

Matematică avansată

Această pagină prezintă trăsături mai avansate ale matematicii din LaTeX. În particular, folosește în mare măsură pachetul `amsmath` oferit de [Societatea Americană de Matematică \(American Mathematical Society\)](#).

Numerotarea ecuațiilor

Mediul `equation` numerotează automat ecuațiile.

```
\begin{equation}
f(x)=(x+a)(x+b)
\end{equation}
```

$$f(x) = (x + a)(x + b) \quad (1)$$

Puteți utiliza de asemenea comenzile `\label` și `\ref` pentru a eticheta, respectiv a face referințe la ecuații. Sau puteți folosi comanda `\eqref`, însă aceasta necesită pachetul `amsmath`. Iată un exemplu ce vă arată cum puteți stabili referințe cu ajutorul comenzii `\ref` (are ca rezultat **1** pentru ecuația 1), iar apoi cum să folosiți `\eqref` din pachetul `amsmath` (are ca rezultat **(1)** pentru ecuația 1):

```
\begin{equation}
\label{ec:o-ec}
5^2 - 5 = 20
\end{equation}
```

$$5^2 - 5 = 20 \quad (1)$$

creează o referință la ecuația `\ref{ec:o-ec}`.

```
\begin{equation}
\label{ec:erl}
a = bq + r
\end{equation}
```

unde `\eqref{ec:erl}` este adevărată dacă a și b sunt întregi, iar

$$a = bq + r \quad (1)$$

unde **(1)** este adevărată dacă a și b sunt întregi, iar $b \neq c$.

$\$b \neq c\$$.

Notă: Pentru afișarea caracterelor românești trebuie să includeți un pachet pentru acestea, cum ar fi `inputenc` cu opțiunea pentru UTF:

```
\usepackage[utf8x]{inputenc}
```

Pentru mai multe informații, vezi capitolul [Etichete și referințe](#).

Pentru ca enumerarea să urmeze tiparul secțiunii sau subsecțiunii curente, trebuie să folosiți pachetul `amsmath` sau documente din clasa AMS. Apoi scrieți

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

în preambul pentru ca enumerarea să corespundă nivelului secțiunii sau

```
\numberwithin{equation}{subsection}
```

pentru ca enumerarea să se adapteze nivelului subsecțiunii.

```
\documentclass[12pt]{article}
\usepackage{amsmath}
\numberwithin{equation}{subsection}
\begin{document}
\section{Prima Secțiune}

\subsection{O subsecțiune}
\begin{equation}
L' = L \sqrt{1 - \frac{v^2}{c^2}}
\end{equation}
\end{document}
```

$$L' = L \sqrt{1 - \frac{v^2}{c^2}} \quad (1.1.1)$$

Dacă stilul folosit pune puncte după numere, comanda `\numberwithin{equation}{subsection}` în preambul va face ca numărul ecuației din exemplul de mai sus să fie afișat în felul următor: (1.1.1).

Pentru numerotarea la nivelul secțiunii, puteți folosi și comanda

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

Numerotarea la nivelul subsecțiunii cu comanda

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

este echivalentă cu versiunea din exemplul anterior, cu condiția ca această comandă să fie în preambul sau înaintea oricăror ecuații.

Notă: Puteți folosi aceste ultime comenzi pentru a schimba numerotarea ecuațiilor în interiorul documentului.

Numerotarea ecuațiilor subordonate

Pentru a numerota ecuațiile subordonate într-un mediu de ecuații numerotate, plasați partea documentului ce le conține într-un mediu `subequations`:

```
\section{Prima Secțiune}
\renewcommand{\theequation}{\thesection.\arabic{equation}}
\begin{subequations}
Ecuatiile lui Maxwell:
\begin{align}
B'&=-\partial \times E, \\
E'&=\partial \times B - 4\pi j,
\end{align}
\end{subequations}
```

Ecuatiile lui Max

$$B' = -\partial \times E,$$

$$E' = \partial \times B - 4\pi j$$

Nu uitați că pentru afișarea caracterelor românești trebuie să includeți un pachet pentru acestea, cum ar fi `inputenc` cu opțiunea pentru UTF:

```
\usepackage[utf8x]{inputenc}
```

Formule matematice în modul de afișare aliniate vertical

O problemă des întâlnită cu mediile de afișare (`displaymath` și `equation`) este lipsa abilității de a se întinde pe mai multe linii. Deși puteți defini liniile individual, acestea nu vor fi aliniate.

