Comment compose les mathématiques avec LaTeX

Édition du 28 octobre 2000

◇ *Ce symbole dans la marge gauche signale une possibilité qui n'est pas offerte directement par le format* `latex`*, mais se trouve définie dans le paquetage* `bruno`*.*

Accolades : $\{$ et $\}$ ne sont accessibles qu'en mode mathématique, et se codent respectivement `\{` et `\}`. C'est l'emploi des accolades pour le groupage qui impose cette contorsion. Pour agrandir les accolades, cf. **Parenthèses**.

Adhérence : cf. **Conjugué**.

Aleph : $\aleph$ se code `\aleph`.

Angles : $\hat p$ se code `\hat p`. Pour un «grand» angle : $\widehat{QKC}$ se code `\widehat{QKC}` ; on peut ainsi couvrir jusqu'à trois caractères.

Appartenance : $x \in B$ et $x \notin B$ se codent respectivement `x \in B` et `x \notin B`. $B \ni x$ se code `B \owns x`.

Barre : cf. **Conjugué**.

Chapeau : utiliser `\hat` pour mettre un «chapeau» sur une lettre ; on peut mettre un «grand chapeau» avec `\widehat` ; cf. **Angles**.

Chiffres romains : utiliser `\romannumeral`, par exemple `\romannumeral 1996` donne mcmxcvi. Pour obtenir des chifferes romains en majuscules, il faut ruser : `\uppercase\expandafter{\romannumeral 1999}` donera MCMXCIX.

Coefficient binomial : C_n^p se code `C_n^p`. Si l'on veut un C «droit», coder `{\rm C}_n^p`, ce qui donne C_n^p ;
◇ avec le paquetage `bruno`, on peut l'obtenir en codant `\Comb_n^p`. Avec le paquetage `amsmath`, $\binom{n}{p}$ se code `\binom{n}{p}`.

Complémentaire : utiliser `\complement` (requiert le package `amssymb`) ; par exemple, `\complement A` donne $\complement A$.

Congru à : pour obtenir $a \equiv b \pmod n$, coder `a equiv b \pmod n`.

Conjugué : $\bar z$ peut se coder `\bar z`, mais il vaut mieux utiliser `\overline`, et, pour avoir un bon alignement des barres au-dessus de lettres de hauteurs différentes, comme dans $\overline{\mathstrut a}+\overline{\mathstrut b}$, coder comme suit :

```
\overline{\mathstrut a}+\overline{\mathstrut b}
```

Contenu dans : cf. **Inclusion**.

Continuité : $\mathcal{C}^n$ se note `{\cal C}^n`. cf. **Cursives (majuscules)**.

Convergence : cf. **Limite**.

Convolution : $f * g$ se code `f*g`.

Coproduit : $\coprod$ se code `\coprod`. Les règles qui s'appliquent à l'opérateur $\prod$ (cf. **Produit**) sont également valables.

Crochets : $[a, b]$ se code bêtement `[a,b]`. Pour avoir de grand crochets : cf. **Parenthèses**. Crochets d'intervalle discret : $[\![p, q]\!]$ se code `\IntervalleDiscret{p}{q}` ; `\IntervalleDiscret` est défini dans le paquetage `bruno`. Sur le Mac, le crochet ouvrant est obtenu par la frappe combinée des touches $\boxed{\text{SHIFT}}$, $\boxed{\text{OPT}}$ et $\boxed{(}$; et le crochet fermant par la frappe des touche $\boxed{\text{SHIFT}}$, $\boxed{\text{OPT}}$ et $\boxed{)}$.

Cursives (majuscules) : utiliser `\cal` pour obtenir des majuscules cursives ; par exemple, $\mathcal{F}(\mathcal{A}, \mathcal{B})$ se code `{\cal F}({\cal A},{\cal B})`.

Curviligne (intégrale) : cf. **Intégrales**.

Degré : $27^\circ\mathrm{C}$ se code `$27^\circ{\mathrm C}$`.

