(La)TeX Documentation

Math mode - v.1.95

Herbert Voß[*]

24th April 2005

Abstract

More than once people say that TeX was designed for mathematical or technical purpose. This maybe true when we remember the reasons why Donald Knuth created TeX. But nowadays there are a lot of examples where TeX was used for publications without any mathematical or technical background. Nevertheless, we have to consider, that writing publications with a lot of mathematical material is one of the important advantages of TeX and it seems that is impossible to know all existing macros and options of (La)TeX and the several additional packages, especially AMSmath. This is the reason why I tried to collect all important facts in this paper.

Please report typos or any other comments to this documentation to voss@perce.de. This document was written with the LaTeX editor `Kile 1.7b2 (Qt 3.2 KDE 3.2)` http://sourceforge.net/projects/kile/ and the PDF output was built with the Linux version of `VTeX/Free`, Version 8.46 (http://www.micropress-inc.com/linux/)

[*] Thanks for the feedback to: Alexander Boronka; Christian Faulhammer; José Luis Gómez Dans; Azzam Hassam; Martin Hensel; Morten Høgholm; Dan Lasley; Angus Leeming; Tim Love; Hendrik Maryns; Heinz Mezera; David Neuway; Joachim Punter; Carl Riehm; Will Robertson; Christoph Rumsmüller; José Carlos Santos; Uwe Siart; Heiko Stamer; Uwe Stöhr; David Weenink; Michael Zedler; and last but not least a special thanks to Monika Hattenbach for her excellent job of proofreading.

Contents

Page

Part I
Standard LaTeX math mode

1 Introduction

The following sections describe all the math commands which are available without any additional package. Most of them also work with special packages and some of them are redefined. At first some important facts for typesetting math expressions.

2 The Inlinemode

As the name says this are always math expressions which are in a standard textline, like this one: $f(x)=\int_a^b\frac{\sin x}{x}dx$. There are no limitations for the height of the math expressions, so that the layout may be very lousy if you insert a big matrix in an inline mode like this: $\underline{A}=\left[\begin{array}{ccc}a&b&c\\d&e&f\\g&h&i\end{array}\right]$. In this case it is better to use the `\smallmatrix` environment $\underline{A}=\left[\begin{smallmatrix}a&b&c\\d&e&f\\g&h&i\end{smallmatrix}\right]$ from the AMSmath package (see section 26.6 on page 61) or the displaymath mode (section 3 on page 13).

This inline mode is possible with three different commands:

$\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$

$\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$

$\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$

```
\(\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)\)\\[10pt]
$\sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)$\\[10pt]
\begin{math}
  \sum_{i=1}^{n}i=\frac{1}{2}n\cdot(n+1)
\end{math}
```

1. `\( ... \)` , the problem is that `\(` is not a robust macro (see section 2.3 on the next page). `\(...\)`
2. `\small $ ... $` `$...$`
3. `\begin{math}` ... `\end{math}`, also not robust `\begin{math}` ... `\end{math}`

In general `$...$` is the best choice, but this does not work in environments like `verbatim` or `alltt`. In this case `\(...\)` works.

2.1 Limits

In the inline mode the limits are by default only in super or subscript mode and the fractions are always in the scriptstyle[1] font size. For example:

$\int_1^\infty \frac{1}{x^2}dx = 1$, which is not too big for the textline. You can change this with the command `\limits`, which must follows a mathoperator[2] like an integral (`\int`), a sum (`\sum`), a product (`\prod`) or a limes (`\lim`). But this $\int\limits_1^\infty \frac{1}{x^2}dx = 1$ looks not very nice in a text line when it appears between two lines, especially when there are multiline limits.[3]

`\limits`
`\int`
`\lim`
`\prod`
`\sum`

2.2 Fraction command

For inlined formulas the fractions are by default in the scriptstyle (see tabular 8 on page 38), which is good for the typesetting $y = \frac{a}{b+1}$, because the linespacing is nearly the same, but not optimal, when the formula shows some important facts. There are two solutions to get a better reading:

`\frac`

1. choose the display mode instead of the inline mode, which is the better one;
2. set the fontstyle to `displaystyle`, which makes the fraction $y = \dfrac{a}{b+1}$ more readable but the linespacing increases which is always a bad solution and should only be used when the first solution makes no sense.[4]

$y = \frac{a}{b+1} = {\displaystyle\frac{a}{b+1}}$

```
$y=\frac{a}{b+1}={\displaystyle\frac{a}{b+1}}$
```

2.3 Math in `\part`, `\chapter`, `\section`, ... titles like $f(x) = \prod_{i=1}^n \left(i - \frac{1}{2i}\right)$

All commands which appear in positions like contents, index, header, ... must be robust[5] which is the case for `$...$` but not for `\(\...\)`. If you do not have any contents, index, a.s.o. you can write the mathstuff in `\chapter`, `\section`, a.s.o without any restriction. Otherwise use `\protect\(` and `\protect\)` or the `$...$` version.

The whole math expression appears in the default font shape and not in bold like the other text. Section 22.1 on page 47 describes how the math expressions can be printed also in bold.

`\texorpdfstring`

There are problems with `hyperref` when there is a non text part in a title. It is possible to tell `hyperref` to use different commands, one for the title and another one for the bookmarks:

[1] See section 12 on page 37.

[2] To define a new operator see section 71

[3] For more information about limits see section 6.1 on page 24 or section 35 on page 68.

[4] For an abbreviation see section 29 on page 63, there is a special `\dfrac` macro.

[5] `robust` means that the macro is not expanded before it is moved into for example the tableofcontents file (`*.toc`). No robustness is often a problem, when a macro is part of another macro.

`\texorpdfstring{<TeX part>}{<hyperref part>}`

E.g.

```
\texorpdfstring{$\int f(x)\,dx$}{Integral function}+.
```

2.4 Equation numbering

It is obvious that the numbering of inline mathstuff makes no sense!

2.5 Framed math

With the `\fbox` macro everything of inline math can be framed, like the following one:

$\boxed{f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)}$

```
\fbox{$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)$}
```

Parameters are the width of `\fboxsep` and `\fboxrule`, the predefined values from `latex.ltx` are:

```
\fboxsep = 3pt
\fboxrule = .4pt
```

The same is possible with the `\colorbox` $f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)$ from the color package.

```
\colorbox{yellow}{$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\
  right)$}
```

2.6 Linebreak

LaTeX can break an inline formula only when a relation symbol $(=,<,>,\dots)$ or a binary operation symbol $(+,-,\dots)$ exists and at least one of these symbols appears at the outer level of a formula. Thus `$a+b+c$` can be broken across lines, but `${a+b+c}$` not.

- The default: $f(x)=a_nx^n+a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\ldots+a_ix^i+a_2x^2+a_1x^1+a_0$
- The same inside a group {...}: $f(x)=a_nx^n+a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\ldots+a_ix^i+a_2x^2+a_1x^1$
- Without any symbol: $f(x)=a_n\left(a_{n-1}\left(a_{n-2}\left(\ldots\right)\ldots\right)\ldots\right)$

If it is not possible to have any mathsymbol, then split the inline formula in two or more pieces (`$...$ $...$`). If you do not want a linebreak for the whole document, you can set in the preamble:

```
\relpenalty=9999
\binoppenalty=9999
```

which is the extreme case of grudgingly allowing breaks in extreme cases.

2.7 Whitespace

LaTeX defines the length `\mathsurround` with the default value of `0pt`. This length is added before and after an inlined math expression (see table 1).

foo $\boxed{f(x)=\int_1^{\infty}\frac{1}{x^2}dx=1}$ bar	`foo \fbox{$ f(x)=\int_1^{\infty}\frac{1}{x^2}dx=1 $} bar`
foo █ $\boxed{f(x)=\int_1^{\infty}\frac{1}{x^2}dx=1}$ █ bar	`foo \rule{20pt}{\ht\strutbox}\fbox{$ f(x)=\int_1^{\infty}\frac{1}{x^2}dx=1 $}\rule{20pt}{\ht\strutbox} bar`
foo $\boxed{\hspace{20pt}f(x)=\int_1^{\infty}\frac{1}{x^2}dx=1\hspace{20pt}}$ bar	`\setlength{\mathsurround}{20pt}foo \fbox{$ f(x)=\int_1^{\infty}\frac{1}{x^2}dx=1 $} bar`

Table 1: Meaning of `\mathsurround`

2.8 $\mathcal{A}_{\mathcal{M}}\mathcal{S}$math for the inline mode

None of the $\mathcal{A}_{\mathcal{M}}\mathcal{S}$math-functions are available in inline mode.

3 Displaymath mode

This means, that every formula gets its own paragraph (line). There are some differences in the layout to the one from the title of 2.3.

3.1 `equation` environment

For example:

$$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right) \qquad (1)$$

```
\begin{equation}
  f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)
\end{equation}
```

The delimiters `\begin{equation} ... \end{equation}` are the only difference to the inline version. There are some equivalent commands for the display-math mode:

1. `\begin{displaymath}`. . . `\end{displaymath}`, same as `\[ . . . \]` `\begin{displaymath}` . . . `\end{displaymath}`
2. `\[...\]`. (see above) the short form of a displayed formula, no number `\[...\]`

$$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right)$$

displayed, no number. Same as 1.

3. `\begin{equation}...\end{equation}`

\begin{equation}
...
\end{equation}

$$f(x)=\prod_{i=1}^{n}\left(i-\frac{1}{2i}\right) \tag{2}$$

displayed, a sequential equation number, which may be reset when starting a new chapter or section.

(a) There is only **one** equation number for the whole environment.

\nonumber

(b) There exists no star-version of the equation environment because `\[...\]` is the equivalent. With the tag `\nonumber` it is possible to suppress the equation number:

$$f(x)=[...]$$

```
\begin{equation}
  f(x)= [...] \nonumber
\end{equation}
```

3.2 eqnarray environment

This is by default an array with three columns and as many rows as you like. It is nearly the same as an array with a `rcl` column definition.

\begin{eqnarray}
...
\end{eqnarray}

It is **not possible** to change the internal behaviour of the `eqnarray` environment without rewriting the environment. It is always an implicit array with **three** columns and the horizontal alignment `right-center-left (rcl)` and small **symbol** sizes for the middle column. All this can not be changed by the user without rewriting the whole environment in `latex.ltx`.

$$\begin{array}{rcl}
\mathrm{left} & \mathrm{middle} & \mathrm{right}\\
\frac{1}{\sqrt{n}}= & \tfrac{\sqrt{n}}{n}= & \frac{n}{n\sqrt{n}}
\end{array}$$

```
\begin{eqnarray*}
\mathrm{left} & \mathrm{middle} & \mathrm{right}\\
\frac{1}{\sqrt{n}}= & \frac{\sqrt{n}}{n}= & \frac{n}{n\sqrt{n}}
\end{eqnarray*}
```

The `eqnarray` environment should not be used as an array. As seen in the above example the typesetting is wrong for the middle column. The numbering of eqnarray environments is always for every row, means, that four lines get four different equation numbers (for the labels see section 3.4):

$$\begin{array}{rcll}
y & = & d & (3)\\
y & = & cx+d & (4)\\
y & = & bx^2+cx+d & (5)\\
y & = & ax^3+bx^2+cx+d & (6)
\end{array}$$

```
\begin{eqnarray}
y & = & d\label{eq:2}\\
y & = & cx+d\\
y & = & bx^{2}+cx+d\\
y & = & ax^{3}+bx^{2}+cx+d\label{eq:5}
\end{eqnarray}
```

Toggling numbering off/on for **all** rows is possible with the starred version of eqnarray.

$$\begin{eqnarray*}
y & = & d\\
y & = & cx+d\\
y & = & bx^{2}+cx+d\\
y & = & ax^{3}+bx^{2}+cx+d
\end{eqnarray*}$$

```
\begin{eqnarray*}
y & = & d\label{eq:3}\\
y & = & cx+d\\
y & = & bx^{2}+cx+d\\
y & = & ax^{3}+bx^{2}+cx+d\label{eq:4}
\end{eqnarray*}
```

Toggling off/on for **single** rows is possible with the above mentioned `\nonumber` tag at the end of a row (before the newline command). For example:

$$\begin{eqnarray}
y & = & d\nonumber\\
y & = & cx+d\nonumber\\
y & = & bx^{2}+cx+d\nonumber\\
y & = & ax^{3}+bx^{2}+cx+d \tag{7}
\end{eqnarray}$$

```
\begin{eqnarray}
 y & = & d\nonumber \\
 y & = & cx+d\nonumber \\
 y & = & bx^{2}+cx+d\nonumber \\
 y & = & ax^{3}+bx^{2}+cx+d
\end{eqnarray}
```

3.2.1 Short commands

It is possible to define short commands for the `eqnarray` environment

```
\makeatletter
\newcommand{\be}{%
  \begingroup
%  \setlength{\arraycolsep}{2pt}
  \eqnarray%
  \@ifstar{\nonumber}{}%
}
\newcommand{\ee}{\endeqnarray\endgroup}
\makeatother
```

Now you can write the whole equation as

$$f(x) = \int\frac{\sin x}{x}dx \tag{8}$$

```
\be
  f(x) &=& \int\frac{\sin x}{x}dx
\ee
```

or, if you do not want to have a numbered equation as

$$f(x) = \int\frac{\sin x}{x}dx$$

```
\be*
  f(x) &=& \int\frac{\sin x}{x}dx
\ee
```

3.3 Equation numbering

`\nonumber`

For all equations which can have one or more equation numbers (for every line/row) the numbering for the whole equation can be disabled with switching from the unstarred to the star version. This is still for the whole formula and doesn't work for single rows. In this case use the `\nonumber` tag.

- This doc is written with the article-class, which counts the equations continuously over all parts/sections. You can change this behaviour in different ways (see the following subsections).
- In standard LaTeX it is a problem with too long equations and the equation number, which may be printed with the equation one upon the other. In this case use the $\mathcal{AMS}$math package, where the number is set above or below of a too long equation (see equation 28 on page 28).
- For counting subequations see section 33.1 on page 67.

3.3.1 Changing the style

`\theequation`

With the beginning of Section 25.2 on page 52 the counting changes from "47" into the new style "II-54". The command sequence is

```
\renewcommand\{\theequation}{%
  \thepart-\arabic{equation}%
}
```

See section 33 on page 66 for the $\mathcal{AMS}$math command.

3.3.2 Resetting a counter style

Removing a given reset is possible with the package **`remreset`**.[6] Write into the preamble

`\@removefromreset`

```
\makeatletter
\@removefromreset{equation}{section}
\makeatother
```

or anywhere in the text.

Now the equation counter is no longer reset when a new section starts. You can see this after section 26.3 on page 58.

3.3.3 Equation numbers on the left side

Choose package **`leqno`**[7] or have a look at your document class, if such an option exists.

[6] CTAN://macros/latex/contrib/supported/carlisle/remreset.sty

[7] CTAN://macros/latex/unpacked/leqno.sty

3.3.4 Changing the equation number style

The number style can be changed with a redefinition of

```
\def\@eqnnum{{\normalfont \normalcolor (\theequation)}}
```

For example: if you want the numbers not in parentheses write

```
\makeatletter
\def\@eqnnum{{\normalfont \normalcolor \theequation}}
\makeatother
```

For $\mathcal{A}\mathcal{M}\mathcal{S}$math there is another macro, see section 33 on page 66.

3.3.5 More than one equation counter

You can have more than the default equation counter. With the following code you can easily toggle between roman and arabic equation counting.

```
%code by Heiko Oberdiek
\makeatletter
%Roman counter
\newcounter{roem}
\renewcommand{\theroem}{\roman{roem}}

% save the original counter
\newcommand{\c@org@eq}{}
\let\c@org@eq\c@equation
\newcommand{\org@theeq}{}
\let\org@theeq\theequation

%\setroem sets roman counting
\newcommand{\setroem}{
  \let\c@equation\c@roem
  \let\theequation\theroem}

%\setarab the arabic counting
\newcommand{\setarab}{
  \let\c@equation\c@org@eq
  \let\theequation\org@theeq}
\makeatother
```

The following examples show how it works:

$$
\begin{align}
f(x) &= \int \sin x dx \tag{9}\\
g(x) &= \int \frac{1}{x} dx \tag{10}
\end{align}
$$

$$
\begin{align}
F(x) &= -\cos x \tag{i}\\
G(x) &= \ln x \tag{ii}
\end{align}
$$

$$
\begin{align}
f'(x) &= \sin x \tag{11}\\
g'(x) &= \frac{1}{x} \tag{12}
\end{align}
$$

```
\begin{align}
f(x) &= \int\sin x dx\label{eq:arab1}\\
g(x) &= \int\frac{1}{x}dx
\end{align}
%
\setroem
%
\begin{align}
F(x) &=-\cos x\\
G(x) &=\ln x\label{eq:rom1}
\end{align}
%
\setarab
%
\begin{align}
f^{\prime} (x) &= \sin x\\
g^{\prime} (x) &= \frac{1}{x}\label{eq:arab2}
\end{align}
```

There can be references to these equations in the usual way, like eq.9, 12 and for the roman one eq.ii.

3.4 Labels

Every numbered equation can have a label to which a reference is possible.

- There is one restriction for the label names, they cannot include one of LaTeX's command characters.[8]
- The label names are replaced by the equation number.

\tag

If you do not want a reference to the equation number but to an self defined name then use the AMSmath command `\tag...`, which is described in section 34 on page 68.

3.5 Frames

Similiar to the inline mode, displayed equations can also be framed with the `\fbox` command, like equation 13. The only difference is the fact, that the equation must be packed into a parbox or minipage. It is nearly the same for a colored box, where the `\fbox{...}` has to be replaced with `\colorbox{yellow}{...}`. The package `color.sty` must be loaded and – important– the `calc.sty` package to get a correct boxwidth.

$$\boxed{f(x) = \int_1^{\inf} \frac{1}{x^2} dx = 1 \qquad (13)}$$

[8] \$ _ ^ \ & % { }

```
\noindent\fbox{\parbox{\linewidth-2\fboxsep-2\fboxrule}{%
\begin{equation}\label{eq:frame0}
 f(x)=\int_1^{\inf}\dfrac{1}{x^2}dx=1
\end{equation}%
}}
```

If the equation number should not be part of the frame, then it is a bit complicated. There is one tricky solution, which puts an unnumbered equation just beside an empty numbered equation. The `\hfill` is only useful for placing the equation number right aligned, which is not the default. The following four equations 14-17 are the same, only the second one written with the `\myMathBox` macro which has the border and background color as optional arguments with the defaults `white` for background and `black` for the frame. If there is only one optional argument, then it is still the one for the frame color (15).

```
\makeatletter
\def\myMathBox{\@ifnextchar[{\my@MBoxi}{\my@MBoxi[black]}}
\def\my@MBoxi[#1]{\@ifnextchar[{\my@MBoxii[#1]}{\my@MBoxii[#1][white]}}
\def\my@MBoxii[#1][#2]#3#4{%
  \par\noindent%
  \fcolorbox{#1}{#2}{%
     \parbox{\linewidth-\labelwidth-2\fboxrule-2\fboxsep}{#3}%
  }%
  \parbox{\labelwidth}{%
    \begin{eqnarray}\label{#4}\end{eqnarray}%
  }%
  \par%
}
\makeatother
```

$$f(x) = x^2 + x \tag{14}$$

$$\boxed{f(x) = x^2 + x} \tag{15}$$

$$\boxed{f(x) = x^2 + x} \tag{16}$$

$$\boxed{f(x) = x^2 + x} \tag{17}$$

```
\begin{equation}\label{eq:frame2}
 f(x)=x^2 +x
\end{equation}
\myMathBox[red]{\[f(x)=x^2 +x\]}{eq:frame3}
```

```
\myMathBox[red][yellow]{\[f(x)=x^2 +x\]}{eq:frame4}
\myMathBox{\[f(x)=x^2 +x\]}{eq:frame5}
```

If you are using the $\mathcal{AMS}$math package, then try the solutions from section 39 on page 75.

4 array environment

\begin{array}
...
\end{array}

This is simply the same as the eqnarray environment only with the possibility of variable rows **and** columns and the fact, that the whole formula has only **one** equation number and that the `array` environment can only be part of another math environment, like `equation` or `displaymath`.

$$\left.\begin{array}{r@{\quad}ccrr}
\textrm{a)} & y & = & c & (constant)\\
\textrm{b)} & y & = & cx+d & (linear)\\
\textrm{c)} & y & = & bx^{2}+cx+d & (square)\\
\textrm{d)} & y & = & ax^{3}+bx^{2}+cx+d & (cubic)
\end{array}\right\} \textrm{Polynomes} \tag{18}$$

```
\begin{equation}
\left.%
\begin{array}{r@{\\
\quad}ccrr}
 \textrm{a}) & y & = & c & (constant)\\
 \textrm{b}) & y & = & cx+d & (linear)\\
 \textrm{c}) & y & = & bx^{2}+cx+d & (square)\\
 \textrm{d}) & y & = & ax^{3}+bx^{2}+cx+d & (cubic)
\end{array}%
\right\}  \textrm{Polynomes}
\end{equation}
```

The horizontal alignment of the columns is the same than the one from the `tabular` environment.

For arrays with delimiters see section 54 on page 95.

4.1 Cases structure

If you do not want to use the $\mathcal{AMS}$math package then write your own cases structure with the `array` environment:

```
\begin{equation}
x=\left\{ \begin{array}{cl}
0 & \textrm{if A=...}\\
1 & \textrm{if B=...}\\
x & \textrm{this runs with as much text as you like, but without an raggeright text.}\end{array}\right.
\end{equation}
```

$$x=\left\{\begin{array}{cl}0 & \text{if A=...}\\ 1 & \text{if B=...}\\ x & \text{this runs with as much text as you like, but without an raggeright text.}\end{array}\right. \tag{19}$$

It is obvious, that we need a `\parbox` if the text is longer than the possible linewidth.

```
\begin{equation}
x = \left\{%
  \begin{array}{l>{\raggedright}p{.5\textwidth}}%
     0 & if A=...\tabularnewline
     1 & if B=...\tabularnewline
     x & \parbox{0.5\columnwidth}{this runs with as much text as you like, %
         because an automatic linebreak is given with %
         an raggedright text. Without this %
         \raggedright command, you'll get a formatted %
         text, like the following one ... but with a parbox ... it works}
   \end{array}%
  \right. %
\end{equation}
```

$$x=\left\{\begin{array}{ll}0 & \text{if A=...}\\ 1 & \text{if B=...}\\ x & \parbox{0.5\columnwidth}{this runs with as much text as you like, because an automatic linebreak is given with an raggedright text. Without this command, you'll get a formatted text, like the following one ... but with a parbox ... it works}\end{array}\right. \tag{20}$$

4.2 arraycolsep

\arraycolsep

All the foregoing math environments use the array to typeset the math expression. The predefined separation between two columns is the length `\arraycolsep`, which is set by nearly all document classes to `5pt`, which seems to be too big. The following equation is typeset with the default value and the second one with `\arraycolsep=1.4pt`

$$\boxed{f(x)} \quad \boxed{=} \quad \boxed{\int \frac{\sin x}{x} dx}$$

$$\boxed{f(x)}\boxed{=}\boxed{\int \frac{\sin x}{x} dx}$$

If this modification should be valid for all arrays/equations, then write it into the preamble, otherwise put it into a group or define your own environment as done in section 3.2.1 on page 15.

```
\bgroup
\arraycolsep=1.4pt
\begin{eqnarray}
f(x) & = & \int\frac{\sin x}{x}dx
\end{eqnarray}
\egroup
```

```
\makeatletter
\newcommand{\be}{%
    \begingroup
    \setlength{\arraycolsep}{1.4pt}
[ ... ]
```

5 Matrix

\matrix
\bordermatrix

TEX knows two macros and LATEX one more for typesetting a matrix:

$$\begin{matrix} A & B & C \\ d & e & f \\ 1 & 2 & 3 \end{matrix}$$

```
$\begin{matrix}
  A & B & C \\
  d & e & f \\
  1 & 2 & 3 \\
\end{matrix}$
```

$$\begin{array}{c c} & \begin{matrix} 0 & 1 & 2 \end{matrix} \\ \begin{matrix} 0 \\ 1 \\ 2 \end{matrix} & \begin{pmatrix} A & B & C \\ d & e & f \\ 1 & 2 & 3 \end{pmatrix} \end{array}$$

```
$\bordermatrix{%
   & 0 & 1 & 2 \cr
 0 & A & B & C \cr
 1 & d & e & f \cr
 2 & 1 & 2 & 3 \cr
}$
```

The first two macros are listed here for some historical reason, because the **array** or especially the AMSmath package offer the same or better macros/environments. Nevertheless it is possible to redefine the **bordermatrix** macro to get other parentheses and a star version which takes the left top part as matrix:

$$\begin{array}{cc}
 & \begin{array}{cc} 1 & 2 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{pmatrix} x1 & x2 \\ x3 & x4 \\ x5 & x6 \end{pmatrix}
\end{array}$$

```
$\bordermatrix{%
  &  1 &  2 \cr
1 & x1 & x2 \cr
2 & x3 & x4 \cr
3 & x5 & x6
}$
```

$$\begin{array}{cc}
 & \begin{array}{cc} 1 & 2 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} x1 & x2 \\ x3 & x4 \\ x5 & x6 \end{bmatrix}
\end{array}$$

```
$\bordermatrix[{[]}]{%
  &  1 &  2 \cr
1 & x1 & x2 \cr
2 & x3 & x4 \cr
3 & x5 & x6
}$
```

$$\begin{array}{cc}
 & \begin{array}{cc} 1 & 2 \end{array} \\
\begin{array}{c} 1 \\ 2 \\ 3 \end{array} & \begin{Bmatrix} x1 & x2 \\ x3 & x4 \\ x5 & x6 \end{Bmatrix}
\end{array}$$

```
$\bordermatrix[\{\}]{%
  &  1 &  2 \cr
1 & x1 & x2 \cr
2 & x3 & x4 \cr
3 & x5 & x6
}$
```

$$\begin{array}{cc}
\begin{pmatrix} x1 & x2 \\ x3 & x4 \\ x5 & x6 \end{pmatrix} & \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \\
\begin{array}{cc} 1 & 2 \end{array} &
\end{array}$$

```
$\bordermatrix*{%
 x1 & x2 & 1 \cr
 x3 & x4 & 2 \cr
 x5 & x6 & 3 \cr
  1 &  2
}$
```

$$\begin{array}{cc}
\begin{bmatrix} x1 & x2 \\ x3 & x4 \\ x5 & x6 \end{bmatrix} & \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \\
\begin{array}{cc} 1 & 2 \end{array} &
\end{array}$$

```
$\bordermatrix*[{[]}]{%
 x1 & x2 & 1 \cr
 x3 & x4 & 2 \cr
 x5 & x6 & 3 \cr
  1 &  2
}$
```

$$\begin{array}{cc}
\begin{Bmatrix} x1 & x2 \\ x3 & x4 \\ x5 & x6 \end{Bmatrix} & \begin{array}{c} 1 \\ 2 \\ 3 \end{array} \\
\begin{array}{cc} 1 & 2 \end{array} &
\end{array}$$

```
$\bordermatrix*[\{\}]{%
 x1 & x2 & 1 \cr
 x3 & x4 & 2 \cr
 x5 & x6 & 3 \cr
  1 &  2
}$
```

There is now an optional argument for the parenthesis with `()` as the default one. To get such a behaviour, write into the preamble:

```
\makeatletter
\newif\if@borderstar
\def\bordermatrix{\@ifnextchar*{%
 \@borderstartrue\@bordermatrix@i}{\@borderstarfalse\@bordermatrix@i*}%
}
\def\@bordermatrix@i*{\@ifnextchar[{\@bordermatrix@ii}{\
  @bordermatrix@ii[()]}}
```

```
\def\@bordermatrix@ii[#1]#2{%
\begingroup
  \m@th\@tempdima8.75\p@\setbox\z@\vbox{%
    \def\cr{\crcr\noalign{\kern 2\p@\global\let\cr\endline }}%
    \ialign {$###$\hfil\kern 2\p@\kern\@tempdima & \thinspace %
    \hfil $###$\hfil && \quad\hfil $###$\hfil\crcr\omit\strut %
    \hfil\crcr\noalign{\kern -\baselineskip}#2\crcr\omit %
    \strut\cr}}%
  \setbox\tw@\vbox{\unvcopy\z@\global\setbox\@ne\lastbox}%
  \setbox\tw@\hbox{\unhbox\@ne\unskip\global\setbox\@ne\lastbox}%
  \setbox\tw@\hbox{%
    $\kern\wd\@ne\kern -\@tempdima\left\@firstoftwo#1%
      \if@borderstar\kern2pt\else\kern -\wd\@ne\fi%
    \global\setbox\@ne\vbox{\box\@ne\if@borderstar\else\kern 2\p@\fi}%
    \vcenter{\if@borderstar\else\kern -\ht\@ne\fi%
      \unvbox\z@\kern-\if@borderstar2\fi\baselineskip}%
      \if@borderstar\kern-2\@tempdima\kern2\p@\else\,\fi\right\@secondoftwo#1 $%
  }\null \;\vbox{\kern\ht\@ne\box\tw@}%
\endgroup
}
\makeatother
```

The `matrix` macro cannot be used together with the $\mathcal{AMS}$math package, it redefines this macro (see section 26.6 on page 61).

