 5.5, we have illustrated the effect of various amounts of horizontal and vertical padding.

5.3.4 Horizontal and vertical alignment

The horizontal and vertical alignment of cells can be specified via Format ▸ Cell ▸ Alignment or Focus ▸ Cell ▸ Horizontal alignment and Vertical alignment. The current horizontal alignment is indicated on the focus bar by one of the icons , , , or . The cell alignment can also be modified via the corresponding icon menu.

Besides the classic left, center, and right alignments, it is also possible to horizontally align on a decimal dot or comma. This is useful for numerical and monetary tables. Similarly, besides the classic top, center, and bottom alignments, cells can be aligned at their "baselines". This kind of vertical alignment is the default and usually indeed most appropriate for ordinary text. Table 5.6 shows the various available types of cell alignment.

	left	center	right	decimal dot	decimal comma
top	$x^2+y^2=z^2$	$e^{\pi i}=-1$	$H=\frac{1}{\frac{1}{x}+\frac{1}{y}}$	0.14285714	0,14285714
center	$H=\frac{1}{\frac{1}{x}+\frac{1}{y}}$	$x^2+y^2=z^2$	$e^{\pi i}=-1$	14.285714	14,285714
baseline	$e^{\pi i}=-1$	$H=\frac{1}{\frac{1}{x}+\frac{1}{y}}$	$x^2+y^2=z^2$	1428.5714	1428,5714
bottom	$e^{\pi i}=-1$	$H=\frac{1}{\frac{1}{x}+\frac{1}{y}}$	$x^2+y^2=z^2$	1428.5714	1428,5714

Table 5.6. Demonstration of the various kinds of cell alignment.

It is easy to modify the alignment of cells using the keyboard. The shortcut ⌘→ moves the horizontal alignment of a cell further to the right: a cell that was aligned at the left will be centered, and a cell that was centered will become right aligned. Similarly, the keystroke ⌘← moves the horizontal alignment further to the left, whereas ⌘↑ and ⌘↓ allow you to adjust the vertical alignment. When applied to a selection of cells, we notice that the selection remains active, so that multiple changes can be made efficiently. If you need to modify the horizontal alignment of many columns in multiple tables, then the keyboard shortcuts ⌘← and ⌘→ are also very efficient in combination with the "column operation mode" (Focus ▸ Cell ▸ Operation mode ▸ Columns).

5.3.5 Line-wrapping and block content

We already discussed the difference between wide tables and smaller inline tables such as matrices inside mathematical formulas. Wide tables allow cells to contain large portions of text. If the contents of a cell do not fit on a single line, then line-wrapping is activated by default in order to prevent the table from running out of the page. To this effect, we recall the existence of the toggle on the focus bar. Notice that line-wrapping only makes sense inside cells or tables for which a maximal width has been specified (see sections 5.3.2 and 5.4.2); otherwise, the cell will simply become as large as necessary for its contents to fit on a single line.

Line-wrapping can also be activated or deactivated using Format ▸ Cell ▸ Line wrapping or Focus ▸ Cell ▸ Line wrapping. These menus additionally allow you to specify the way in which line-wrapped cells should be aligned with respect to neighboring cells. For example, consider the following list of equations, created using Insert ▸ Mathematics ▸ Several equations:

$$\begin{array}{lcl}
\begin{array}[b]{l} a+b+c+d+e+f+g+h+i+j+k+\\ l+m+n+o+p+q+r+s+t+u+v+\\ w+x+y+z \end{array} & = & \begin{array}[t]{l} z+y+x+w+v+u+t+s+r+q+p+\\ o+n+m+l+k+j+i+h+g+f+e+\\ d+c+b+a \end{array}\\
 & = & \begin{array}[t]{l} a+z+b+y+c+x+d+w+e+v+f+\\ u+g+t+h+s+i+r+j+q+k+p+\\ l+o+m+n. \end{array}
\end{array}$$

Both the first and the last column of this table are line-wrapped. However, the first column is aligned at the bottom whereas the last column is aligned at the top.

By default, cells in which line-wrapping is activated allow for "block content". Such cells may for instance contain several paragraphs, theorems, enumerations, or entire computer algebra sessions. In certain situations you may wish to allow for block content inside cells that are not line-wrapped. Inversely, you may want to disallow block content in cells where line wrapping is active. This can be specified via the menus Format ▸ Cell ▸ Block content or Focus ▸ Cell ▸ Block content.

5.4 Table properties

The global properties of a table can be edited after opening the "Table properties" editor via Format ▸ Table. Alternatively, you may use the submenus under Focus ▸ Table or the icon on the focus bar.