Deasupra și dedesubt

Comenzile `\overset` și `\underset` scriu simboluri deasupra și dedesubtul expresiilor. Fără AMS în TeX, același rezultat al `\overset` poate fi obținut cu `\stackrel`. Acest lucru poate fi deosebit de util pentru a crea noi relații binare:

$$\begin{array}{l} \backslash \\ A \overset{!}{=} B; A \stackrel{!}{=} B \\ \backslash \end{array} \quad \left| \quad A \overset{!}{=} B; A \underset{!}{=} B$$

sau pentru a prezenta utilizarea [legii lui L'Hôpital](#):

$$\begin{array}{l} \backslash \\ \lim_{x \rightarrow 0} \left\{ \frac{e^x - 1}{2x} \right\} \\ \overset{\left[\frac{0}{0} \right]}{\underset{\mathrm{H}}{=}} \\ \lim_{x \rightarrow 0} \left\{ \frac{e^x}{2} \right\} = \frac{1}{2} \\ \backslash \end{array} \quad \left| \quad \lim_{x \rightarrow 0} \frac{e^x - 1}{2x} \overset{\left[\frac{0}{0} \right]}{\underset{\mathrm{H}}{=}} \lim_{x \rightarrow 0} \frac{e^x}{2}$$

Este convenabil să definim un nou operator ce va seta semnul de egalitate cu H și fracția:

```
\newcommand{\Heg}[1]{\overset{\left[#1\right]}{\underset{\mathrm{H}}{=}}{}}
```

pentru a reduce exemplul de mai sus la:

$$\begin{array}{l} \backslash \\ \lim_{x \rightarrow 0} \left\{ \frac{e^x - 1}{2x} \right\} \\ \Heg \left\{ \frac{0}{0} \right\} \\ \lim_{x \rightarrow 0} \left\{ \frac{e^x}{2} \right\} = \frac{1}{2} \\ \backslash \end{array}$$

Dacă vrei să scrieți comentarii în anumite părți ale unei ecuații, comenzile `\overbrace` și `\underbrace` pot fi mai utile, cu toate acestea, ele au o sintaxă diferită:

```

\l
z = \overbrace{
  \underbrace{x}_{\text{real}} +
  \underbrace{iy}_{\text{imaginar}}
}^{\text{număr complex}}
\l

```

$$z = \overbrace{\underbrace{x}_{\text{real}} + \underbrace{iy}_{\text{imaginar}}}^{\text{număr complex}}$$

Câteodată comentariile sunt mai lungi decât formula comentată, ceea ce poate genera probleme de spațiere. Acestea nu sunt înlăturate cu comanda `\mathclap`^[2]:

```

\l
y = a +
f(\underbrace{
  e^{bx}_{
    \ge 0 \text{
conform
ipotezei}}
}
= a +
f(\underbrace{
  e^{bx}_{
    \mathclap{\
ge 0 \text{
conform
ipotezei}}}}
\l

```

$$y = a + f(\underbrace{bx}_{\ge 0 \text{ conform ipotezei}}) = a + f(\underbrace{\quad}_{\ge 0 \text{ conform ipotezei}})$$

Ca o alternativă, pentru a folosi paranteze drepte în loc de acolade, utilizați comenzile `\underbracket` și `\overbracket`^[2]:

```

\l
z = \overbracket[3pt]{
  \underbracket{x}_{\text{real}} +
  \underbracket[0.5pt][7pt]{iy}_{\text{imaginar}}
}^{\mbox{număr complex}}
\l

```

$$z = \overbracket[3pt]{\underbracket{x}_{\text{real}} + \underbracket[0.5pt][7pt]{iy}_{\text{imaginar}}}$$

Argumentele opționale setează grosimea liniei, respectiv înălțimea parantezei drepte:

```
\underbracket[grosime linie][înălțime paranteză dreaptă]{argument}_ {text dedesubt}
```

Comenzile `\xleftarrow` și `\xrightarrow`^[1] produc săgeți care se extind pe toată lungimea textului. Din nou, sintaxa este diferită: argumentul opțional (scris cu ajutorul `[` și `]`) specifică partea de sub săgeată, iar argumentul obligatoriu (între `{` și `}`) specifică partea de deasupra săgeții.

```
\[
A \xleftarrow{\text{așa}} B
\xrightarrow[\text{sau așa}]{} C
\]
```

$$A \xleftarrow{\text{așa}} B \xrightarrow{\text{sau așa}} C$$

Pentru săgeți mai extensibile, trebuie să utilizați pachetul `mathtools`:

```
\begin{align*}
A \xleftrightharpoons[sub]{peste} B \\
B \xLeftarrow[sub]{peste} C \\
C \xRrightarrow[sub]{peste} D \\
D \xLeftrightarrow[sub]{peste} E \\
E \xhookleftarrow[sub]{peste} F \\
F \xhookrightarrow[sub]{peste} G \\
G \xmapsto[sub]{peste} H
\end{align*}
```

$$\begin{array}{c}
 A \xleftrightharpoons[sub]{peste} B \\
 B \xLeftarrow[sub]{peste} C \\
 C \xRrightarrow[sub]{peste} D \\
 D \xLeftrightarrow[sub]{peste} E \\
 E \xhookleftarrow[sub]{peste} F \\
 F \xhookrightarrow[sub]{peste} G \\
 G \xmapsto[sub]{peste} H
 \end{array}$$