6 Super/Subscript and limits

Writing a_{min} and a_{max} gives the same depth for the subscript, but writing them in upright mode with `\mbox` gives a different depth: $a_{\mbox{min}}$ and $a_{\mbox{max}}$. The problem is the different height, which can be modified in several ways

- `$a_{\mbox{\vphantom{i}max}}`: $a_{\mbox{min}}$ and $a_{\mbox{\vphantom{i}max}}$;
- `$a_{\mathrm{max}}`: a_{min} and a_{max};
- `$a_{\max}`: $a_{\min}$ and $a_{\max}$. Both are predefined operators (see section 16 on page 42).

6.1 Multiple limits

\atop

For general information about limits read section 2.1 on page 10. With the `\atop` command multiple limits for a sum or prod are possible. The syntax is:

$${above \atop below}$$

```
\[ {above \atop below} \]
```

which is nearly the same as a fraction without a rule. This can be enhanced to `a\atop b\atop c` and so on. For equation 21 do the following steps:

$$\sum_{1\le j\le p \atop {1\le j\le q \atop 1\le k\le r}} a_{ij}b_{jk}c_{ki} \tag{21}$$

```
\begin{equation}\label{eq:atop}
\sum_{{1\le j\le p\atop {%
  {1\le j\le q\atop 1\le k\le r}}}%
}a_{ij}b_{jk}c_{ki}
\end{equation}
```

There are other solutions to get multiple limits, e.g. an array, which is not the best solution because the space between the lines is too big. The $\mathcal{AMS}$math package provides several commands for limits (section 35) and the `\underset` and `\overset` commands (see section 41).

6.2 Problems

$$\boxed{\sum_{1\le j\le p \atop {1\le j\le q \atop 1\le k\le r}}}\boxed{a_{ij}b_{jk}c_{ki}} \tag{22}$$

The equation 22 shows that the horizontal alignment is not optimal, because the math expression on the right follows at the end of the limits which are a unit together with the sum symbol. There is an elegant solution with $\mathcal{AMS}$math, described in subsection 35.2 on page 69. If you do not want to use $\mathcal{AMS}$math, then use `\makebox`. But there is a problem when the general fontsize is increased, `\makebox` knows nothing about the actual math font size. Equation 23a shows the effect and equation 23b the view without the boxes.

$$\sum_{\boxed{1\le j\le p \atop {1\le j\le q \atop 1\le k\le r}}}\boxed{a_{ij}b_{jk}c_{ki}} \tag{23a}$$

$$\sum_{1\le j\le p \atop {1\le j\le q \atop 1\le k\le r}} a_{ij}b_{jk}c_{ki} \tag{23b}$$

```
\begin{equation}
\sum_{\makebox[0pt]{$%
 {{\scriptscriptstyle 1\le j\le p\atop {%
 {1\le j\le q\atop 1\le k\le r}}}}%
  $}}a_{ij}b_{jk}c_{ki}
\end{equation}
```

7 Roots

The square root `\sqrt` is is the default for LaTeX and the n-th root can be inserted with the optional parameter `\sqrt[n]....` . `\sqrt`

`\sqrt{x}`	$\sqrt{x}$
`\sqrt[3]{x}`	$\sqrt[3]{x}$

There is a different typesetting in roots. Equation 24 has different heights for the roots, whereas equation 25 has the same one. This is possible with the `\vphantom` command, which reserves the vertical space (without a horizontal one) of the parameter height.

`\vphantom`

$$\sqrt{a}\,\sqrt{T}\,\sqrt{2\alpha k_{B_1}T^i} \tag{24}$$

```
\begin{equation}
  \sqrt{a}\,%
  \sqrt{T}\,%
  \sqrt{2\alpha k_{B_1}T^i}\label{eq:root1}
\end{equation}
```

$$\sqrt{a\vphantom{T}\vphantom{{}_{B_1}}}\,\sqrt{\vphantom{a}T\vphantom{{}_{B_1}}}\,\sqrt{2\alpha k_{B_1}T^i} \tag{25}$$

```
\bgroup
\begin{equation}\label{eq:root2}
  \sqrt{a\vphantom{T}\vphantom{{}_B{_1}}}\,%
  \sqrt{\vphantom{a}T\vphantom{{}_B{_1}}}\,%
  \sqrt{2\alpha k_{B_1}T^i}
\end{equation}
\egroup
```

The typesetting looks much more better, especially when the formula has different roots in a row, like equation 24. Using $\mathcal{AMS}$math with the `\smash` command[9] gives some more possibilities for typesetting of roots (see section 30 on page 64).

8 Brackets, braces and parentheses

This is one of the major problems inside the math mode, because there is often a need for different brackets, braces and parentheses in different size. At first we had to admit, that there is a difference between the characters “$()\,[]\,/\backslash\,\{\}\,|\,\|\,\lfloor\rfloor\,\lceil\rceil\,\langle\rangle\,\uparrow\Uparrow\,\downarrow\Downarrow\,\updownarrow\Updownarrow$” and their use as an argument of the `\left` and `\right` command, where LaTeX stretches the size in a way that all between the pair of left and right parentheses is smaller than the parentheses. In some cases[10] it may be useful to choose a fixed height, which is possible with the `\big`-series. Instead of writing `\leftX` or `\rightX` one of the following commands can be chosen:

`\leftX`
`\rightX`

`\bigX`
`\BigX`
`\biggX`
`\BiggX`

default	$()[]/\backslash\{\}|\|\lfloor\rfloor\lceil\rceil\langle\rangle\uparrow\Uparrow\downarrow\Downarrow\updownarrow\Updownarrow$
`\bigX`	$\big(\big)\big[\big]\big/\big\backslash\big\{\big\}\big|\big\|\big\lfloor\big\rfloor\big\lceil\big\rceil\big\langle\big\rangle\big\uparrow\big\Uparrow\big\downarrow\big\Downarrow\big\updownarrow\big\Updownarrow$
`\BigX`	$\Big(\Big)\Big[\Big]\Big/\Big\backslash\Big\{\Big\}\Big|\Big\|\Big\lfloor\Big\rfloor\Big\lceil\Big\rceil\Big\langle\Big\rangle\Big\uparrow\Big\Uparrow\Big\downarrow\Big\Downarrow\Big\updownarrow\Big\Updownarrow$
`\biggX`	$\bigg(\bigg)\bigg[\bigg]\bigg/\bigg\backslash\bigg\{\bigg\}\bigg|\bigg\|\bigg\lfloor\bigg\rfloor\bigg\lceil\bigg\rceil\bigg\langle\bigg\rangle\bigg\uparrow\bigg\Uparrow\bigg\downarrow\bigg\Downarrow\bigg\updownarrow\bigg\Updownarrow$
`\BiggX`	$\Bigg(\Bigg)\Bigg[\Bigg]\Bigg/\Bigg\backslash\Bigg\{\Bigg\}\Bigg|\Bigg\|\Bigg\lfloor\Bigg\rfloor\Bigg\lceil\Bigg\rceil\Bigg\langle\Bigg\rangle\Bigg\uparrow\Bigg\Uparrow\Bigg\downarrow\Bigg\Downarrow\Bigg\updownarrow\Bigg\Updownarrow$

[9] The `\smash` command exists also in LaTeX but without an optional argument, which makes the use for roots possible.

[10] See section 8.1.1 on page 28 for example.

Only a few commands can be written in a short form like `\big(`. The “X” has to be replaced with one of the following characters or commands from table 3, which shows the parentheses character, its code for the use with one of the “big” commands and an example with the code for that.

There exist for all commands a left/right version `\bigl`, `\bigr`, `\Bigl` and so on, which only makes sense when writing things like:

`\biglX`
`\bigrX`

$$\biggl)\times\frac{a}{b}\times\biggr(\tag{26}$$

$$\bigg)\times\frac{a}{b}\times\bigg(\tag{27}$$

```
\begin{align}
  \biggl)\times \frac{a}{b} \times\biggr(
\end{align}
\begin{align}
  \bigg)\times \frac{a}{b} \times\bigg(
\end{align}
```

LaTeX takes the `\biggl)` as a mathopen symbol, which has by default another horizontal spacing.

In addition to the above additional commands there exists some more: `\bigm`, `\Bigm`, `\biggm` and `\Biggm`, which work as the standard ones (without the addtional “m”) but add some more horizontal space between the delimiter and the formula before and after (see table 2).

`\bigmX`
`\bigmX`

$3\bigg\|a^2-b^2-c^2\bigg\|+2$	`$3\bigg\|a^2-b^2-c^2\bigg\|+2$`
$3\biggm\|a^2-b^2-c^2\biggm\|+2$	`$3\biggm\|a^2-b^2-c^2\biggm\|+2$`

Table 2: Difference between the default `\bigg` and the `\biggm` command

Char	Code	Example	Code
( )	( )	$3\Big(a^2+b^{c^2}\Big)$	`3\Big( a^2+b^{c^2}\Big)`
[ ]	[ ]	$3\Big[a^2+b^{c^2}\Big]$	`3\Big[ a^2+b^{c^2}\Big]`

/ \	`/\backslash`	$3\Big/ a^2+b^{c^2}\Big\backslash$	`3\Big/`
			`a^2+b^{c^2}\Big\backslash`

{ }	`\{\}`	$3\Big\{ a^2+b^{c^2}\Big\}$	`3\Big\{ a^2+b^{c^2}\Big\}`
| ‖	`| \Vert`	$3\Big\|a^2+b^{c^2}\Big\Vert$	`3\Big|a^2+b^{c^2}\Big\Vert`

⌊ ⌋	`\lfloor`
	`\rfloor`
		$3\Big\lfloor a^2+b^{c^2}\Big\rfloor$	`3\Big\lfloor a^2+b^{c^2}`
			`\Big\rfloor`

⌈ ⌉	`\lceil\rceil`	$3\Big\lceil a^2+b^{c^2}\Big\rceil$	`3\Big\lceil a^2+b^{c^2}`
			`\Big\rceil`

Char	Code	Example	Code

$\langle\ \rangle$	`\langle\rangle`	$3\Big\langle a^2+b^{c^2}\Big\rangle$	`3\Big\langle`
			`a^2+b^{c^2}\Big\rangle`

$\uparrow\ \Uparrow$	`\uparrow`
	`\Uparrow`
		$3\Big\uparrow a^2+b^{c^2}\Big\Uparrow$	`3\Big\uparrow`
			`a^2+b^{c^2}\Big\Uparrow`

$\downarrow\ \Downarrow$	`\downarrow`
	`\Downarrow`
		$3\Big\downarrow a^2+b^{c^2}\Big\Downarrow$	`3\Big\downarrow a^2+b^{c^2}`
			`\Big\Downarrow`

$\updownarrow\ \Updownarrow$	`\updownarrow`
	`\Updownarrow`
		$3\Big\updownarrow a^2+b^{c^2}\Big\Updownarrow$	`3\Big\updownarrow`
			`a^2+b^{c^2} \Big\Updownarrow`

Table 3: Use of the different parentheses for the "big" commands

8.1 Examples

8.1.1 Braces over several lines

The following equation in the single line mode looks like

$$\frac{1}{2}\Delta(f_{ij}f^{ij}) = 2\left(\sum_{i<j}\chi_{ij}(\sigma_i-\sigma_j)^2 + f^{ij}\nabla_j\nabla_i(\Delta f) + \nabla_k f_{ij}\nabla^k f^{ij} + f^{ij}f^k[2\nabla_i R_{jk} - \nabla_k R_{ij}]\right) \tag{28}$$

and is too long for the text width and the equation number has to be placed under the equation. [11] With the array environment the formula can be split in two smaller pieces:

$$\begin{array}{rcl}\frac{1}{2}\Delta(f_{ij}f^{ij}) & = & 2\left(\sum\limits_{i<j}\chi_{ij}(\sigma_i-\sigma_j)^2 + f^{ij}\nabla_j\nabla_i(\Delta f)+\right.\\ & & \left.+\nabla_k f_{ij}\nabla^k f^{ij} + f^{ij}f^k[2\nabla_i R_{jk} - \nabla_k R_{ij}]\right)\end{array} \tag{29}$$

It is obvious that there is a problem with the right closing parentheses. because of the two pairs "`\left( ... \right.`" and "`\left. ... \right)`" they have a different size because every pair does it in its own way. Using the `\Bigg` command changes this into a better typesetting:

[11] In standard LaTeX the equation and the number are printed one over the other for too long formulas. Only AMSmath puts it one line over (left numbers) or under (right numbers) the formula.

$$\begin{array}{rcl}
\frac{1}{2}\Delta(f_{ij}f^{ij}) & = & 2\Bigg({\displaystyle\sum_{i<j}}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\\
& & +\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}[2\nabla_{i}R_{jk}-\nabla_{k}R_{ij}]\Bigg)
\end{array} \tag{30}$$

```
{\arraycolsep=2pt
\begin{equation}
\begin{array}{rcl}
 \frac{1}{2}\Delta(f_{ij}f^{ij}) & = & 2\Bigg({\displaystyle
 \sum_{i<j}}\chi_{ij}(\sigma_{i}-\sigma_{j})^{2}+f^{ij}%
 \nabla_{j}\nabla_{i}(\Delta f)+\\
& & +\nabla_{k}f_{ij}\nabla^{k}f^{ij}+f^{ij}f^{k}[2
 \nabla_{i}R_{jk}-\nabla_{k}R_{ij}]\Bigg)
\end{array}
\end{equation}
}
```

Section 26.4 on page 60 shows another solution for getting the right size for parentheses when breaking the equation in smaller pieces.

$$\begin{aligned}
B(r,\phi,\lambda) = & \,\dfrac{\mu}{r}\Bigg[\sum_{n=2}^{\infty} \Bigg(\left(\dfrac{R_e}{r} \right)^n J_nP_n(s\phi) \\
& +\sum_{m=1}^n \left(\dfrac{R_e}{r} \right)^n (C_{nm}\cos m\lambda+S_{nm}\sin m\lambda)P_{nm}(s\phi) \Bigg)\Bigg]
\end{aligned}$$

```
\begin{align*}
  B(r,\phi,\lambda) = & \,\dfrac{\mu}{r}
    \Bigg[\sum_{n=2}^{\infty} \Bigg( \left( \dfrac{R_e}{r} \right)^n J_nP_n(s\phi) \\
  & +\sum_{m=1}^n \left( \dfrac{R_e}{r} \right) ^n
   (C_{nm}\cos m\lambda+S_{nm}\sin m\lambda)P_{nm}(s\phi) \Bigg)\Bigg]
\end{align*}
```

8.1.2 Middle bar

See section 52 on page 93 for examples and the use of package `braket.sty`.

8.2 New delimiters

The default delimiters are defined in the file `fontmath.ltx` which is stored in general in `[TEXMF]/tex/latex/base/fontmath.ltx`. If we need for example a thicker vertical symbol than the existing `\verb` symbol we can define in the preamble:

```
\DeclareMathDelimiter{\Norm}
  {\mathord}{largesymbols}{"3E}{largesymbols}{"3E}
```

The character number $3E_{16}$ (decimal 62) from the `cmex10` font is the small thick vertical rule. Now the new delimiter `\Norm` can be used in the usual way:

$$\left| *BLA* \right| \qquad \left| \dfrac{*BLA*}{*BLUB*} \right|$$

```
$\left\Norm *BLA* \right\Norm$

$\left\Norm \dfrac{*BLA*}{*BLUB*} \right\Norm$
```

8.3 Problems with parentheses

\delimitershortfall
\delimiterfactor

It is obvious that the following equation has not the right size of the parenthesis in the second integral, the innerone should be a bit smaller than the outer one.

$$\int_\gamma F'(z)dz = \int_\alpha^\beta F'\left(\gamma(t)\right)\cdot\gamma'(t)dt$$

```
\[
\int_\gamma F'(z) dz =\int_\alpha^\beta
F'\left(\gamma (t)\right)\cdot\gamma '(t)dt
\]
```

The problem is that TEX controlls the height of the parenthesis with `\delimitershortfall` and `\delimiterfactor`, with the default values

```
\delimitershortfall=5pt
\delimiterfactor=901
```

`\delimiterfactor/1000` is the relative size of the parenthesis for a given formula environment. They could be of `\delimitershortfall` too short. These values are valid at the end of the formula, the best way is to set them straight before the math environment or global for all in the preamble.

$$\int_\gamma F'(z)dz = \int_\alpha^\beta F'\bigl(\gamma(t)\bigr)\cdot\gamma'(t)dt$$

```
{\delimitershortfall=-1pt
\[
\int_\gamma F'(z) dz =\int_\alpha^\beta
F'\left(\gamma (t)\right)\cdot\gamma '(t)dt
\]}
```

9 Text in math mode

Standard text in math mode should be written in upright shape and not in the italic one which is reserved for the variable names: $\boxed{I\ am\ \mathrm{text}\ inside\ math.}$ or one of table 7 on page 33. There are different ways to write text inside math.

\textstyle
\mbox
\mathrm

- `\mathrm`. It is like math mode (no spaces), but in upright mode

- `\textrm`. Upright mode with printed spaces (real textmode)

- `\mbox`. The font size is still the one from `\textstyle` (see section 12 on page 37), so that you have to place additional commands when you use `\mbox` in a super- or subscript for limits.

Inserting long text is possible with a `parbox`, which can be aligned as usual to the top, bottom or center, e.g.

$$a+b+c+d+ef \quad = \quad g+h+i+j+k \qquad \text{this is a very long description of a formula} \tag{31}$$

```
\begin{eqnarray}
  a+b+c+d+ef  &  =  &  g+h+i+j+k  %
  \qquad\textrm{\parbox[t]{.25\linewidth}{%
      this is a very long description of a formula}%
  }
\end{eqnarray}
```

Additional commands for text inside math are provided by $\mathcal{AMS}$math (see section 37 on page 71).

10 Font commands

10.1 Old-style font commands

Should never be used, but are still present and supported by LaTeX. The default syntax for the old commands is

```
{\XX test}
```

Table 4 shows what for the `XX` has to be replaced. The major difference to the new style is that these `\XX` are toggling the actual math mode into the "XX" one, whereas the new commands starts a group which switches at its end back to the mode before.

`\bf`	$\mathbf{test}$	`\cal`	$\mathcal{TEST}$	`\it`	test	`\rm`	test	`\tt`	$\mathtt{test}$

Table 4: Old font style commands

10.2 New-style font commands

`\mathrm`
`\mathfrak`
`\mathcal`
`\mathsf`
`\mathbb`
`\mathtt`
`\mathit`
`\mathbf`

The default syntax is

```
\mathXX{test}
```

Table 5 shows what for the XX have to be replaced. See section 57 on page 97 for additional packages.

Command	**Test**
default	$ABCDEFGHIJKLMNOPQRSTUVWXYZ$ $abcdefghijklmnopqrstuvwxyz$
`\mathfrak`	$\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$ $\mathfrak{abcdefghijklmnopqrstuvwxyz}$
`\mathcal`[a]	$\mathcal{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
`\mathsf`	$\mathsf{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$ $\mathsf{abcdefghijklmnopqrstuvwxyz}$
`\mathbb`[a]	$\mathbb{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$
`\mathtt`	$\mathtt{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$ $\mathtt{abcdefghijklmnopqrstuvwxyz}$
`\mathit`	$\mathit{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$ $\mathit{abcdefghijklmnopqrstuvwxyz}$
`\mathrm`	$\mathrm{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$ $\mathrm{abcdefghijklmnopqrstuvwxyz}$
`\mathbf`	$\mathbf{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$ $\mathbf{abcdefghijklmnopqrstuvwxyz}$
`\mathds`[b]	$\mathds{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$

[a] Not available for lower letters. For mathcal exists a non free font for lower letters (http://www.yandy.com)

[b] Needs package `dsfont.sty`

Table 5: Fonts in math mode

11 Space

11.1 Math typesetting

`\thinmuskip`
`\medmuskip`
`\thickmuskip`

LaTeXdefines the three math lengths[12] with the following values[13]:

```
\thinmuskip=3mu
\medmuskip=4mu plus 2mu minus 4mu
\thickmuskip=5mu plus 5mu
```

where `mu` is the abbreviation for `math unit`.

$$1\mathrm{mu} = \frac{1}{18}\mathrm{em}$$

These lengths can have all glue and are used for the horizontal spacing in math expressions where TeX puts spaces between symbols and operators. The meaning of these different horizontal skips is shown in the table 6. For a better typesetting LaTeX inserts different spaces between the symbols.

`\thinmuskip` space between ordinary and operator atoms

`\medmuskip` space between ordinary and binary atoms in display and text styles

[12] For more information see: http://www.tug.org/utilities/plain/cseq.html

[13] see `fontmath.ltx`

default	$\boxed{f(x) = x^2 + 3x_0 \cdot \sin x}$
`\thinmuskip=0mu`	$\boxed{f(x) = x^2 + 3x_0 \cdot \mathrm{sin}x}$
`\medmuskip=0mu`	$\boxed{f(x) = x^2{+}3x_0{\cdot}\sin x}$
`\thickmuskip=0mu`	$\boxed{f(x){=}x^2 + 3x_0 \cdot \sin x}$
all set to zero	$\boxed{f(x){=}x^2{+}3x_0{\cdot}\mathrm{sin}x}$

Table 6: The meaning of the math spaces

`\thickmuskip` space between ordinary and relation atoms in display and text styles

11.2 Additional horizontal spacing

`\thinspace`
`\medspace`
`\thickspace`
`\negthinspace`
`\negmedspace`
`\negthickspace`

Positive Space		Negative Space	
`$ab$`	$\boxed{a}\boxed{b}$		
`$a b$`	$\boxed{a}\boxed{b}$		
`$a\ b$`	$\boxed{a}\ \boxed{b}$		
`$a\mbox{\textvisiblespace}b$`	$\boxed{a}\text{␣}\boxed{b}$		
`$a\,b$` (`$a\thinspace b$`)	$\boxed{a}\,\boxed{b}$	`$a\! b$`	$\boxed{a}\!\boxed{b}$
`$a\: b$` (`$a\medspace b$`)	$\boxed{a}\:\boxed{b}$	`$a\negmedspace b$`	$\boxed{a}\negmedspace\boxed{b}$
`$a\; b$` (`$a\thickspace b$`	$\boxed{a}\;\boxed{b}$	`$a\negthickspace b$`	$\boxed{a}\negthickspace\boxed{b}$
`$a\quad b$`	$\boxed{a}\quad\boxed{b}$		
`$a\qquad b$`	$\boxed{a}\qquad\boxed{b}$		
`$a\hspace{0.5cm}b$`	$\boxed{a}\hspace{0.5cm}\boxed{b}$	`$a\hspace{-0.5cm}b$`	$\boxed{a}\hspace{-0.5cm}\boxed{b}$
`$a\kern0.5cm b$`	$\boxed{a}\kern0.5cm\boxed{b}$	`$a\kern-0.5cm b$`	$\boxed{a}\kern-0.5cm\boxed{b}$
`$a\hphantom{xx}b$`	$\boxed{a}\hphantom{xx}\boxed{b}$		
`$axxb$`	$\boxed{a}xx\boxed{b}$		

Table 7: Spaces in math mode

LaTeX defines the following short commands:

```
\def\>{\mskip\medmuskip}
\def\;{\mskip\thickmuskip}
\def\!{\mskip-\thinmuskip}
```

In math mode there is often a need for additional tiny spaces between variables, e.g. $L\frac{di}{dt}$ written with a tiny space between L and $\frac{di}{dt}$ looks nicer: $L\,\frac{di}{dt}$. Table 7 shows a list of all commands for horizontal space which can be used in math mode. The "space" is seen "between" the boxed a and b. For all examples a is `\boxed{a}` and b is `\boxed{b}`. The short forms for some spaces may cause problems with other packages. In this case use the long form of the commands.

`\hspace`
`\hphantom`
`\kern`

11.3 Problems

Using `\hphantom` in mathmode depends to the object. `\hphantom` reserves only the space of the exact width without any additional space. In the following example the second line is wrong: `& \hphantom{\rightarrow} b\\`. It does not reserve any additional space.

$$\begin{aligned}
a & \rightarrow b\\
  & \hphantom{\rightarrow} b\\
  & \mkern\thickmuskip\hphantom{\rightarrow}\mkern\thickmuskip b\\
  & \mathrel{\hphantom{\rightarrow}} b
\end{aligned}$$

```
\begin{align*}
a & \rightarrow b\\
  & \hphantom{\rightarrow} b\\
  & \mkern\thickmuskip\hphantom{\rightarrow}\mkern\thickmuskip b\\
  & \mathrel{\hphantom{\rightarrow}} b
\end{align*}
```

This only works when the math symbol is a mathrel one, otherwise you have to change the horizontal space to `\medmuskip` or `\thinmuskip`. For more informations about the math objects look into `fontmath.ltx` or `amssymb.sty` or use the `\show` macro, which prints out the type of the math-symbol, e.g.: `\show\rightarrow` with the output:

```
> \rightarrow=\mathchar"3221.
l.20 \show\rightarrow
```

The first digit represents the type:
0 : ordinary
1 : large operator
2 : binary operation
3 : relation
4 : opening
5 : closing
6 : punctuation
7 : variable family

Grouping a math symbol can change the behaviour in horizontal spacing. Compare 50×10^{12} and $50{\times}10^{12}$, the first one is typeset with `$50\times10^{12}$` and the second one with `$50{\times}10^{12}$`. Another possibilty is to use the `numprint` package.[14]

11.4 Dot versus comma

`\mathpunct`
`\mathord`

In difference to a decimal point and a comma as a marker of thousends a lot of countries prefer it vice versa. To get the same behaviour the meaning of

[14] CTAN://macros/latex/contrib/numprint/

dot and comma has to be changed:

$$1,234,567.89 \text{ default} \tag{32}$$

$$1.234.567,89 \text{ vice versa, wrong spacing} \tag{33}$$

$$1\mathpunct{.}234\mathpunct{.}567{,}89 \text{ correct spacing} \tag{34}$$

```
1,234,567.89 & \textrm{ default}\\
1.234.567,89 & \textrm{ vice versa, wrong spacing}\\
1\mathpunct{.}234\mathpunct{.}567{,}89 & \textrm{ correct
  spacing}
```

The original definitions from `fontmath.ltx`[15] are

```
\DeclareMathSymbol{,}{\mathpunct}{letters}{"3B}
\DeclareMathSymbol{.}{\mathord}{letters}{"3A}
```

and can be changed for a documentwide other behaviour. In the above equation 34 the comma is only set in a pair of braces `{,}`, which is the same as writing `\mathord{,}` because LaTeX handles everything inside of parenthises as a formula, which gets the same spacing.

It is also possible to use the package `icomma.sty`[16] for a documentwide correct spacing.

11.5 Vertical whitespace

11.5.1 Before/behind math expressions

There are four predefined lengths, which control the vertical whitespace of displayed formulas:

```
\abovedisplayskip=12pt plus 3pt minus 9pt
\abovedisplayshortskip=0pt plus 3pt
\belowdisplayskip=12pt plus 3pt minus 9pt
\belowdisplayshortskip=7pt plus 3pt minus 4pt
```

The short skips are used if the formula starts behind the end of the foregoing last line. Only for demonstration in the following examples the shortskips are set to `0pt` and the normal skips to `20pt` without any glue:

[15] Located in `texmf/tex/latex/base/`

[16] CTAN:// macros/latex/contrib/was/

The line ends before.

$$f(x) = \int \frac{\sin x}{x} dx \tag{35}$$

The line doesn't end before the formula.

$$f(x) = \int \frac{\sin x}{x} dx \tag{36}$$

And the next line starts as usual with some text ...

```
\abovedisplayshortskip=0pt
\belowdisplayshortskip=0pt
\abovedisplayskip=20pt
\belowdisplayskip=20pt
\noindent The line ends before.
\begin{equation}
 f(x) = \int\frac{\sin x}{x}dx
\end{equation}
\noindent The line doesn't end before the formula.
\begin{equation}
 f(x) = \int\frac{\sin x}{x}dx
\end{equation}
\noindent And the next line starts as usual with some text ...
```

11.5.2 Inside math expressions

`\\[<length>]` This works inside the math mode in the same way as in the text mode.