5.4.1 Number of rows and columns

We have already discussed the insertion and removal of rows and columns using the keyboard shortcuts ⌥←, ⌥→, ⌥↑, ⌥↓, or the icons , , , , , , , and on the focus bar. It is also possible to specify an exact number of rows or columns using Focus ▸ Table ▸ Size ▸ Set number of rows and Set number of columns. Moreover, the Focus ▸ Table ▸ Size menu allows you to specify a minimum or maximum number of rows or columns in a table. This is especially useful for the design of macros for customized tables (see section 5.7). For instance, the macro eqnarray* for aligned lists of equations

$$\begin{aligned} \mu &= \frac{a_1 + \cdots + a_n}{n} \\ \sigma &= \sqrt{\frac{a_1^2 + \cdots + a_n^2}{n}} \end{aligned}$$

specifies that "tables" of this kind should contain at least and at most three columns. Newly created lists of equations therefore contain three columns instead of one, but no new columns can be inserted using ⌥← or ⌥→. Similarly, the keys ⌫ and ⌦ suppress the table as soon as its cells are empty and there are only one row and three columns left.

5.4.2 Width and height

The width and the height of a table can be specified using Format ▸ Table ▸ Width and Height or Focus ▸ Table ▸ Width and Height, much in the same way as for cells (see section 5.3.2).

Two settings for the global table size are most common: wide tables on separate lines usually extend over the entire width of a paragraph, whereas the sizes of smaller tables such as matrices are typically determined automatically so as to make the contents of all cells fit. Toggling between these two possibilities is so common that the focus bar contains a special icon for this purpose. You may also use Focus ▸ Table ▸ Width ▸ Paragraph or specify `1par` as Format ▸ Table ▸ Width or Focus ▸ Table ▸ Width.

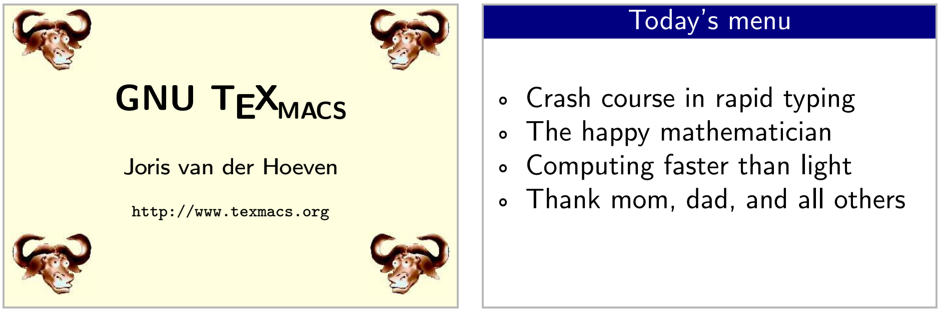


Figure 5.12. Example of two miniscreens for a presentation.

Of course, other table sizes can sometimes be useful. For instance, one may use a fixed table size for mimicking a minipage or a miniscreen, as in Figure 5.12.

5.4.3 Borders and padding

You may specify the width of the left, right, top, and bottom borders of a table using Format ▸ Table ▸ Border or Focus ▸ Table ▸ Border. This feature has been provided for consistency, but is really redundant with the possibility to specify the borders of selections of cells (see section 5.3.3): it suffices to select all cells. On the other hand, padding is applied around the table, instead of inside the cells. You may change the padding at the left, right, top, and bottom of the table using Format ▸ Table ▸ Padding or Focus ▸ Table ▸ Padding.

5.4.4 Alignment with respect to the surrounding text

The global alignment properties of a table specify how to align it with respect to the surrounding text. These properties can be modified using Format ▸ Table ▸ Alignment or Focus ▸ Table ▸ Horizontal alignment and Vertical alignment. Customized alignments are mainly needed for "inline" tables, which are embedded into a formula or paragraph with other text.

By default, tables are vertically aligned at the *axis*. The axis is located halfway between the top and the bottom of the character "x". Tables that are centered along the axis roughly descend as much below the bottom of the text as they exceed the top. When selecting Top, Middle or Bottom for the vertical alignment, the top, center, or top of the table will instead be aligned with the *baseline* of the surrounding text: see Figure 5.13.

Vertical alignment at the $\boxed{\begin{array}{c}\text{axis}\\\text{axis}\end{array}}$ or $\boxed{\begin{array}{c}\text{top}\\\text{top}\end{array}}$ or $\boxed{\begin{array}{c}\text{middle}\\\text{middle}\end{array}}$ or $\boxed{\begin{array}{c}\text{bottom}\\\text{bottom}\end{array}}$.