Pentru cârlige:

```

\begin{align*}
H \xrightarrow[\text{sub}]{\text{peste}} I \\
I \xrightarrow[\text{sub}]{\text{peste}} J \\
J \xrightarrow[\text{sub}]{\text{peste}} K \\
K \xrightarrow[\text{sub}]{\text{peste}} L \\
L \xrightarrow[\text{sub}]{\text{peste}} M \\
M \xrightarrow[\text{sub}]{\text{peste}} N
\end{align*}

```

$$\begin{array}{ccc}
H & \xrightarrow[\text{sub}]{\text{peste}} & I \\
I & \xrightarrow[\text{sub}]{\text{peste}} & J \\
J & \xrightarrow[\text{sub}]{\text{peste}} & K \\
K & \xrightarrow[\text{sub}]{\text{peste}} & L \\
L & \xrightarrow[\text{sub}]{\text{peste}} & M \\
M & \xrightarrow[\text{sub}]{\text{peste}} & N
\end{array}$$

align și align*

Mediile `align` și `align*`^[1] sunt folosite pentru aranjarea ecuațiilor pe mai multe linii. La fel ca în cazul matricilor și tabelelor, `\` specifică o linie nouă, iar `&` se folosește pentru a indica punctul în care liniile ar trebui aliniate.

Mediul `align*` se folosește la fel ca `displaymath` sau `equation*`:

```

\begin{align*}
f(x) &= (x+a)(x+b) \\
&= x^2 + (a+b)x + ab
\end{align*}

```

$$\begin{aligned}
f(x) &= (x+a)(x+b) \\
&= x^2 + (a+b)x + ab
\end{aligned}$$

Pentru a forța numerotarea unei linii specifice, utilizați comanda `\tag{...}` înainte de capătul de linie.

Mediul `align` este similar, însă numerotează automat fiecare linie, la fel ca mediul `equation`. Se poate face referire la liniile individuale plasând o comandă `\label{...}` înainte de capătul de linie. Puteți folosi una din comenzile `\nonumber` sau `\notag` pentru a suprima numărul unei linii date:

```

\begin{align}
f(x) &= x^4 +
7x^3 + 2x^2
\nonumber \\
&\quad \{ +
10x + 12
\end{align}

```

$$f(x) = x^4 + 7x^3 + 2x^2 + 10x + 12 \quad (1)$$

Notăți că am adăugat o anumită indentare la a doua linie. De asemenea, trebuie să inserăm acolade `{}` înainte de semnul `+` întrucât altfel LaTeX nu va crea spațierea corectă după `+`. Motivul pentru asta este că fără acolade, LaTeX interpretează semnul `+` ca pe un operator unar, în locul operatorului binar, după cum este de așteptat.

Acolade care se întind pe mai multe linii

Mai multe caractere `&` pe o singură linie vor specifica mai multe coloane, fiecare dintre ele fiind aliniată corespunzător. Dacă vreți ca o acoladă să continue pe linia următoare, faceți următoarele:

```

\begin{align}
f(x) &= \pi
\left\{ x^4 +
7x^3 + 2x^2
\right.\nonumber \\
&\quad \left.
\{ + 10x + 12
\right\}
\end{align}

```

$$f(x) = \pi \left\{ x^4 + 7x^3 + 2x^2 + 10x + 12 \right\} \quad (1)$$

Dimensiunile celor două acolade devin egale și corespund mărimii tipice a simbolurilor dintre ele prin comenzile `\left\{` și `\right\}`. Însă deoarece aceste două comenzi apar pe linii diferite, trebuie să le echilibrăm cu comenzile `\right.` și `\left.`. (În mod normal acestea nu sunt necesare dacă formula este pe o singură linie.) Ca o alternativă, puteți controla manual dimensiunea acoladelor cu ajutorul comenzilor `\big`, `\Big`, `\bigg`, `\Bigg`.

Pentru a potrivi automat mărimea celor două acolade pentru o ecuație înaltă, utilizați comanda `\vphantom`:

```

\begin{align}
A &= \left(\int_t
XXX
\right.\nonumber\!
&\quad
\left.\vphantom{\int_t}
\right) \dots \right)
\end{align}

```

$$A = \left(\int_t XXX \right. \quad \left. \begin{matrix} \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \end{matrix} \right) \dots \right) \quad (1)$$

Mediul `cases`

Mediul `cases`^[1] permite scrierea de funcții pe ramuri:

```

\[
u(x) =
\begin{cases}
\exp{x} & \text{dacă} \\
x \geq 0 \\
1 & \text{dacă } x \\
< 0
\end{cases}
\]

```

$$u(x) = \begin{cases} \exp x & \text{dacă } x \geq 0 \\ 1 & \text{dacă } x < 0 \end{cases}$$

La fel ca mai înainte, nu trebuie să aveți grijă de definiția sau alinierea coloanelor, LaTeX va face asta automat.