`\jot`

`\jot` The vertical space between the lines for all math expressions which allow multiple lines can be changed with the length `\jot`, which is predefined as

`\newdimen\jot \jot=3pt`

The following three formulas show this for the default value, `\jot=0pt` and `\jot=10pt`.

$$\begin{array}{rcl} y & = & d \\ y & = & c\frac{1}{x} + d \\ y & = & b\frac{1}{x^2} + cx + d \end{array} \qquad \begin{array}{rcl} y & = & d \\ y & = & c\frac{1}{x} + d \\ y & = & b\frac{1}{x^2} + cx + d \end{array} \qquad \begin{array}{rcl} y & = & d \\[10pt] y & = & c\frac{1}{x} + d \\[10pt] y & = & b\frac{1}{x^2} + cx + d \end{array}$$

Defining a new environment with a parameter makes things easier, because changes to the length are locally.

```
\newenvironment{mathspace}[1]{%
  \setlength{\jot}{#1}%
  \ignorespaces%
}{%
  \ignorespacesafterend%
}
```

\arraystretch

\arraystretch The vertical space between the lines for all math expressions which contain an array environment can be changed with the command \arraystretch, which is predefined as

```
\def\arraystretch{1}
```

Renewing this definition is global to all following math expressions, so it should be used in the same way than \jot.

Package setspace To have all formulas with another vertical spacing, one can choose the package setspace and redefining some of the math macros, e.g.

```
\newcommand*\Array[2][1]{\setstretch{#1}\array{#2}}
\let\endArray\endarray
```

$$\begin{array}{cc} a = & b \\[2ex] a = & b \\[2ex] a = & b \end{array}$$

text $\begin{array}{cc} a = & b \\ a = & b \\ a = & b \end{array}$ text

```
\[
\begin{Array}[2]{cc}
a =&b\\
a =&b\\
a =&b
\end{Array}
\]

text $\begin{Array}{cc}
a =&b\\
a =&b\\
a =&b
\end{Array}$ text
```

12 Styles

\textstyle
\displaystyle
\scriptstyle
\scripscriptstyle

This depends on the environment in which they are used. An inline formula has a default math fontsize called \textstyle, which is smaller than the one for a display formula (see section 3), which is called \displaystyle. Below this predefinition there are two other special fontstyles for math, \scriptstyle and \scriptscriptstyle. They are called “style” in difference to “size”, because they have a dynamic character, their real fontsize belongs to the environment in which they are used. A fraction for example is by default in scriptstyle when it is in an inline formula like this $\frac{a}{b}$, which

Mode	Inline	Displayed
default	$f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\displaystyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\displaystyle`	$\displaystyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\displaystyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\scriptstyle`	$f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\scriptstyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\scriptscriptstyle`	$\scriptscriptstyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\scriptscriptstyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$
`\textstyle`	$\textstyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$	$\textstyle f(t)=\frac{T}{2\pi}\int\frac{1}{\sin\frac{\omega}{t}}dt$

Table 8: Math styles

can be changed to $\displaystyle\frac{a}{b}$. This maybe in some cases useful but it looks in general ugly because the line spacing is too big. These four styles are predefined and together in a logical relationship. It is no problem to use the other styles like `\large`, `\Large`, ... **outside** the math environment. For example a fraction written with `\Huge`: $\frac{a}{b}$ (`\Huge$\frac{a}{b}$`). This may cause some problems when you want to write a displayed formula in another fontsize, because it also affects the interline spacing of the preceding part of the paragraph. If you end the paragraph, you get problems with spacing and page breaking above the equations. So it is better to declare the font size and then restore the baselines:

$$\int_1^2\frac{1}{x^2}\,dx=0.5 \tag{37}$$

```
\makeatletter
\newenvironment{smallequation}[1]{%
  \skip@=\baselineskip
  #1%
  \baselineskip=\skip@
  \equation
}{\endequation \ignorespacesafterend}
\makeatother

\begin{smallequation}{\tiny}
\int_1^2\,\frac{1}{x^2}\,dx=0.5
\end{smallequation}
```

If you use this the other way round for huge fontsizes, don't forget to load package `exscale` (see section 58 on page 98). Also ee this section for diffent symbol sizes.

13 Dots

\cdots
\dots
\dotsb
\dotsc
\dotsi
\dotsm
\dotso
\ldots
\vdots

In addition to the above decorations there are some more different dots which are single commands and not by default over/under a letter. It is not easy to see the differences between some of them. Dots from lower left to upper right are possible with `\reflectbox{$\ddots$}` ⋰

`\cdots`	$\cdots$	`\ddots`	$\ddots$	`\dotsb`	$\cdots$	`\dotsc`	$\ldots$	`\dotsi`	$\cdots$
`\dotsm`	$\cdots$	`\dotso`	$\ldots$	`\ldots`	$\ldots$	`\vdots`	$\vdots$		

Table 9: Dots in math mode

14 Accents

The letter “a” is only for demonstration. The table 10 shows all in standard LaTeX available accents and the ones which are placed under a character, too. With package `amssymb` it is easy to define new accents. For more information see section 31 on page 65 or other possibilities at section 47 on page 90.

`\acute`	$\acute{a}$	`\bar`	$\bar{a}$	`\breve`	$\breve{a}$
`\bar`	$\bar{a}$	`\breve`	$\breve{a}$		
`\check`	$\check{a}$	`\dddot`	$\dddot{a}$	`\ddot`	$\ddot{a}$
`\dot`	$\dot{a}$	`\grave`	$\grave{a}$	`\hat`	$\hat{a}$
`\mathring`	$\mathring{a}$	`\overbrace`	$\overbrace{a}$	`\overleftarrow`	$\overleftarrow{a}$
`\overleftrightarrow`	$\overleftrightarrow{a}$	`\overline`	$\overline{a}$	`\overrightarrow`	$\overrightarrow{a}$
`\tilde`	$\tilde{a}$	`\underbar`	$\underline{\mathrm{a}}$	`\underbrace`	$\underbrace{a}$
`\underleftarrow`	$\underleftarrow{a}$	`\underleftrightarrow`	$\underleftrightarrow{a}$	`\underline`	$\underline{a}$
`\underrightarrow`	$\underrightarrow{a}$	`\vec`	$\vec{a}$	`\widehat`	$\widehat{a}$
`\widetilde`	$\widetilde{a}$				

Table 10: Accents in math mode

The letters `i` and `j` can be substituted with the macros `\imath` and `\jmath` when an accents is placed over these letters and the dot should disappear: $\vec{\imath}\ \dddot{\jmath}$ (`$\vec{\imath}\ \dddot{\jmath}$`).

Accents can be used in different ways, e.g. strike a single chracter with a horizontal line like `$\mathaccent‘-A$`: $\rlap{\,-}A$ or `$\mathaccent\mathcode‘-A$`: $\rlap{-}A$. In section 53 on page 94 is a better solution for more than one character.

14.1 Over- and underbrackets

There are no `\underbracket` and `\overbracket` commands in the list of accents. They can be defined in the preamble with the following code.

```
\makeatletter
\def\underbracket{%
 \@ifnextchar[{\@underbracket}{\@underbracket [\@bracketheight]}%
}
\def\@underbracket[#1]{%
 \@ifnextchar[{\@under@bracket[#1]}{\@under@bracket[#1][0.4em]}%
}
\def\@under@bracket[#1][#2]#3{%\message {Underbracket: #1,#2,#3}
 \mathop{\vtop{\m@th \ialign {##\crcr $\hfil \displaystyle {#3}\hfil $%
 \crcr \noalign {\kern 3\p@ \nointerlineskip }\upbracketfill {#1}{#2}
     \crcr \noalign {\kern 3\p@ }}}}\limits}
\def\upbracketfill#1#2{$\m@th \setbox \z@ \hbox {$\braceld$}
                  \edef\@bracketheight{\the\ht\z@}\bracketend{#1}{#2}
                  \leaders \vrule \@height #1 \@depth \z@ \hfill
                  \leaders \vrule \@height #1 \@depth \z@ \hfill \bracketend{#1}{#2}$}
\def\bracketend#1#2{\vrule height #2 width #1\relax}
\makeatother
```

1. `\underbrace{...}` is an often used command:

$$\underbrace{x^2+2x+1}_{(x+1)^2} = f(x) \tag{38}$$

2. Sometimes an under**bracket** is needed, which can be used in more ways than `\underbrace{...}` an example for `\underbracket{...}`:

$$\text{Hate Science}\quad \underbracket{1\rightarrow 2\rightarrow 3\rightarrow 4}_{\text{low}}\rightarrow \underbracket[1pt]{5\rightarrow 6\rightarrow 7}_{\text{medium}}\rightarrow \underbracket[2pt]{8\rightarrow 9\rightarrow 10}_{\text{high}}\quad \text{Love Science}$$

14.1.1 Use of `\underbracket{...}`

The `\underbracket{...}` command has two optional parameters:

- the line thickness in any valid latex unit, e.g. `1pt`
- the height of the edge brackets, e.g. `1em`

using without any parameters gives the same values for thickness and height as predefined for the `\underbrace` command.

1.	`$\underbracket {foo\ bar}$`	$\underbracket{foo\ bar}$
2.	`$\underbracket[2pt] {foo\ bar}$`	$\underbracket[2pt]{foo\ bar}$
3.	`$\underbracket[2pt] [1em] {foo\ bar}$`	$\underbracket[2pt][1em]{foo\ bar}$

14.1.2 Overbracket

In addition to the underbracket an overbracket is also useful, which can be used in more ways than `\overbrace{...}`. For example:

$$\text{Hate Science}\quad \underset{\text{low}}{\overbracket{1\to2\to3\to4}}\underset{\text{medium}}{\overbracket[1pt]{\to5\to6\to}}\underset{\text{high}}{\overbracket[2pt]{7\to8\to9\to10}}\quad\text{Love Science}$$

The `\overbracket{...}` command has two optional parameters:

- the line thickness in any valid latex unit, e.g. `1pt`
- the height of the edge brackets, e.g. `1em`

using without any parameters gives the same values for thickness and height as predefined for the `\overbrace` command.

1.	`$\overbracket {foo\ bar}$`	$\overbracket{foo\ bar}$
2.	`$\overbracket[2pt] {foo\ bar}$`	$\overbracket[2pt]{foo\ bar}$
3.	`$\overbracket[2pt] [1em] {foo\ bar}$`	$\overbracket[2pt][1em]{foo\ bar}$

14.2 Vectors

Especially for vectors there is the `esvect.sty`[17] package, which looks better than the `\overrightarrow`, f.ex:

\vv{...}	\overrightarrow{...}
$\vv{a}$	$\overrightarrow{a}$
$\vv{abc}$	$\overrightarrow{abc}$
$\vv{\imath}$	$\overrightarrow{\imath}$
$\vv{A}_x$	$\overrightarrow{A}_x$

Table 11: Vectors with package `esvect.sty` (in the right column the default one from LaTeX)

Look into the documentation for more details about `esvect.sty`.

[17] CTAN://macros/latex/contrib/esvect/

15 Exponents and indices

The two active characters `_` and `^` can only be used in math mode. The **following** character will be printed as an index (`$y=a_1x+a_0$`: $y = a_1x + a_0$) or as an exponent (`$x^2+y^2=r^2$`: $x^2+y^2 = r^2$). For more than the next character put it inside of `{}`, like `$a_{i-1}+a_{i+1}<a_i$`: $a_{i-1}+a_{i+1} < a_i$.

Especially for multiple exponents there are several possibilities. For example:

$$((x^2)^3)^4 = {({(x^2)}^3)}^4 = {\left({\left(x^2\right)}^3\right)}^4 \tag{39}$$

```
((x^2)^3)^4 =
{({(x^2)}^3)}^4 =
{\left({\left(x^2\right)}^3\right)}^4
```

For variables with both exponent and indices index the order is not important, `$a_1^2$` is exactly the same than `$a^2_1$`: $a_1^2 = a_1^2$

16 Operators

They are written in upright font shape and are placed with some additional space before and behind for a better typesetting. With the $\mathcal{AMS}$math package it is possible to define one's own operators (see section 36 on page 71). Table 12 and 13 on the following page show a list of the predefined ones for standard LaTeX.

`\coprod`	$\coprod$	`\bigvee`	$\bigvee$	`\bigwedge`	$\bigwedge$
`\biguplus`	$\biguplus$	`\bigcap`	$\bigcap$	`\bigcup`	$\bigcup$
`\intop`	$\intop$	`\int`	$\int$	`\prod`	$\prod$
`\sum`	$\sum$	`\bigotimes`	$\bigotimes$	`\bigoplus`	$\bigoplus$
`\bogodot`	$\bigodot$	`\ointop`	$\oint$	`\oint`	$\oint$
`\bigsqcup`	$\bigsqcup$	`\smallint`	$\smallint$		

Table 12: The predefined operators of `fontmath.ltx`

The difference between `\intop` and `\int` is that the first one has by default over/under limits and the second subscript/superscript limits. Both can be changed with the `\limits` or `\nolimits` command. The same behaviour happens to the `\ointop` and `\oint` Symbols.

For more predefined operator names see table 20 on page 92. It is easy to define a new operator with

```
\makeatletter
\newcommand\foo{\mathop{\operator@font foo}\nolimits}
\makeatother
```

Now you can use `\foo` in the usual way:

`\log`	$\log$	`\lg`	$\lg$	`\ln`	$\ln$
`\lim`	$\lim$	`\limsup`	$\limsup$	`\liminf`	$\liminf$
`\sin`	$\sin$	`\arcsin`	$\arcsin$	`\sinh`	$\sinh$
`\cos`	$\cos$	`\arccos`	$\arccos$	`\cosh`	$\cosh$
`\tan`	$\tan$	`\arctan`	$\arctan$	`\tanh`	$\tanh$
`\cot`	$\cot$	`\coth`	$\coth$	`\sec`	$\sec$
`\csc`	$\csc$	`\max`	$\max$	`\min`	$\min$
`\sup`	$\sup$	`\inf`	$\inf$	`\arg`	$\arg$
`\ker`	$\ker$	`\dim`	$\dim$	`\hom`	$\hom$
`\det`	$\det$	`\exp`	$\exp$	`\Pr`	$\Pr$
`\gcd`	$\gcd$	`\deg`	$\deg$	`\bmod`	$\bmod$
`\pmod{a}`	$\pmod{a}$				

Table 13: The predefined operators of `latex.ltx`

$$\mathrm{foo}_1^2 = x^2$$

```
\[ \foo_1^2 = x^2 \]
```

In this example `\foo` is defined with `\nolimits`, means that limits are placed in superscript/subscript mode and not over under. This is still possible with `\limits` in the definition or the equation:

$$\mathop{\mathrm{foo}}\limits_1^2 = x^2$$

```
\[ \foo\limits_1^2 = x^2 \]
```

AMSmath has an own macro for a definition, have a look at section 36 on page 71.

17 Greek letters

The AMSmath package simulates a bold font for the greek letters, it writes a greek character twice with a small kerning. The `\mathbf{<character>}` doesn't work with lower greek character. See section 40 on page 76 for the `\pmb` macro, which makes it possible to print bold lower greek letters. Not all upper case letters have own macro names. If there is no difference to the roman font, then the default letter is used, e.g.: A for the upper case of α. Table 14 shows only those upper case letters which have own macro names. Some of the lower case letters have an additional `var` option for an alternative.

lower	default	upper	default	`\mathbf`	`\mathit`
`\alpha`	α				
`\beta`	β				
`\gamma`	γ	`\Gamma`	Γ	$\mathbf{\Gamma}$	$\mathit{\Gamma}$
`\delta`	δ	`\Delta`	Δ	$\mathbf{\Delta}$	$\mathit{\Delta}$
`\epsilon`	ϵ				

lower	default	upper	default	`\mathbf`	`\mathit`
`\varepsilon`	ε				
`\zeta`	ζ				
`\eta`	η				
`\theta`	θ	`\Theta`	Θ	$\mathbf{\Theta}$	$\mathit{\Theta}$
`\vartheta`	ϑ				
`\iota`	ι				
`\kappa`	κ				
`\delta`	λ	`\Lambda`	Λ	$\mathbf{\Lambda}$	$\mathit{\Lambda}$
`\mu`	μ				
`\nu`	ν				
`\xi`	ξ	`\Xi`	Ξ	$\mathbf{\Xi}$	$\mathit{\Xi}$
`\pi`	π	`\Pi`	Π	$\mathbf{\Pi}$	$\mathit{\Pi}$
`\varpi`	ϖ				
`\rho`	ρ				
`\varrho`	ϱ				
`\sigma`	σ	`\Sigma`	Σ	$\mathbf{\Sigma}$	$\mathit{\Sigma}$
`\varsigma`	ς				
`\tau`	τ				
`\upsilon`	υ	`\Upsilon`	Υ	$\mathbf{\Upsilon}$	$\mathit{\Upsilon}$
`\phi`	ϕ	`\Phi`	Φ	$\mathbf{\Phi}$	$\mathit{\Phi}$
`\varphi`	φ				
`\chi`	χ				
`\psi`	ψ	`\Psi`	Ψ	$\mathbf{\Psi}$	$\mathit{\Psi}$
`\omega`	ω	`\Omega`	Ω	$\mathbf{\Omega}$	$\mathit{\Omega}$

Table 14: The greek letters

Bold greek letters are possible with the package `bm` (see section 51 on page 93) and if they should also be upright with the package `upgreek`:

```
\bm{\upalpha}, $bm{\upbeta} ...
```

18 Pagebreaks

`\allowdisplaybreaks`

By default a displayed formula cannot have a pagebreak. This makes some sense, but sometimes it gives a better typesetting when a pagebreak is possible.

```
\allowdisplaybreaks
```

This macro enables TEX to insert pagebreaks into displayed formulas whenever a newline command appears. With the command `\displaybreak` it is also possible to insert a pagebreak at any place.

19 \stackrel

`\stackrel` puts a character on top of another one which may be important if a used symbol is not predefined. For example “$\stackrel{\triangle}{=}$” (`\stackrel{\wedge}{=}`). The syntax is

```
\stackrel{top}{base}
```

Such symbols may be often needed so that a macro definition in the preamble makes some sense:

```
\newcommand{\eqdef}{%
 \ensuremath{%
  \stackrel{\mathrm{def}}{=}%
 }%
}
```

With the `\ensuremath` command we can use the new `\eqdef` command in text and in math mode, LaTeX switches automatically in math mode, which saves some keystrokes like the following command, which is written without the delimiters (`$...$`) for the math mode $\stackrel{\mathrm{def}}{=}$, only `\eqdef` with a space at the end. In math mode together with another material it may look like $\vec{x}\stackrel{\mathrm{def}}{=}\left(x_{1},\ldots,x_{n}\right)$ and as command sequence

```
$\vec{x}\eqdef\left(x_{1},\ldots,x_{n}\right)$
```

The fontsize of the top is one size smaller than the one from the base, but it is no problem to get both in the same size, just increase the top or decrease the base.

20 \choose

`\choose` is like `\atop` with delimiters or like `\frac` without the fraction line and also with delimiters. It is often used for binoms and has the following syntax:

```
{above \choose below}
```

The two braces are not really important but it is safe to use them.

$$\binom{m+1}{n}=\binom{m}{n}+\binom{m}{k-1} \tag{40}$$

```
{{m+1 \choose n}}={{m \choose n}}+{{m \choose k-1}}\label{eq:
  choose}
```

See section 29.2 on page 64 for the $\mathcal{AMS}$math equivalents and enhancements.

21 Color in math expressions

There is no difference in using coloured text and colored math expressions. With

`\usepackage{color}`

in the preamble the macro `\textcolor{<color>}{<text or math>}` exists.

$$f(x) = \int\limits_1^{\infty} \frac{1}{x^2}\,dx = 1 \tag{41}$$

```
\begin{equation}
\textcolor{blue}{f(x)} = \int\limits_1^{\infty}\textcolor{red}{\frac{1}{x^2}}\,dx=1
\end{equation}
```

If all math expressions should be printed in the same color, then it is better to use the everydisplay macro (section 24 on page 48).

22 Boldmath

\mathversion
\boldmath
\unboldmath

Writing a whole formula in bold is possible with the command sequence `\boldmath ... \unboldmath`, which itself must be written in textmode (outside the formula) or with the command `{\mathversion{bold} ... }`.

$$\sum_{\substack{1\le j\le p\\ 1\le j\le q\\ 1\le k\le r}} a_{ij}b_{jk}c_{ki} \qquad \boldsymbol{\sum_{\substack{1\le j\le p\\ 1\le j\le q\\ 1\le k\le r}} a_{ij}b_{jk}c_{ki}}$$

```
\boldmath
\[
\sum_{%
 \makebox[0pt]{$%
  {{\scriptscriptstyle 1\le j\le p\atop {%
  {1\le j\le q\atop 1\le k\le r}}}}%
 $}%
 }a_{ij}b_{jk}c_{ki}
\]
\unboldmath
```

The `\mathversion` macro defines a math style which is valid for all following math expressions. If you want to have all math in bold then use this macro instead of `\boldmath`. But it is no problem to put `\mathversion` inside a group to hold the changes locally.

$$\boldsymbol{y(x) = ax^3 + bx^2 + cx + d} \tag{42}$$

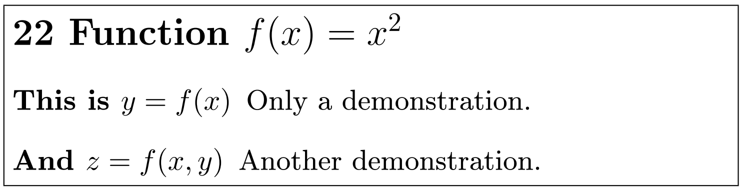
```
{\mathversion{bold}%
\begin{equation}
y(x) = ax^3+bx^2+cx+d
\end{equation}}
```

Single characters inside a formula can be written in bold with `mathbf`, but only in upright mode, which is in general not useful as shown in equation 43. It is better to use package `bm.sty` (see section 51 on page 93).

$$\sum_{\substack{1\le j\le p\\1\le j\le q\\1\le k\le r}} a_{ij}\mathbf{b_{jk}}c_{ki} \tag{43}$$

22.1 Bold math expressions as part of titles and items

By default the titles in sections, subsections, a.s.o. are printed in bold. Same for the `description` environment. The problem is that a math expression in one of these environments is printed in default font shape, like the following example for a `section` and `description` environment:

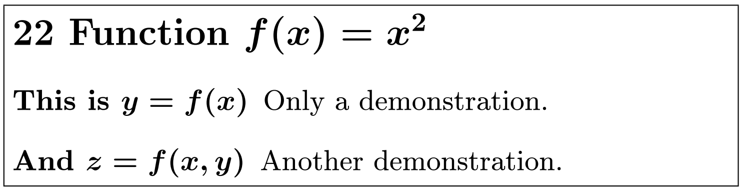


22 Function $f(x) = x^2$

This is $y = f(x)$ Only a demonstration.

And $z = f(x, y)$ Another demonstration.

With a redefinition of the `section` and `item` macros it is possible to get everything in bold font.



22 Function $\boldsymbol{f(x) = x^2}$

This is $\boldsymbol{y = f(x)}$ Only a demonstration.

And $\boldsymbol{z = f(x, y)}$ Another demonstration.

```
\let\itemOld\item
\makeatletter
\renewcommand\item[1][]{%
    \def\@tempa{#1}
    \ifx\@tempa\@empty\itemOld\else\boldmath\itemOld[#1]\unboldmath\fi%
}
\makeatother
\let\sectionOld\section
\renewcommand\section[2][\empty]{%
 \boldmath\sectionOld[#1]{#2}\unboldmath%
}
```

23 Multiplying numbers

When the dot is used as the decimal marker as in the United States, the preferred sign for the multiplication of numbers or values of quantities is a cross (`\times` $\times$), not a half-high and centered dot (`\cdot` $\cdot$).

When the comma is used as the decimal marker as in Europe, the preferred sign for the multiplication of numbers is the half-high dot. The multiplication of quantity symbols (or numbers in parentheses or values of quantities in parentheses) may be indicated in one of the following ways: ab, $a \cdot b$, $a \times b$.

For more information see "Nist Guide to SI Units -More on Printing and Using Symbols and Numbers in Scientific and Technical Documents"[18] or the German DIN 1304, Teil 1.

24 Other macros

`\everymath`
`\everydisplay`
`\underline`

There are some other macros which are not mentioned in the foregoing text. Here comes a not really complete list of these macros.

`\everymath` puts the argument before any inlined math expression, e.g. `\everymath{\small}`.

`\everydisplay` puts the argument before any displayed math expression, e.g. `\everydisplay{\color{blue}}`.

`\underline` underlines a math expression and has to be used inside the math mode.

$$\underline{F(x) = \int f(x)\,dx}$$

[18] http://physics.nist.gov/Pubs/SP811/sec10.html

Part II
$\mathcal{AMS}$math package

In general the $\mathcal{AMS}$ packages are at least a collection of three different ones:

1. `amsmath.sty`
2. `amssymb.sty`
3. `amsfonts.sty`

In the following only the first one is described in detail.

The $\mathcal{AMS}$math has the following options:

`centertags`	(default) For a split equation, place equation numbers vertically centered on the total height of the equation.
`tbtags`	'Top-or-bottom tags' For a split equation, place equation numbers level with the last (resp. first) line, if numbers are on the right (resp. left).
`sumlimits`	(default) Place the subscripts and superscripts of summation symbols above and below, in displayed equations. This option also affects other symbols of the same type – $\prod$, $\coprod$, $\bigotimes$, $\bigoplus$, and so forth – but excluding integrals (see below).
`nosumlimits`	Always place the subscripts and superscripts of summation-type symbols to the side, even in displayed equations.
`intlimits`	Like sumlimits, but for integral symbols.
`nointlimits`	(default) Opposite of intlimits.
`namelimits`	(default) Like sumlimits, but for certain 'operator names' such as `det`, `inf`, `lim`, `max`, `min`, that traditionally have subscripts placed underneath when they occur in a displayed equation.
`nonamelimits`	Opposite of namelimits.

To use one of these package options, put the option name in the optional argument, e.g., `\usepackage[intlimits]{\amsmath}`. The $\mathcal{AMS}$math also recognises the following options which are normally selected (implicitly or explicitly) through the `documentclass` command, and thus need not be repeated in the option list of the `\usepackage{\amsmath}` statement.

`leqno`	Place equation numbers on the left.
`reqno`	(default) Place equation numbers on the right.
`fleqn`	Position equations at a fixed indent from the left margin rather than centered in the text column. $\mathcal{AMS}$math defines the length `\mathindent` and uses it when the equations have only one tabbing character (`&`).

All math environments are displayed ones, so there is no special inline math.

25 align environments

There are four different align environments, described in the following subsections. Their behaviour is shown in table 15. The code for all align environments was:

```
\begin{<name>}
   <name> &= x & x &= x\\
   <name> &= x & x &= x
\end{<name>}
```

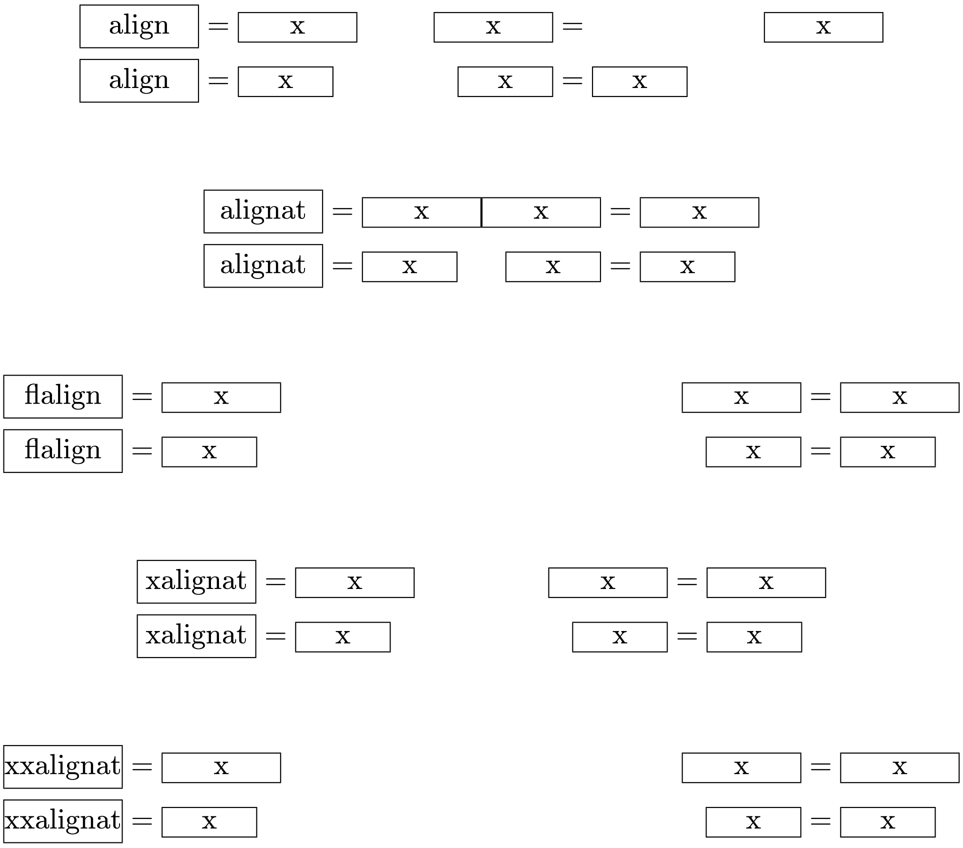


Table 15: Comparison between the different align environments with the same code, where the first three can have an equation number

In difference to the `eqnarray` environment from standard LaTeX (section 3.2), the “three” parts of one equation `expr.-symbol-expr.` are devided by only one ampersand in two parts. In general the ampersand should be before the symbol to get the right spacing, e.g. `y &= x`. Compare the following three equations, the second has a wrong spacing.

$$y = x \qquad (44)$$
$$y =x \qquad (45)$$
$$y = x \qquad (46)$$

```
  y &= x
  y =& x
  y ={}& x
```

25.1 The default `align` environment

The `eqnarray` environment has a not so good spacing between the cells. Writing the equations no. **??** to **??** with the `align` environment gives:

$$\begin{aligned} y &= d && (47)\\ y &= cx + d && (48)\\ y_{12} &= bx^{2} + cx + d && (49)\\ y(x) &= ax^{3} + bx^{2} + cx + d && (50) \end{aligned}$$

The code looks like:

```
\begin{align}
 y & =d\label{eq:IntoSection}\\
 y & =cx+d\\
 y_{12} & =bx^{2}+cx+d\\
 y(x) & =ax^{3}+bx^{2}+cx+d
\end{align}
```

- The `align` environment has an implicit **{rlrl...}** horizontal alignment with a vertical column-alignment, e.g.:

12 3

```
\begin{align*}
 1 & 2 & 3
\end{align*}
```

- A nonumber-version `\begin{align*}...\end{align*}` exists.