Figure 5.13. Basic types of vertical table alignment with respect to surrounding text.

Baseline alignment at the	top	or	middle	or	bottom	or	second row	.
	top		middle		bottom		second row	
	top		middle		bottom		second row	
	top		middle		bottom		second row	

Figure 5.14. Various kinds of vertical baseline alignment of tables.

It is also possible to align the baseline of a particular row in the table with the baseline of the surrounding text, by selecting Top baseline, Middle baseline, Bottom baseline or Specific baseline in Focus ▸ Table ▸ Vertical alignment. This kind of alignment is illustrated in Figure 5.14. When the number of rows of a table is even, the middle baseline is located exactly at the middle between the baselines of the two central rows. For the rightmost table in Figure 5.14, we chose "2" for Focus ▸ Table ▸ Vertical alignment ▸ Specific baseline.

The default horizontal alignment of tables is at the left. Other alignments are supported for completeness, but rarely needed, since they usually result in overlapping text: see Figure 5.15.

5.4.5 Breaking up large tables

Unfortunately, $\text{T}_{\text{E}}\text{X}_{\text{MACS}}$ is currently not very efficient for editing large tables. In particular, it is recommended to manually break up tables that do not fit on a single page. Smaller tables sometimes do not nicely fit on a page either. This typically happens for long lists of equations such as

$$\begin{aligned}
x_1 &= \Phi_1(x_1, x_2, \ldots, x_{n-1}, x_n) \\
x_2 &= \Phi_2(x_1, x_2, \ldots, x_{n-1}, x_n) \\
&\vdots \\
x_{n-1} &= \Phi_{n-1}(x_1, x_2, \ldots, x_{n-1}, x_n) \\
x_n &= \Phi_n(x_1, x_2, \ldots, x_{n-1}, x_n).
\end{aligned}$$

If you want to allow for page breaks inside such tables, then you should use Format ▸ Table ▸ Large tables ▸ Enable page breaking or Focus ▸ Table ▸ Special ▸ Table breaking. Notice that you should set all interline and interparagraph spacing to zero (see section 3.9) whenever the cells of such a table admit borders.

Horizontal alignment at the	left	or at the	middle	or, even, at the	right	.
	left		middle		right	

Figure 5.15. Horizontal table alignment with respect to surrounding text.

		2011		2012		Total	
		jan–jun	jul–dec	jan–jun	jul–dec		
Beverages	Milk	120.10	116.20	128	126.15	490.45	850.65
	Cola	85.40	105.30	74.15	95.35	360.20	
Fruit	Apples	65.10	72.85	82.90	83.10	303.95	728.40
	Cherries	34.20	55.45	23.85	57.55	171.05	
	Bananas	83	47.20	72	51.20	253.40	
Total		387.80	397	380.90	413.35	1579.05	

Table 5.7. Spending table that illustrates how to use joined cells.

5.5 Joined cells and subtables

It is sometimes useful to merge several cells into a single cell. After making a selection of cells, this can be done by clicking on the icon or Focus ▸ Cell ▸ Join selected cells. Cells that have been joined together can be dissociated using or Focus ▸ Cell ▸ Dissociate joined cells. Table 5.7 gives a typical example of a table with a few joined cells. Cell-joining can also be an efficient way to create certain diagrams: see Figure 5.16. In this example the widths and heights of all cells were set to `0.25in`.

Remark 5.2. When joining selected cells $C_{11}, \ldots, C_{kl}$, the top left cell C_{11} will assume the role of the "leader" and contains the content of the joined cell. The other cells $C_{12}, \ldots, C_{kl}$ remain part of the table, but become invisible to the user until the joined cell is dissociated.

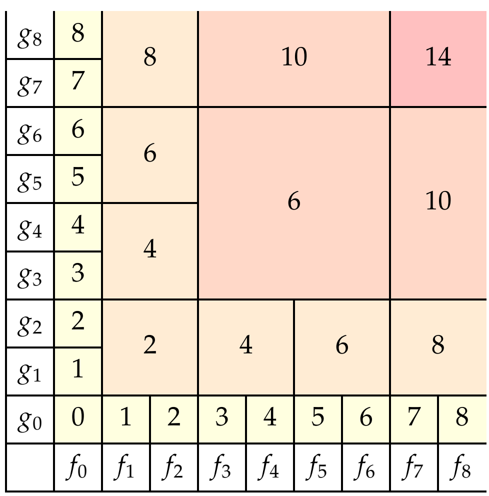


Figure 5.16. Fast relaxed multiplication of formal power series.