Din păcate, setează mediul matematic intern la stilul text, ducând la un astfel de rezultat:

$$a = \left\{ \int x dx \right. \\ \left. b^2 \right.$$

Pentru a forța stilul de afișare pentru ecuații în cadrul acestei construcții, utilizați mediul `dcases`^[2]:

```

\[
a =
\begin{dcases}
\int x \, dx \\
b^2
\end{dcases}

```

$$a = \begin{cases} \int x dx \\ b^2 \end{cases}$$

```

    b^2
\end{dcases}
\]

```

Deseori a doua coloană constă mai ales din text normal. Pentru a o seta la fontul *roman* normal al documentului, folosiți mediul `dcases*`^[2].

```

\[
f(x) =
\begin{dcases*}
x & \text{pentru } \$x\$
\par\
-x & \text{pentru } \$x\$
\impar
\end{dcases*}
\]

```

$$f(x) = \begin{cases} x & \text{pentru } x \text{ par} \\ -x & \text{pentru } x \text{ impar} \end{cases}$$

Alte medii

Deși mediile `align` și `align*` sunt cele mai utile, sunt multe alte medii ce pot fi și ele de interes:

Mediu	Descriere	Note
<code>eqnarray</code> și <code>eqnarray*</code>	Similar cu <code>align</code> și <code>align*</code>	Nerecomandat deoarece spațierea este inconsistentă
<code>multline</code> și <code>multline*</code> ^[1]	Prima linie este aliniată la stânga, ultima linie este aliniată la dreapta	Numărul ecuației este aliniat vertical cu prima linie, nefiind centrat ca în cazul altor medii.
<code>gather</code> și <code>gather*</code> ^[1]	Ecuatii consecutive fără aliniere	
<code>flalign</code> și <code>flalign*</code> ^[1]	Similar cu <code>align</code> , însă aliniază la stânga prima coloană a ecuației și la dreapta ultima coloană	
<code>alignat</code> și <code>alignat*</code> ^[1]	Ia un argument ce specifică numărul de coloane. Permite controlul explicit al spațiului orizontal dintre ecuații	Puteți calcula numărul de coloane numărând caracterele & de pe o linie, la care adăugați 1, iar rezultatul îl împărțiți la 2

Sunt și câteva medii ce nu reprezintă medii matematice prin ele însele, dar care pot fi utilizate pentru a construi structuri mai elaborate:

Mediu matematic	Descriere	Note
<code>gathered</code> ^[1]	Permite strângerea laolaltă a mai multor ecuații, care urmează să fie așezate una sub alta și care vor primi un singur număr de ecuație	
<code>split</code> ^[1]	Similar cu <code>align*</code> , însă folosit într-un alt mediu matematic de afișare	
<code>aligned</code> ^[1]	Similar cu <code>align</code> , însă folosit într-un alt mediu matematic de afișare	
<code>alignedat</code> ^[1]	Similar cu <code>alignat</code> , și la fel ca el, ia un argument adițional ce specifică numărul de coloane de setat pentru ecuații	

Spre exemplu:

```

\begin{equation}
\left.\begin{aligned}
& B'&=- \\
& \partial \times E, \\
& E'&= \partial \times B - \\
& 4\pi j,
\end{aligned}
\right\}
\end{equation}

```

$$\left. \begin{aligned} B' &= -\partial \times E, \\ E' &= \partial \times B - 4\pi j, \end{aligned} \right\} \text{ Ecuațiile lui Maxwell}$$

Sfârșitul de pagină în medii matematice

Pentru a sugera sistemului LaTeX să treacă la pagina următoare într-unul din mediile `amsmath`, puteți folosi comanda `\displaybreak` înaintea unui capăt de linie. La fel ca în cazul `\pagebreak`, `\displaybreak` poate lua un argument opțional între 0 și 4, care denotă nivelul de dezirabilitate de a trece la pagina următoare în acea situație dată. Valoarea 0 înseamnă "se permite trecerea la pagina următoare aici", 4 forțează sfârșitul de pagină. Nici un argument înseamnă valoarea 4.

Ca o alternativă, puteți activa sfârșitul de linie automat în mediile matematice cu comanda `\allowdisplaybreaks`. Și aceasta are un argument opțional, care denotă în ce măsură se permite trecerea la pagina următoare în ecuații. În mod similar, 1 înseamnă "se permite sfârșitul de pagină, însă în general se evită", iar 4 înseamnă "treci la pagina următoare când dorești". Puteți împiedica sfârșitul paginii după o linie dată cu ajutorul comenzii `*`.

LaTeX va insera un sfârșit de pagină într-o ecuație lungă (cu mediul `align`, de exemplu) dacă adaugă ceva text în plus cu comanda `\intertext{text}`, fără alte comenzi.