- Not numbered single rows are possible with `\nonumber`.

- The `align` environment takes the whole horizontal space if you have more than two columns:

$$\begin{aligned} y &= d & z &= 1 && (51)\\ y &= cx + d & z &= x + 1 && (52)\\ y_{12} &= bx^{2} + cx + d & z &= x^{2} + x + 1 &&\\ y(x) &= ax^{3} + bx^{2} + cx + d & z &= x^{3} + x^{2} + x + 1 && (53) \end{aligned}$$

The code for this example looks like

```
\begin{align}
 y & =d & z & =1\\
 y & =cx+d & z & =x+1\\
 y_{12} & =bx^{2}+cx+d & z & =x^{2}+x+1\nonumber \\
 y(x) & =ax^{3}+bx^{2}+cx+d & z & =x^{3}+x^{2}+x+1
\end{align}
```

25.2 alignat environment

\begin{align}
...
\end{align}

> From now the counting of the equation changes. It is introduced with a foregoing command, which doesn't really make sense, it is only for demonstration:
> `\renewcommand{\theequation}{\thepart-\arabic{equation}}`.

This means "align at several places" and is something like more than two **align** environment side by side. Parameter is the number of the **align** environments, which is not important for the user. The above last **align** example looks like:

$$\begin{aligned} y &= d & z &= 1 && \text{(II-54)}\\ y &= cx+d & z &= x+1 && \text{(II-55)}\\ y_{12} &= bx^2+cx+d & z &= x^2+x+1 &&\\ y(x) &= ax^3+bx^2+cx+d & z &= x^3+x^2+x+1 && \text{(II-56)} \end{aligned}$$

The parameter was 2 and is for the following example 3:

$$\begin{aligned} i_{11} &= 0.25 & i_{12} &= i_{21} & i_{13} &= i_{23} &&\\ i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31} && \text{(II-57)}\\ i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11} && \text{(II-58)} \end{aligned}$$

For this example the code is:

```
\begin{alignat}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22}\quad & i_{32} & =0.15i_{32}\quad & i_{33} & =i_{11}
\end{alignat}
```

With the **alignat** environment one can easily align equations vertically at more than one marker:

$$\begin{alignat}{3}
abc &= xxx &&= xxxxxxxxxxxx &&= aaaaaaaaa \tag{II-59}\\
ab &= yyyyyyyyyyyyyyy &&= yyyy &&= ab \tag{II-60}
\end{alignat}$$

```
\begin{alignat}{3}
   abc &= xxx              &&= xxxxxxxxxxxx  &&= aaaaaaaaa \\
   ab  &= yyyyyyyyyyyyyyy &&= yyyy          &&= ab
\end{alignat}
```

- The **alignat** environment has an implicit {rlrl...rlrl} horizontal alignment with a vertical column alignment.
- A nonumber-version `\begin{alignat*}...\end{alignat*}` exists.
- Not numbered single rows are possible with `\nonumber`.

25.3 flalign environment

\begin{flalign}
...
\end{flalign}

This is the new replacement for the **xalignat** and **xxalignat** environments. It is nearly the same as the **xalignat** environment, only more "out spaced" and "left aligned".

$$\begin{align}
i_{11} &= 0.25 \nonumber\\
i_{21} &= \frac{1}{3}i_{11} \tag{II-61}\\
i_{31} &= 0.33i_{22} \tag{II-62}
\end{align}$$

```
\begin{flalign}
i_{11} & =0.25\nonumber \\
i_{21} & =\frac{1}{3}i_{11}\\
i_{31} & =0.33i_{22}
\end{flalign}
```

As seen, the equations are not really left aligned, when they have only one ampersand. In this case **flalign** has the same behaviour as the **align** environment.

When there are more than one tabbing characters (&), then the equations are really left aligned. This is also an easy way to get an equation with only one ampersand left aligned, see equation II-66 below.

$$\begin{flalign}
i_{11} &= 0.25 & i_{12} &= i_{21} & i_{13} &= i_{23} \nonumber\\
i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31} \tag{II-63}\\
i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11} \tag{II-64}
\end{flalign}$$

The code looks like:

```
\begin{flalign}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22}\quad & i_{32} & =0.15i_{32}\quad & i_{33} & =i_{11}
\end{flalign}
```

This environment can be used to mix centered and left aligned equations without using the document wide valid option `fleqn`.

$$f(x) = \int \frac{1}{x^2}\,dx \tag{II-65}$$

$$f(x) = \int \frac{1}{x^2}\,dx \tag{II-66}$$

Equation II-66 is left aligned in fact of the second tabbing character `&`.

```latex
\begin{align}\label{eq:centered}
 f(x) & = \int\frac{1}{x^2}\,dx
\end{align}

\begin{flalign}\label{eq:leftaligned}
 f(x) & = \int\frac{1}{x^2}\,dx &
\end{flalign}
```

Another case is placing text left aligned, whereas the formulas should be right aligned.

$$\begin{aligned} && 12(x-1)+20(y-3)+14(z-2) &= 0\\ \text{same as} && 6x+10y+7z &= 0 \end{aligned}$$

```latex
\begin{flalign*}
                 && 12(x-1)+20(y-3)+14(z-2)  &= 0\\
\text{same as } &&                6x+10y+7z  &= 0
\end{flalign*}
```

25.4 `xalignat` environment

`\begin{xalignat}`
`...`
`\end{xalignat}`

This is an obsolete macro but still supported by the $\mathcal{AMS}$math package. Same as `alignat` environment, only a little more "out spaced".

$$\begin{alignedat}{3} i_{11} &= 0.25 &\qquad i_{12} &= i_{21} &\qquad i_{13} &= i_{23}\\ i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31} && \text{(II-67)}\\ i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11} && \text{(II-68)} \end{alignedat}$$

The same code looks like:

```latex
\begin{xalignat}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22}\quad & i_{32} & =0.15i_{32}\quad & i_{33} & =i_{11}
\end{xalignat}
```

25.5 xxalignat environment

```
\begin{xxalignat}
...
\end{xxalignat}
```

Like xalignat an obsolete macro but still supported by the $\mathcal{AMS}$math package. Same as align environment, only extremely “out spaced”, therefore no equation number!

$$\begin{alignat*}{3}
i_{11} &= 0.25 &\qquad\qquad i_{12} &= i_{21} &\qquad\qquad i_{13} &= i_{23}\\
i_{21} &= \frac{1}{3}i_{11} & i_{22} &= 0.5i_{12} & i_{23} &= i_{31}\\
i_{31} &= 0.33i_{22} & i_{32} &= 0.15i_{32} & i_{33} &= i_{11}
\end{alignat*}$$

The same code looks like:

```
\begin{xxalignat}{3}
  i_{11} & =0.25 & i_{12} & =i_{21} & i_{13} & =i_{23}\nonumber\\
  i_{21} & =\frac{1}{3}i_{11} & i_{22} & =0.5i_{12}& i_{23} & =i_{31}\\
  i_{31} & =0.33i_{22} & i_{32} & =0.15i_{32} & i_{33} & =i_{11}
\end{xxalignat}
```

25.6 aligned environment

```
\begin{aligned}
...
\end{aligned}
```

In difference to the split environment (section 26.3 on page 58), the aligned environment allows more than one horizontal alignment but has also only one equation number:

$$\begin{aligned}
2x+3 &= 7 & 2x+3-3 &= 7-3\\
2x &= 4 & \frac{2x}{2} &= \frac{4}{2}\\
x &= 2
\end{aligned} \tag{II-69}$$

```
\begin{equation}
\begin{aligned}
    2x+3 &= 7   &      2x+3-3 &= 7-3    \\
    2x   &= 4   & \frac{2x}2 &= \frac42\\
    x    &= 2
\end{aligned}
\end{equation}
```

The aligned environment is similar to the array environment, there exists no starred version and it has only one equation number and has to be part of another math environment, which should be equation environment. The advantage of aligned is the much more better horizontal and vertical spacing.

25.7 Problems

When using one of the align environments, there should be no \\ at the end of the last line, otherwise you’ll get another equation number for this ”empty“ line:

$$\begin{aligned} 2x+3 &= 7 && \text{(II-70)} \\ & && \text{(II-71)} \end{aligned}$$

```
\begin{align}
   2x+3 &= 7\\
\end{align}
```

$$2x+3=7 \qquad \text{(II-72)}$$

```
\begin{align}
   2x+3 &= 7
\end{align}
```

26 Other environments

26.1 gather environment

\begin{gather}
...
\end{gather}

This is like a multi line environment with no special horizontal alignment. All rows are centered and can have an own equation number:

$$\begin{gathered} i_{11} = 0.25 \qquad \text{(II-73)} \\ i_{21} = \frac{1}{3} i_{11} \\ i_{31} = 0.33 i_{22} \qquad \text{(II-74)} \end{gathered}$$

For this example the code looks like:

```
\begin{gather}
  i_{11} = 0.25\\
  i_{21} = \frac{1}{3}i_{11}\nonumber\\
  i_{31} =0.33i_{22}
\end{gather}
```

- The `gather` environment has an implicit `{c}` horizontal alignment with no vertical column alignment. It is just like an one column array/table.
- A nonumber-version `\begin{gather*}...\end{gather*}` exists. Look at section 26.3 on page 58 for an example.

26.2 multline environment

\begin{multline}
...
\end{multline}

This is also like a multi line[19] environment with a special vertical alignment. The **first** row is **left aligned**, the second and all following ones except the last one are **centered** and the **last** line is **right aligned**. It is often used to write extremely long formulas:

```
\begin{multline}
  A = \lim _{n\rightarrow \infty }\Delta x\left( a^{2}+\left( a^{2}+2a\Delta x
         +\left( \Delta x\right) ^{2}\right)\right.\\
  +\left( a^{2}+2\cdot 2a\Delta x+2^{2}\left( \Delta x\right) ^{2}\right)\\
  +\left( a^{2}+2\cdot 3a\Delta x+3^{2}\left( \Delta x\right) ^{2}\right)\\
```

[19] It is no typo, the name of the environment is `multline`, no missing i here!

```
  + \ldots\\
  \left.+\left( a^{2}+2\cdot (n-1)a\Delta x +(n-1)^{2}\left( \Delta x\right)^{2}\right) \right)\\
  = \frac{1}{3}\left( b^{3}-a^{3}\right)
\end{multline}
```

$$\begin{multline} A=\lim_{n\to\infty}\Delta x\left(a^2+\left(a^2+2a\Delta x+(\Delta x)^2\right)\right.\\ +\left(a^2+2\cdot 2a\Delta x+2^2\left(\Delta x\right)^2\right)\\ +\left(a^2+2\cdot 3a\Delta x+3^2\left(\Delta x\right)^2\right)\\ +\ldots\\ \left.+\left(a^2+2\cdot(n-1)a\Delta x+(n-1)^2\left(\Delta x\right)^2\right)\right)\\ =\frac{1}{3}\left(b^3-a^3\right) \tag{II-75}\end{multline}$$

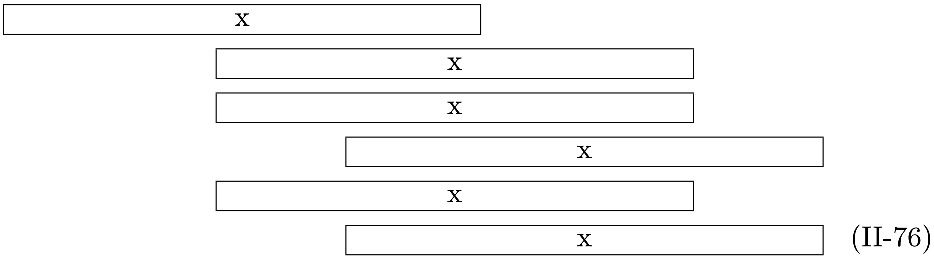


Figure 1: **multline** Alignment demo (the fourth row is shifted to the right with **\shoveright**)

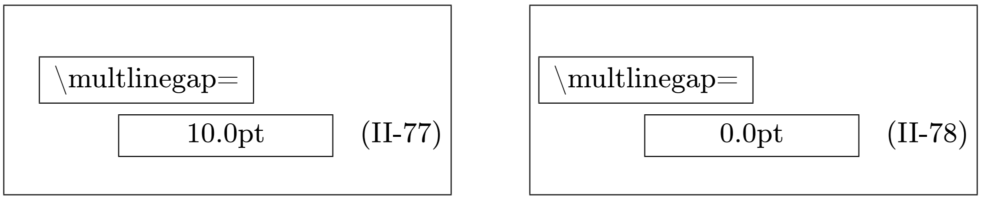


Figure 2: Demonstration of **\multlinegap** (default is 0pt)

- A nonumber-version **\begin{multline*}**...**\end{multline*}** exists.
- By default only the last line (for right equation numbers) or the first line (for left equation numbers) gets a number, the others can't.

- The alignment of a single line can be changed with the command `\shoveright` (figure 1 on the preceding page)
- The first line and the last line have a small gap to the text border.[20] See figure 2, where the length of `\multlinegap` is set to `0pt` for the right one.

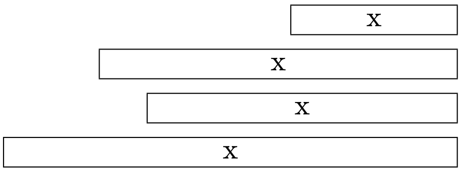
26.3 `split` environment

```
\begin{split}
...
\end{split}
```

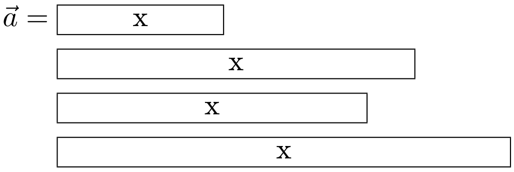
> From now the counting of the equation changes. It is introduced with a foregoing command, which doesn't really make sense, it is only for demonstration:
>
> ```
> \makeatletter
> \@removefromreset{equation}{section}
> \makeatother
> ```

The `split` environment is like the `multline` or `array` environment for equations longer than the column width. Just like the array environment and in contrast to `multline`, `split` can only be used as **part of another environment**. `split` itself has no own numbering, this is given by the other environment. Without an ampersand all lines in the split environment are right-aligned and can be aligned at a special point by using an ampersand. In difference to the `aligned` environment (section 25.6 on page 55), the `split` environment permits more than one horizontal alignment.

Important is that the split environment has another behaviour when used inside one of the "old" LaTeX environments `\[...\]` or `\begin{equation} ... \end{equation}`, in this case more than one horizontal alignment tabs are possible.



```
\[
\begin{split}
   \framebox[0.35\columnwidth]{x}\\
   \framebox[0.75\columnwidth]{x}\\
   \framebox[0.65\columnwidth]{x}\\
   \framebox[0.95\columnwidth]{x}
\end{split}
\]
```



```
\[
\begin{split}
   \vec{a} = {}&\framebox[0.35\columnwidth]{x}\\
             &\framebox[0.75\columnwidth]{x}\\
             &\framebox[0.65\columnwidth]{x}\\
             &\framebox[0.95\columnwidth]{x}
\end{split}
\]
```

[20] When the first (numbers left) or last line (numbers right) has an equation number then `\multlinegap` is not used for these ones, only for the line without a number.

The following example shows the `split` environment as part of the `equation` environment:

$$\begin{split}
A_{1} & = \left| \int_{0}^{1}(f(x)-g(x))dx\right| +\left| \int_{1}^{2}(g(x)-h(x))dx\right| \\
& = \left| \int_{0}^{1}(x^{2}-3x)dx\right| +\left| \int_{1}^{2}(x^{2}-5x+6)dx\right| \\
& = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right|_{0}^{1}+\left| \frac{x^{3}}{3}-\frac{5}{2}x^{2}+6x\right|_{1}^{2}\\
& = \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-\left(\frac{1}{3}-\frac{5}{2}+6\right) \right| \\
& = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right| =\frac{7}{6}+\frac{5}{6}=2\,\textrm{FE}
\end{split} \qquad \text{(II-79)}$$

```
\begin{equation}
  \begin{split}
    A_{1} & = \left| \int _{0}^{1}(f(x)-g(x))dx\right| +\left|
   \int _{1}^{2}(g(x)-h(x))dx\right| \\
          & = \left| \int _{0}^{1}(x^{2}-3x)dx\right| +\left|
   \int _{1}^{2}(x^{2}-5x+6)dx\right| \\
          & = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right| _{0}^{1}+
   \left| \frac{x^{3}}{3}-
                \frac{5}{2}x^{2}+6x\right| _{1}^{2}\\
          & = \left| \frac{1}{3}-\frac{3}{2}\right| +\left|
   \frac{8}{3}-\frac{20}{2}+12-
                \left( \frac{1}{3}-\frac{5}{2}+6\right) \right| \\
          & = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}
   \right| =\frac{7}{6}+\frac{5}{6}=2\, \textrm{FE}
  \end{split}
\end{equation}
```

The same using the `array` environment with `{rl}`-alignment instead of `split` gives same horizontal alignment but another vertical spacing[21] and the symbols only in scriptsize and not textsize:[22]

$$\begin{array}{rl}
A_{1} & = \left| \int_{0}^{1}(f(x)-g(x))dx\right| +\left| \int_{1}^{2}(g(x)-h(x))dx\right| \\
& = \left| \int_{0}^{1}(x^{2}-3x)dx\right| +\left| \int_{1}^{2}(x^{2}-5x+6)dx\right| \\
& = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right|_{0}^{1}+\left| \frac{x^{3}}{3}-\frac{5}{2}x^{2}+6x\right|_{1}^{2}\\
& = \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-\left(\frac{1}{3}-\frac{5}{2}+6\right) \right| \\
& = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right| =\frac{7}{6}+\frac{5}{6}=2\,\textrm{FE}
\end{array} \qquad \text{(II-80)}$$

- There exists no star version (`\begin{split*}`) of the `split` environment.

[21] Can be changed with `\renewcommand\arraystretch{1.5}`

[22] See section 12 on page 37

26.4 Specials for `multline` and `split` environments

With the multline environment the equation 28 on page 28 looks like:

$$\frac{1}{2}\Delta(f_{ij}f^{ij}) = 2\left(\sum_{i<j}\chi_{ij}(\sigma_i-\sigma_j)^2 + f^{ij}\nabla_j\nabla_i(\Delta f)+\right.$$

$$\left. + \nabla_k f_{ij}\nabla^k f^{ij} + f^{ij}f^k\left[2\nabla_i R_{jk} - \nabla_k R_{ij}\right]\right) \tag{II-81}$$

which is again a bad typesetting because of the two unequal parentheses. Each one has a size which is correct for the line but not for the whole formula. LaTeX accepts only pairs of parentheses for one line and has an "empty" parentheses, the dot "`\left.`" or "`\right.`" to get only one of the "pair". There are different solutions to get the right size of the parentheses. One of them is to use the `\vphantom` command, which reserves the vertical space without any horizontal one, like a vertical rule without any thickness. The sum symbol from the first line is the biggest one and responsible for the height, so this one is the argument of `\vphantom` which has to be placed anywhere.

$$\frac{1}{2}\Delta(f_{ij}f^{ij}) = 2\left(\sum_{i<j}\chi_{ij}(\sigma_i-\sigma_j)^2 + f^{ij}\nabla_j\nabla_i(\Delta f)+\right.$$

$$\left.+\nabla_k f_{ij}\nabla^k f^{ij} + f^{ij}f^k\left[2\nabla_i R_{jk} - \nabla_k R_{ij}\right]\vphantom{\sum_{i<j}}\right) \tag{II-82}$$

```
\begin{multline}
\frac{1}{2}\Delta(f_{ij}f^{ij})=
 2\left(\sum_{i<j}\chi_{ij}(\sigma_{i}-
 \sigma_{j})^{2}+f^{ij}\nabla_{j}\nabla_{i}(\Delta f)+\right.\\
 \left.+\nabla_{k}f_{ij}\nabla^{k}f^{ij}+
 f^{ij}f^{k}\left[2\nabla_{i}R_{jk}-
 \nabla_{k}R_{ij}\right]\vphantom{\sum_{i<j}}\right)
\end{multline}
```

Instead of using the `\vphantom` command it is also possible to use fixed-width parentheses, which is described in section 8 on page 26.

26.5 `cases` environment

This gives support for an often used mathematical construct. You can also choose the more than once described way to convert some text into math, like

```
$x=\begin{cases}
 0 & \text{if A=...}\\
 1 & \text{if B=...}\\
 x & \textrm{this runs with as much text as you like,
      but without an automatic linebreak, it runs out
      of page....}
\end{cases}$
```

which gives equation II-83. It is obvious what the problem is.

$$x \quad = \quad \begin{cases} 0 & \text{if A=...}\\ 1 & \text{if B=...}\\ x & \text{this runs with as much text as you like, but without a linebreak, it runs out of page....} \end{cases} \tag{II-83}$$

In this case it is better to use a parbox for the text part with a `flushleft` command for a better view.

$$x = \begin{cases} 0 & \text{if A=...}\\ 1 & \text{if B=...}\\ x & \parbox{5cm}{\flushleft this runs with as much text as you like, but without an automatic linebreak, it runs out of page....} \end{cases} \tag{II-84}$$

```
\begin{equation}
x=\begin{cases}
  0 & \text{if A=...}\\
  1 & \text{if B=...}\\
  x & \parbox{5cm}{%
      \flushleft%
      this runs with as much text as you like,
      but without an automatic linebreak,
      it runs out of page....}%
\end{cases}
\end{equation}
```

From now the counting of the equation changes. It is introduced with a foregoing command, which doesn't really make sense, it is only for demonstration:

```
\renewcommand\theequation{\arabic{equation}}
```

26.6 Matrix environments

All matrix environments can be nested and an element may also contain any other math environment, so that very complex structures are possible.

\Vmatrix	$\begin{Vmatrix} a & b \\ c & d \end{Vmatrix}$	\Bmatrix	$\begin{Bmatrix} a & b \\ c & d \end{Bmatrix}$	\matrix	$\begin{matrix} a & b \\ c & d \end{matrix}$
\vmatrix	$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$	\bmatrix	$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$	\pmatrix	$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$
				\smallmatrix	$\begin{smallmatrix} a & b \\ c & d \end{smallmatrix}$

Table 16: Matrix environments

By default all cells have a centered alignment, which is often not the best when having different decimal numbers or plus/minus values. Changing the alignment to right is possible with

matrix
vmatrix
Vmatrix
bmatrix
Bmatrix
pmatrix
smallmatrix

```
\makeatletter
\def\env@matrix{\hskip -\arraycolsep
  \let\@ifnextchar\new@ifnextchar
  \array{*\c@MaxMatrixCols r}}
\makeatother
```

The special matrix environment `smallmatrix`, which decreases horizontal and vertical space is typeset in scriptstyle. The `smallmatrix` environment makes some sense in the inline mode to decrease the line height. For dots over several columns look for `\hdotsfor` in the following section.

27 Vertical whitespace

See section 11.5 on page 35 for the lengths which control the vertical whitespace. There is no difference to $\mathcal{AMS}$math.

28 Dots

In addition to section 13 on page 39 $\mathcal{AMS}$math has two more commands for dots: `\dddot{...}`[23] and `\ddddot{...}`

`$\dddot{y}$`: $\dddot{y}$

`$\ddddot{y}$`: $\ddddot{y}$

Another interesting dot command is `\hdotsfor` with the syntax:

```
\hdotsfor[<spacing factor>]{<number of columns>}
```

With the spacing factor the width of the dots can be stretched or shrinked. The number of columns allows a continuing dotted line over more columns. Equation 85 shows the definition of a tridiagonal matrix.

[23] already mentioned in section 14

$$\underline{A}=\left[\begin{array}{ccccccc}
a_{11} & a_{12} & 0 & \ldots & \ldots & \ldots & 0\\
a_{21} & a_{22} & a_{23} & 0 & \ldots & \ldots & 0\\
0 & a_{32} & a_{33} & a_{34} & 0 & \ldots & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
\hdotsfor{7}\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & \ldots & 0 & a_{n-2,n-3} & a_{n-2,n-2} & a_{n-2,n-1} & 0\\
0 & \ldots & \ldots & 0 & q_{n-1,n-2} & a_{n-1,n-1} & a_{n-1,n}\\
0 & \ldots & \ldots & \ldots & 0 & a_{n,n-1} & a_{nn}
\end{array}\right] \tag{85}$$

```
\begin{equation}
\underline{A}=\left[\begin{array}{ccccccc}
a_{11} & a_{12} & 0 & \ldots & \ldots & \ldots & 0\\
a_{21} & a_{22} & a_{23} & 0 & \ldots & \ldots & 0\\
0 & a_{32} & a_{33} & a_{34} & 0 & \ldots & 0\\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
\hdotsfor{7}\cr\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\
0 & \ldots & 0 & a_{n-2,n-3} & a_{n-2,n-2} & a_{n-2,n-1} & 0\\
0 & \ldots & \ldots & 0 & q_{n-1,n-2} & a_{n-1,n-1} & a_{n-1,n}\\
0 & \ldots & \ldots & \ldots & 0 & a_{n,n-1} & a_{nn}
\end{array}\right]
\end{equation}
```

29 fraction commands

29.1 Standard

Additional to the font size problem described in subsection 2.2 on page 11 $\mathcal{AMS}$math supports some more commands for fractions. The `\frac` command described in [7], does no more exists in $\mathcal{AMS}$math.

- The global fraction definition has five parameters

```
\genfrac{<left delim>}{<right  delim>}{<thickness>}{<mathstyle>}{<nominator>}{<denominator>}
```

  where thickness can have any length with a valid unit like `genfrac{}{}{1pt}{}{x^2+x+1}{3x-2}` → $\genfrac{}{}{1pt}{}{x^2+x+1}{3x-2}$

- `\cfrac` (continued fraction) which is by default set in the display mathstyle and useful for fractions like

$$\cfrac{1}{\sqrt{2}+\cfrac{1}{\sqrt{3}+\cfrac{1}{\sqrt{4}+\cfrac{1}{\dots}}}} \tag{86}$$

which looks with the default `\frac` command like

$$\frac{1}{\sqrt{2}+\frac{1}{\sqrt{3}+\frac{1}{\sqrt{4}+\frac{1}{\dots}}}} \tag{87}$$

where the mathstyle decreases for every new level in the fraction. The `\cfrac` command can be called with an optional parameter which defines the placing of the nominator, which can be `[l]`eft, `[r]`ight or `[c]`enter (the default - see equation 86):

$$\cfrac[l]{1}{\sqrt{2}+\cfrac[l]{1}{\sqrt{3}+\cfrac[l]{1}{\sqrt{4}+\cfrac[l]{1}{\dots}}}} \qquad \cfrac[r]{1}{\sqrt{2}+\cfrac[r]{1}{\sqrt{3}+\cfrac[r]{1}{\sqrt{4}+\cfrac[r]{1}{\dots}}}}$$

- `\dfrac` which takes by default the displaystyle, so that fractions in inline mode $\dfrac{1}{2}$ have the same size than in display mode.
- `\tfrac` (vice versa to `\dfrac`) which takes by default the scriptstyle, so that fractions in display mode have the same size than in inline mode.