Dérivées : f' se code `f'` tout simplement. Pour la dérivée seconde f'', utiliser `f''`, où l'on tape deux apostrophes (éviter le caractère «guillemet»). Pour les physiciens : $\dot x$ se code `\dot x`, et $\ddot x$ se code `\ddot x`. Dérivée n-ième : mettre en exposant, avec `f^{(n)}` pour obtenir $f^{(n)}$. Dérivée partielle : ∂ se code `\partial`. Enfin, si vous voulez écrire $\frac{\mathrm{d}f}{\mathrm{d}x}$, n'oubliez pas que le `d` est en caractères romains, et se code donc `{\rm d}`.

Déterminants : paquetage `amsmath`. Coder comme pour une matrice (cf. **Matrices**), mais avec l'environnement `vmatrix` au lieu de `pmatrix`. Exemple : $\begin{vmatrix} a & b \\ c & d \end{vmatrix}$ se code `\begin{vmatrix} a&b\\ c&d \end{vmatrix}`.

Différence : $A \setminus B$ se code `A \setminus B`. cf. **Moins**.

Différent : $\neq$ se code `\not=`.

Différentielle : n'oubliez pas que dans $\mathrm{d}f$, le d est en caractères romains ... Donc codez `{\rm d}f`.

Divergence : utiliser `\mathop{\rm div}` pour avoir un opérateur div. Il est raisonnable de définir `\divergence` comme suit (la deuxième syntaxe est préférable) :

```
\def\divergence{\mathop{\rm div}\nolimits}
\newcommand{\divergence}{\mathop{\rm div}\nolimits}
```

L'emploi de `\div` est déconseillé, car celui-ci est défini comme un opérateur *binaire*.

Division : l'opérateur $\div$ se code `\div`. On peut bien entendu utiliser la barre oblique /, comme dans 1/2. On dispose d'un opérateur binaire `\bmod` : ainsi, `a \bmod b` donne $a \bmod b$; et d'un opérateur unaire `\pmod`, qui permet d'obtenir $a \equiv b \pmod q$ en codant `$a \equiv b \pmod q$`.

Ensembles : cf. **Vide (ensemble)**, **Appartenance**, **Intersection**, **Réunion**, **Inclusion**.

Entière (partie) : cf. **Partie entière**.

Epsilon : `\epsilon` donne ϵ, il vaut donc mieux utiliser `\varepsilon` qui donne ε.

Équivalence : $\iff$ se code `\iff`, tandis que $\approx$, $\sim$ et $\equiv$ se codent respectivement `\approx`, `\sim` et `\equiv`.

Équivalent à : pour obtenir $f(x) \mathop{\widetilde{\scriptstyle x\to 0}} x$, je propose la construction `f(x) \equivalent_{x\to0} x` où la commande `\equivalent` est définie par

```
\def\equivalent_#1{\mathop{\lower4pt\hbox{$\widetilde{\scriptstyle{\#1}}$}}}
```

Etc : utiliser `\cdots` dans $a_1 + a_2 + \cdots + a_n$ et `\ldots` dans $a_1, a_2, \ldots, a_n$; pour composer des matrices et des déterminants, $\vdots$ et $\ddots$ sont obtenus respectivement avec `\vdots` et `\ddots`.

Étoile : en plus du caractère `*` du clavier, qui donne $\boxed{\text{*}}$ en mode texte et $\boxed{*}$ en mode mathématique, on dispose de $\star$, obtenu par `\star`. Lorsque l'on est en mode mathématique, `\ast` est synonyme de `*`.

Exposants : utiliser `^`, en pensant à grouper pour éviter les ambiguïtés. Par exemple, e^{x^2} se code `e^{x^2}`. Si l'exposant est «compliqué», il faut le grouper ; ainsi, x^{n+1} sera codé `x^{n+1}`. Si la «base» comporte un indice, penser à ajouter au besoin un `{}` avant l'exposant ; ainsi $x_n{}^2$ sera codé `x_n{}^2` ; si l'on omet cette précaution, on obtient x_n^2, qui n'est pas forcément le résultat escompté.