Un caz specific de utilizare poate arăta în felul următor:

```
\begin{align*}
&\vdots\!
&=12+7 \int_0^2
\left(
-
\frac{1}{4}\left(e^
{-4t_1}+e^{4t_1-
8}\right)

\right)\,dt_1\displa
ybreak[3]\!
&= 12-
\frac{7}{4}\int_0^2
e^{-4t_1}+e^{4t_1-
8}\,dt_1\!
&\vdots\!
&\end{align*}
```

$$\begin{aligned} & \vdots \\ & = 12 + 7 \int_0^2 \left(-\frac{1}{4} (e^{-4t_1} + e^{4t_1-8}) \right) dt_1 \\ & \hline & = 12 - \frac{7}{4} \int_0^2 e^{-4t_1} + e^{4t_1-8} dt_1 \\ & \vdots \end{aligned}$$

Ecuatii cuprinse într-un cadru

Pentru o singură ecuație, cu eticheta în afara căsuței, utilizați comanda `\boxed{}`:

```
\begin{equation}
\boxed{x^2+y^2 =
z^2}
\end{equation}
```

$$\boxed{x^2 + y^2 = z^2} \quad (1)$$

Dacă vrei ca întreaga linie sau mai multe ecuații să fie încadrate, folosiți un mediu `minipage` într-o căsuță `\fbox{}`:

```
\fbox {
\addtolength{\line
width}{-
2\fboxsep}%
\addtolength{\line
width}{-
2\fboxrule}%
\begin{minipage}{\
linewidth}
\begin{equation}
x^2+y^2=z^2
\end{equation}
\end{minipage}
}
```

$$\boxed{x^2 + y^2 = z^2}$$

Operatori personalizați

Deși LaTeX pune la dispoziție mulți [operatori](#) de uz comun, câteodată veți avea nevoie să creați unul nou, spre exemplu pentru a scrie operatorul [argmax](#). Comenzile `\operatorname` și `\operatorname*`^[1] afișează astfel de operatori, versiunea cu stea * setează opțiunea cu underscore dedesubt, ca pentru operatorul `\lim`:

```
\[
\operatorname{arg}\max
]_a f(a)
=
```

**Nu s-a putut interpreta
(Conversie în PNG eşuată;
verificați corectitudinea instalării
sistemelor LaTeX sau dvipng (sau**

```
\operatorname*{arg\,max}
}_b f(b)
\]
```

```
dvips + gs + convert)):
\operatorname{arg\,max}_a f(a)
=
\underset{b}\operatorname{arg\,
max} f(b)
```

Totuși, dacă operatorul este des folosit, este de preferat ca noul operator să fie încorporat în sistemul LaTeX, pentru documentul curent. Comenzile `\DeclareMathOperator` și `\DeclareMathOperator*`^[1] sunt specificate în preambulul documentului:

```
\DeclareMathOperator*{\argmax} {arg\,max}
```

Asta definește o nouă comandă, ce poate fi folosită în corpul documentului:

```
\[
\argmax_c f(c)
\]
```

$$\operatorname*{argmax}_c f(c)$$

Formatare avansată

Limite

Există anumite setări implicite pentru plasarea indicilor și exponenților. Spre exemplu, limitele pentru operatorul `\lim` se așează de obicei sub simbol, în felul următor:

```
\begin{equation}
\lim_{a\to\infty} \tfrac{1}{a}
\end{equation}
```

$$\lim_{a\rightarrow\infty} \frac{1}{a}$$

Pentru a modifica acest comportament implicit, utilizați operatorul `\nolimits`:

```
\begin{equation}
\lim\nolimits_{a\to\infty} \tfrac{1}{a}
\end{equation}
```

$$\lim_{a\rightarrow\infty} \frac{1}{a}$$

Un operator `\lim` în text (în cadrul $\$...\$$) va plasa limitele pe lateral. Dacă vreți ca argumentul comenzii `_` să fie plasat sub limită, folosiți comanda `\limits`, plasată imediat după operator:

$$\$ \lim \limits_{a \to \infty} \frac{1}{a} \$ \quad \left| \quad \lim_{a \rightarrow \infty} \frac{1}{a}$$

În mod asemănător, puteți așeza indicii care de obicei sunt plasați lateral:

$$\begin{equation} \int_a^b x^2 \end{equation}$$

deasupra și dedesubtul unui simbol:

$$\begin{equation} \int_a^b x^2 \end{equation}$$

Pentru a seta așezarea implicită laterală pentru toate cazurile unui simbol de însumare (în modul de afișare), adăugați opțiunea `nosumlimits` pachetului `amsmath`. Pentru a schimba așezarea implicită a limitelor intervalului pentru integrale adăugați parametrul `intlimits` la opțiuni și `nonamelimits` pentru a modifica plasarea implicită pentru operatorii *named*, de genul `det`, `min`, `lim`, etc.