$\tfrac{2}{3}$	`\tfrac{2}{3}`
$\dfrac{2}{3}$	`\frac{2}{3}`

29.2 Binoms

`\binom`
`\dbinom`
`\tbinom`

They are like fractions without a rule and its syntax is different to the `\choose` command from standard LaTeX (see section 2.2 on page 11). AMSmath provides three different commands for binoms just like the ones for fractions.

30 Roots

`\leftroot`
`\uproot`

The typesetting for roots is sometimes not the best. Some solutions for better typesetting are described in section 7 on page 25 for standard LaTeX. AMSmath has some more commands for the n-th root:

Command	Inlinemath	Displaymath
\binom{m}{n}	$\binom{m}{n}$	$\dbinom{m}{n}$
\dbinom{m}{n}	$\dbinom{m}{n}$	$\dbinom{m}{n}$
\tbinom{m}{n}	$\tbinom{m}{n}$	$\tbinom{m}{n}$

Table 17: binom commands

```
\sqrt[\leftroot{<number>}\uproot{<number>}<root>]{< ... >}
```

`<number>` indicates a value for the points[24] of which the root can be adjusted to the left and/or to the top, e.g.: $\sqrt[k_n]{a}$ (`$\sqrt[k_n]{a}$`) has a too deep exponent, whereas $\sqrt[\uproot{2}k_n]{a}$ `$\sqrt[\uproot{2}k_n]{a}$` looks nicer.

30.1 Roots with \smash command

\smash

The default for a root with λ_{k_i} as root argument looks like $\sqrt{\lambda_{k_i}}$, which maybe not the best typesetting. It is possible to reduce the lowest point of the root to the baseline with the `\smash` command: $\sqrt{\lambda_{k_i}} \xrightarrow{\text{with \textbackslash smash}} \sqrt{\smash[b]{\lambda_{k_i}}}$

The syntax of the with the $\mathcal{A}\kern-.1667em\lower.5ex\hbox{$\mathcal{M}$}\kern-.125em\mathcal{S}$math package renewed `\smash` command[25] is

```
\smash[<position>]{<argument>}
```

The optional argument for the position can be:

t keeps the bottom and annihilates the top

b keeps the top and annihilates the bottom

tb annihilates top and bottom (the default)

31 Accents

With the macro `\mathaccent` it is easy to define new accent types, for example

```
\def\dotcup{$\mathaccent\cdot\cup$}
```

$\dot{\cup}$

Overwriting of two symbols is also possible:

$\curvearrowleft\curvearrowright\curvearrowleft\!\!\!\curvearrowright$

[24] In PostScript units (bp – Big Points).

[25] In `latex.ltx` `\smash` is defined without an optional argument.

In this case the second symbol has to be shifted to left for a length of $5mu$ (mu: math unit).

```
\def\curvearrowleftright{%
 \ensuremath{%
  \mathaccent\curvearrowright{\mkern-5mu\curvearrowleft}%
 }%
}
```

For other possibilities to define new accent see section 47 on page 90.

32 `\mod` command

The modulo command is in standard LaTeX not an operator, though it is often used in formulas. $\mathcal{AMS}$math provides two (three) different commands for modulo, which are listed in tabular 18.

- They all insert some useful space before and behind the mod-operator.

a\mod{n^2}=b	→	$a \mod n^2 = b$
a\pmod{n^2}=b	→	$a \pmod{n^2} = b$
a\pod{n^2}=b	→	$a \pod{n^2} = b$

Table 18: The modulo commands and their meaning

33 Equation numbering

\numberwithin

See section 3.3 on page 16 for equation numbering. It is mostly the same, only one command is new to $\mathcal{AMS}$math. If you want a numbering like "47" then write in the preamble or like this example anywhere in your doc:

```
\numberwithin{equation}{section}
```

From now the numbering looks like equation 47 on page 51. For the book-class you can get the same for chapters.

If you want to get rid of the parentheses then write in preamble:

```
\makeatletter
\def\tagform@#1{\maketag@@@{\ignorespaces#1\unskip\
  @@italiccorr}}
\makeatother
```

Now the following four subequation numbers have no parentheses.

33.1 Subequations

Amsmath supports this with the environment `subequation`. For example:

$$
\begin{align*}
y &= d \tag*{33.88a}\\
y &= cx+d \tag*{33.88b}\\
y &= bx^{2}+cx+d \tag*{33.88c}\\
y &= ax^{3}+bx^{2}+cx+d \tag*{33.88d}
\end{align*}
$$

```
\begin{subequations}
\begin{align}
y & =  d\\
y & =  cx+d\\
y & =  bx^{2}+cx+d\\
y & =  ax^{3}+bx^{2}+cx+d
\end{align}
\end{subequations}
```

Inside of subequations only complete other environments (`\begin{...}` ... `\end{...}`) are possible.

```
\renewcommand{\theequation}{%
  \theparentequation{}-\arabic{equation}%
}
```

$$
\begin{align*}
y &= d \tag{33.89-1}\\
y &= cx+d \tag{33.89-2}\\
y &= bx^{2}+cx+d \tag{33.89-3}\\
y &= ax^{3}+bx^{2}+cx+d \tag{33.89-4}
\end{align*}
$$

A ref to a subequation is possible like the one to equation 33.89-2. The environment chooses the same counter "`equation`" but saves the old value into "`parentequation`".

It is also possible to place two equations side by side with counting as subfigures:

$$y=f(x) \qquad (33.90\text{a}) \qquad\qquad y=f(z) \qquad (33.90\text{b})$$

In this case, the $\mathcal{AMS}$math internal subfigure counter cannot be used and an own counter has to be defined:

```
\newcounter{mySubCounter}
\newcommand{\twocoleqn}[2]{
    \setcounter{mySubCounter}{0}%
    \let\OldTheEquation\theequation%
    \renewcommand{\theequation}{\OldTheEquation\alph{mySubCounter}}%
    \noindent%
    \begin{minipage}{.49\textwidth}
          \begin{equation}\refstepcounter{mySubCounter}
            #1
        \end{equation}
    \end{minipage}\hfill%
    \addtocounter{equation}{-1}%
    \begin{minipage}{.49\textwidth}
        \begin{equation}\refstepcounter{mySubCounter}
            #2
        \end{equation}
    \end{minipage}%
    \let\theequation\OldTheEquation
}
[ ... ]
\twocoleqn{y=f(x)}{y=f(z)}
```

34 Labels and tags

\tag

For the `\label` command see section 3.4 on page 18, it is just the same behaviour. $\mathcal{A}\mathcal{M}\mathcal{S}$math allows to define own single “equation numbers” with the `\tag` command.

$$\begin{align} f(x) &= a \tag{linear}\\ g(x) &= dx^2+cx+b \tag{quadratic}\\ h(x) &= \sin x \tag*{trigonometric} \end{align}$$

```
\begin{align}
f(x) & =a\tag{linear}\label{eq:linear}\\
g(x) & =dx^{2}+cx+b\tag{quadratic}\label{eq:quadratic}\\
h(x) & =\sin x\tag*{trigonometric}
\end{align}
```

- The `\tag` command is also possible for unnumbered equations, LaTeX changes the behaviour when a tag is detected.
- There exists a star version `\tag{*}{...}`, which supresses any annotations like parentheses for equation numbers.
- There exists two package options for tags, `ctagsplit` and `righttag` (look at the beginning of this part on page 49).

35 Limits

By default the sum/prod has the limits above/below and the integral at the side. To get the same behaviour for all symbols which can have limits load the packags $\mathcal{A}\mathcal{M}\mathcal{S}$math in the preamble as

```
\usepackage[sumlimits,intlimits]{amsmath}
```

There exists also options for the vice versa (see page 49). See also section 41 for the additional commands `\underset` and `\overset`.

35.1 Multiple limits

For general information about limits read section 2.1 on page 10. Standard LaTeX provides the `\atop` command for multiple limits (section 6.1 on page 24). $\mathcal{A}\mathcal{M}\mathcal{S}$math has an additional command for that, which can have several lines with the following syntax:

\substack
`\begin{Sb}`
...
`\end{Sb}`
`\begin{Sp}`
...
`\end{Sp}`

```
\substack{...\\...\\...}
```

The environments described in [7]

```
\begin{Sb} ... \end{Sb}
\begin{Sp} ... \end{Sp}
```

are obsolete and no more part of $\mathcal{A}\mathcal{M}\mathcal{S}$math.

The example equation 21 on page 25 with the `\substack` command looks like:

$$\sum_{\substack{1\le i\le p\\ 1\le j\le q\\ 1\le k\le r}} a_{ij}b_{jk}c_{ki} \tag{35.1}$$

Insert these limits in the following way:

```
\begin{equation}
 \sum_{%
  \substack{1\le i\le p\\
   1\le j\le q\\
   1\le k\le r}
  }%
 a_{ij}b_{jk}c_{ki}
\end{equation}
```

35.2 Problems

There are still some problems with limits and the following math expression. For example:

$$X = \sum_{1\le i\le j\le n} X_{ij}$$

```
\[
X = \sum_{1\le i\le j\le n}X_{ij}
\]
```

does not look nice because of the long limit. Using a `\makebox` also does not really solve the problem, because `\makebox` is in TEX horizontal mode and knows nothing about the appropriate math font size, because limits have a smaller font size. It is better to define a `\mathclap` macro, similiar to the two macros `\llap` and `\rlap` and uses the also new defined `\mathclap` macro:

```
\def\mathllap{\mathpalette\mathllapinternal}
\def\mathllapinternal#1#2{%
 \llap{$\mathsurround=0pt#1{#2}$}%
}
\def\clap#1{\hbox to 0pt{\hss#1\hss}}
\def\mathclap{\mathpalette\mathclapinternal}
\def\mathclapinternal#1#2{%
 \clap{$\mathsurround=0pt#1{#2}$}%
}
\def\mathrlap{\mathpalette\mathrlapinternal}
\def\mathrlapinternal#1#2{%
 \rlap{$\mathsurround=0pt#1{#2}$}%
}
```

Now we can write limits which have a boxwidth of 0pt and the right font size and the following math expression appears just behind the symbol:

$$X = \sum_{\mathclap{1\le i\le j\le n}} X_{ij}$$

```
\[
X = \sum_{\mathclap{1\le i\le j\le n}}X_{ij}
\]
```

35.3 `\sideset`

This is a command for a very special purpose, to combine over/under limits with superscript/subscripts for the sum-symbol. For example: it is not possible to place the prime for the equation 35.2 near to the sum symbol, because it becomes an upper limit when writing without an preceeding `{}`. `\sideset`

$$\sum_{\substack{n<k\\ n\ \mathrm{odd}}} {}'nE_n \tag{35.2}$$

The command `\sideset` has the syntax

```
\sideset{<before>}{<behind>}
```

It can place characters on all four corners of the sum-symbol:

$$\sideset{_{LowerLeft}^{UpperLeft}}{_{LowerRight}^{UpperRight}}\sum_{B}^{T}$$

```
\[
\sideset{_{LowerLeft}^{UpperLeft}}{_{LowerRight}^{UpperRight
  }}\sum_{B}^{T}
\]
```

Now it is possible to write the equation 35.2 in a proper way with the command `\sideset{}{'}` before the sum symbol:

$$\sideset{}{'}\sum_{\substack{n<k\\ n\ \text{odd}}} nE_n \tag{35.3}$$

36 Operator names

\operatorname

By default variables are written in italic and operator names in upright mode, like $y = \sin(x)$.[26] This happens only for the known operator names, but creating a new one is very easy with:

```
\newcommand{\mysin}{\operatorname{mysin}}
```

Now `\mysin` is also written in upright mode $y = \operatorname{mysin}(x)$ and with some additional space before and behind.

It is obvious, that only those names can be defined as new operator names which are not commands in another way. Instead of using the new definition as an operator, it is also possible to use the text mode. But it is better to have all operators of the same type, so that changing the style will have an effect for all operators.

\operatornamewithlimi

The new defined operator names cannot have limits, only superscript/subscript is possible. `amsopn.sty` has an additional command `\operatornamewithlimits`, which supports over/under limits like the one from `\int` or `\sum`.

\mathop

It is also possible to use the macro `\mathop` to declare anything as operator, like

$${}_1\mathrm{B}$$

```
\[ \sideset{_1}{}{\mathop{\mathrm{B}}} \]
```

With this definition it is possible to use `\sideset` for a forgoing index, which is only possible for an operator.

For a real LaTeX definition have a look at section 16 on page 42.

37 Text in math mode

If you need complex structures between formulas, look also at section 71.

37.1 `\text` command

`\text`
`\mbox`
`\textnormal`
`\mathrm`

This is the equivalent command to `\mathrm` or `\mbox` from the standard LaTeX (section 9 on page 30) with the exception, that `\mathrm` always uses the roman font and `\text` the actual one and that the font size is different when used in super- and subscript.

For example: $\boxed{f(x)=x\quad\text{this was math}}$.

$$A^{\text{text}}_{\text{text}}\qquad A^{\text{text}}_{\text{text}}\qquad A^{\text{text}}_{\text{text}}\qquad A^{\mathrm{text}}_{\mathrm{text}}$$

```
$\boxed{f(x)=x\quad\text{this was math}}$

{\sffamily\huge
$A^{\mbox{text}}_{\mbox{text}}$\quad
$A^{\text{text}}_{\text{text}}$\quad
$A^{\textnormal{text}}_{\textnormal{text}}$\quad
$A^{\mathrm{text}}_{\mathrm{text}}$
}
```

The `\text` macro can be used at any place and can be in some cases a better solution as `\intertext` (see section 37.2).

$$\begin{aligned} &12(x-1)+20(y-3)+14(z-2)=0\\ \text{and}\quad &6x+10y+7z=0\end{aligned}$$

$$12(x-1)+20(y-3)+14(z-2)=0 \qquad (37.1)$$

$$\text{and}\qquad 6x+10y+7z=0 \qquad (37.2)$$

```
\begin{flalign*}
            && 12(x-1) + 20(y-3) + 14(z-2) & = 0 &&\\
  \text{and} &&                6x + 10y + 7z & = 0  &&
\end{flalign*}

\begin{align}
           && 12(x-1) + 20(y-3) + 14(z-2) & = 0\\
  \text{and} &&                6x + 10y + 7z & = 0
\end{align}
```

[26] See section 16 on page 42, where all for standard LaTeX known operator names are listed. Package AMSmath has some more (see documentation).

37.2 \intertext command

This is useful when you want to place some text between two parts of math stuff without leaving the math mode, like the name ”intertext” says. For example we write the equation II-79 on page 59 with an additional command after the second line.

$$\begin{aligned}
A_1 &= \left|\int_0^1 (f(x)-g(x))dx\right| + \left|\int_1^2 (g(x)-h(x))dx\right| \\
&= \left|\int_0^1 (x^2-3x)dx\right| + \left|\int_1^2 (x^2-5x+6)dx\right|
\end{aligned}$$

Now the limits of the integrals are used

$$\begin{aligned}
&= \left|\frac{x^3}{3}-\frac{3}{2}x^2\right|_0^1 + \left|\frac{x^3}{3}-\frac{5}{2}x^2+6x\right|_1^2 \\
&= \left|\frac{1}{3}-\frac{3}{2}\right| + \left|\frac{8}{3}-\frac{20}{2}+12-\left(\frac{1}{3}-\frac{5}{2}+6\right)\right| \\
&= \left|-\frac{7}{6}\right| + \left|\frac{14}{3}-\frac{23}{6}\right| = \frac{7}{6}+\frac{5}{6} = 2\,\mathrm{FE}
\end{aligned}$$

The code looks like:

```
\begin{equation}
  \begin{split}
    A_{1} & = \left| \int _{0}^{1}(f(x)-g(x))dx\right| +\left| \int _{1}^{2}(g(x)-h(x))dx\right| \\
          & = \left| \int _{0}^{1}(x^{2}-3x)dx\right| +\left| \int _{1}^{2}(x^{2}-5x+6)dx\right| \\
          \intertext{Now the limits of the integrals are used}
          & = \left| \frac{x^{3}}{3}-\frac{3}{2}x^{2}\right| _{0}^{1}+\left| \frac{x^{3}}{3}-
              \frac{5}{2}x^{2}+6x\right| _{1}^{2}\\
          & = \left| \frac{1}{3}-\frac{3}{2}\right| +\left| \frac{8}{3}-\frac{20}{2}+12-
              \left( \frac{1}{3}-\frac{5}{2}+6\right) \right| \\
          & = \left| -\frac{7}{6}\right| +\left| \frac{14}{3}-\frac{23}{6}\right| =\frac{7}{6}+
              \frac{5}{6}=2\, \textrm{FE}
  \end{split}
\end{equation}
```

Writing very long text is possible with using a `parbox`, see section 9 on page 30 for an example with `\textrm`, which behaves in the same way as `\text`.

38 Extensible arrows

\xrightarrow
\xleftarrow
\xmapsto

To write something like $\xrightarrow[\text{below}]{\text{above the arrow}}$ you can use the following macro

```
$\xrightarrow[\text{below}]{\text{above the arrow}}$
```

and the same with `\xleftarrow`. You can define your own extensible arrow macros if you need other than these two predefined ones. To get a doublelined extensible arrow like `$\Longleftrightarrow$` ($\Longleftrightarrow$) but with the same behaviour than an extensible one, write in preamble

```
\newcommand\xLongLeftRightArrow[2][]{%
    \ext@arrow 0055{\LongLeftRightArrowfill@}{#1}{#2}}
\def\LongLeftRightArrowfill@{%
 \arrowfill@\Leftarrow\Relbar\Rightarrow}
\newcommand\xlongleftrightarrow[2][]{%
    \ext@arrow 0055{\longleftrightarrowfill@}{#1}{#2}}
\def\longleftrightarrowfill@{%
 \arrowfill@\leftarrow\relbar\rightarrow}
```

The three parts `\Leftarrow\Relbar\Rightarrow` define left|middle|right of the arrow, where the middle part would be stretched in a way that the arrow is at least as long as the text above and/or below it. This macro has one optional and one standard parameter. The optional one is written below and the standard above this arrow. Now we can write

```
$\xLongLeftRightArrow[\text{below}]{\text{above the arrow}}$
$\xlongleftrightarrow[\text{below}]{\text{above the arrow}}$
```

to get $\xLeftrightarrow[\text{below}]{\text{above the arrow}}$ or $\xleftrightarrow[\text{below}]{\text{above the arrow}}$. The "number" 0055 after `\ext@arrow` defines the position relative to the extended error and is not a number but four parameters for additional space in the math unit `mu`.

```
\def\mapstofill@{%
  \arrowfill@{\mapstochar\relbar}\relbar\rightarrow}
\newcommand*\xmapsto[2][]{%
  \ext@arrow <four digits>\mapstofill@{#1}{#2}}
```

`$\ext@arrow 0000$` $\xmapsto[\boxed{\text{under}}]{\boxed{\text{over}}}$

`$\ext@arrow 9000$` $\xmapsto[\qquad\boxed{\text{under}}]{\qquad\boxed{\text{over}}}$

`$\ext@arrow 0900$` $\xmapsto[\boxed{\text{under}}\qquad]{\boxed{\text{over}}\qquad}$

`$\ext@arrow 0009$` $\xmapsto[\boxed{\text{under}}]{\boxed{\text{over}}}$

`$\ext@arrow 0090$` $\xmapsto[\boxed{\text{under}}]{\boxed{\text{over}}}$

`$\ext@arrow 0099$` $\xmapsto[\boxed{\text{under}}]{\boxed{\text{over}}}$

`$\ext@arrow 9999$` $\xmapsto[\boxed{\text{under}}]{\boxed{\text{over}}}$

- 1st digit: space left
- 2nd digit: space right
- 3rd digit: space left and right
- 4th digit: space relativ to the tip of the “arrow”

The two macros `\xrightarrow` and `\xleftarrow` are defined as:

```
\newcommand{\xrightarrow}[2][]{\ext@arrow 0359\
  rightarrowfill@{#1}{#2}}
\newcommand{\xleftarrow}[2][]{\ext@arrow 3095\leftarrowfill@
  {#1}{#2}}
```

39 Frames

`\boxed`

$\mathcal{A}\mathcal{M}\mathcal{S}$math knows the macro `\boxed` which can be used for inline $a\boxed{b+c}$ and displayed math expressions:

$$\boxed{f(x)=\int_1^{\infty}\frac{1}{x^2}\,dt=1} \tag{39.1}$$

```
\begin{align}
\boxed{f(x)=\int_1^{\infty}\frac{1}{x^2}\,dt=1}
\end{align}
```

For coloured boxes use package **empheq**. For an example see section 55 on page 95.

40 Greek letters

`\pmb`

The $\mathcal{A}\kern-.1667em\lower.5ex\hbox{$\mathcal{M}$}\kern-.125em\mathcal{S}$math package simulates a bold font for the greek letters, it writes a greek character twice with a small kerning. This is done with the macro `\pmb{<letter>}`. The `\mathbf{<character>}` doesn't work with lower greek character.

α	$\boldsymbol{\alpha}$
β	$\boldsymbol{\beta}$
γ	$\boldsymbol{\gamma}$
δ	$\boldsymbol{\delta}$
ϵ	$\boldsymbol{\epsilon}$
$\ldots$	$\ldots$

41 Miscellaneous commands

`\overset`
`\underset`
`\boxed`

There are several commands which can be used in math mode:

Some examples are shown in table 19.

`$\underset{under}{baseline}$`	$\underset{under}{baseline}$
`$\overset{over}{baseline}$`	$\overset{over}{baseline}$

Table 19: Different mathcommands

`\underset` is a useful macro for having limits under non operators (see section 91).

Part III
TeX and math

There is in general no need to use the TeX macros, because the ones defined with LaTeX or with AMSmath are much more usefull. Nevertheless there maybe situations, where someone has to use one of the TeX macros or special TeX math length. One can not expect, that all macros work in the usual way, a lot of it are redefined by LaTeX or AMSmath. On the other hand some of these basic macros or length definitions are used in the TeX way, so it might be interesting to have all declared in a short way for some information.

42 Length registers

42.1 `\abovedisplayshortskip`

A length with glue, see section 11.5.1 for an example.

42.2 `\abovedisplayskip`

A length with glue, see section 11.5.1 for an example.

42.3 `\belowdisplayshortskip`

A length with glue, see section 11.5.1 for an example.

42.4 `\belowdisplayskip`

A length with glue, see section 11.5.1 for an example.

42.5 `\delimiterfactor`

The height of a delimiter is often not optimal calculated by TeX. In some cases it is too short. With `\delimiterfactor` one can correct this height. The delimiterheight is $<$ calculated height $> \cdot < \#1 > /1000$ where `#1` is the parameter of `\delimiterfactor`. The default value is 901.

$$y=\left\{\begin{array}{ll} x^2+2x & \text{if } x<0,\\ x^3 & \text{if } 0\le x<1,\\ x^2+x & \text{if } 1\le x<2,\\ x^3-x^2 & \text{if } 2\le x.\end{array}\right.$$

```
\[
y = \left\{%
 \begin{array}{ll}
   x^2+2x  &\textrm{if }x<0,\\
   x^3     &\textrm{if }0\le x<1,\\
   x^2+x   &\textrm{if }1\le x<2,\\
   x^3-x^2 &\textrm{if }2\le x.
 \end{array}%
\right.
\]
```

$$y=\left\{\begin{array}{ll}x^2+2x & \text{if }x<0,\\ x^3 & \text{if }0\le x<1,\\ x^2+x & \text{if }1\le x<2,\\ x^3-x^2 & \text{if }2\le x.\end{array}\right.$$

```
\[
\delimiterfactor=1500
y = \left\{%
 \begin{array}{ll}
   x^2+2x  &\textrm{if }x<0,\\
   x^3     &\textrm{if }0\le x<1,\\
   x^2+x   &\textrm{if }1\le x<2,\\
   x^3-x^2 &\textrm{if }2\le x.
 \end{array}%
\right.
\]
```

42.6 \delimitershortfall

Additionally to the forgoing `\delimiterfactor` one can modify the height of the delimiter with another value. TEX makes the delimiter larger than the values of $<$ calculated height $>\cdot<$ delimiterfactor $>/1000$ and $<$ calculated height $>-<$ delimitershortfall $>$. This makes it possible to get always different heights of a sequence of delimiters.

$x\cdot\left(\left(x^2-y^2\right)-3\right)$

$x\cdot\Bigg(\bigg(x^2-y^2\bigg)-3\Bigg)$

```
$x\cdot\left(\left(x^2-y^2\right)-3\right)$\\[7pt]

$
\delimitershortfall-1pt
x\cdot\left(\left(x^2-y^2\right)-3\right)$
```

$(((A)))$

$\Big(\big((A)\big)\Big)$

```
$\left(\left(\left(A\right)\right)\right)$\\[7pt]

$\delimitershortfall-1pt
\left(\left(\left(A\right)\right)\right)$
```

42.7 \displayindent

This is the left shift amount of a line holding displayed equation. By default it is $0pt$ but gets the value of an indented paragraph when there is an environment like the quotation one.

The following formula is typeset in the usual way without modifying anything.

$$f(x)=\int\frac{\sin x}{x}dx$$

Now we start an quotation environment which sets `\labelwidth` to new values for a greater left margin.

- The following formula is typeset in the usual way without modifying anything.

$$f(x)=\int\frac{\sin x}{x}dx$$

- Now we write the same equation, but now with modifying `\displayindent`, it is set to the negative `\labelwidth`:

$$f(x) = \int \frac{\sin x}{x} dx$$

```
\[
  \displayindent=-\leftskip
  f(x) = \int \frac{\sin x}{x}dx
\]
```

42.8 \displaywidth

The he width of the line holding a displayed equation, which is by default `\linewidth`. In the second example the formula is centered for a display width of `0.5\linewidth`.

$$f(x) = \int \frac{\sin x}{x} dx$$

$$f(x) = \int \frac{\sin x}{x} dx$$

```
\[ f(x) = \int \frac{\sin x}{x}dx \]
\[
  \displaywidth=0.5\linewidth
  f(x) = \int \frac{\sin x}{x}dx
\]
```

42.9 \mathsurround

Extra space added when switching in and out of the inline math mode (see section 11.5).

42.10 \medmuskip

See section 11.1 for an example.

42.11 \mkern

Similiar to `\kern`, but adds a math kern item to the current math list. Length must be a math unit.

42.12 \mskip

Similiar to `\skip`, but adds math glue to the current math list. Length must be a math unit.

42.13 `\muskip`

Assigns a length with a math unit to one of the 256 `\muskip` register.

42.14 `\muskipdef`

Defines a symbolic name for a `\muskip` register.

42.15 `\nonscript`

Ignores immediately following glue or kern in script and scriptscript styles, which makes a redefinition of `\mathchoice` superfluous.

42.16 `\nulldelimiterspace`

This is the width of a null or missing delimiter, e.g. `\right.` or for the left one.

42.17 `\predisplaysize`

Is the effective width of the line preceeding a displayed equation, whether `\abedisplayskip` or `abedisplayshortskip` is used for the vertical skip.

42.18 `\scriptspace`

The space inserts after an exponent or index, predefined as `\scriptspace=0.5pt`

42.19 `\thickmuskip`

See section 11.1.

42.20 `\thinmuskip`

The short version for positive skip is defined as `\def\,{\mskip\thinmuskip}` and the one for a negative skip as `\def\!{\mskip-\thinmuskip}` (see also section 11.1).

$$\begin{gathered}
\sqrt 2 x - \sqrt 2\,x\\
\sqrt{\log x} - \sqrt{\,\log x}\\
P\left({1/\sqrt n}\right) - P\left({1/ \sqrt n}\,\right)\\[8pt]
[0,1) - [\,0,1)\\
x^2/2 - x^2\!/2
\end{gathered}$$

```
$\sqrt 2 x$ -- $\sqrt 2\,x$\\
$\sqrt{\log x}$ -- $\sqrt{\,\log x}$\\
$P\left({1/\sqrt n}\right)$ -- $P\left({1/ \
  sqrt n}\,\right)$\\[8pt]
 $[0,1)$ -- $[\,0,1)$\\
$x^2/2$ -- $x^2\!/2$\\
```

$$\int\int_D dxdy \quad \int\!\int_D dx\,dy$$

$$\int\!\!\int_D dx\,\,dy \quad \int\!\!\!\int_D dx\,\,\,dy$$

$$\int\!\!\!\!\int_D dx\,\,\,\,dy \quad \int\!\!\!\!\!\int_D dx\,\,\,\,\,dy$$

$$\int\!\!\!\int_D dx\,dy$$

```
$$\int\int_D dxdy \quad
  \int\!\int_D dx\,dy$$
$$\int\!\!\int_D dx\,\,dy \quad
  \int\!\!\!\int_D dx\,\,\,dy$$
$$\int\!\!\!\!\int_D dx\,\,\,\,dy \quad
  \int\!\!\!\!\!\int_D dx\,\,\,\,\,dy$$
$$\int\!\!\!\int_D dx\,dy$$
```

42.21 \thinmuskip

See section 11.1.