Flèche : $x \to +\infty$ se code `x \to +\infty`. $f : x \mapsto x^2 - 1$ se code `f:x \mapsto x^2-1`. $a \leftarrow y$ se code `x \leftarrow y`. cf. **Vecteurs**, **Gradient**, **Implique**, **Limite**.

Flux (intégrale de) : cf. **Intégrales**.

Fonctions : Ne pas oublier de mettre un `\` devant le nom d'une fonction usuelle, pour que celui-ci soit composé en romain et non en italique, et soit correctement séparé de ce qui le précède et de ce qui le suit. Si LaTeX proteste parce que vous utilisez la fonction `\toto` par exemple, c'est que celle-ci n'est définie nulle part. Mettez alors la définition suivante (adaptée à vos besoins) au début de votre texte :

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

`\mathop` signale à LaTeX qu'il s'agit d'un opérateur, pour qu'il répartisse des espaces adéquatement ; et `\nolimits` évite qu'un exposant ou un indice soit mal placé, dans un *display math*.

Sont définies dans LaTeX les fonctions suivantes :

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

Bien entendu, le `\nolimits` est inclus dans la définition de `\cos`, par exemple, mais pas dans celle de `\lim`.

◇ Sont définies, en outre, dans le paquetage `bruno` :

`\argch`	`\argsh`	`\argsh`	`\Card`	`\com`	`\cotan`	`\ch`	`\im`
`\rg`	`\sh`	`\th`	`\tr`	`\val`	`\Vect`		

ce qui complète les besoins (par exemple pour l'algèbre linéaire) et prend en compte certaines conventions différentes de celles en vigueur dans les pays anglo-saxons.

Fraction : $\frac{a-b}{c-d}$ se code `\frac{a-b}{c-d}`. Pour avoir des caractères de taille habituelle, comme dans $\displaystyle\frac{a-b}{c-d}$ faire `\displaystyle \frac{a-b}{c-d}`. On peut utiliser `/` dans le texte ; ainsi `1/p` donne $1/p$ qui est plus lisible que $\frac{1}{p}$.

Gamma : la fonction Γ se code `\Gamma`, bien entendu.

Gothiques (lettres) : cf. **Idéaux**.

Grand O : coder `{\cal O}(n)` pour obtenir $\mathcal{O}(n)$.

Gradient : l'opérateur $\overrightarrow{\rm grad}$ sera codé `\overrightarrow{\rm grad}`. En fait, comme c'est un *opérateur*, il est souhaitable de le mettre dans un groupe auquel on applique `\mathop`.

Grecs (caractères) : Utiliser `\alpha` pour α, etc. Les majuscules grecques qui n'existent pas dans l'alphabet latin s'obtiennent de même : `\Delta` donne Δ. Les minuscules grecques sont en caractères italiques, les majuscules en caractères romains. Si l'on veut obtenir en italiques les quelques majuscules grecques qui ne sont pas dans l'alphabet habituel, utiliser `\mit` ; ainsi, $\mit\Gamma$ s'obtient par `{\mit Gamma}`. LaTeX offre également les «variations» de certains caractères grecs :

ε	`\varepsilon`	ϑ	`\vartheta`	ϖ	`\varpi`
ϱ	`\varrho`	ς	`\varsigma`	φ	`\varphi`

Bien noter que toutes ces commandes ne sont utilisables que dans les modes mathématiques.

Heaviside (fonction de) : la lettre grecque Υ se code `\Upsilon`.

Idéaux : il est de tradition de désigner les idéaux d'un anneau au moyen de lettres qualifiées à tort de «gothiques» ; si vous disposez des fontes de l'AMS, chargez-les au moyen du paquetage `amsfonts`, puis utilisez `\mathfrak{abcdefg}` pour obtenir $\mathfrak{abcdefg}$. Voici une partie des caractères disponibles :

$\mathfrak{abcdefghijklmnopqrstuvwxyz}$
$\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$

Implique : $\Rightarrow$ se code `\Rightarrow`.