Indici și exponenți

Deși puteți așeza caractere pe poziție de indici sau de exponenți în cazul simbolurilor de genul sumelor cu comanda prezentată mai devreme `\nolimits`:

$$\begin{equation} \sum \nolimits C_n \end{equation}$$

este imposibil să le îmbinați cu utilizarea tipică a acestor simboluri:

$$\begin{equation} \sum_{n=1} \nolimits C_n \end{equation}$$

Pentru a adăuga atât apostroful, cât și limita la un simbol, trebuie să utilizați comanda `\sideset`:

<pre>\begin{equation} \sideset{}{}\sum_{n=1}C_n \end{equation}</pre>	$\sum_{n=1}^f c_n$
--	--------------------

Este foarte flexibilă, spre exemplu, poate pune litere în fiecare colț al simbolului sumei:

<pre>\begin{equation} \sideset{_a^b}{_c^d}\sum \end{equation}</pre>	${}_a\sum_c^b{}_d$
---	--------------------

Dacă vreți să faceți asta pentru un simbol arbitrar, puteți folosi comanda `\fourIdx` din pachetul [fouridx](#):

<pre>\begin{equation} \fourIdx{a}{b}{c}{d}{\left(\dfrac{x}{y}\right)} \end{equation}</pre>	${}_b\left(\frac{x}{y}\right)_d^a{}_c$
--	--

Indici pe mai multe linii

Pentru a produce indici care se pot întinde pe mai multe linii, utilizați comanda `\substack`:

<pre>\begin{equation} \prod_{\substack{1\leq i\leq n\\1\leq j\leq m}} M_{i,j} \end{equation}</pre>	$\prod_{\substack{1\leq i\leq n\\1\leq j\leq m}} M_{i,j}$
--	---

Text în modul matematic de afișare aliniat

Pentru a adăuga explicații scurte în mediile matematice, folosiți comanda `\intertext`:

```

\begin{minipage}{3i
n}
\begin{align*}
\intertext{Dacă}
A &=
\sigma_1 + \sigma_2 \\
B &=
\rho_1 + \rho_2 \\
\intertext{atunci}
C(x) &=
e^{Ax^2 + \pi} + B
\end{align*}
\end{minipage}

```

Dacă

$$A = \sigma_1 + \sigma_2$$

$$B = \rho_1 + \rho_2$$

atunci

$$C(x) = e^{Ax^2 + \pi} + .$$

Notați că utilizarea acestei comenzi nu schimbă alinierea, așa cum ar face mai multe medii `align` intercalate.

Modificarea dimensiunii fontului

Se pot ivi unele situații în care să preferați un control mai mare asupra dimensiunii fontului pentru formule matematice. Spre exemplu, mediile matematice în modul `text` —`$...$` sau `\(...\)` — afișează implicit o fracție simplă (`\frac`) în felul următor: $\frac{a}{b}$. Într-o asemenea situație, ar fi de preferat ca fracția să fie mai mare, ca în modul de afișare —`$$...$$` sau `\[...\]`, dar să-și păstreze poziția pe linia curentă: $\bar{\frac{a}{b}}$.

O abordare simplă constă în utilizarea unor dimensiuni predefinite pentru elementele matematice:

Comandă	Descriere
<code>\displaystyle</code>	Dimensiunea ecuațiilor în modul de afișare
<code>\textstyle</code>	Dimensiunea ecuațiilor în modul <code>text</code>
<code>\scriptstyle</code>	Mărimea primului indice sau exponent
<code>\scriptscriptstyle</code>	Mărimea indicelui sau exponentului următor

Un exemplu clasic în acest sens îl constituie fracțiile continue (deși e mai bine să folosiți comanda `\cfrac`^[1] descrisă în capitolul [Matematică](#)):

```

\begin{equation}
x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + a_4}}}
\end{equation}

```

$$x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + a_4}}}$$

După cum vedeți, pe măsură ce fracțiile continuă, devin mai mici (aici au atins limita `\scriptstyle`). Dacă vreți ca dimensiunea să rămână consistentă, puteți declara stilul de afișare pentru fiecare fracție în parte:

```

\begin{equation}
x = a_0 +
\frac{1}{\displaystyle
a_1
+
\frac{1}{\displaystyle
a_2
+
\frac{1}{\displaystyle
a_3 + a_4}}}
\end{equation}

```

$$x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + a_4}}}$$

O altă abordare constă în utilizarea comenzii `\DeclareMathSizes` pentru a selecta dimensiunile preferate. Puteți redefini doar dimensiunile `\displaystyle`, `\textstyle`, `\scriptstyle`. Un posibil dezavantaj este că această comandă setează valorile globale, întrucât poate fi folosită numai în preambulul documentului.

Totuși, este foarte ușor de folosit: `\DeclareMathSizes{df}{da}{dt}{ds}`, unde *df* înseamnă dimensiunea implicită a fontului (pentru documentul curent), *da* este dimensiunea în modul de afișare (`\displaystyle`), *dt* dimensiunea în modul text (`\textstyle`), iar *ds* dimensiunea corespunzătoare modului `\scriptstyle`. Valorile furnizate se consideră a fi date în puncte (pt).