43 Math font macros

43.1 \delcode

Each character has not only a `\catcode` and `\mathcode` but also a `\delcode` which defines for a single chracter how it should look when used as a math delimiter.

43.2 \delimiter

Every character can be declared as a delimiter, but TEX must know which characters should be used for the default and the big size. For LATEX the macro `\DeclareMathDelimiter` should be used (see section 8.2).

In the following example `\tdela` is the character $0x22$ ($\uparrow$) from font number 2 (`csmy`) and character $0x78$ from font number 3 (`cmex`) for the big version. `\tdelb` is the same vice versa ($\downarrow$).

$$\uparrow x-y\downarrow(x+y)=x^2-y^2$$

$$\uparrow\sum_{n=0}^{\infty}\frac{1}{2^n}\downarrow^2=4$$

$$\left\uparrow\sum_{n=0}^{\infty}\frac{1}{2^n}\right\downarrow^2=4$$

```
\def\tdela{\delimiter"4222378\relax}
\def\tdelb{\delimiter"5223379\relax}

$\tdela x-y\tdelb(x+y)=x^2-y^2$

\[\tdela\sum_{n=0}^\infty {1\over2^n}\tdelb^2 = 4\]

\[\left\tdela\sum_{n=0}^\infty {1\over2^n}\right\tdelb^2 = 4\]
```

43.3 \displaystyle

See section 12 for an example.

43.4 `\fam`

When TeX switches into the math mode, it typesets everything using one of the 16 possible families of fonts. `\fam` in an internal register where other macros can check which font is the actual one. At the beginning TeX starts with `\fam=-1`.

`\fam=-1`	$123abcABC\alpha\beta\gamma$
`\fam=0`	$\mathrm{123abcABC}\alpha\beta\gamma$
`\fam=1`	$\mathnormal{123abcABC}\alpha\beta\gamma$
`\fam=2`	$\infty\in\ni\dashv\lfloor\rfloor\mathcal{ABC}\alpha\beta\gamma$
`\fam=3`	$\Bigg\backslash\prod\coprod\widehat{\quad}\Bigg)''^{\alpha\beta\gamma}$
`\fam=4`	$\geqslant\curlyeqprec\curlyeqsucc\frown\Subset\Supset\sqsupset\rhd\lhd\alpha\beta\gamma$
`\fam=5`	$\nVdash\nvDash\nVDash\Game\mathbb{ABC}\alpha\beta\gamma$

$\mathrm{123abcABC}\alpha\beta\gamma(0)$

$\mathbf{123abcABC}\alpha\beta\gamma\mathbf{(6)}$

$\mathit{123abcABC}\alpha\beta\gamma\mathit{(11)}$

$\mathtt{123abcABC}\alpha\beta\gamma\mathtt{(12)}$

$\mathsf{123abcABC}\alpha\beta\gamma\mathsf{(13)}$

$\mathnormal{123abcABC}\alpha\beta\gamma(1)$

```
$\mathrm{123abcABC\alpha\beta\gamma (\the\fam)
  }$\\[5pt]
$\mathbf{123abcABC\alpha\beta\gamma (\the\fam)
  }$\\[5pt]
$\mathit{123abcABC\alpha\beta\gamma (\the\fam)
  }$\\[5pt]
$\mathtt{123abcABC\alpha\beta\gamma (\the\fam)
  }$\\[5pt]
$\mathsf{123abcABC\alpha\beta\gamma (\the\fam)
  }$\\[5pt]
$\mathnormal{123abcABC\alpha\beta\gamma (\the\
  fam)}$
```

43.5 `\mathaccent`

Requires three parameter as one number, the class, the font family and the character.

$\breve{A}$

```
\def\dA{\mathaccent"7015\relax}
{\Large $\dA{A}$}
```

43.6 `\mathbin`

Declares a following character as a binary symbol with another spacing before and behind such a symbol.

$a|b \quad a \mathbin| b$

```
{\Large
$a|b \quad a\mathbin| b$}
```

43.7 \mathchar

Declares a math character by three parameter as one integer number, giving its class, font family, and font position. In the following example `\mathchar` defines a character of class 1 (big operators), font family 3 (math extension font) and number 58 (big sum character).

$$a\sum\limits_{i=1}^{\infty} b \quad a\mathop{\displaystyle\sum}\limits_{i=1}^{\infty} b$$

```
{\Large
$a\sum\limits_{i=1}^{\infty} b \quad
 a\mathchar"1358\limits_{i=1}^{\infty} b$}
```

43.8 \mathchardef

This is in principle the same as `\mathchar`, it allows only to make such definitions permanent.

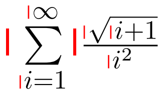
$$a\mathop{\displaystyle\sum}\limits_{i=1}^{\infty}\sqrt{i+1}$$

$$a\sum\limits_{i=1}^{\infty}\sqrt{i+1}$$

```
\bgroup
\mathchardef\sum="1358
$a\sum\limits_{i=1}^{\infty}\sqrt{i+1}$\\[5pt]
\egroup

$a\sum\limits_{i=1}^{\infty}\sqrt{i+1}$
```

43.9 \mathchoice

Specifies specific subformula sizes for the 4 main styles: `displaystyle` – `textstyle` – `scriptstyle` – `scriptscriptstyle`.



```
\Large
\def\myRule{{%
  \color{red}%
  \mathchoice{\rule{2pt}{20pt}}{\rule{1pt}{10pt}}%
    {\rule{0.5pt}{5pt}}{\rule{0.25pt}{2.5pt}}%
    \mkern2mu}}
$\myRule\sum\limits_{\myRule i=1}^{\myRule\infty}%
 \myRule\frac{\myRule\sqrt{\myRule i+1}}{\myRule i^2}$
```

43.10 \mathclose

Assigns class 5 (closing character) to the following parameter, which can hold a single character or a subformula.

$$A:\frac{B}{C}:D$$

$$A\mathopen{:}\frac{B}{C}\mathclose{:}D$$

```
{\large
$A:\frac{B}{C}:D$\\[5pt]
$A\mathopen:\frac{B}{C}\mathclose: D $}
```

43.11 \mathcode

A math font is far different from a text font. A lot of the characters has to be defined with `\mathcode`, which defines the character with its class, font family and character number, e.g. `` \mathcode`\<="313C ``. It defines the character "<" as a realtion symbol (class 3) from the font family 1 and the character number 0x3C, which is 60 decimal.

43.12 \mathop

Assigns class 1 (large operator) to the parameter, which can be a single character or a subformula.

$$A_{i=1}^{\infty}$$

$$\mathop{A}_{i=1}^{\infty}$$

```
\[ A_{i=1}^{\infty} \]
\[ \mathop{A}_{i=1}^{\infty} \]
```

43.13 \mathopen

Vice versa to `\mathclose` (see section 43.10).

43.14 \mathord

Assigns class 0 (ordinary character) to the following parameter, which can be a single character or a subformula.

$y = f(x)$

$y \mathord{=} f(x)$

```
{\large
$y = f(x)$\\[5pt]
$y \mathord= f(x)$}
```

43.15 \mathpunct

Assigns class 6 (punctuation) to the following parameter, which can be a single character or a subformula (see section 11.4 for an example).

43.16 \mathrel

Assigns class 3 (relation) to the following parameter, which can be a single character or a subformula.

$x_1 \rightarrow x_2 \rightarrow x_3$

$x_1\mathpunct{\rightarrow} x_2\mathpunct{\rightarrow} x_3$

```
{\large
$x_1\rightarrow x_2\rightarrow x_3$\\[5pt]
$x_1\mathpunct\rightarrow x_2\mathpunct\rightarrow x
  _3$}
```

43.17 \scriptfont

Specifies the scriptstyle font (used dor super/subscript) for a family.

A_1 A_1

```
$A_1$
\font\tenxii=cmr12
\scriptfont0=\tenxii
$A_1$
```

43.18 \scriptscriptfont

Specifies the scriptscriptstyle font for a family.

43.19 `\scriptscriptstyle`

Selects scriptscript style for the following characters.

43.20 `\scriptstyle`

Selects script style for the following characters.

43.21 `\skew`

Especially for italic characters double accents are often misplaced. `\skew` has three arguments

horizontal shift: A value in math units for the additional shift of the accent.

the accent: The symbol which is placed above the character.

the character: This is in general a single character, but can also include itself an accent.

$\mathcal{AMS}$math redefines the setting of double accents. This is the reason why there are only a few cases where someone has to use `\skew` when `amsmath.sty` is loaded, like in this document.

$$\tilde{\tilde{i}} \qquad \tilde{\tilde{A}}$$

$$\skew{3}{\tilde}{\tilde{i}} \qquad \skew{7}{\tilde}{\tilde{A}}$$

```
\large
$\tilde i$ \qquad $\tilde{A}$\\[5pt]
$\skew{3}{\tilde}{i}$ \qquad $\skew{7}{\tilde}{A}$
```

43.22 `\skewchar`

Is -1 or the character (reference symbol) used to fine-tune the positioning of math accents.

43.23 `\textfont`

Specifies the text font for a family.

43.24 `\textstyle`

Selects the text style for the following characters.

44 Math macros

44.1 `\above`

$\genfrac{}{}{0pt}{1}{a}{b}$

$\genfrac{}{}{1pt}{1}{a}{b}$

$\genfrac{}{}{2.5pt}{1}{a}{b}$

$\genfrac{}{}{0pt}{0}{a}{b}$

```
$a\above0pt b$\\[8pt]

${a\above1pt b}$\\[8pt]

${a\above2.5pt b}$\\[8pt]

$\displaystyle{a\above0pt b}$
```

44.2 \abovewithdelims

$\genfrac{(}{)}{0pt}{1}{a}{b}$

$\genfrac{\{}{)}{1.0pt}{1}{a}{b}$

$\genfrac{[}{]}{2.0pt}{1}{a}{b}$

$\genfrac{\{}{.}{0pt}{0}{a}{b}$

```
$a\abovewithdelims()0pt b$\\[8pt]

\def\fdelimA{\abovewithdelims\{)1.0pt}
${a\fdelimA b}$\\[8pt]

\def\fdelimB{\abovewithdelims[]2.0pt}
${a\fdelimB b}$\\[8pt]

\def\fdelimC{\abovewithdelims\{.0pt}
$\displaystyle{a\fdelimC b}$
```

44.3 \atop

${a \atop b}$

$({n \atop k}) = {n! \above 1pt k!(n-k)!}$

$\displaystyle{a \atop b}$

```
$a\atop b$\\[8pt]

$({n \atop k}) = {n!\above1pt k!(n-k)!}$\\[8pt]

$\displaystyle{a\atop b}$
```

44.4 \atopwithdelims

$\binom{a}{b}$

$\binom{n}{k} = \frac{n!}{k!(n-k)!}$

$\displaystyle\genfrac{\{}{.}{0pt}{}{a}{b}$

```
$a\atopwithdelims() b$\\[8pt]

${n \atopwithdelims() k} = {n!\above1pt k!(n-k)!}$\\[8pt]

$\displaystyle{a\atopwithdelims\{. b}$
```

44.5 \displaylimits

Resets the conventions for using limits with operators to the standard for the used environment.

44.6 \eqno

Puts an equation number at the right margin, the parameter can hold anything. \eqno places only the parameter, but doesn't increase any equation counter.

$$y = f(x) \qquad\qquad (A12)$$

```
\[ y=f(x) \eqno{(A12)} \]
```

44.7 \everydisplay

Inserts the parameter at the start of every switch to display math mode.

$$f(x) = \int \frac{\sin x}{x}dx$$

$$g(x) = \int \frac{\sin^2 x}{x^2}dx$$

```
\everydisplay{\color{red}
}
\[ f(x) = \int \frac{\sin x}{x}dx \]
\[ g(x) = \int \frac{\sin^2 x}{x^2}dx \]
```

44.8 \everymath

Same as \everydisplay, but now for the inline mode. In the following example the displaystyle is used (beside using color red) for every inline math expression.

$$f(x) = \int \frac{\sin x}{x}dx$$

Instead of $\displaystyle\frac{\sin x}{x}$ now with $\displaystyle\frac{\cos x}{x}$:

$$g(x) = \int \frac{\cos x}{x}dx$$

```
\everymath{\color{red}%
  \displaystyle}
\[ f(x) = \int \frac{\sin x}{x}dx \]
Instead of $\frac{\sin x}{x}$
  now with $\frac{\cos x}{x}$:
\[ g(x) = \int \frac{\cos x}{x}dx \]
```

44.9 \left

TeX calculates the size of the following delimiter needed at the left side of a formula. Requires an additional right.

44.10 \leqno

Vice versa to \eqno (see section 44.6).

44.11 \limits

Typesets limits above and/or below operators (see section 6).

44.12 \mathinner

Defines the following parameter as subformula.

44.13 \nolimits

The opposit of \limits, instead of above/below limits are placed to the right of large operators (class 1).

44.14 `\over`

Is equivalent to the fraction macro of LaTeX and equivalent to the `\overwithdelims`, see section 44.16.

${a\over b} \qquad {{m\over n}\over{a+b}}$

$$ {m\over n}\over{a+b} $$

```
$ {a\over b} \qquad {{m\over n}\over{a+b}} $
\[ {m\over n}\over{a+b} \]
```

44.15 `\overline`

Puts a line over the following character or subformula and has the same problems with different heights as underlines (see section 44.19).

$\overline{x}+\overline{y}=\overline{z}$

$\overline{x} + \overline{A} = \overline{z}$

$\overline{x\vphantom{A}} + \overline{A} = \overline{z\vphantom{A}}$

```
$\overline{x}+\overline{y}=\overline{z}$\\
\let\ol\overline
$ \ol{x} + \ol{A} = \ol{z} $\\[5pt]
\def\yPh{\vphantom{A}}
$ \ol{x\yPh} + \ol{A} = \ol{z\yPh} $
```

44.16 `\overwithdelims`

is a generalized fraction command with preset fraction bar thickness.

${a\overwithdelims() b} \qquad {{m\over n}\overwithdelims[]{a+b}}$

$$ {m\over n}\overwithdelims\{.{a+b} $$

```
$ {a\overwithdelims() b} \qquad {{m\over n}\overwithdelims[]{a+b}} $
\[ {m\over n}\overwithdelims\{.{a+b} \]
```

44.17 `\radical`

Makes a radical atom from the delimiter (27-bit number) and the math field.

$\sqrt{\frac{1}{7}}$

$\sqrt{\frac{1}{7}}$

$\sqrt{\frac{1}{7}}$

$\sqrt{\frac{1}{7}}$

```
\def\mySqrt{\radical"0270371\relax}
$ \mySqrt{\frac{1}{7}} $\\[5pt]

\def\mySqrt{\radical"0270372\relax}
$ \mySqrt{\frac{1}{7}} $\\[5pt]

\def\mySqrt{\radical"0270373\relax}
$ \mySqrt{\frac{1}{7}} $\\[5pt]

\def\mySqrt{\radical"0270374\relax}
$ \mySqrt{\frac{1}{7}} $\\[5pt]
```

44.18 `\right`

Opposite to `\left`, makes TeX calculate the size of the delimiter needed at the right of a formula.

44.19 \underline

When there is a combination of variables with and without an index, the underline are typset with a different depth. Using \vphantom in this case is a good choice.

$$\underline{x}+\underline{y}=\underline{z}$$
$$\underline{x\vphantom{y}}+\underline{y}=\underline{z\vphantom{y}}$$
$$\underline{x_1}+\underline{y_2}=\underline{z_3}$$

```
$\underline{x}+\underline{y}=\underline{z}$\\

\let\ul\underline
\def\yPh{\vphantom{y}}
$ \ul{x\yPh} + \ul{y} = \ul{z\yPh} $\\

$ \ul{x_1} + \ul{y_2} = \ul{z_3} $
```

44.20 \vcenter

Centers vertical material with respect to the axis.

45 Math penalties

45.1 \binoppenalty

A penalty for breaking math expressions between lines in a paragraph. TeX breaks lines only when the binary symbol is not the last one and when the penalty is below 10,000.

45.2 \displaywidowpenalty

The penalty which is added after the penultimate line immediately preceeding a display math formula.

45.3 \postdisplaypenalty

Is dded immediately after a math display ends.

45.4 \predisplaypenalty

Is added immediately before a math display starts.

45.5 \relpenalty

The penalty for a line break after a relation symbol (if a break is possible).

Part IV
Other packages

The following sections are not a replacement for the package documentation!

46 List of available math packages

accents
alphalph
amsart
amsbook
amsbsy
amscd
amscls
amsfonts
amslatex
amsltx11
amsmath
amsppt
amsppt1
amsproc
amssym (plain TeX)
amssymb (LaTeX)
amstex (Plain TeX)
amstext
amsthm
bez123
bitfield
brclc
breqn
cancel
cases
comma
datenumber
diagxy
doublestroke
easyeqn
easybmat
easymat
eqnarray
esvect
fixmath
ftlpoint
icomma
leftidx
mathdots
mathematica
mil3
mtbe
Nath
numprint
random
romannum
TeXaide

The following examples depend to the listed versions of the packages:

```
  amsopn.sty    1999/12/14 v2.01 operator names
      bm.sty    1999/07/05 v1.0g Bold Symbol Support (DPC/FMi)
  empheq.sty    2004/08/03 v2.11 Emphasizing equations (MH)
   amscd.sty    1999/11/29 v2.0
 accents.sty    2000/08/06 v1.2 Math Accent Tools
  framed.sty    2002/12/29 v 0.5: framed or shaded text with page breaks
  pstcol.sty    2001/06/20 v1.1 PSTricks color colompatibility (DPC)
pstricks.sty    2004/05/06 v0.2k LaTeX wrapper for ‘PSTricks’ (RN,HV)
pstricks.tex    2003/03/07 v97 patch 15 ‘PSTricks’ (tvz)
pst-node.sty    1997/03/25 package wrapper for PSTricks pst-node.tex
delarray.sty    1994/03/14 v1.01 array delimiter package (DPC)
   xypic.sty    1999/02/16 Xy-pic version 3.7
 exscale.eps    Graphic file (type veps)
```

47 accents

If you want to write for example an underlined M, then you can do it as

`\underline{$M$}`	$\underline{M}$
`\underbar{$M$}`	$\underline{M}$
`\underaccent{\bar}{M}`	$\underaccent{\bar}{M}$

As seen, there is no difference in `\underline` and `\underbar`. For some reasons it may be better to use the package `accents.sty` with the `\underaccents` macro.

48 amscd – commutative diagrams

`amscd.sty` is part of the $\mathcal{AMS}$math bundle or available at CTAN[27] and has no options for the `\usepackage` command. `amscd.sty` does not support diagonal arrows but is much more easier to handle than the complex pstricks or the xypic package. On the other hand simple diagrams can be written with the array environment or look at [23].

$$\begin{tikzcd}
R\times S\times T \arrow[r, "\text{restriction}"] \arrow[d, "proj"'] & S\times T \arrow[d, "proj"] \\
R\times S & S \arrow[l, "\text{inclusion}"]
\end{tikzcd}$$

```
\[
\begin{CD}
  R\times S\times T @>\text{restriction}>> S\times T \\
        @VprojVV                            @VVprojV \\
  R\times S         @<<\text{inclusion}<        S
\end{CD}
\]
```

49 amsopn

With this package it is very easy to declare new math operators, which are written in upright mode:

$\underset{s=p}{Res}$ versus $\underset{s=p}{\operatorname{Res}}$

```
\documentclass[10pt]{article}
\usepackage{amsmath}
\usepackage{amsopn}
\DeclareMathOperator{\Res}{Res}
\begin{document}
$\underset{s=p}{Res}\quad\underset{s=p}{\Res}$
\end{document}
```

Table 20 shows the predefined operatornames of `amsopn`.

[27] CTAN://macros/latex/required/amslatex/math/amscd.dtx

`\arccos`	arccos	`\arcsin`	arcsin	`\arctan`	arctan
`\arg`	arg	`\cos`	cos	`\cosh`	cosh
`\cot`	cot	`\coth`	coth	`\csc`	csc
`\deg`	deg	`\det`	det	`\dim`	dim
`\exp`	exp	`\gcd`	gcd	`\hom`	hom
`\inf`	inf	`\injlim`	inj lim	`\ker`	ker
`\lg`	lg	`\lim`	lim	`\liminf`	lim inf
`\limsup`	lim sup	`\ln`	ln	`\log`	log
`\max`	max	`\min`	min	`\Pr`	Pr
`\projlim`	proj lim	`\sec`	sec	`\sin`	sin
`\sinh`	sinh	`\sup`	sup	`\tan`	tan
`\tabh`	tanh				

Table 20: The predefined operators of `amsopn.sty`

50 bigdel

This is a very useful package together with the `multirow.sty` package. In the following example we need additional parentheses for a different number of rows. This is also possible with the `array` environment, but not as easy as with `bigdelim.sty`. The trick is that you need one separate column for a big delimiter, but with empty cells in all rows, which the delimiter spans.

$$\begin{pmatrix}
\begin{array}{c} \\ \end{array} & \begin{array}{cccc} x_{11} & x_{12} & \dots & x_{1p} \end{array} & \\
\text{text}\left[\begin{array}{c} \\ \\ \\ \\ \end{array}\right. & \left.\begin{array}{cccc} x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & & & \\ x_{n_1 1} & x_{n_1 2} & \dots & x_{n_1 p} \end{array}\right\}\text{some text} & \\
 & \left.\begin{array}{cccc} x_{n_1+1,1} & x_{n_1+1,2} & \dots & x_{n_1+1,p} \\ \vdots & & & \\ x_{n_1+n_2,1} & x_{n_1+n_2,2} & \dots & x_{n_1+n_2,p} \end{array}\right\}\text{some more text} & \\
 & \begin{array}{c} \vdots \end{array} &
\end{pmatrix}$$

```
\[
  \begin{pmatrix}
     & x_{11} & x_{12} & \dots & x_{1p} & \rdelim\}{4}{3cm}[some text]\\
     \ldelim[{5}{1cm}[text] & x_{21} & x_{22} & \dots & x_{2p} \\
     & \vdots\\
     & x_{n_1 1}& x_{n_1 2} & \dots & x_{n_1 p}\\
     & x_{n_1+1,1}&x_{n_1+1,2} & \dots & x_{n_1+1, p} &
         \rdelim\}{3}{3cm}[some more text]\\
     & \vdots\\
     & x_{n_1+n_2, 1} & x_{n_1+n_2,2} & \dots & x_{n_1+n_2,p}\\
     & \vdots \\
  \end{pmatrix}
\]
```

As seen in the above listing the left big delimiter is placed in the first column, all other rows start with second column. It is possible to use all columns above and below the delimiter. For the `array` environment there must be two more columns defined, in case of a big delimiter left and right. The syntax of `\ldelim` and `\rdelim` is:

```
\ldelim<delimiter>{<n rows>}{<added horizontal space>}[<text>]
\rdelim<delimiter>{<n rows>}{<added horizontal space>}[<text>]
```

Any delimiter which is possible for the `\left` or `\right` command are allowed, e.g.: "`()[]{}|`". The text is an optional argument and always typeset in text mode.

51 bm

By default the math macro **mathbf** writes everything in bold and in upright mode $\mathbf{y=f(x)}$ (`$\mathbf{y=f(x)}$`), but it should be in italic mode especially for variables $\boldsymbol{y=f(x)}$ (`$\bm{y=f(x)}$`). For writing a whole formula in bold have a look at section 22 on page 46.

52 braket

It is available at CTAN://macros/latex/contrib/other/misc/braket.sty and provides several styles for writing math expressions inside brakets. For example:

$$\left\{ x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}\right\}$$

```
\[ \left\{ x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}\right\} \]
```

looks not quit right and it is not really easy to get the first vertical line in the same size as the outer braces. Some solution maybe using `\vphantom`:

$$\left\{\vphantom{\frac{5}{3}}x\in\mathbf{R} \right|\left. 0<{|x|}<\frac{5}{3}\right\}$$

```
\[
\left\{\vphantom{\frac{5}{3}}x\in\mathbf{R} \right|\left
  . 0<{|x|}<\frac{5}{3}\right\}
\]
```

`braket.sty` has the macros

```
\Bra{<math expression>}
\Ket{<math expression>}
\Braket{<math expression>}
\Set{<math expression>}
```

and the same with a leading lower letter, which are not really interesting.

$$\left\langle x \in \mathbf{R} | 0 < |x| < \frac{5}{3} \right|$$

$$\left| x \in \mathbf{R} | 0 < |x| < \frac{5}{3} \right\rangle$$

$$\left\langle x \in \mathbf{R} \,\middle|\, 0 < |x| < \frac{5}{3} \right\rangle$$

$$\left\{ x \in \mathbf{R} \,\middle|\, 0 < |x| < \frac{5}{3} \right\}$$

```
\[ \Bra{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
\[ \Ket{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
\[ \Braket{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
\[ \Set{x\in\mathbf{R} | 0<{|x|}<\frac{5}{3}} \]
```

The difference between the `\Set` and the `\Braket` macro is the handling of the vertical lines. In `\Set` only the first one gets the same size as the braces and in `\Braket` all.

$$\left\langle \phi \,\middle|\, \frac{\partial^2}{\partial t^2} | \psi \right\rangle$$

```
\[  \Braket{ \phi | \frac{\partial^2}{\partial t^2} | \psi
   }\]
```

`\Bra` and `\Ket` do nothing with the inner vertical lines.

53 cancel

This is a nice package for canceling anything in mathmode with a slash, backslash or a `X`. To get a horizontal line we can define an additional macro called `hcancel` with an optional argument for the line color (requires package `color`):

```
\newcommand\hcancel[2][black]{\setbox0=\hbox{#2}%
 \rlap{\raisebox{.45\ht0}{\textcolor{#1}{\rule{\wd0}{1pt
   }}}}#2}
```

It is no problem to redefine the `cancel` macros to get also colored lines. A horizontal line for single characters is also decribed in section 14 on page 39.

`\cancel`: $f(x) = \frac{\left(x^2+1\right)\cancel{(x-1)}}{\cancel{(x-1)}(x+1)}$

`\bcancel`: $\bcancel{3}$ $\qquad \cancel{1234567}$

\xcancel: $\xcancel{3}\qquad\xcancel{1234567}$

\hcancel: $\hcancel{3}\qquad\hcancel{1234567}$

```
$f(x)=\dfrac{\left(x^2+1\right)\cancel{(x-1)}}{\cancel{(x-1)
  }(x+1)}$\\[0.5cm]
$\bcancel{3}\qquad\bcancel{1234567}$\\[0.5cm]
$\xcancel{3}\qquad\xcancel{1234567}$\\[0.5cm]
$\hcancel{3}\qquad\hcancel[red]{1234567}$
```

54 delarray

Package `delarray.sty`[28] supports different delimiters which are defined together with the beginning of an array:

```
\begin{array}<delLeft>{cc}<delRight>
...
```

defines an array with two centered columns and the delimiters "`<delLeft><delRight>`", e.g. "()".

```
\[
A=\begin{array}({cc})
  a & b\\
  c & d
\end{array}
\]
```

$$A=\left(\begin{array}{cc} a & b\\ c & d \end{array}\right)$$

`delarray.sty` expects a pair of delimiters. If you need only one (like the cases structure) then use the dot for an "empty" delimiter, e.g.

```
\[
A=\begin{array}\{{cc}.
  a & b\\
  c & d
\end{array}
\]
```

$$A=\left\{\begin{array}{cc} a & b\\ c & d \end{array}\right.$$

which is a useful command for a cases structure without the $\mathcal{A}\!\mathcal{M}\!\mathcal{S}$math package, which is described in the $\mathcal{A}\!\mathcal{M}\!\mathcal{S}$math part.

55 empheq

This package supports different frames for math environments of the $\mathcal{A}\!\mathcal{M}\!\mathcal{S}$math package. It doesn't support all the environments from standard LaTeXwhich are not modified by $\mathcal{A}\!\mathcal{M}\!\mathcal{S}$math, e.g. `eqnarray`.