Inclusion : $A \subset B$, $A \subseteq B$, $A \not\subset B$ et $A \not\subseteq B$ se codent respectivement `A \subset B`, `A \subseteq B`, `A \not\subset B` et `A \not\subsetneq B`.

Indices : utiliser `_`, et penser à grouper (cf. **Exposants**). Ainsi, $P_k^{a_k}$ se code `P_k^{a_k}`. On peut faire un peu mieux : `P{}_k^{a_k}` donnera $P{}_k^{a_k}$; l'indice et l'exposant ont leurs bords gauches alignés sur une même verticale, ce qui peut être préférable. cf. **Tenseurs**.

◇ **Infini** : ∞ se code `\infty`. Avec les définitions du paquetage `bruno`, on peut se contenter, sur le Mac, de ∞, obtenu sur le clavier avec les touches $\boxed{\text{OPT}}$ et $\boxed{?\atop ;}$.

Intégrale : $\int_0^{x-1} \cos t \, dt$ se code `\int_0^{x-1} \cos t \, dt`. Noter : la borne supérieure est codée `{x-1}` ; la fonction cosinus est codée `\cos` pour être en caractères romains, et non italiques (et aussi, pour que LaTeX mette un peu d'espace avant et après) ; et `\,` qui place un petit espace entre le t et le dt.

Par défaut, les bornes sont placées à côté du $\int$; on peut les mettre au-dessus et en-dessous en utilisant `\limits` : par exemple, en style *texte*, on obtient $\int\limits_a^b$ en codant `\int\limits_a^b`.

Intégrale curviligne : $\oint_\Gamma f(t)\,dt$ se code `\oint_Gamma f(t)\,dt`.

Intégrale de flux : une intégrale de flux doit être construite à la main (hélas) ; je propose, pour obtenir $\oiint$, la construction suivante :

```
\setbox0=\hbox{$\displaystyle \int\!\!\!\!\!\int$}
\setbox1=\hbox to \wd0{\hfill$\bigcirc$\hfill}
\setbox0=\hbox to \wd0{\copy0\hss\copy1}
\mathop{\copy0}
```

qui donnera un résultat acceptable dans un *display*. On aura avantage à placer cette construction dans un `\def\ointint` au début du texte.

Intégrales doubles et triples : $\iint$ et $\iiint$ se codent respectivement `\iint` et `\iiint`. Pour aller au-delà des intégrales triples, on dispose de `\iiiint` et `\idotsint` qui donnent respectivement $\iiiint$ et *idotsint*.

Intersection : $A \cap B$ se code `A \cap B`. Pour une famille : $\bigcap_{n \geqslant 0} A_n$ se code `A \bigcap_{n \ge 0} A_n`.

Intervalles : cf. **Crochets**.

Kronecker (symbole de) : ce n'est jamais qu'un delta, donc δ_{ij} se code `\delta_{ij}`.

l : si ℓ vous semble meilleur que l, utilisez `\ell`.

Legendre (symbole de) : pour obtenir $\genfrac(){0.2pt}0ab$, coder `\genfrac(){0.2pt}0ab`. Il peut être économique (et mnémotechnique) de définir

```
\def\legendre#1#2{\genfrac(){0.2pt}0{#1}{#2}}
```

pour coder tout simplement `{a \legendre b}`.

Limite : $\lim\limits_{x \to +\infty}$ se code `\lim\limits_{x \to +\infty}`. On dipose également de $\limsup$ et $\liminf$, codés respectivement `\limsup` et (vous aviez deviné) `\liminf`. Pour obtenir $f(x) \mathop{\longrightarrow}\limits_{x\to+\infty} 1$, codez

```
f(x) \mathop{\longrightarrow}\limits_{x\to+\infty} 1
```

cf. **Tend vers**. On obtiendra $f_n \buildrel{\rm CVS}\over\longrightarrow g$ en codant :

```
f_n\buildrel{\rm CVS}\over\longrightarrow g
```

Logarithme: $\ln x$ se code `\ln x`. Si vous préférez $\ell\mathrm{n}\, x$, utilisez par exemple `\lnn x` après avoir défini `\lnn` par

```
\def\lnn{\mathop \ell{\rm n} \nolimits}
```

Le `\nolimits` permet d'obtenir un indice correctement placé, en cas de besoin.