Notă: modificările valorilor globale au loc numai dacă valoarea primului argument corespunde dimensiunii textului din documentul curent. Se obișnuiește de aceea ca în preambul să fie mai multe declarații de genul acesta, în eventualitatea că se va modifica dimensiunea fontului implicit. Spre exemplu:

```

\DeclareMathSizes{10}{18}{12}{8} % Pentru text de mărime 10

```

```
\DeclareMathSizes{11}{19}{13}{9} % Pentru dimensiunea 11 a textului
\DeclareMathSizes{12}{20}{14}{10} % Pentru text de mărime 12pt
```

Note

1. ↑ [1.00](#) [1.01](#) [1.02](#) [1.03](#) [1.04](#) [1.05](#) [1.06](#) [1.07](#) [1.08](#) [1.09](#) [1.10](#) [1.11](#) [1.12](#) [1.13](#) [1.14](#) necesită pachetul `amsmath`
2. ↑ [2.0](#) [2.1](#) [2.2](#) [2.3](#) necesită pachetul `mathtools`

Teoreme

Prin "[teoremă](#)" înțelegem orice tip de enunț etichetat pe care vrem să-l vedem ca fiind separat de restul textului și care să aibă numere secvențiale. Această abordare se folosește în mod obișnuit pentru teoreme matematice, însă poate fi utilizată pentru orice. LaTeX oferă o comandă care facilitează definirea oricărui enunț în genul unei teoreme.

Teoreme de bază

Mai întâi, includeți pachetul `amsthm`:

```
\usepackage{amsthm}
```

Modul cel mai ușor de a defini o teoremă este să scrieți:

```
\newtheorem{nume}{Output generat}
```

în preambul. Primul argument este numele pe care îl veți folosi în document pentru a face referire la teoremă, iar al doilea argument este outputul generat de LaTeX la fiecare utilizare a teoremei. Spre exemplu:

```
\newtheorem{o-def}{Definiție}
```

va defini mediul `def`; dacă îl folosiți în felul următor:

```
\begin{o-def}
Iată o definiție nouă
\end{o-def}
```

va arăta așa:

Definiție 1. *Iată o definiție nouă*

separată prin linii libere de restul textului.

Numerotarea teoremelor

Adesea numerele teoremelor sunt determinate de secțiune, spre exemplu "Theorem 2.3" se referă la a treia teoremă din a doua secțiune a documentului. În cazul de față, specificați teorema după cum urmează:

```
\newtheorem{nume}{Output generat}[numără după]
```

unde *numără după* specifică [nivelul de împărțire în secțiuni](#) (secțiunea/subsecțiunea/etc.) la care numerotarea are loc.

Implicit, fiecare teoremă are propriul contor. Totuși, se obișnuiește ca tipuri similare de teoreme (de exemplu, teoreme, leme și corolari) să folosească în comun un contor. În acest caz, definiți teoremele următoare astfel:

```
\newtheorem{nume}[contor]{Output generat}
```

unde *contor* este numele contorului de utilizat. De obicei, acesta va fi numele teoremei principale.

Puteți de asemenea să creați un mediu pentru teoreme care nu este numerotat cu ajutorul comenzii `newtheorem*`^[1]. Spre exemplu,

```
\newtheorem*{o-def}{Definiție}
```

definește mediul `o-def`, care va genera definiții fără numerotare.

Demonstrații

Mediul `proof`^[1] poate fi folosit pentru a adăuga demonstrația unei teoreme. Utilizarea de bază este:

```
\begin{proof}  
  Demonstrația mea  
\end{proof}
```

Nu face decât să adauge *Proof* cu caractere italice la începutul textului dat ca argument și un pătrat gol pe dinăuntru, simbolul [\(Q.E.D.\)](#), cunoscut și sub

denumirea de [lespede \(tombstone\)](#) la sfârșit. Dacă scrieți în altă limbă decât engleza, folosiți pachetul [babel](#) cu argumentul corespunzător și cuvântul *Proof* afișat va fi tradus în această limbă; oricum, în codul sursă, numele mediului rămâne `proof`.

Dacă vreți să denumiți explicit demonstrația, includeți numele în paranteze drepte:

```
\begin{proof}[Demonstrația unei teoreme importante]
  Iată demonstrația mea importantă
\end{proof}
```

Dacă ultima linie a demonstrației este în modul matematic de afișare, atunci simbolul Q.E.D. va apare pe o linie liberă următoare. Pentru a pune simbolul Q.E.D. la sfârșitul ultimei linii, folosiți comanda `\qedhere`:

```
\begin{proof}
  Iată demonstrația mea:
  \[
  a^2 + b^2 = c^2 \qedhere
  \]
\end{proof}
```

Metoda de mai sus nu merge cu mediul *deprecated* (care nu mai este în uz, urmând a fi eliminat într-o versiune viitoare de LaTeX) [eqnarray*](#). Iată o posibilă soluție:

```
\begin{proof}
  Iată demonstrația mea:
  \begin{eqnarray*}
    a^2 + b^2 = c^2
  \end{eqnarray*}
  \vspace{-1.4cm}\[\qedhere\]
\end{proof}
```

Pentru a folosi un alt simbol Q.E.D., redefiniți comanda `\qedsymbol`. Pentru a ascunde apariția simbolului Q.E.D., redefiniți-l să nu fie nimic:

```
\renewcommand{\qedsymbol}{{}}
```

Stiluri de teoreme

Acestea adaugă posibilitatea de a schimba outputul mediilor definite de `\newtheorem` cu ajutorul comenzii `\theoremstyle`^[1] în preambul:

```
\theoremstyle{denumire_stil}
```