[28] CTAN://macros/latex/required/tools/delarray.dtx

With the optional argument of the environment **`empheq`** the preferred box type can be specified. A simple one is **`\fbox`**

$$\boxed{f(x)=\int_1^{\infty}\frac{1}{x^2}\,dt=1} \tag{55.1}$$

```
\begin{empheq}[box=\fbox]{align}
 f(x)=\int_1^{\infty}\frac{1}{x^2}\,dt=1
\end{empheq}
```

The same is possible with the macro **`\colorbox`**:

$$f(x)=\int_1^{\infty}\frac{1}{x^2}\,dt=1 \tag{55.2}$$

```
\begin{empheq}[box={\fboxsep=10pt\colorbox{yellow}}]{align}
 f(x)=\int_1^{\infty}\frac{1}{x^2}\,dt=1
\end{empheq}
```

The key **`box`** can hold any possible LaTeX command sequence. Boxing subequations is also no problem, the **`empheq`** environment works in the same way:

$$\begin{aligned} f(x)&=\int_1^{\infty}\frac{1}{x^1}\,dt=1 && (55.3a)\\ f(x)&=\int_2^{\infty}\frac{1}{x^2}\,dt=0.25 && (55.3b) \end{aligned}$$

```
\begin{subequations}
\begin{empheq}[box={\fboxsep=10pt\colorbox{cyan}}]{align}
 f(x)  & =\int_1^{\infty}\frac{1}{x^2}\,dt=1\\
 f(x)  & =\int_2^{\infty}\frac{1}{x^2}\,dt=0.25
\end{empheq}
\end{subequations}
```

For more information on **`empheq`** have a look at the documentation of the package which is available at any CTAN server.

56 esint

This is a very useful package when you want nice double or triple integral or curve integral symbols. The ones from **`wasysym`**[29] are not the best. **`esint`**[30] supports the following symbols:

[29] CTAN://macros/latex/contrib/wasysym/

[30] CTAN://macros/latex/contrib/esint/ CTAN://fonts/ps-type1/esint/

$$\texttt{\textbackslash int}: \int \tag{56.1}$$

$$\texttt{\textbackslash iint}: \iint \tag{56.2}$$

$$\texttt{\textbackslash iiintop}: \iiint \tag{56.3}$$

$$\texttt{\textbackslash iiiintop}: \iiiint \tag{56.4}$$

$$\texttt{\textbackslash dotsintop}: \idotsint \tag{56.5}$$

$$\texttt{\textbackslash ointop}: \oint \tag{56.6}$$

$$\texttt{\textbackslash oiint}: \oiint \tag{56.7}$$

$$\texttt{\textbackslash sqint}: \sqint \tag{56.8}$$

$$\texttt{\textbackslash sqiint}: \sqiint \tag{56.9}$$

$$\texttt{\textbackslash ointctrclockwise}: \ointctrclockwise \tag{56.10}$$

$$\texttt{\textbackslash ointclockwise}: \ointclockwise \tag{56.11}$$

$$\texttt{\textbackslash varointclockwise}: \varointclockwise \tag{56.12}$$

$$\texttt{\textbackslash varointctrclockwise}: \varointctrclockwise \tag{56.13}$$

$$\texttt{\textbackslash fint}: \fint \tag{56.14}$$

$$\texttt{\textbackslash varoiint}: \varoiint \tag{56.15}$$

$$\texttt{\textbackslash landupint}: \landupint \tag{56.16}$$

$$\texttt{\textbackslash landdownint}: \landdownint \tag{56.17}$$

57 eucal and euscript.sty

These packages should be part of your local TeX installation, because they come with the AMSmath packages. Otherwise get them from CTAN[31]. They support a scriptwriting of only upper letters

[31] CTAN://fonts/amsfonts/latex/euscript.sty

`\mathscr{...}` $\mathscr{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$

Read the documentation of the docs for the interdependence to the `\mathcal` command. For the above example the package `eucal.sty` was loaded with the option `mathscr`.

58 exscale

The following formula is written with the default fontsize where everything looks more or less well:

$$\int_{-1}^{+1}\frac{f(x)}{\sqrt{1-x^2}}\,dx \approx \frac{\pi}{n}\sum_{i=1}^{n} f\left(\cos\left(\frac{2i-1}{2n}\right)\right)$$

Writing the same with the fontsize `\huge` gives a surprising result, which belongs to the historical development of LaTeX, the int and sum symbols are not stretched. This extreme fontsize is often needed for slides and not only written “just for fun”.

$$\mathrm{Z}\,{}^{+1}_{-1}\frac{f(x)}{\sqrt{1-x^2}}\,dx \approx \frac{\pi}{n}\overset{n}{\mathrm{X}}_{i=1} f \begin{matrix}0\\ \mathrm{B}\\ \mathrm{B}\\ \mathrm{B}\\ @\end{matrix}\cos\begin{matrix}0\\ \mathrm{B}\\ \mathrm{B}\\ \mathrm{B}\\ @\end{matrix}\frac{2i-1}{2n}\begin{matrix}1\\ \mathrm{C}\\ \mathrm{C}\\ \mathrm{C}\\ \mathrm{A}\end{matrix}\begin{matrix}1\\ \mathrm{C}\\ \mathrm{C}\\ \mathrm{C}\\ \mathrm{A}\end{matrix}$$

Using the `exscale.sty`[32] package, which should be part of any local TeX installation, all symbols get the right size.

$$\int_{-1}^{+1}\frac{f(x)}{\sqrt{1-x^2}}\,dx \approx \frac{\pi}{n}\sum_{i=1}^{n} f\left(\cos\left(\frac{2i-1}{2n}\right)\right)$$

59 relsize

Often consecutives math operators are used, like two sum symbols, e.g.:

$$\sum\sum_{i=1}^{n} i^2$$

As seen the sums are of the same size. To increase the first operator size, someone can use the `\scalebox` macro from `graphicx` and writing an own macro `\Sum`, e.g.:

[32] CTAN://macros/latex/base/

```
\def\Sum{\ensuremath\mathop{\scalebox{1.2}{$\displaystyle\sum$}}}
\[ \Sum_{j=1}\sum_{i=1}^\infty i \]
```

$$\sum_{j=1}\sum_{i=1}^{\infty} i$$

Another solution is to use the **relsize**[33] package together with the **exscale** one. **relsize** defines a useful macro **mathlarger**:

$$\mathlarger{\sum}\sum_{i=1}^{n} i^2$$

```
\[ \mathlarger{\sum}\sum_{i=1}^n i^2 \]
```

60 xypic

The **xymatrix** macro is part of the **xypic** package[34] which can be loaded with several options which are not so important.[35].

$$\begin{tikzcd}
A \arrow[d, dash, squiggly] \arrow[dr, dash] & B & C \\
D \arrow[r, dash, dotted] & E \arrow[r, dash, squiggly] & F \arrow[dl, dash, squiggly] \\
G & H & I
\end{tikzcd} \tag{60.1}$$

This matrix was created with

```
\[
\xymatrix{ A\POS [];[d]**\dir {~},[];[dr]**\dir {-} & B & C\\
 D & E\POS [];[l]**\dir {.},[];[r]**\dir {~} & F\POS [];[dl]**\dir {~}\\
 G & H & I}
\]
```

[33] CTAN://macros/latex/ltxmisc/

[34] CTAN://macros/generic/diagrams/xypic/xy-3.7/

[35] For more information look at the style file **xy.sty**, which is often saved in **/usr/share/texmf/tex/generic**

Part V
Special symbols

In this section there are only those symbols defined, which are not part of the list of all available symbols: CTAN://info/symbols/comprehensive/symbols-a4.pdf. LaTeX itself defines with `fontmath.ltx` the following special symbols for using inside math:

Name	Meaning
`\mathparagraph`	¶
`\mathsection`	§
`\mathdollar`	$
`\mathsterling`	£
`\mathunderscore`	_
`\mathellipsis`	...

Table 21: Predefined math symbols from `fontmath.ltx`

61 Integral symbols

Name	Symbol
`\dashint`	⨍
`\ddashint`	⨎
`\clockint`	∲
`\counterint`	∳

For all new integral symbols limits can be used in the usual way:

$$\ddashint_0 1 = \dashint_1 0 < \oint\limits_{-\infty}^{\infty} = \clockint\counterint_A \tag{61.1}$$

```latex
\ddashint_01=\dashint_10<\oint\limits_{-\infty}^\infty = \clockint\counterint_A
```

Put the following definitions into the preamble to use one or all of these new integral symbols.

```latex
\def\Xint#1{\mathchoice
   {\XXint\displaystyle\textstyle{#1}}%
   {\XXint\textstyle\scriptstyle{#1}}%
   {\XXint\scriptstyle\scriptscriptstyle{#1}}%
   {\XXint\scriptscriptstyle\scriptscriptstyle{#1}}%
   \!\int}
\def\XXint#1#2#3{{\setbox0=\hbox{$#1{#2#3}{\int}$}
     \vcenter{\hbox{$#2#3$}}\kern-.5\wd0}}
```

```
\def\ddashint{\Xint=}
\def\dashint{\Xint-}
\def\clockint{\Xint\circlearrowright} % GOOD!
\def\counterint{\Xint\rotcirclearrowleft} % Good for Computer Modern!
\def\rotcirclearrowleft{\mathpalette{\RotLSymbol{-30}}\circlearrowleft}
\def\RotLSymbol#1#2#3{\rotatebox[origin=c]{#1}{$#2#3$}}
```

62 Harpoons

LaTeX knows no stretchable harpoon symbols, like `\xrightarrow`. The following code defines several harpoon symbols.

\xrightharpoondown
\xrightharpoonup
\xleftharpoondown
\xleftharpoonup
\xleftrightharpoons
\xrightleftharpoons

```
\def\rightharpoondownfill@{%
 \arrowfill@\relbar\relbar\rightharpoondown}
\def\rightharpoonupfill@{%
 \arrowfill@\relbar\relbar\rightharpoonup}
\def\leftharpoondownfill@{%
 \arrowfill@\leftharpoondown\relbar\relbar}
\def\leftharpoonupfill@{%
 \arrowfill@\leftharpoonup\relbar\relbar}
\newcommand{\xrightharpoondown}[2][]{%
 \ext@arrow 0359\rightharpoondownfill@{#1}{#2}}
\newcommand{\xrightharpoonup}[2][]{%
 \ext@arrow 0359\rightharpoonupfill@{#1}{#2}}
\newcommand{\xleftharpoondown}[2][]{%
 \ext@arrow 3095\leftharpoondownfill@{#1}{#2}}
\newcommand{\xleftharpoonup}[2][]{%
 \ext@arrow 3095\leftharpoonupfill@{#1}{#2}}
\newcommand{\xleftrightharpoons}[2][]{\mathrel{%
 \raise.22ex\hbox{%
  $\ext@arrow 3095\leftharpoonupfill@{\phantom{#1}}{#2}$}%
 \setbox0=\hbox{%
  $\ext@arrow 0359\rightharpoondownfill@{#1}{\phantom{#2}}$}%
 \kern-\wd0 \lower.22ex\box0}%
}
\newcommand{\xrightleftharpoons}[2][]{\mathrel{%
 \raise.22ex\hbox{%
  $\ext@arrow 3095\rightharpoonupfill@{\phantom{#1}}{#2}$}%
 \setbox0=\hbox{%
  $\ext@arrow 0359\leftharpoondownfill@{#1}{\phantom{#2}}$}%
 \kern-\wd0 \lower.22ex\box0}%
}
```

`\xrightharpoondown[under]{over}`	$\xrightharpoondown[under]{over}$
`\xrightharpoonup[under]{over}`	$\xrightharpoonup[under]{over}$
`\xleftharpoondown[under]{over}`	$\xleftharpoondown[under]{over}$
`\xleftharpoonup[under]{over}`	$\xleftharpoonup[under]{over}$
`\xleftrightharpoons[under]{over}`	$\xleftrightharpoons[under]{over}$
`\xrightleftharpoons[under]{over}`	$\xrightleftharpoons[under]{over}$

63 Bijective mapping arrow

To get something like ⤖ we can define:

```
\def\bijmap{%
 \ensuremath{%
  \mathrlap{\rightarrowtail}\rightarrow%
 }%
}
```

This uses the `\mathrlap` definition from section 35.2 on page 69. With this definition a huge symbol is also possible: `{\Huge\bijmap}` ⤖.

64 Stacked equal sign

There are several symbols stacked with an equal sign, e.g. `\doteq`, `\equiv` or `\cong` ($\doteq$, $\equiv$, $\cong$). But there are still some missing, which are shown in table 22 and the following definitions.

`\eqdef`	$\stackrel{\mathrm{def}}{=}$
`\eqexcl`	$\stackrel{!}{=}$
`\eqhat`	$\widehat{=}$

Table 22: New symbols in combination with the equal sign

```
\newcommand{\eqdef}{\ensuremath{\stackrel{\mathrm{def}}{=}}}
\newcommand{\eqexcl}{\ensuremath{\stackrel{\mathrm{def}}{!}}}
\newcommand{\eqhat}{\ensuremath{\widehat{=}}}
```

65 Other symbols

```
\newcommand*{\threesim}{%
  \mathrel{\vcenter{\offinterlineskip
  \hbox{$\sim$}\vskip-.35ex\hbox{$\sim$}\vskip-.35ex\hbox{$\sim$}}}}
$\threesim ABC$
```

≋ ABC

Part VI
Examples

66 Identity matrix

There are several possibilities to write this matrix. Here is a solution with the default array environment.

$$\left(\begin{array}{ccccc}
1\\
 & 1 & & \text{\huge 0}\\
 & & 1\\
 & \text{\huge 0} & & 1\\
 & & & & 1
\end{array}\right)$$

```
\[
\left(\begin{array}{ccccc}
1\\
 & 1 &  & \text{{\huge{0}}}\\
 &  & 1\\
 & \text{{\huge{0}}} &  & 1\\
 &  &  &  & 1\end{array}\right)
\]
```

67 Cases structure

Sometimes it is better to use the array environment instead of amsmaths cases environment. To get optimal horizontal spacing for the conditions, there are two matrixes in series, one 3×1 followed by 3×3 matrix. To minimize the horizontal space around the variable z a

```
\addtolength{\arraycolsep}{-3pt}
```

is a useful command.

$$I(z)=\delta_{0}\left\{
\begin{array}{lcrcl}
D+z & \quad & -D & \le z\le & -p\\
D-\frac{1}{2}\left(p-\frac{z^{2}}{p}\right) & \quad & -p & \le z\le & p\\
D-z & \quad & p & \le z\le & D
\end{array}\right. \tag{67.1}$$

```
\addtolength{\arraycolsep}{-3pt}
I(z)=\delta_{0}\left\{%
 \begin{array}{lcrcl}
  D+z & \quad & -D & \le z\le & -p\\
  D-\frac{1}{2}\left(p-\frac{z^{2}}{p}\right)%
    & \quad & -p & \le z\le & \phantom{-}p\\
  D-z & \quad & p & \le z\le & \phantom{-}D
 \end{array}\right.
\end{equation}
```

The `\phantom` command replaces exactly that place with whitespace which the argument needs.

67.1 Cases with numbered lines

This is not possible in an easy way, because `cases` uses the array environment for typesetting which has by default no numbering. However, there are some tricky ways to get numbered lines. The following three examples use the `tabular`, the `tabularx` and the `array` environment.

$$\text{some text hier}\begin{cases} x = 2 & \text{if } y > 2 \\ x = 3 & \text{if } y \le 2 \end{cases} \qquad \begin{matrix}(67.2)\\(67.3)\end{matrix}$$

```
\begin{tabular}{rc}
\ldelim\{{2}{2.75cm}[some text hier] &
   \parbox{{\linewidth-3cm-4\tabcolsep}}{
   \vspace*{1ex}
   \begin{flalign}
     x & = 2\quad\text{if }y >2 &\\
     x & = 3\quad\text{if }y \le 2&
   \end{flalign}}
\end{tabular}
```

$$\text{some text hier}\begin{cases} x = 2 & \text{if } y > 2 \\ x = 3 & \text{if } y \le 2 \end{cases} \qquad \begin{matrix}(67.4)\\(67.5)\end{matrix}$$

```
\begin{tabularx}{\linewidth}{rXc}
\ldelim\{{2}{2.75cm}[some text hier]
  & $ x = 2\quad\text{if }y > 2  $ & \refstepcounter{equation}(\theequation)\\
  & $ x = 3\quad\text{if }y \le 2$ &\refstepcounter{equation}(\theequation)
\end{tabularx}
```

$$\text{some text hier}\begin{cases} x = 2 & \text{if } y > 2 \qquad (67.6) \\ x = 3 & \text{if } y \le 2 \qquad (67.7) \end{cases}$$

```
\[
\begin{array}{rc@{\qquad}c}
\ldelim\{{2}{2.75cm}[some text hier]
  & x = 2\quad\text{if }y > 2  & \refstepcounter{equation}(\theequation)\\
  & x = 3\quad\text{if }y \le 2& \refstepcounter{equation}(\theequation)
\end{array}
\]
```

68 Arrays

There is a general rule that a lot of mathematical stuff should be divided in smaller pieces. But sometimes it is difficult to get a nice horizontal alignment when splitting a formula. The following ones uses the `array` environment to get a proper alignment.

68.1 Quadratic equation

$$\begin{array}{rcll}
y & = & x^{2}+bx+c\\
& = & x^{2}+2\cdot\dfrac{b}{2}x+c\\
& = & \underbrace{x^{2}+2\cdot\dfrac{b}{2}x+\left(\dfrac{b}{2}\right)^{2}}_{}-\left(\dfrac{b}{2}\right)^{2}+c\\
& & \qquad\left(x+\dfrac{b}{2}\right)^{2}\\
& = & \left(x+\dfrac{b}{2}\right)^{2}-\left(\dfrac{b}{2}\right)^{2}+c & \left|+\left(\dfrac{b}{2}\right)^{2}-c\right.\\
y+\left(\dfrac{b}{2}\right)^{2}-c & = & \left(x+\dfrac{b}{2}\right)^{2} & \left|(\text{Scheitelpunktform})\right.\\
y-y_{S} & = & (x-x_{S})^{2}\\
S(x_{S};y_{S}) & \,\text{bzw.}\, & S\left(-\dfrac{b}{2};\,\left(\dfrac{b}{2}\right)^{2}-c\right)
\end{array} \tag{68.1}$$

```
\begin{equation}
\begin{array}{rcll}
y & = & x^{2}+bx+c\\
  & = & x^{2}+2\cdot{\displaystyle\frac{b}{2}x+c}\\
  & = & \underbrace{x^{2}+2\cdot\frac{b}{2}x+\left(\frac{b}{2}\right)^{2}}-{\displaystyle%
 \left(\frac{b}{2}\right)^{2}+c}\\
 &  & \qquad\left(x+{\displaystyle \frac{b}{2}}\right)^{2}\\
 & = & \left(x+{\displaystyle \frac{b}{2}}\right)^{2}-\left({\displaystyle%
\frac{b}{2}}\right)^{2}+c & \left|+\left({\displaystyle%
\frac{b}{2}}\right)^{2}-c\right.\\
y+\left({\displaystyle \frac{b}{2}}\right)^{2}-c & = & \left(x+{\displaystyle%
\frac{b}{2}}\right)^{2} & \left|(\textrm{Scheitelpunktform})\right.\\
y-y_{S} & = & (x-x_{S})^{2}\\
S(x_{S};y_{S}) & \,\textrm{bzw.}\, & S\left(-{\displaystyle%
\frac{b}{2};\,\left({\displaystyle \frac{b}{2}}\right)^{2}-c}\right)
\end{array}
\end{equation}
```

68.2 Vectors and matrices

$$
\begin{array}{rcl}
\underline{RS} & = & \left(\begin{array}{cccccccc}
01 & a4 & 55 & 87 & 5a & 58 & db & 9e\\
a4 & 56 & 82 & f3 & 1e & c6 & 68 & e5\\
02 & a1 & fc & c1 & 47 & ae & 3d & 19\\
a4 & 55 & 87 & 5a & 58 & db & 9e & 03\end{array}\right)\\
\\
\left(\begin{array}{c}
s_{i,0}\\
s_{i,1}\\
s_{i,2}\\
s_{i,3}
\end{array}\right) & = & \underline{RS}\cdot
\left(\begin{array}{c}
m_{8i+0}\\
m_{8i+1}\\
\cdots\\
m_{8i+6}\\
m_{8i+7}
\end{array}\right)\\
\\
S_{i} & = & \sum_{j=0}^{3}s_{i,j}\cdot2^{8j}\qquad i=0,1,...,k-1\\
\\
S & = & \left(S_{k-1},S_{k-2},...,S_{1},S_{0}\right)
\end{array}
\tag{68.2}
$$

```latex
\begin{equation}
\begin{array}{rcl}
\underline{RS} & = & \left(\begin{array}{cccccccc}
01 & a4 & 55 & 87 & 5a & 58 & db & 9e\\
a4 & 56 & 82 & f3 & 1e & c6 & 68 & e5\\
02 & a1 & fc & c1 & 47 & ae & 3d & 19\\
a4 & 55 & 87 & 5a & 58 & db & 9e & 03\end{array}\right)\\
\\
\left(\begin{array}{c}
 s_{i,0}\\
 s_{i,1}\\
 s_{i,2}\\
 s_{i,3}
\end{array}\right) & = & \underline{RS}\cdot%
\left(\begin{array}{c}
 m_{8i+0}\\
 m_{8i+1}\\
 \cdots\\
 m_{8i+6}\\
 m_{8i+7}
\end{array}\right)\\
\\
S_{i} & = & \sum_{j=0}^{3}s_{i,j}\cdot2^{8j}\qquad i=0,1,...,k-1\\
\\
S & = & \left(S_{k-1},S_{k-2},...,S_{1},S_{0}\right)
\end{array}
\end{equation}
```

68.3 Cases with (eqn)array environment

This solution is important when $\mathcal{A}_{\mathcal{M}}\mathcal{S}$math couldn't be used.

$$\lim\limits_{n->\infty}q^{n}=\left\{\begin{array}{lc@{\kern2pt}c@{\kern2pt}r}\textrm{divergent}\ & q & \le & -1\\ 0 & |q| & < & 1\\ 1 & q & = & 1\\ \infty & q & > & 1\end{array}\right.$$

```
$\lim\limits_{n->\infty}q^{n}=\left\{%
 \begin{array}{lc@{\kern2pt}c@{\kern2pt}r}
  \textrm{divergent}\  & q & \le & -1\\
  0 & |q| & < & 1\\
  1 &  q  & = & 1\\
  \infty  & q & > & 1
 \end{array}\right.$
```

68.4 Arrays inside arrays

The array environment is a powerful one because it can be nested in several ways:

$$\left(\begin{array}{c@{}c@{}c}\begin{array}{|cc|}\hline a_{11} & a_{12} \\ a_{21} & a_{22} \\\hline\end{array} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \begin{array}{|ccc|}\hline b_{11} & b_{12} & b_{13}\\ b_{21} & b_{22} & b_{23}\\ b_{31} & b_{32} & b_{33}\\\hline\end{array} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \begin{array}{|cc|}\hline c_{11} & c_{12} \\ c_{21} & c_{22} \\\hline\end{array} \\\end{array}\right)$$

```
\[
\left(
\begin{array}{c@{}c@{}c}
 \begin{array}{|cc|}\hline
  a_{11} & a_{12} \\
  a_{21} & a_{22} \\\hline
 \end{array} & \mathbf{0} & \mathbf{0} \\
 \mathbf{0} &
 \begin{array}{|ccc|}\hline
  b_{11} & b_{12} & b_{13}\\
  b_{21} & b_{22} & b_{23}\\
  b_{31} & b_{32} & b_{33}\\\hline
 \end{array} & \mathbf{0} \\
 \mathbf{0} & \mathbf{0} &
 \begin{array}{|cc|}\hline
  c_{11} & c_{12} \\
  c_{21} & c_{22} \\\hline
 \end{array} \\
\end{array}
\right)
\]
```

$$Y^1 = \begin{array}{c} \left[\begin{array}{rrrr} 0 & 0 & 1 & 0\\ 1 & 0 & 1 & 0\\ 1 & 1 & 1 & 1 \end{array}\right]\\[3ex]\hline \begin{array}{rrrr} 2 & 1 & 3 & 1 \end{array} \end{array}$$

```
\[
Y^1=
\begin{array}{c}
 \null\\[1ex]% only vor vertical alignment
 \left[\begin{array}{rrrr}
  0 & 0 & 1 & 0\\
  1 & 0 & 1 & 0\\
  1 & 1 & 1 & 1
 \end{array}\right]\\[3ex]\hline
 \begin{array}{rrrr}
%  \hdotsfor{4}\\%( needs \AmSmath) instead of \\[3ex]\hline
  2 & 1 &3 & 1
 \end{array}
\end{array}
\]
```

68.5 Colored cells

In general there is no difference in coloring tabular or array cells. The following example shows how one can put colors in rows, columns and cells.

$$\left[\begin{array}{ccccc}
h_{k,1,0}(n) & h_{k,1,1}(n) & h_{k,1,2}(n) & 0 & 0\\
h_{k,2,0}(n) & h_{k,2,1}(n) & h_{k,2,2}(n) & 0 & 0\\
h_{k,3,0}(n) & h_{k,3,1}(n) & h_{k,3,2}(n) & 0 & 0\\
h_{k,4,0}(n) & h_{k,4,1}(n) & h_{k,4,2}(n) & 0 & 0\\
0 & h_{k,1,0}(n-1) & h_{k,1,1}(n-1) & h_{k,1,2}(n-1) & 0\\
0 & h_{k,2,0}(n-1) & h_{k,2,1}(n-1) & h_{k,2,2}(n-1) & 0\\
0 & h_{k,3,0}(n-1) & h_{k,3,1}(n-1) & h_{k,3,2}(n-1) & 0\\
0 & h_{k,4,0}(n-1) & h_{k,4,1}(n-1) & h_{k,4,2}(n-1) & 0\\
0 & 0 & h_{k,1,0}(n-2) & h_{k,1,1}(n-2) & h_{k,1,2}(n-2)\\
0 & 0 & h_{k,2,0}(n-2) & h_{k,2,1}(n-2) & h_{k,2,2}(n-2)\\
0 & 0 & h_{k,3,0}(n-2) & h_{k,3,1}(n-2) & h_{k,3,2}(n-2)\\
0 & 0 & h_{k,4,0}(n-2) & h_{k,4,1}(n-2) & h_{k,4,2}(n-2)
\end{array}\right]_{12\times 5}$$

```
...
\usepackage{array}
\usepackage{colortbl}
\definecolor{umbra}{rgb}{0.8,0.8,0.5}
\def\zero{\multicolumn{1}{>{\columncolor{white}}c}{0}}
```

```
\def\colCell#1#2{\multicolumn{1}{>{\columncolor{#1}}c}{#2}}
\begin{document}
\[\left[\,
\begin{array}{*{5}{>{\columncolor[gray]{0.95}}c}}
  h_{k,1,0}(n) & h_{k,1,1}(n) & h_{k,1,2}(n) & \zero & \zero\\
  h_{k,2,0}(n) & h_{k,2,1}(n) & h_{k,2,2}(n) & \zero & \zero\\
  h_{k,3,0}(n) & h_{k,3,1}(n) & h_{k,3,2}(n) & \zero & \zero\\
  h_{k,4,0}(n)} & \colCell{umbra}{h_{k,4,1}(n)} & h_{k,4,2}(n) & \zero & \zero\\
  \zero & h_{k,1,0}(n-1) & h_{k,1,1}(n-1) & h_{k,1,2}(n-1) & \zero\\
  \zero & h_{k,2,0}(n-1) & h_{k,2,1}(n-1) & h_{k,2,2}(n-1) & \zero\\
  \zero & h_{k,3,0}(n-1) & h_{k,3,1}(n-1) & h_{k,3,2}(n-1) & \zero\\
  \zero & \colCell{umbra}{h_{k,4,0}(n-1)} & h_{k,4,1}(n-1) & h_{k,4,2}(n-1) & \zero\\
  \zero & \zero & h_{k,1,0}(n-2) & h_{k,1,1}(n-2) & h_{k,1,2}(n-2)\\
  \zero & \zero & h_{k,2,0}(n-2) & h_{k,2,1}(n-2) & h_{k,2,2}(n-2)\\
  \zero & \zero & h_{k,3,0}(n-2) & h_{k,3,1}(n-2) & h_{k,3,2}(n-2)\\
  \zero & \zero & h_{k,4,0}(n-2) & h_{k,4,1}(n-2) & h_{k,4,2}(n-2)
\end{array} \,\right]_{12\times 5}\]
...
```

69 Over- and underbraces

69.1 Braces and roots

To put an underbrace in a root without enlarging the root symbol is possible with the `\makebox` macro:

$$z = \underbrace{\sqrt{x^2+y^2}}_{=z^2}$$

```
\[
z =\;\;\underbrace{%
  \makebox[\widthof{~$x^2+y^2$}][r]{%
     $\sqrt{x^2+y^2}$}}_{=z^2}
\]
```

69.2 Overlapping braces

Overlapping under- and overbraces like $\underbrace{\qquad\quad\overbrace{\quad\;}^{}}_{u1}\!\!\!\!\!\!\overbrace{\qquad\qquad\quad}^{o}\!\!\!\!\!\!\underbrace{\overbrace{\quad\;}^{}\qquad\quad}_{u2}$ needs some tricky code, because we cannot have parts of the argument inside `overbrace` and also `underbrace`. The following equation 69.1 is an example for such a construction:

$$\begin{aligned}
y &= 2x^2 - 3x + 5\\
&= 2\left(\underbrace{x^2 - \frac{3}{2}x + \left(\frac{3}{4}\right)^2}\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\overbrace{\qquad\qquad\qquad\qquad}^{=0}\!\!\!\!\!\!\!\!\!\!\!\!\!\!\!\underbrace{- \left(\frac{3}{4}\right)^2 + \frac{5}{2}}\right) \qquad (69.1)\\
&= 2\left(\qquad\left(x - \frac{3}{4}\right)^2 \qquad + \ \frac{31}{16}\qquad\right)\\
y-\frac{31}{8} &= 2\left(x-\frac{3}{4}\right)^2
\end{aligned}$$

```
y &= 2x^2 -3x +5\nonumber\\
  & \hphantom{= \ 2\left(x^2-\frac{3}{2}\,x\right. }%
        \textcolor{blue}{%
             \overbrace{\hphantom{+\left(\frac{3}{4}\right)^2- %
                 \left(\frac{3}{4}\right)^2}}^{=0}}\nonumber\\[-11pt]
  &= 2\left(\textcolor{red}{%
     \underbrace{%
         x^2-\frac{3}{2}\,x + \left(\frac{3}{4}\right)^2}%
     }%
     \underbrace{%
        - \left(\frac{3}{4}\right)^2 + \frac{5}{2}}%
     \right)\\
  &= 2\left(\qquad\textcolor{red}{\left(x-\frac{3}{4}\right)^2}
     \qquad + \ \frac{31}{16}\qquad\right)\nonumber\\
y\textcolor{blue}{-\frac{31}{8}}
  &= 2\left(x\textcolor{cyan}{-\frac{3}{4}}\right)^2\nonumber
\end{align}
```

69.3 Vertical alignment of different braces

When having several braces in one formula line, then it looks better when all braces are also on the same line, e.g.:

$$\binom{x_R}{y_R} = \underbrace{r\vphantom{\binom{A}{B}}}_{\text{scaling}}\cdot\underbrace{\begin{pmatrix} \sin\gamma & -\cos\gamma \\ \cos\gamma & \sin\gamma \end{pmatrix}}_{\text{Rotation}}\binom{x_K}{y_K} + \underbrace{\binom{t_x}{t_y}}_{\text{Translation}} \tag{69.2}$$

```latex
\begin{equation}
 \binom{x_R}{y_R} = \underbrace{r\vphantom{\binom{A}{B}}}_{\text{Skalierung}}\cdot%
    \underbrace{%
  \begin{pmatrix}
      \sin \gamma & -\cos \gamma \\
      \cos \gamma & \sin \gamma \\
  \end{pmatrix}%
    }_{\text{Rotation}}
 \binom{x_K}{y_K} +
    \underbrace{\binom{t_x}{t_y}}_{\text{Translation}}
\end{equation}
```

It is again the \vphantom macro which reserves the needed vertical space. Nevertheless the horizontal space around the r of the first underbrace and the last + should be decreased to get a better typesetting. This is possible with \hspace or simply \kern:

$$\binom{x_R}{y_R} = \kern-10pt\underbrace{r\vphantom{\binom{A}{B}}}_{\text{Skalierung}}\kern-10pt\cdot\underbrace{\begin{pmatrix} \sin\gamma & -\cos\gamma \\ \cos\gamma & \sin\gamma \end{pmatrix}}_{\text{Rotation}}\binom{x_K}{y_K} +\kern-5pt \underbrace{\binom{t_x}{t_y}}_{\text{Translation}}$$

```latex
$ \binom{x_R}{y_R} = %
    \kern-10pt\underbrace{r\vphantom{\binom{A}{B}}}_{\text{Skalierung}}\kern-10pt%
    \cdot\underbrace{%
      \begin{pmatrix}
        \sin \gamma & -\cos \gamma \\
        \cos \gamma & \sin \gamma \\
      \end{pmatrix}%
    }_{\text{Rotation}}
  \binom{x_K}{y_K} +\kern-5pt%
  \underbrace{\binom{t_x}{t_y}}_{\text{Translation}} $
```

69.4 Vertical and horizontal alignment

The forgoing example simply uses \hspace to decrease the horizontal width between two underbraces. This maybe okay for a single solution, but in general it is better to have some code which works in any case.

The following example looks simple but it need some tricky code to get vertical and horizontal alignment.

$$\frac{300}{5069}\underbrace{\longmapsto}_{\substack{\Delta a=271\\[2pt] \Delta b=4579\\[2pt] \text{1 iteration}}}\frac{29}{490}\underbrace{\longmapsto\frac{19}{321}\longmapsto}_{\substack{\Delta a=10\ =\langle271\rangle_{29}\\[2pt] \Delta b=169=\langle4579\rangle_{490}\\[2pt] \text{2 iterations}}}\frac{9}{152}\underbrace{\longmapsto\frac{8}{135}\longmapsto\dots\longmapsto}_{\substack{\Delta a=1\ =\langle10\rangle_{9}\\[2pt] \Delta b=17=\langle169\rangle_{152}\\[2pt] \text{8 iterations}}}\frac{1}{16}\underbrace{\longmapsto\dots\longmapsto}_{\substack{\Delta a=0=\langle1\rangle_{1}\\[2pt] \Delta b=1=\langle17\rangle_{16}\\[2pt] \text{8 iterations}}}\frac{1}{1}$$

It uses the in section 35.2 on page 69 defined macro `\mathclap`, which gives a better result. It is also possible to use `\makebox[0pt]{...}` but it works only in text mode and this needs some more `$...$`.

```
\def\num#1{\hphantom{#1}}
\def\vsp{\vphantom{\rangle_1}}

\begin{equation*}
    \frac{300}{5069}%
    \underbrace{\longmapsto\vphantom{\frac{1}{1}}}_{%
    \mathclap{\substack{%
      \Delta a=271\num9\vsp \\[2pt]
      \Delta b=4579\vsp\\[2pt]
      \text{$1$ iteration}%
    }}} \frac{29}{490}%
 \underbrace{\longmapsto \frac{19}{321}\longmapsto}_{%
      \mathclap{\substack{%
      \Delta a=10\num{9}=\langle271\rangle_{29}\num{20}\\[2pt]
         \Delta b=169=\langle4579\rangle_{490}\\[2pt]
         \text{$2$ iterations}
      }}} \frac{9}{152}
    \underbrace{\longmapsto \frac{8}{135}\longmapsto\dots\longmapsto}_{%
      \substack{%
     \Delta a=1\num{7}=\langle10\rangle_{9}\num{119}\\[2pt]
     \Delta b=17=\langle169\rangle_{152}\\[2pt]
     \text{$8$ iterations}
      }} \frac{1}{16}
    \underbrace{\longmapsto\dots\longmapsto\vphantom{\frac{8}{135}}}_{%
      \substack{%
      \Delta a=0=\langle1\rangle_{1}\num{76} \\[2pt]
      \Delta b=1=\langle17\rangle_{16} \\[2pt]
         \text{$8$ iterations}
      }} \frac{1}{1}
\end{equation*}
```

70 Integrals

The *first theorem of Green* is:

$$\iiint\limits_{\mathcal{G}}\left[u\nabla^2 v+(\nabla u,\nabla v)\right]d^3V=\oiint\limits_{\mathcal{S}}u\frac{\partial v}{\partial n}d^2A$$

The *second theorem of Green* is:

$$\underset{\mathcal{G}}{\iiint}\left[u\nabla^{2}v-v\nabla^{2}u\right]d^{3}V=\underset{\mathcal{S}}{\oiint}\left(u\frac{\partial v}{\partial n}-v\frac{\partial u}{\partial n}\right)d^{2}A$$

They are both written with the **esint.sty** package[36], which gives nice integral symbols. The LaTeX code for the first equation is:

```
\[
\underset{\mathcal{G}\quad}\iiint\!%
 \left[u\nabla^{2}v+\left(\nabla u,\nabla v\right)\right]d^{3}V%
 =\underset{\mathcal{S}\quad}\oiint u\Q{v}{n}d^{2}A
\]
```

with the following definition in the preamble for the partial derivation:

```
\def\Q#1#2{\frac{\partial#1}{\partial #2}}
```

which makes things easier to write.

71 Vertical alignment

71.1 Example 1

Sometimes it maybe useful to have a vertical alignment over the whole page with a mix of formulas and text. Section 37 shows the use of `\intertext`. There is another trick to get all formulas vertical aligned. Let's have the following formulas distributed over the whole page:

$$\begin{aligned}
f(x)&=a\\
g(x)&=x2-4x\\
f(x)-g(x)&=x2+x3+x\\
g&=x2+x3+x4+x5+b
\end{aligned}$$

They all have a different length of the left and right side. Now we want to write some text and other objects between them, but let the alignment untouched. We choose the longest left and the longest right side and take them for scaling with the `\hphantom` command:

```
\hphantom{\mbox{$f(x)-g(x)$}} & \hphantom{\mbox{$= x^2+x^3+x^4+x^5+b$}}
```

This is the first (empty) line in every equation where now all other lines are aligned to this one. For example:

[36] See section 70 on the previous page.

---

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

$$
\begin{aligned}
f(x) &= a \qquad (71.1)\\
g(x) &= x^2 - 4x \qquad (71.2)
\end{aligned}
$$

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

$$f(x) - g(x) = x^2 + x^3 + x \tag{71.3}$$

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

$$g(x) = x^2 + x^3 + x^4 + x^5 + b \tag{71.4}$$

blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah blah

---

The phantom line is empty but leaves the vertical space for a line. This could be corrected with decreasing the `\abovedisplayshortskip` length and restoring them after the whole sequence of commands. The code of the above looks like:

```latex
\newcommand{\x}{blah blah blah blah blah blah blah blah }
\addtolength{\abovedisplayshortskip}{-1cm} % decrease the skip
\addtolength{\abovedisplayskip}{-1cm}
\x\x\x\x\x
\begin{align}
\hphantom{\mbox{$f(x)-g(x)$}} & \hphantom{\mbox{$= x^2+x^3+x^4+x^5+b$}}\nonumber\\
f(x) &= a\\
g(x) &= x^2-4x
\end{align}

\x\x\x\x\x
\begin{align}
\hphantom{\mbox{$f(x)-g(x)$}} & \hphantom{\mbox{$= x^2+x^3+x^4+x^5+b$}}\nonumber\\
f(x)-g(x) &= x^2+x^3+x
\end{align}
\x\x\x\x\x

```

```
\begin{align}
\hphantom{\mbox{$f(x)-g(x)$}} & \hphantom{\mbox{$= x^2+x^3+x^4+x^5+b$}}\nonumber\\
g(x) &= x^2+x^3+x^4+x^5+b
\end{align}
\x\x\x\x\x
% restore old values
\addtolength{\abovedisplayshortskip}{1cm}
\addtolength{\abovedisplayskip}{1cm}
```

Another case of aligning equations inside an itemize environment is the following one. With the `\makebox` macro one can have the same size on the left side of the equal sign to get a vertical alignment.

- first function
  $\displaystyle P_1 = \sum_a \in A$
- but another one
  $\sin\left(P_1\right) = blabla$
- or perhaps
  $P_3 + P_2 - P_1 = blablub$

```
\newsavebox\lW
\sbox\lW{$P_{3}+P_{2}-P_{1}$}

\begin{itemize}
\item first function \\
  $\displaystyle\makebox[\wd\lW][r]{$P_1$}=\sum_a \in A$
\item but another one \\
  $\makebox[\wd\lW][r]{$\sin\left(P_1\right)$}=blabla$
\item or perhaps \\
  $P_{3}+P_{2}-P_{1}=blablub$
\end{itemize}
```

71.2 Example 2

This one comes from Hartmut Henkel and offers a special form of placing additional text between the equation and the equationnumber. This makes only sense when you load the documentclass with the option `fleqn`. The example places the additional text at `0.5\textwidth`, changing this value is no problem.

---

text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text

$$\varepsilon = \frac{E \cdot 4 \cdot \pi \cdot \varepsilon_0 \cdot a_0 \cdot \left(Z_i^{\frac{2}{3}} + Z_{Si}^{\frac{2}{3}} \right)^{-\frac{1}{2}}}{Z_i \cdot Z_{Si} \cdot e2 \cdot \left(1 + \frac{m_i}{m_{Si}} \right)}; \tag{71.5}$$

a_0	Bohrscher Radius (= 0,53 Å)
e	Elementarladung
N_{si}	Anzahl der Siliziumatome
	pro Einheitsvolumen
m	Atomgewicht
Z	Kernladungszahl

$$a2 + b2 = c2 \qquad \text{abc} \tag{71.6}$$

$$z = 9 \tag{71.7}$$

text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text

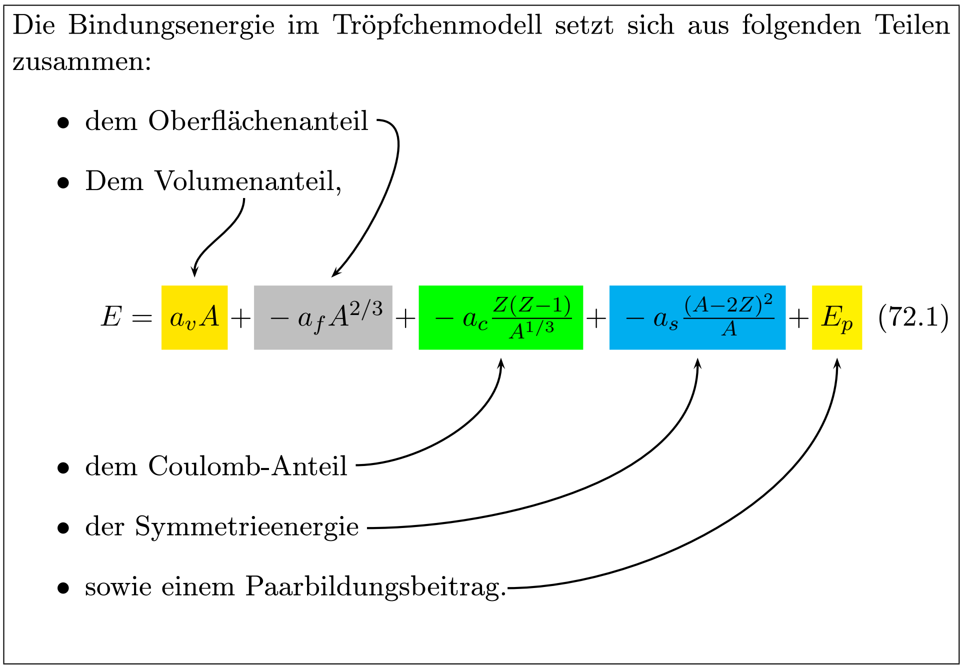
---

This solution works only with $\mathcal{AMS}$math, without you have to redefine the LaTeX macro, which creates the equation number.

```
\newsavebox{\myendhook} % hier gehen die Tabellen rein
\def\tagform@#1{{(\maketag@@@{\ignorespaces#1\unskip\@@italiccorr)}
  \makebox[0pt][r]{% hinter der Zeilennummer aufgehaengt
    \makebox[0.4\textwidth][l]{\usebox{\myendhook}}%
  }%
  \global\sbox{\myendhook}{}% Box wird geleert
}}

[ ... ]

\sbox{\myendhook}{%
\begin{footnotesize}%
\begin{tabular}{@{}ll}
$a_0$ & Bohrsche Radiuns ($\mathrm{= 0{,}53\,\mbox{\AA}}$)\\
$e$ & Elementarladung\\
$N_{si}$ & Anzahl der Siliziumatome\\
& pro Einheitsvolumen\\
$m$ & Atomgewicht\\
$Z$ & Kernladungszahl
\end{tabular}
\end{footnotesize}}

\begin{equation}
\varepsilon = \frac{E \cdot 4 \cdot \pi \cdot \varepsilon_{0}
\cdot a_0 \cdot \left( Z_i^{\frac{2}{3}} + Z_{Si}^{\frac{2}{3}}
\right)^{-\frac{1}{2}}} {Z_i \cdot Z_{Si} \cdot e2 \cdot \left( 1
+ \frac{m_i}{m_{Si}} \right)}\,;
\end{equation}

\sbox{\myendhook}{abc}

\begin{equation}
a2+b2=c2
\end{equation}

\begin{equation}
z = 9
\end{equation}
```

72 Node connections

This is a typical application for PSTricks and it needs the package `pst-node` and doesn't work with `pdflatex`. Use `VTeX`, `ps4pdf` or `ps2pdf`.



Die Bindungsenergie im Tröpfchenmodell setzt sich aus folgenden Teilen zusammen:

- dem Oberflächenanteil
- Dem Volumenanteil,

$$E = a_v A + -a_f A^{2/3} + -a_c \frac{Z(Z-1)}{A^{1/3}} + -a_s \frac{(A-2Z)^2}{A} + E_p \qquad (72.1)$$

- dem Coulomb-Anteil
- der Symmetrieenergie
- sowie einem Paarbildungsbeitrag.

```
\psset{nodesep=3pt}
\newrgbcolor{lila}{0.6 0.2 0.5}
\newrgbcolor{darkyellow}{1 0.9 0}
Die Bindungsenergie im Tröpfchenmodell setzt sich aus
folgenden Teilen zusammen:
\begin{itemize}
\item dem \rnode{b}{Oberflächenanteil}
\item Dem \rnode{a}{Volumenanteil},\\[1cm]
\def\xstrut{\vphantom{\frac{(A)^1}{(B)^1}}}
\begin{equation}
E =
\rnode[t]{ae}{\psframebox*[fillcolor=darkyellow,
  linestyle=none]{\xstrut a_vA}} +
\rnode[t]{be}{\psframebox*[fillcolor=lightgray,
  linestyle=none]{\xstrut -a_fA^{2/3}}} +
\rnode[t]{ce}{\psframebox*[fillcolor=green,
  linestyle=none]{\xstrut -a_c\frac{Z(Z-1)}{A^{1/3}}}} +
\rnode[t]{de}{\psframebox*[fillcolor=cyan,
  linestyle=none]{\xstrut -a_s\frac{(A-2Z)^2}{A}}} +
\rnode[t]{ee}{\psframebox*[fillcolor=yellow,
  linestyle=none]{\xstrut E_p}}
\end{equation}\\[0.25cm]
\item dem \rnode{c}{Coulomb-Anteil}
\item der \rnode{d}{Symmetrieenergie}
\item sowie einem \rnode{e}{Paarbildungsbeitrag}.
\end{itemize}
\nccurve[angleA=-90,angleB=90]{->}{a}{ae}
\nccurve[angleB=45]{->}{b}{be}
\nccurve[angleB=-90]{->}{c}{ce}
\nccurve[angleB=-90]{->}{d}{de}
\nccurve[angleB=-90]{->}{e}{ee}
```

73 Special placement of displayed equations

73.1 Formulas side by side

Sometimes it may be useful to have numbered formulas side by side like the following ones:

$$\oint Eds = 0 \qquad (73.1.a) \qquad \nabla \cdot B = 0 \qquad (73.1.b)$$

$$a = \frac{c}{d} \qquad (73.2.a) \qquad b = 1 \qquad (73.2.b)$$

$$c = 1 \qquad (73.3.a) \qquad \int 2xdx = x^2 + C \qquad (73.3.b)$$

And again a default display equation:

$$F(x) = \int_0^\infty \frac{1}{x}\,dx \tag{73.4}$$

```
\begin{mtabular}{*{2}{m{0.35\linewidth}m{0.15\linewidth}}}
\begin{align*} \oint E ds=0 \end{align*} & \eqnCnt %
 & \begin{align*} \nabla\cdot B=0 \end{align*} & \eqnCnt[\
   label{blah}]\\
\begin{align*} a =\frac{c}{d} \end{align*} & \eqnCnt %
 & \begin{align*} b = 1 \end{align*} & \eqnCnt\\
\begin{align*} c =1 \end{align*} & \eqnCnt[\label{blub}]
```

```
 & \begin{align*} \int 2x dx = x^2+C \end{align*} & \eqnCnt
\end{mtabular}
```

The new environment **mtabular** has two arguments, one optional and one which is the same than the one from the **tabular** environment. With the option `long` it is possible to have all the formulas in a `longtable` environment, which allows a pagebreak. The new macro `\eqnCnt` controls the counting of these equations as subequations for one tabular line. This macro can have an optional argument for a label. At least it counts the equations. If the equation number is not centered to the foregoing equation, then it needs some more horizontal space in the tabular column.

`\eqnCnt[<optional label>]`

The vertical space is controlled by the length **mtabskip**, which is by default `-1.25cm` and can be modified in the usual way.

To define all these macros write into the preamble:

```
\usepackage{amsmath}
\newcounter{subequation}
%
\newlength\mtabskip\mtabskip=-1.25cm
%
\newcommand\eqnCnt[1][]{%
 \refstepcounter{subequation}%
 \begin{align}#1\end{align}%
 \addtocounter{equation}{-1}%
}
\def\mtabLong{long}
\makeatletter
\newenvironment{mtabular}[2][\empty]{%
 \def\@xarraycr{%
  \stepcounter{equation}%
  \setcounter{subequation}{0}%
  \@ifnextchar[\@argarraycr{\@argarraycr[\mtabskip]}%
 }
 \let\theoldequation\theequation%
 \renewcommand\theequation{\theoldequation.\alph{subequation}}
 \edef\mtabOption{#1}
 \setcounter{subequation}{0}%
 \tabcolsep=0pt
 \ifx\mtabOption\mtabLong\longtable{#2}\else\tabular{#2}\fi%
}{%
 \ifx\mtabOption\mtabLong\endlongtable\else\endtabular\fi%
 \let\theequation\theoldequation%
 \stepcounter{equation}
}
\makeatother
```

As seen in equation 73.3.a and equation 73.1.b, everything is nonsense ... And the following tabular is defined as a longtable to enable pagebreaks.

$$\oint Eds = 0 \qquad (73.5.a) \qquad\qquad \nabla \cdot B = 0 \qquad (73.5.b)$$

$$a = \frac{c}{d} \qquad (73.6.a) \qquad\qquad b = 1 \qquad (73.6.b)$$

$$c = 1 \qquad (73.7.a) \qquad\qquad \int 2xdx = x^2 + C \qquad (73.7.b)$$

$$\oint Eds = 0 \qquad (73.8.a) \qquad\qquad \nabla \cdot B = 0 \qquad (73.8.b)$$

$$a = \frac{c}{d} \qquad (73.9.a) \qquad\qquad b = 1 \qquad (73.9.b)$$

$$c = 1 \qquad (73.10.a) \qquad\qquad \int 2xdx = x^2 + C \qquad (73.10.b)$$

$$\oint Eds = 0 \qquad (73.11.a) \qquad\qquad \nabla \cdot B = 0 \qquad (73.11.b)$$

$$a = \frac{c}{d} \qquad (73.12.a) \qquad\qquad b = 1 \qquad (73.12.b)$$

$$c = 1 \qquad (73.13.a) \qquad\qquad \int 2xdx = x^2 + C \qquad (73.13.b)$$

$$\oint Eds = 0 \qquad (73.14.a) \qquad\qquad \nabla \cdot B = 0 \qquad (73.14.b)$$

$$a = \frac{c}{d} \qquad (73.15.a) \qquad\qquad b = 1 \qquad (73.15.b)$$

$$c = 1 \qquad (73.16.a) \qquad\qquad \int 2xdx = x^2 + C \qquad (73.16.b)$$

As seen in equation 73.13.a and equation 73.11.b, everything is nonsense …

And again a default display equation:

$$F(x) = \int_0^\infty \frac{1}{x}\,dx \tag{73.17}$$

```
\begin{mtabular}[long]{*{2}{m{0.375\linewidth}m{0.125\linewidth}}}
\begin{align*} \oint E ds=0 \end{align*} & \eqnCnt %
 & \begin{align*} \nabla\cdot B=0 \end{align*} & \eqnCnt\\
\begin{align*} a =\frac{c}{d} \end{align*} & \eqnCnt %
 & \begin{align*} b = 1 \end{align*} & \eqnCnt\\
\begin{align*} c =1 \end{align*} & \eqnCnt
 & \begin{align*} \int 2x dx = x^2+C \end{align*} & \eqnCnt\\

[ ... ]
```

73.2 Formulas inside an itemize enviroment

Without any modification it is not possible to get a numbered equation at the same height as the symbol of the itemize environment. This depends to the `\abovedisplayskip`. The formula has to be raised up for exactly this length.

```
\def\itemMath#1{%
    \raisebox{-\abovedisplayshortskip}{%
        \parbox{0.75\linewidth}{%
            \begin{equation}#1\end{equation}}}}
%
\begin{itemize}
\item \itemMath{ f = l }
\item \itemMath{ g(x) = \int f(x) dx }
\end{itemize}
```

- $$f = l \tag{73.18}$$
- $$g(x) = \int f(x)dx \tag{73.19}$$

Part VII
Lists, bibliography and index

List of Figures

List of Tables

References

[1] Paul W. Abrahams, Karl Berry, and Kathryn Hargreaves. *TEX for the Impatient*. http://tug.org/ftp/tex/impatient/book.pdf, 2003.

[2] Claudio Beccari. Typesetting mathematics for science and technology according to iso 31/xi. *TUGboat Journal*, 18(1):39–47, 1997.

[3] Thierry Bouche. Diversity in math fonts. *TUGboat Journal*, 19(2):121–135, 1998.

[4] David Cobac. *Atelier documents mathématiques*. http://crdp.ac-lille.fr/crdp2003/archives/latex/Ateliers/Atelier2/Presentation4.pdf, 2004.

[5] David Cobac. *Ecrire des mathématiques avec LATEX*. http://crdp.ac-lille.fr/crdp2003/archives/latex/Ateliers/Atelier2/prepDocMaths.pdf, 2004.

[6] Michael Downes. *Technical Notes on the **amsmath** package*. American Mathematical Society, ftp://ftp.ams.org/pub/tex/doc/amsmath/technote.pdf, 1999.

[7] Michael Downes. *Short Math Guide for LATEX*. American Mathematical Society, http://www.ams.org/tex/short-math-guide.html, 2002.

[8] Victor Eijkhout. *TEX by Topic*. http://www.eijkhout.net/tbt/, 1992.

[9] J. Anthony Fitzgerald. *Web Math Formulas Using TEX*. http://www.unb.ca/web/Sample/math/, 1997.

[10] Michel Goosens, Frank Mittelbach, and Alexander Samarin. *The LATEX Companion*. Addison Wesley, 13 edition, 1994.

[11] George Grätzer. *Math into LATEX*. Birkhäuser Boston, third edition, 2000.

[12] Donald E. Knuth. *The TEXbook*. Addison Wesley Professional, 21 edition, 1986.

[13] Donald E. Knuth, Tracy Larrabee, and Paul M. Roberts. *Mathematical Writing*. Stanford University, Computer Science Department, http://sunburn.stanford.edu/~knuth/papers/mathwriting.tex.gz, 1987.

[14] R. Kuhn, R. Scott, and L. Andreev. *An Introduction to using LaTeX in the Harvard Mathematics Department*. Harvard University, Department of Mathematics, http://abel.math.harvard.edu/computing/latex/manual/texman.html.

[15] Johannes Küster. *Designing Math Fonts*. http://www.typoma.com/publ/20040430-bachotex.pdf, apr 2004. Vortrag auf der polnischen TeX-Konferenz »BachoTeX«.

[16] Johannes Küster. *Fonts for Mathematics*. http://www.typoma.com/publ/20041002-atypi.pdf, oct 2004. Vortrag auf der ATypI-Konferenz in Prag.

[17] Richard Lawrence. Math=Typography? *TUGboat Journal*, 24(2):165–168, 2003.

[18] NIST. *Typefaces for Symbols in Scientific Manuscripts*. http://physics.nist.gov/Document/typefaces.pdf, 2004.

[19] Luca Padovani. Mathml formatting with tex rules and tex fonts. *TUGboat Journal*, 24(1):53–61, 2003.

[20] Sebastian Rahtz and Leonor Barroca. A style option for rotated objects in LaTeX. *TUGboat Journal*, 13(2):156–180, July 1992.

[21] Steve Seiden. *Math cheat sheet*. TUG, http://www.tug.org/texshowcase/#math, 2000.

[22] Carole Siegfried and Herbert Voß. Mathematik im Inline-modus. *Die TeXnische Komödie*, 3/04:25–32, November 2004.

[23] Paul Taylor. *Commutative Diagrams in TeX*. Department of Computer Science, Queen Mary and Westfield College, http://www.dcs.qmw.ac.uk/~pt/diagrams/, 2000.

[24] Herbert Voß. Farbige Mathematik. *Die TeXnische Komödie*, 2/04:81–87, March 2004.

Index