Lois de composition: voici les symboles connus de LaTeX comme désignant des opérateurs binaires, donc des lois de composition, et la façon de les coder.

$+$	`+`	$-$	`-`	$\div$	`\div`
$\pm$	`\pm`	$\cap$	`\cap`	$\vee$	`\vee`
$\mp$	`\mp`	$\cup$	`\cup`	$\wedge$	`\wedge`
$\setminus$	`\setminus`	$\uplus$	`\uplus`	$\oplus$	`\oplus`
$\cdot$	`\cdot`	$\sqcap$	`\sqcap`	$\ominus$	`\ominus`
$\times$	`\times`	$\sqcup$	`\sqcup`	$\otimes$	`\otimes`
$*$	`*`	$\triangleleft$	`\triangleleft`	$\oslash$	`\oslash`
$\star$	`\star`	$\triangleright$	`\triangleright`	$\odot$	`\odot`
$\diamond$	`\diamond`	$\wr$	`\wr`	$\dagger$	`\dagger`
$\circ$	`\circ`	$\bigcirc$	`\bigcirc`	$\ddagger$	`\ddagger`
$\bullet$	`\bullet`	$\bigtriangledown$	`\bigtriangledown`	$\amalg$	`\amalg`
		$\bigtriangleup$	`\bigtriangleup`		

Pour définir une autre loi, par exemple avec le caractère ϕ, procédez en deux temps: au début de votre texte, mettez `\def\loiphi{\mathbin{\phi}}`; puis, pour coder $a \mathbin{\phi} b$, faites `a \loiphi b`. Comparez la répartition des espaces avec celle que donne $a\phi b$, codé bêtement `a \phi b`.

Vous pouvez également utiliser comme lois de composition les symboles de relation (cf. **Relations**); les espaces placés avant et après sont légèrement différents.

Matrices: paquetage `amsmath`. Utiliser l'environnement `pmatrix` pour une matrice délimitée automatiquement par des parenthèses; par exemple, `\begin{pmatrix} a&b\\ c&d \end{pmatrix}` donne $\begin{pmatrix} a&b\\ c&d \end{pmatrix}$. Les `&` séparent les coefficients sur une ligne, les `\\` séparent les lignes. Autres environnements matriciels: `bmatrix` (crochets au lieu des parenthèses), `vmatrix` (barres verticales, cf. **Déterminants**), `Vmatrix` (doubles barres verticales), `matrix` (pas de délimiteurs).

Module: cf. **Valeur absolue**.

Modulo: cf. **Équivalence**, **Division**.

Moins: le signe $-$ se code (bêtement) `-`. cf. **Différence**.

Nabla: ∇ se code `\nabla`.

Négation: pour écrire la relation contraire d'une relation donnée, la faire précéder de `\not`. Par exemple, $\not\subset$ sera codé `\not\subset`. Une exception : $\notin$ sera codé `\notin` dont l'apparence est meilleure. L'opérateur logique $\neg$ est codé `\neg`.

Norme: $\|\vec u\|$ se code `\|\vec u\|`. Pour agrandir, cf. **Parenthèses**.

$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}\ldots$: ces caractères sont dans la fonte `msbm`; pour y accéder, chargez le package `amsfonts` puis utilisez `{\mathbb N}` pour obtenir $\mathbb{N}$.

Orthogonal: utiliser `\perp` pour l'orthogonalité de deux droites. Pour désigner l'orthogonal $F^\bot$, faire `F^\bot`. Pour F°, remplacer `\bot` par `\circ`.

Parallèle: $D \parallel D'$ se code `D \parallel D'`.

Parenthèses : `\left(` et `\right)` permettent de placer autour d'une expression des parenthèses de taille adaptée. Ceci fonctionne aussi avec les crochets `[` et `]`, les accolades `{` et `}`, les barres verticales simples `|` et `|` et doubles `\|` et `\|` ... Il se peut que la taille calculée par LaTeX ne soit pas la plus judicieuse. Dans ce cas, utiliser `\bigl(` ... `\bigr)` pour obtenir $\bigl(\ldots\bigr)$; ceci vaut aussi avec `[` ... `]`, `{` ... `}`, `\|` ... `\|`. On dispose de trois autres tailles, par ordres croissants : `Big`, `bigg` et `Bigg`.

Chaque `\left` doit être associé à un `\right`. Si l'on ne veut obtenir que l'un des délimiteurs, remplacer l'autre par un point ; ainsi `\left.\genfrac{}{}{0pt}012\right]` donnera $\left.\genfrac{}{}{0pt}{0}{1}{2}\right]$. Par contre, les options `big` peuvent être placées sans se soucier de leur appariement.

Partie entière : $E(x)$ se code bêtement `E(x)`, tandis que $\mathrm{E}(x)$ sera codé `{\rm E}(x)` ; de même, $[x]$ se code `[x]`. Enfin, $\lfloor x\rfloor$ et $\lceil x\rceil$ se codent respectivement `\lfloor x\rfloor` et `\lceil x\rceil`. On peut appliquer à `\lfloor` ... `\rfloor` et à `\lceil` ... `\rceil` les mêmes modifications qu'au couple `(` ... `)` ; cf. **Parenthèses** à ce sujet.

Perpendiculaire : $D \perp D'$ se code `D \perp D'`.

PGCD : $a \wedge b$ se code `a \wedge b`.

Phi : `\phi` donne ϕ, je préfère `\varphi`, qui donne φ. La majuscule Φ est obtenue avec `\Phi`.

Plus : le signe $+$ se code (bêtement) `+` ; `\oplus` donne $\oplus$ (mode mathématique uniquement), et l'on a aussi `\bigoplus`, qui donne $\textstyle\bigoplus$ (style `text`) ou $\displaystyle\bigoplus$ (style `display`). cf. **Plus ou moins**, **Moins**, **Somme**.

Plus ou moins : le signe $\pm$ se code `\pm` ; pour le signe $\approx$ (qui signifie «à peu près égal à»), coder `\approx`.

Pour cent : comme le caractère `%` joue un rôle spécial dans LaTeX (il signale que ce qui suit est un commentaire), il est nécessaire pour l'obtenir de le faire précéder d'un `\` ; ainsi, `\%` donnera bien %.

PPCM : $p \vee q$ se note `p \vee q`.

Produit : $\prod_{k=1}^{n}$: cf. **Intégrale**, remplacer `\int` par `\prod`. Pour un produit de deux objets : $A \times B$ se code `A \times B`, et $A \cdot B$ se code `A \cdot B`. Si l'on veut un «gros» point : $a \bullet b$ se code `a\bullet b`, ou plus simplement `a•b` grâce à la définition du caractère • dans le paquetage `bruno`. L'opérateur `\*` a une interprétation particulière : en temps normal, il sera remplacé par `\times`, donnant $\times$, mais, si LaTeX a besoin de couper la formule à cet endroit précis, l'opérateur sera ignoré. Le produit scalaire $\langle f, g\rangle$ se code `\langle f,g\rangle` ; `\langle` et `\rangle` peuvent grandir comme les parenthèses, avec `\big` ; cf. **Parenthèses**.

Produit tensoriel : $a \otimes b$ se code `a \otimes b` ; et $\bigotimes\limits_{i\in I} E_i$ se code `\bigotimes\limits_{i\in I} E_i`.

Produit vectoriel : cf. **Vectoriel (produit)**.

Produit scalaire : pour obtenir $\langle f, g\rangle$, coder `\langle f,g\rangle`. On peut trouver avantage à utiliser la macro avec paramètres balisés :

```
\def\proscal<f,g>{\langle\#1,\#2\rangle}
```

Quantificateurs : $\forall$ et $\exists$ sont codés espectivement `\forall` et `\exists`.

Quotient (ensemble, groupe ...) : A/B se code `A/B`.

Racines et radicaux : $\sqrt{a+b}$ se code `\sqrt{a+b}`, tandis que $\sqrt[3]{2}$ se code `\root 3 \of 2`. Pour avoir des radicaux bien alignés dans $\sqrt{\mathstrut a} + \sqrt{\mathstrut b}$, utiliser `\mathstrut`, cf. **Conjugué**.

Relations : voici les symboles connus de LaTeX comme des relations, et la façon de les coder.

$<$	`<`	$>$	`>`	$=$	`=`
$\le$	`\le`	$\ge$	`\ge`	$\equiv$	`\equiv`
$\prec$	`\prec`	$\succ$	`\succ`	$\sim$	`\sim`
$\preceq$	`\preceq`	$\succeq$	`\succeq`	$\simeq$	`\simeq`
$\ll$	`\ll`	$\gg$	`\gg`	$\asymp$	`\asymp`
$\subset$	`\subset`	$\supset$	`\supset`	$\approx$	`\approx`
$\subseteq$	`\subseteq`	$\supseteq$	`\supseteq`	$\cong$	`\cong`
$\sqsubseteq$	`\sqsubseteq`	$\sqsupseteq$	`\sqsupseteq`	$\bowtie$	`\bowtie`
$\in$	`\in`	$\ni$	`\ni`	$\propto$	`\propto`
$\vdash$	`\vdash`	$\dashv$	`\dashv`	$\models$	`\models`
$\smile$	`\smile`	$\mid$	`\mid`	$\doteq$	`\doteq`
$\frown$	`\frown`	$\parallel$	`\parallel`	$\perp$	`\perp`

`\mid` est également une relation ; exemple : `\{x\in A\mid x \ne 0\}` donne $\{x \in A \mid x \neq 0\}$.

◇ Le paquetage `bruno` permet d'utiliser les caractères $\leqslant$ et $\geqslant$ du clavier du Mac, obtenus en combinant la touche $\boxed{\text{OPT}}$ avec $\boxed{<}$ ou $\boxed{+}$; les formes $\le$ et $\ge$ restent accessibles avec `\le` et `\ge`.

Pour définir une autre relation, par exemple avec le caractère α, procédez en deux temps : au début de votre texte, mettez `\def\avant{\mathrel{\alpha}}` ; puis, pour coder $a \mathrel{\alpha} b$, faites `a \avant b`. Comparez la répartition des espaces avec celle que donne $a\alpha b$, codé bêtement `a \alpha b`.

Restriction : $f|_E$ se code `f|_E`.

Réunion : cf. **Intersection**, remplacer `\cap` par `\cup` et `\bigcap` par `\bigcup` respectivement. Un moyen mnémotechnique pour retenir le nom : cup = coupe ....

Romains (chiffres) : cf. **Chiffres roamins**.

Rond : cf. **Intérieur** ; pour la composition des applications : $f \circ g$ se code `f\circ g` ; si vous trouvez qu'il y a trop d'espace autour du $\circ$, faites `f \mathord\circ g` pour obtenir $f{\circ}g$.

Somme : cf. **Intégrale**, en remplaçant `\int` par `\sum`.

Somme directe : $A \oplus B$ se code `A \oplus B`.

Souligner : $\underline{abc}$ se code `\underline{abc}`. Noter qu'on ne peut souligner qu'en mode mathématique ; pour mettre en valeur un mot dans du texte usuel, on passera en italique (ou en caractères gras).

Surligner : cf. **Conjugué**.

Symétrique (groupe) : pour obtenir $\mathfrak{S}_n$, composer `\mathop{\mathfrak{S}}\nolimits_n` ; vous devrez avoir chargé le paquetage `amsfonts` ; cf. **Idéaux**.

Tend vers : cf. **Limite**. Voici une solution lorsque la flèche semble trop courte : $e^x \xrightarrow[x \to -\infty]{} 0$ sera obtenu par `e^x \tendvers_{x \to -\infty} 0`, à condition de définir `\tendvers` par :

```
\def\tendvers_#1{\mathrel{\mathop{\hbox{\rightarrowfill}}\limits_{\>#1\>}}}
```

Tenseurs : ${G_{i,j}}^k{}_p$ se code `G_{i,j}{}^k{}_p`. Noter l'emploi de groupes vides. Voir aussi **Produit tensoriel**.

◇ **Théorème** : déclarer un environnement `theorem`, dans le préambule :

```
\newtheorem{theorem}{Théorème}
```

La numérotation est automatique :

```
\theorem (Stone \& Weierstrass) toute sous-algèbre ...
\theorem (Pythagore) la somme des carrés ...
\theorem (PPDA) le théorème de Fermat dit que $a^2+b^2=c^2$ ...
```

vous donnera ceci :

Théorème 1 (Stone & Weierstrass) toute sous-algèbre ...
Théorème 2 (Pythagore) la somme des carrés ...
Théorème 3 (PPDA) le théorème de Fermat dit que $a^2 + b^2 = c^2$ ...
Consultez la documentation du paquetage `theorem` pour savoir comment définir de nouveaux environnements, avec numérotation automatique.

Tilda : $\tilde p$ se code `\tilde p`. Pour un grand tilda : $\widetilde{F_{pq}}$ se code `\widetilde{F_{pq}}` ; on peut ainsi couvrir jusqu'à trois caractères.

◇ **Transposée** : ${}^t A$ se code `{}^t A`. Sur cet exemple, le groupe `{}` est inutile, mais s'il y a quelque chose avant le ^{t}, on risque des ennuis en l'oubliant ... On peut améliorer la présentation avec `\transpose A`, qui donne ${}^t\!A$. `\transpose` est défini par `\def\transpose{{}^t\!}`.

Union : cf. **Réunion**.

Unités : n'oubliez pas que les noms d'unités doivent être composés en caractères romains (donc avec `\rm` à l'intérieur d'une formule mathématique), et qu'ils ne prennent pas la marque du pluriel ... Seuls les journalistes écrivent «le 400 ms haies».

Valeur absolue : $|x - y|$ se code `|x-y|`. Sur le Mac, les barres verticales sont obtenues par la frappe combinée des touches $\boxed{\text{SHIFT}}$, $\boxed{\text{OPT}}$ et $\boxed{\text{L}}$. Elles peuvent grandir, cf. **Parenthèses**.

Vecteurs : $\vec u$ et $\vec\imath$ se codent respectivement `\vec u` et `\vec \imath`. Notez l'emploi de `\imath` au lieu de `i`, pour ôter le point sur le `i`. On dispose aussi de `\jmath`. Utiliser `\overrightarrow` pour les vecteurs plus importants ; ainsi, $\overrightarrow{AB}$ se code `\overrightarrow{AB}`. Pour écrire un vecteur donné par ses composantes en ligne : $(U_1, U_2, \ldots, U_n)$ se code `(U_1,U_2,\ldots,U_n)`. Si les composantes sont en colonne, cf. **Matrices**.

Vectoriel (produit) : $\vec u \wedge \vec v$ se code `\vec u \wedge \vec v`.

Vide (ensemble) : $\emptyset$ se code `\emptyset`.

Virgule : pour séparer la partie entière et la partie décimale d'un nombre, il faudra placer la virgule dans un groupe : `123{,}456` donne 123,456, alors que `123,456` donne $123, 456$. La raison en est que la virgule est considérée comme un symbole de ponctuation, et, à ce titre, est suivie d'un peu d'espace.

Zeta (fonction) : la fonction ζ de Riemann se code `\zeta` tout simplement.