Argumentul este stilul pe care vrei să-l folosești. Toate teoremele definite mai târziu vor folosi acest stil. Iată o listă cu stilurile predefinite:

denumire_stil	Descriere
plain	Folosit pentru teoreme, leme, propoziții, etc. (stilul implicit)
definition	Utilizat pentru definiții și exemple
remark	Folosit pentru remarce și note

Stiluri personalizate

Pentru a defini propriul stil, folosești comanda `\newtheoremstyle`^[1]:

```
\newtheoremstyle{denumire stil}% numele stilului folosit
  {spațiu înainte}% spațiu liber deasupra teoremei, de
exemplu: 3pt
  {spațiu după}% spațiu liber sub teoremă, de exemplu: 3pt
  {font corp}% fontul din corpul teoremei
  {indentare}% implicit fără indentare
  {font antet}% fontul pentru antet
  {punctuație după antet}% punctuația dintre antet și corp
  {spațiu după antet}% spațiu după antetul teoremei; " " =
spațiu normal dintre cuvinte
  {antet}% antet specificat manual
```

(Orice argumente lăsate libere vor lua valorile implicite). Iată un exemplu de specificații pentru antet:

```
{\thmname{\textbf{#1}}\thmnumber{ \textbf{#2}}\thmnote{ (#3)}}
```

Exemplul complet:

```
\documentclass[a4paper,12pt]{article}

\usepackage[utf8x]{inputenc} % pentru caractere Unicode
\usepackage{amsthm}
\usepackage[pdfborder={0 0
0},colorlinks=true,linkcolor=blue]{hyperref}

\newtheoremstyle{plain}
  {3pt} % spațiu liber înaintea teoremei
  {3pt} % spațiu liber după teoremă
  {\itshape} % fontul din corpul teoremei
  {\parindent} % indentare
```

```

{\rmfamily} % fontul pentru antet
{:} % punctuația dintre antet și corp
{.5em} % spațiu după antetul teoremei
{\thmname{\textbf{#1}}\thmnumber{\textbf{#2}}\thmnote{
(#3)}} % antet
\newtheorem{defn}{Definiția}

\begin{document}

\begin{defn}[Topologie]
\href{http://ro.wikipedia.org/wiki/Topologie}{Topologia} este
o ramură a matematicii,
mai precis o extensie a geometriei care studiază deformările
spațiului prin
transformări continue.
\end{defn}

\end{document}

```

(Argumentul definiției, în cazul de față Topologie, este întotdeauna opțional, însă nu va apare implicit, dacă nu-l specificați ca mai sus în specificația antetului).

Output:

Definiția 1 (Topologie): *Topologia este o ramură a matematicii, mai precis o extensie a geometriei care studiază deformările spațiului prin transformări continue.*

Conflicte

Mediul pentru teoreme (`theorem`) intră în conflict cu alte medii, spre exemplu cu `wrapfigure`. O soluție posibilă constă în redefinirea teoremei, spre exemplu în modul următor:

```

% Ajustează spațiile verticale din LaTeX
\def\smallskip{\vskip\smallskipamount}
\def\medskip{\vskip\medskipamount}
\def\bigskip{\vskip\bigskipamount}

% Teoremă cu aspect personalizat
\newcounter{thm}[section]
\renewcommand{\thethm}{\thesection.\arabic{thm}}
\def\claim#1{\par\medskip\noindent\refstepcounter{thm}\hbox{\b
f

```

```

\arabic{chapter}.\arabic{section}.\arabic{thm}. #1.}
\it\ % \ignorespaces
}
\def\endclaim{\par\medskip}
\newenvironment{thm}{\claim}{\endclaim}

```

În acest caz, teorema arată în felul următor:

```

\documentclass[a4paper,12pt]{book}

```

```

\usepackage[utf8x]{inputenc} % pentru
caractere Unicode
...

```

```

\renewcommand{\chaptername}{Capitolul}

```

```

\begin{document}

```

```

\chapter{Matematică}

```

```

\section{Teoreme}

```

```

\begin{thm}{Teoremă
de
probabilitate}\label{
prob}

```

```

Lasă să se
întâmples.
Apoi vei afla.
\end{thm}

```

```

\end{document}

```

Capitolul 1

Matematică

1.1 Teoreme

1.1.1. Teoremă de probabilitate. *I*

Note

- ↑ [1.0](#) [1.1](#) [1.2](#) [1.3](#) necesită pachetul `amsthm`

Legături externe

- Documentația `amsthm`,
<ftp://ftp.ams.org/pub/tex/doc/amscls/amsthdoc.pdf>

Etichete și referințe

Introducere

Un alt avantaj al sistemului LaTeX este acela că puteți crea cu ușurință referințe la aproape orice este numerotat (secțiuni, figuri, formule), iar LaTeX va avea grijă de numerotarea referințelor, actualizând aceste date când este necesar. Comenzile utilizate nu depind de obiectele la care faceți trimiteri, aceste comenzi fiind: