Le package `lymath` : des formules plus sémantiques

Code source disponible sur `https://github.com/bc-latex/ly-math`.

Version `0.5.0-beta` développée et testée sur Mac OS X.

Christophe BAL

2019-09-29

Table des matières

1 Introduction

LATEX est un excellent langage, pour ne pas dire le meilleur, pour rédiger des documents contenant des formules mathématiques. Malheureusement toute la puissance de LATEX permet d'écrire des codes très peu sémantiques. Le modeste but du package `lymath` est de fournir quelques macros sémantiques pour la rédaction de formules mathématiques élémentaires. Considérons le code LATEX suivant.

```
Sachant que $\frac{df}{dx}(x) = 4 cos(x^2)$ sur $[a ; b]$ , nous avons :

$\int_a^b cos(x^2) dx = \left[ \frac{1}{4} f(x) \right]_a^b$.
```

Avec `lymath`, vous pouvez écrire le code suivant.

```
Sachant que $\derfrac{f}{x}(x) = 4 cos(x^2)$ sur $\intervalC{a}{b}$, nous avons :

$\int_a^b cos(x^2) \dd{x} = \hook{\frac{1}{4} f(x)}{a}{b}$.
```

Même si certaines commandes sont plus longues à écrire que ce que permet LATEX, il y a trois avantages à utiliser des commandes sémantiques.

1. La mise en forme dans votre document sera consistante.
2. Il est facile de changer une mise en forme sur l'ensemble d'un document.
3. `lymath` résout certains problèmes "complexes" pour vous.

2 Comment lire cette documentation ?

Le choix a été fait de fournir des exemples comme documentation du package suivis de fiches techniques des macros-commandes. Les exemples se présentent comme ci-dessous *(un code LATEX suivi de sa mise en forme)*.

```
Sachant que $\displaystyle \frac{df}{dx}(x) = 4 cos(x^2)$ sur $[a ; b]$ , nous avons :
$\displaystyle \int_a^b cos(x^2) dx = \left[ \frac{1}{4} f(x) \right]_a^b$.
```

Sachant que $\displaystyle \frac{df}{dx}(x) = 4 cos(x^2)$ sur $[a ; b]$, nous avons : $\displaystyle \int_a^b cos(x^2) dx = \left[\frac{1}{4} f(x) \right]_a^b$.

3 Versions étoilées

Généralement les versions étoilées proposent des mises en forme faisant un peu moins de travail que la version non étoilée *(il y a tout de même quelques exceptions)*. Par exemple une macro utilisant des parenthèses extensibles aura sa version étoilée qui n'utilisera que des parenthèses non extensibles.

4 A propos des arguments LATEX, une convention à connaître

`lymath` propose des macros avec un **nombre fixé d'arguments**. Dans ce cas, la syntaxe LATEX est gardée comme dans `\derfrac{f}{x}` .

Par contre, pour les macros avec un **nombre variable d'arguments**, la convention sera toujours d'utiliser un seul argument, au sens LATEX, dont le contenu sera formé de morceaux séparés par des traits verticaux | comme dans `\coord{3 | -4 | 0}` où l'unique argument `3 | -4 | 0` contient les trois morceaux `3` , `-4` et `0` .

5 Deux séparateurs d'arguments par défaut

La macro `\lymathsep` définit le séparateur d'arguments de premier niveau, et `\lymathsubsep` celui des arguments de deuxième niveau. Cette documentation utilisant l'option `french` de `babel`, la valeur de `\lymathsep` est $\boxed{;}$ et celle de `\lymathsubsep` est $\boxed{,}$. Sans ce choix, les valeurs de `\lymathsep` et `\lymathsubsep` seront $\boxed{,}$ et $\boxed{;}$ respectivement.

6 Quelques gestions d'espaces

6.1 Espace et fraction

Si vous utilisez `\frac` ou `\dfrac` alors de petits espaces sont automatiquement ajoutés pour éviter d'avoir des traits de fraction trop petits. Le comportement par défaut se retrouve en utilisant les macros `\stdfrac` et `\stddfrac` . Voici un exemple.

```
Vous avez $\frac{2}{3} = \stdfrac{2}{3}$ et $\dfrac{2}{3} = \stddfrac{2}{3}$ .
```

Vous avez $\frac{2}{3} = \frac{2}{3}$ et $\dfrac{2}{3} = \dfrac{2}{3}$.

6.2 Espace et racines n-ièmes d'un réel

`\sqrt` a été redéfini pour ajouter un peu d'espaces. Le comportement par défaut se retrouve en utilisant la macro `\stdsqrt` . Voici un exemple.

```
Vous avez $\sqrt{2} = \stdsqrt{2}$ et $\sqrt[n]{45} = \stdsqrt[n]{45}$ .
```

Vous avez $\sqrt{2} = \sqrt{2}$ et $\sqrt[n]{45} = \sqrt[n]{45}$.

6.3 Sommes et produits en mode ligne

Pour limiter l'espace, LaTeX affiche $\sum_{k=1}^{n}$ et non $\displaystyle\sum_{k=1}^{n}$ sauf si l'on utilise la commande `\displaystyle`. Les macros `\dsum` et `\dprod` permettent de se passer de `\displaystyle`. Voici un exemple.

```
$\dsum_{k=1}^{n} 2^k = \sum_{k=1}^{n} 2^k = 2^{n+1} - 1$

$\dprod_{k=1}^{n} k = \prod_{k=1}^{n} k = k !$
```

$\displaystyle\sum_{k=1}^{n} 2^k = \textstyle\sum_{k=1}^{n} 2^k = 2^{n+1} - 1$

$\displaystyle\prod_{k=1}^{n} k = \textstyle\prod_{k=1}^{n} k = k!$

6.4 Espace et point-virgule avec l'option `french` de `babel`

Seulement si vous utilisez `babel` avec l'option `french`, comme c'est le cas dans cette documentation, alors vous verrez le même espacement autour du point-virgule dans $A(x\,;y)$. Que c'est beau !

7 Logique et fondements

7.1 Différents types d'égalités « standard »

D'un point de vue pédagogique, il peut être intéressant de disposer de différentes façon d'écrire une égalité, une non égalité ou inégalité. Voici ce qui est proposé.

7.1.1 Définir quelque chose

L'exemple suivant montre trois façons de rédiger une égalité signifiant une définition [1] *(la section 7.2 explique comment est définit le texte* « déf » *).*

```
La fonction $f$ est définie sur $\RR$ par $f(x) \eqdef x^3 + 1$ ou avec les écritures
symboliques $f(x) \eqdef* x^3 + 1$ et $f(x) \eqdef** x^3 + 1$.
```

La fonction f est définie sur $\mathbb{R}$ par $f(x) \overset{\text{déf}}{=} x^3+1$ ou avec les écritures symboliques $f(x) \coloneqq x^3+1$ et $f(x) \hat{=} x^3+1$.

7.1.2 Indiquer une identité

L'exemple suivant montre deux façons de rédiger des identités, la noptation symbolique n'étant pas standard *(la section 7.2 explique comment est défini le texte* « id » *).*

```
$\forall (a ; b) \in \RR^2$, nous avons : $(a + b)^2 \eqid a^2 + b^2 + 2 a b$ .
On peut utiliser une écriture plus symbolique : $(a + b)^2 \eqid* a^2 + b^2 + 2 a b$ .
```

$\forall(a\,;b) \in \mathbb{R}^2$, nous avons : $(a+b)^2 \overset{\text{id}}{=} a^2+b^2+2ab$. On peut utiliser une écriture plus symbolique : $(a+b)^2 \risingdotseq a^2+b^2+2ab$.

7.1.3 Une égalité à vérifier ou non, une hypothèse, une condition

Se reporter à la section 7.2 pour savoir comment sont définis les textes *« cond »* et *« hyp »*.

1. Le symbole peu courant $\hat{=}$ est utilisé par le langage B qui permet de spécifier et prouver certains programmes.

```
Est-il vrai que $(a + b)^3 \eqtest a^3 + b^3 + 3 a b$ ?

A-t-on $x \neqhyp 0$ pour pouvoir écrire $\dfrac{1}{x}$ ?

Comme $x$ doit être non nul, je sait que $x \neqcond 0$.

Une autre condition $x \eqcond 0$ ou une autre hypothèse $x \eqhyp 0$ ?
```

Est-il vrai que $(a+b)^3 \overset{?}{=} a^3+b^3+3ab$?
A-t-on $x \overset{\text{hyp}}{\neq} 0$ pour pouvoir écrire $\dfrac{1}{x}$?
Comme x doit être non nul, je sait que $x \overset{\text{cond}}{\neq} 0$.
Une autre condition $x \overset{\text{cond}}{=} 0$ ou une autre hypothèse $x \overset{\text{hyp}}{=} 0$?

7.1.4 Différents types d'inéquations

Le principe reste le même pour les symboles d'inéquations excepté qu'il n'y a ici aucune écriture purement symbolique. Voici un code « fourre-tout » pour voir ce que vous avez à votre disposition.

```
A-t-on $x \leqtest x^2$ ou $x \ltest x^2$ ?

A moins que ce ne soit $x \geqtest x^2$ ou $x \gtest x^2$ qu'il faille vérifier.

On peut supposer $x \leqhyp 1$ ou avoir la condition $x \gcond 2$.
```

A-t-on $x \overset{?}{\leq} x^2$ ou $x \overset{?}{<} x^2$?
A moins que ce ne soit $x \overset{?}{\geq} x^2$ ou $x \overset{?}{>} x^2$ qu'il faille vérifier.
On peut supposer $x \overset{\text{hyp}}{\leq} 1$ ou avoir la condition $x \overset{\text{cond}}{>} 2$.

7.2 Textes utilisés

Voici les macros définissant les textes utilisés qui tiennet compte de l'utilisation ou non de l'option `french` de `babel`.

1. `\textopdef` donne *« déf »* en français et *« def »* sinon.
2. `\textopid` donne *« id »* dans tous les cas.
3. `\textopcond` donne *« cond »* dans tous les cas.
4. `\textophyp` donne *« hyp »* dans tous les cas.

7.3 Équivalences et implications

7.3.1 Des symboles supplémentaires

En plus des opérateurs `\iff` et `\implies` proposés par LaTeX, il a été ajouté l'opérateur `\liesimp`, le verlan de `\implies` pour obtenir $\impliedby$. Voici un exemple d'utilisation *(penser aussi aux preuves d'équivalence par double implication)*.

```
Un théorème de la logique : $(A \implies B) \iff (B \liesimp A)$
```

Un théorème de la logique : $(A \implies B) \iff (B \Longleftarrow A)$

Tout comme pour les égalités, il existe des versions de type test, hypothèse et condition.

```
Équivalences : $A \ifftest B \iffhyp C \iffcond D$

Implications directes : $A \impliestest B \implieshyp C \impliescond D$

Implications réciproques : $A \liesimptest B \liesimphyp C \liesimpcond D$
```

Équivalences : $A \overset{?}{\iff} B \overset{\text{hyp}}{\iff} C \overset{\text{cond}}{\iff} D$
Implications directes : $A \overset{?}{\implies} B \overset{\text{hyp}}{\implies} C \overset{\text{cond}}{\implies} D$
Implications réciproques : $A \overset{?}{\Longleftarrow} B \overset{\text{hyp}}{\Longleftarrow} C \overset{\text{cond}}{\Longleftarrow} D$

7.3.2 Équivalences et implications verticales

Dans la section 7.4 est expliqué comment détailler les étapes d'un raisonnement. Avec cet outil, il devient utile d'avoir des versions verticales des symboles d'équivalence et d'implication. Voici comment les obtenir.

```
\begin{tabular}{ccc}
    $A$      &   $B$          &   $C$         \\
    $\viff$  &   $\vimplies$  &   $\vliesimp$ \\
    $D$      &   $E$          &   $F$
\end{tabular}
```

A	B	C
$\Updownarrow$	$\Downarrow$	$\Uparrow$
D	E	F

7.4 Détailler un raisonnment

Un exemple sur une petite étape

La macro `\explain` prend deux arguments : le premier est un symbole et le second une courte explication. Commençons par une simple étape de raisonnement détaillée comme suit.

```
$0 \leq a < b$

$\explain{\vimplies}{Par croissance de la fonction carrée sur $\RRp$.}$

$a^2 \leq b^2$
```

$0 \leq a < b$
$\Downarrow$ *{ Par croissance de la fonction carrée sur $\mathbb{R}_+$. }*
$a^2 \leq b^2$

Si vous souhaitez un espace devant le symbole, il suffit d'utiliser la version étoilée comme suit.

```
$0 \leq a < b$

$\explain*{\vimplies}{Par croissance de la fonction carrée sur $\RRp$.}$

$a^2 \leq b^2$
```

$0 \leq a < b$

$\Downarrow \qquad$ *{ Par croissance de la fonction carrée sur* $\mathbb{R}_+$*. }*

$a^2 \leq b^2$

`\explain` et `\explain*` utilisent les macros constantes suivantes.

- `\textexplainleft` et `\textexplainright` qui donnent { et } respectivement par défaut.
- `\textexplainspacebefore` est l'espace entre le symbole et la courte explication. Par défaut, cette macro vaut `\qquad`.
- `\textexplainspacein` est l'espace supplémentaire ajouté par `\explain*` avant le symbole. Par défaut, cette macro vaut `\qquad` aussi.

Détailler des calculs

Pour finir, voici un exemple avec l'environnement `flalign` du package `amsmath` *(qui est automatiquement chargé par* `lymath`*)*. Il est alors relativement rapide de détailler un calcul.

```
\begin{flalign*}
        & (a + b)^2
        &&\\
        & \explain{=}{On utilise $x^2 = x \cdot x$.}
        &&\\
        & (a + b) (a + b)
        &&\\
        & \explain{=}{Double développement depuis la parenthèse gauche.}
        &&\\
        & a^2 + a b + b a + b^2
        &&\\
        & \explain{=}{Commutativité du produit.}
        &&\\
        & a^2 + 2 a b + b^2
        &&\\
\end{flalign*}
```

$(a+b)^2$

$= \qquad$ *{ On utilise* $x^2 = x \cdot x$*. }*

$(a+b)(a+b)$

$= \qquad$ *{ Double développement depuis la parenthèse gauche. }*

$a^2 + ab + ba + b^2$

$= \qquad$ *{ Commutativité du produit. }*

$a^2 + 2ab + b^2$

Fiches techniques

\explain (2 Arguments)
\explain* (2 Arguments)

— Argument 1 : un symnole.

— Argument 2 : une courte explication.

8 Ensembles

8.1 Différents types d'ensembles

8.1.1 Ensembles versus accolades

Exemple d'utilisation 1

```
Un ensemble de beaux nombres : $\geneset{1 ; 3 ; 5}$ .
```

Un ensemble de beaux nombres : $\{1\,;3\,;5\}$.

Exemple d'utilisation 2

```
Choisissez votre camp :
$\displaystyle \geneset{\frac{1}{3} ; \frac{5}{7} ; \frac{9}{11}}$
ou
$\displaystyle \geneset*{\frac{1}{3} ; \frac{5}{7} ; \frac{9}{11}}$ .
```

Choisissez votre camp : $\displaystyle \left\{\frac{1}{3}\,;\frac{5}{7}\,;\frac{9}{11}\right\}$ ou $\displaystyle \{\frac{1}{3}\,;\frac{5}{7}\,;\frac{9}{11}\}$.

Fiches techniques

\geneset (1 Argument)
\geneset* (1 Argument)

— Argument : la définition de l'ensemble.

8.1.2 Ensembles pour la géométrie

Exemple d'utilisation 1

```
Vous pouvez écrire sémantiquement $\geoset{C}$, $\geoset{D}$ et $\geoset{d}$ mais pas
taper \verb+$\geoset{ABC}$+.
```

Vous pouvez écrire sémantiquement $\mathscr{C}$, $\mathscr{D}$ et $\mathscr{d}$ mais pas taper `$\geoset{ABC}$`.

Exemple d'utilisation 2

```
Pour les indices, utilisez $\geoset*{C}{1}$, $\geoset*{C}{2}$ \dots
```

Pour les indices, utilisez $\mathscr{C}_1$, $\mathscr{C}_2$ ...

Fiches techniques

`\geoset (1 Argument)`

— `Argument`: un seul caractère `ASCII` indiquant un ensemble géométrique.

`\geoset* (2 Arguments)`

— `Argument 1`: un seul caractère `ASCII` indiquant $\mathscr{U}$ dans le nom $\mathscr{U}_d$ d'un ensemble géométrique.

— `Argument 2`: un texte donnant d dans le nom $\mathscr{U}_d$ d'un ensemble géométrique.

8.1.3 Ensembles probabilistes

Exemple d'utilisation 1

```
Vous pouvez écrire sémantiquement $\probaset{E}$ et $\probaset{G}$ mais pas taper
\verb+$\probaset{ABC}$+.
```

Vous pouvez écrire sémantiquement $\mathcal{E}$ et $\mathcal{G}$ mais pas taper `$\probaset{ABC}$`.

Exemple d'utilisation 2

```
Pour les indices, utilisez $\probaset*{E}{1}$, $\probaset*{E}{2}$ \dots
```

Pour les indices, utilisez $\mathcal{E}_1$, $\mathcal{E}_2$ ...

Fiches techniques

`\probaset (1 Argument)`

— `Argument`: un seul caractère `ASCII` majuscule indiquant un ensemble probabiliste.

`\probaset* (2 Arguments)`

— `Argument 1`: un seul caractère `ASCII` majuscule indiquant $\mathcal{U}$ dans le nom $\mathcal{U}_d$ d'un ensemble probabiliste.

— `Argument 2`: un texte donnant d dans le nom $\mathcal{U}_d$ d'un ensemble probabiliste.

8.1.4 Ensembles pour l'algèbre générale

Exemple d'utilisation 1

```
Vous pouvez écrire sémantiquement $\algeset{A}$, $\algeset{K}$, $\algeset{h}$ et
$\algeset{k}$ mais pas taper \verb+$\algeset{ABC}$+.
```

Vous pouvez écrire sémantiquement $\mathbb{A}$, $\mathbb{K}$, $\mathbb{h}$ et $\mathbb{k}$ mais pas taper `$\algeset{ABC}$`.

Exemple d'utilisation 2

```
Pour les indices, utilisez $\algeset*{k}{1}$, $\algeset*{k}{2}$ \dots
```

Pour les indices, utilisez $\mathbb{k}_1$, $\mathbb{k}_2$ $\dots$

Fiches techniques

`\algeset (1 Argument)`

— `Argument`: soit l'une des lettres `h` et `k`, soit un seul caractère `ASCII` majuscule indiquant un ensemble de type anneau ou corps.

`\algeset* (2 Arguments)`

— `Argument 1`: un seul caractère `ASCII` indiquant $\mathbb{U}$ dans le nom $\mathbb{U}_d$ d'un ensemble de type anneau ou corps.

— `Argument 2`: un texte donnant d dans le nom $\mathbb{U}_d$ d'un ensemble de type anneau ou corps.

8.1.5 Ensembles classiques

```
Vous pouvez utiliser directement $\nullset$, $\NN$, $\ZZ$, $\DD$, $\QQ$, $\RR$, $\CC$,
mais aussi $\PP$ pour l'ensemble des nombres premiers, $\HH$ pour les quaternions et
enfin $\OO$ pour les octonions.
```

Vous pouvez utiliser directement $\varnothing$, $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{D}$, $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$, mais aussi $\mathbb{P}$ pour l'ensemble des nombres premiers, $\mathbb{H}$ pour les quaternions et enfin $\mathbb{O}$ pour les octonions.

8.1.6 Ensembles classiques suffixés

```
Il est facile de taper $\RRn$, $\RRp$, $\RRs$, $\RRsn$ et $\RRsp$.
```

Il est facile de taper $\mathbb{R}_-$, $\mathbb{R}_+$, $\mathbb{R}^*$, $\mathbb{R}_-^*$ et $\mathbb{R}_+^*$.

Nous avons utilisé les suffixes `n` pour `Negatif`, `p` pour `Positif`, et `s` pour `star`, soit "étoile" en anglais. Il y a aussi les suffixes composites `sn` et `sp`.

Notez qu'il est interdit d'utiliser `$\CCn$` pour $\mathbb{C}_-$ car l'ensemble $\mathbb{C}$ ne possède pas de structure ordonnée standard. Jetez un oeil à la section suivante pour apprendre à taper $\mathbb{C}_-$ si vous en avez besoin. L'interdiction est ici purement sémantique !

Remarque. La table 1 de la présente page montre les associations autorisées entre ensembles classiques et suffixes.

TABLE 1 – Suffixes

	n	p	s	sn	sp
N			×		
P					
Z	×	×	×	×	×
D	×	×	×	×	×
Q	×	×	×	×	×
R	×	×	×	×	×
C			×		
H			×		
O			×		

8.1.7 Des suffixes à la carte

Exemple d'utilisation

```
Il est tout de même possible d'écrire $\specialset{\CC}{n}$ ou $\specialset{\HH}{sp}$.
Il y a aussi $\specialset*{\probaset{P}}{n}$ avec une autre mise en forme.
```

Il est tout de même possible d'écrire $\mathbb{C}_-$ ou $\mathbb{H}_+^*$. Il y a aussi $\mathcal{P}_{\leq 0}$ avec une autre mise en forme.

Fiches techniques

`\specialset (2 Arguments)`
`\specialset* (2 Arguments)`

— `Argument 1 :` l'ensemble à "suffixer".

— `Argument 2 :` l'un des suffixes `n`, `p`, `s`, `sn` ou `sp`.

8.2 Intervalles

8.2.1 Intervalles réels - Notation française (?)

Exemple d'utilisation 1

Dans cet exemple, la syntaxe fait référence à **O**-pened et **C**-losed pour "ouvert" et "fermé" en anglais. Nous verrons que **CC** et **OO** sont contractés en **C** et **O**.

```
Dans $I = ]a ; b] = \intervalOC{a}{b}$, vous constatez que la macro utilisée résout
un problème d'espacement vis à vis du signe $=$ .
```

Dans $I =]a\,;b] =]a\,;b]$, vous constatez que la macro utilisée résout un problème d'espacement vis à vis du signe $=$.

Exemple d'utilisation 2

Les crochets s'étendent verticalement automatiquement. Pour empêcher cela, il suffit d'utiliser la version étoilée de la macro. Dans ce cas, les crochets restent tout de même un peu plus grands que des crochets utilisés directement. Voici un exemple.

```
$\displaystyle \intervalC{ \frac{1}{2} }{ 1^{2^{3}} }
            = [ \frac{1}{2} ; 1^{2^{3}} ]
            = \intervalC*{ \frac{1}{2} }{ 1^{2^{3}} }$
```

$\displaystyle \left[\frac{1}{2} \,; 1^{2^{3}} \right] = [\frac{1}{2} ; 1^{2^{3}}] = \big[\frac{1}{2} \,; 1^{2^{3}} \big]$

Fiches techniques

Pour toutes les macros ci-dessous, la version non étoilée produit des délimiteurs qui s'étirent si besoin verticalement, tandis que la version étoilée ne le fait pas.

`\intervalCO (2 Arguments)`
`\intervalCO* (2 Arguments)`

— `Argument 1:` borne inférieure a de l'intervalle $[a\,;b[$.

— `Argument 2:` borne supérieure b de l'intervalle $[a\,;b[$.

`\intervalC (2 Arguments)`
`\intervalC* (2 Arguments)`

— `Argument 1:` borne inférieure a de l'intervalle $[a\,;b]$.

— `Argument 2:` borne supérieure b de l'intervalle $[a\,;b]$.

`\intervalO (2 Arguments)`
`\intervalO* (2 Arguments)`

— `Argument 1:` borne inférieure a de l'intervalle $]a\,;b[$.

— `Argument 2:` borne supérieure b de l'intervalle $]a\,;b[$.

`\intervalOC (2 Arguments)`
`\intervalOC* (2 Arguments)`

— `Argument 1:` borne inférieure a de l'intervalle $]a\,;b]$.

— `Argument 2:` borne supérieure b de l'intervalle $]a\,;b]$.

8.2.2 Intervalles réels - Notation américaine

Exemple d'utilisation

Dans cet exemple, la syntaxe fait référence à **P**-arenthèse.

```
Aux États Unis, un intervalle semi-fermé s'écrit $\intervalPC{a}{b} = (a ; b]$ et
un intervalle ouvert se tape $\intervalP{a}{b} = (a ; b)$.
```

Aux États Unis, un intervalle semi-fermé s'écrit $(a\,;b] = (a\,;b]$ et un intervalle ouvert se tape $(a\,;b) = (a\,;b)$.

Fiches techniques

Pour toutes les macros ci-dessous, la version non étoilée produit des délimiteurs qui s'étirent si besoin verticalement, tandis que la version étoilée ne le fait pas.

`\intervalCP (2 Arguments)`
`\intervalCP* (2 Arguments)`

— `Argument 1 :` borne inférieure a de l'intervalle $[a\,;b)$.

— `Argument 2 :` borne supérieure b de l'intervalle $[a\,;b)$.

`\intervalP (2 Arguments)`
`\intervalP* (2 Arguments)`

— `Argument 1 :` borne inférieure a de l'intervalle $(a\,;b)$.

— `Argument 2 :` borne supérieure b de l'intervalle $(a\,;b)$.

`\intervalPC (2 Arguments)`
`\intervalPC* (2 Arguments)`

— `Argument 1 :` borne inférieure a de l'intervalle $(a\,;b]$.

— `Argument 2 :` borne supérieure b de l'intervalle $(a\,;b]$.

8.2.3 Intervalles discrets d'entiers

Exemple d'utilisation

Dans l'exemple, la syntaxe fait référence à $\mathbb{Z}$ l'ensemble des entiers relatifs.

```
Par définition, $\ZintervalC{-1}{4} = \{ -1 ; 0 ; 1 ; 2 ; 3 ; 4 \}$. Donc nous avons
$\ZintervalC{-1}{4} = \ZintervalO{-2}{5}$.
```

Par définition, $[\![-1\,;4]\!] = \{-1\,;0\,;1\,;2\,;3\,;4\}$. Donc nous avons $[\![-1\,;4]\!] =]\!]-2\,;5[\![$.

Fiches techniques

Pour toutes les macros ci-dessous, la version non étoilée produit des délimiteurs qui s'étirent si besoin verticalement, tandis que la version étoilée ne le fait pas.

`\ZintervalCO (2 Arguments)`
`\ZintervalCO* (2 Arguments)`

— `Argument 1 :` borne inférieure a de l'intervalle $[\![a\,;b[\![$.

— `Argument 2 :` borne supérieure b de l'intervalle $[\![a\,;b[\![$.

`\ZintervalC (2 Arguments)`
`\ZintervalC* (2 Arguments)`

— `Argument 1 :` borne inférieure a de l'intervalle $[\![a\,;b]\!]$.

— `Argument 2 :` borne supérieure b de l'intervalle $[\![a\,;b]\!]$.

`\ZintervalO (2 Arguments)`
`\ZintervalO* (2 Arguments)`

— `Argument 1 :` borne inférieure a de l'intervalle $]\!]a\,;b[\![$.

— `Argument 2:` borne supérieure b de l'intervalle $\rrbracket a\,;b\llbracket$.

`\ZintervalOC (2 Arguments)`
`\ZintervalOC* (2 Arguments)`

— `Argument 1:` borne inférieure a de l'intervalle $\rrbracket a\,;b\rrbracket$.

— `Argument 2:` borne supérieure b de l'intervalle $\rrbracket a\,;b\rrbracket$.

8.2.4 Intervalles discrets d'entiers à la sauce « informatique »

Exemple d'utilisation

Dans l'exemple, la syntaxe fait référence à « Computer Science » soit « Informatique Théorique ».

```
Par définition, $\CSinterval{4}{7} = \{ 4 ; 5 ; 6 ; 7 \}$.
```

Par définition, $4\,..\,7 = \{4\,;5\,;6\,;7\}$.

Fiches techniques

`\CSinterval (2 Arguments)`

— `Argument 1:` borne inférieure a de l'intervalle $a\,..\,b$.

— `Argument 2:` borne supérieure b de l'intervalle $a\,..\,b$.

9 Analyse

9.1 Constantes

9.1.1 Constantes classiques

La liste complète

```
Voici la liste des constantes classiques où $\ttau = 2 \ppi$ est la benjamine :
$\ggamma$, $\ppi$, $\ttau$, $\ee$, $\ii$, $\jj$ and $\kk$.
```

Voici la liste des constantes classiques où $\tau = 2\pi$ est la benjamine : γ, π, τ, $\mathbf{e}$, $\mathbf{i}$, $\mathbf{j}$ and $\mathbf{k}$.

Remarque. Faites attention car `{\Large $\ppi \neq \pi$}` produit $\pi \neq \pi$. Comme vous le constatez, les symboles ne sont pas identiques. Ceci est vraie pour toutes les constantes grecques.

9.1.2 Constantes latines personnelles

Exemple d'utilisation

```
Il est aisé d'écrire $\ct{a} x^2 + \ct{b} x + \ct{c}$ au lieu de $a x^2 + b x + c$
afin de souligner que $\ct{a}$, $\ct{b}$ et $\ct{c}$ sont des constantes.
```

Il est aisé d'écrire $\mathbf{a}x^2 + \mathbf{b}x + \mathbf{c}$ au lieu de $ax^2 + bx + c$ afin de souligner que $\mathbf{a}$, $\mathbf{b}$ et $\mathbf{c}$ sont des constantes.

Fiche technique

`\ct (1 Argument)`

— `Argument`: un texte utilisant l'alphabet latin.

9.2 La fonction valeur absolue

Un exemple d'utilisation

```
Il est facile d'écrire $\abs{2}$ ou $\displaystyle \abs{\frac{3}{5}}$ voire aussi
$\displaystyle \abs*{\frac{3}{5}}$ si vous préférez des petits traits verticaux.
```

Il est facile d'écrire $|2|$ ou $\displaystyle \left|\frac{3}{5}\right|$ voire aussi $\displaystyle |\frac{3}{5}|$ si vous préférez des petits traits verticaux.

Remarque. Le code LaTeX vient directement de ce poste : `https://tex.stackexchange.com/a/43009/6880`.

Fiches techniques

`\abs (1 Argument)`
`\abs* (1 Argument)`

— `Argument`: l'expression à laquelle on applique la fonction valeur absolue.

9.3 Fonctions nommées spéciales

9.3.1 Un exemple d'utilisation

```
Quelques fonctions nommées supplémentaires (voir la liste ecomplète ci-dessous) :
$\ch x \neq ch x$ , $\ppcm(x;y)$ , $\lg x =\logb{2} x$ and $\expb{6} y$.
```

Quelques fonctions nommées supplémentaires (voir la liste ecomplète ci-dessous) : $\operatorname{ch} x \neq ch x$, $\operatorname{ppcm}(x\,;y)$, $\lg x = \log_2 x$ and $\exp_6 y$.

9.3.2 Fonctions nommées sans paramètre

Toutes les macros suivantes n'ont aucun paramètre.

`\ch`	`\sh`	`\th`
`\ach`	`\ash`	`\ath`
`\arccosh`	`\arcsinh`	`\arctanh`
`\acos`	`\asin`	`\atan`
`\pgcd`	`\ppcm`	

9.3.3 Fonctions nommées avec un paramètre

La liste complète

Toutes les macros suivantes ont un seul paramètre.

`\expb` (1 paramètre) `\logb` (1 paramètre)

Fiches techniques

`\expb (1 Argument)`
`\logb (1 Argument)`

— `Argument:` la base de l'exponentielle ou du logarithme.

9.4 Des notations complémentaires pour des suites spéciales

Exemple d'utilisation

```
Parfois nous avons besoin d'écrire $\seqplus{F}{1}{2}$ ou $\hypergeo{F}{1}{2}$ et
le fou (?\,) aime vraiment $\suprageo{F}{1}{2}{3}{4}$.
```

Parfois nous avons besoin d'écrire F_1^2 ou ${}_1F_2$ et le fou (?) aime vraiment ${}_1^4F_2^3$.

Fiches techniques

`\seqplus (2 Arguments)`

— `Argument 1:` l'exposant à droite.

— `Argument 2:` l'indice à droite.

`\hypergeo (2 Arguments)`

— `Argument 1:` l'indice à gauche.

— `Argument 2:` l'indice à droite.

`\suprageo (4 Arguments)`

— `Argument 1:` l'indice à gauche.

— `Argument 2:` l'indice à droite.

— `Argument 3:` l'exposant à droite.

— `Argument 4:` l'exposant à gauche.

9.5 Calcul différentiel

9.5.1 Les opérateurs ∂ et d

Exemple d'utilisation

```
Vous pouvez écrire $\dd{f}$ et $\pp{t}$ et aussi $ \dd[5]{x}$ ou $\pp[n]{x}$.
```

Vous pouvez écrire $\mathrm{d}f$ et ∂t et aussi d^5x ou $\partial^n x$.

Fiches techniques

`\dd [1 Option] (1 Argument)`
`\pp [1 Option] (1 Argument)`

— `Option:` utilisée, cette option sera mise en exposant du symbole ∂ ou d.

— `Argument:` la variable de différentiation à droite du symbole ∂ ou d.

9.5.2 Dérivation totale

Exemple d'utilisation 1

```
$\displaystyle \derpow*{f} (a) = \derpow{f} (a)
                               = \derfrac{f}{x} (a)
                               = \dersub{f}{x} (a)$
```

$$f'(a) = f^{(1)}(a) = \frac{\mathrm{d}f}{\mathrm{d}x}(a) = \mathrm{d}_x f(a)$$

Exemple d'utilisation 2

```
$\displaystyle \derpow*[3]{f}(a) = \derpow[3]{f} (a)
                                 = \derfrac[3]{f}{x} (a)
                                 = \dersub[3]{f}{x} (a)$
et $\displaystyle \derpow*[10]{\cos} a = \derfrac[10]{\cos}{x} (a)$ avec beaucoup trop
de primes à gauche (mais le compte y est).
```

$\displaystyle f'''(a) = f^{(3)}(a) = \frac{\mathrm{d}^3 f}{\mathrm{d}x^3}(a) = \mathrm{d}_x^3 f(a)$ et $\displaystyle \cos'''''''''' a = \frac{\mathrm{d}^{10}\cos}{\mathrm{d}x^{10}}(a)$ avec beaucoup trop de primes à gauche (mais le compte y est).

Exemple d'utilisation 3

```
Si $\displaystyle f(x) = \frac{1}{x^2+3}$ alors nous avons :
   $\displaystyle \derpow[3]{f} (a)
                = \derfrac*[3]{\left( \frac{1}{x^2+3} \right)}{x} (a)$.
```

Si $\displaystyle f(x) = \frac{1}{x^2+3}$ alors nous avons : $\displaystyle f^{(3)}(a) = \frac{\mathrm{d}^3}{\mathrm{d}x^3}\left(\frac{1}{x^2+3}\right)(a)$.

Fiches techniques

`\derpow [1 Option] (1 Argument)`
`\derpow* [1 Option] (1 Argument)`

— `Option`: utilisée, cette option sera l'exposant de dérivation mis entre des parenthèses pour la version non étoilée, et le nombre de primes pour la version étoilée.

— `Argument`: la fonction à différencier.

`\derfrac [1 Option] (2 Arguments)`
`\derfrac* [1 Option] (2 Arguments)`
`\dersub [1 Option] (2 Arguments)`

— `Option`: utilisée, cette option sera l'exposant de dérivation.

— `Argument 1`: la fonction à dériver.

— `Argument 2`: la variable.

9.5.3 Dérivation partielle

Exemple d'utilisation 1

```
$\displaystyle \partialfrac{f}{x} (a;b)
             = \partialsub{f}{x} (a;b)
             = \partialprime{f}{x} (a;b)$
```

$\dfrac{\partial f}{\partial x}(a\,;b) = \partial_x f(a\,;b) = f'_x(a\,;b)$

Exemple d'utilisation 2

```
$\displaystyle \partialfrac[3]{G}{f^2 | v} (a;b)
             = \partialfrac{G}{f^2 | v} (a;b)
             = \partialsub{G}{f^2 | v} (a;b)
             = \partialprime{G}{f^2 | v} (a;b)$
```

$\dfrac{\partial^3 G}{\partial f^2\,\partial v}(a\,;b) = \dfrac{\partial G}{\partial f^2\,\partial v}(a\,;b) = \partial_{f(2)\,v} G(a\,;b) = G'_{f(2)\,v}(a\,;b)$

Exemple d'utilisation 3

```
Si $\displaystyle f(x;y) = \frac{cos(x y)}{x^2+y^2}$ alors nous avons
   $\displaystyle \partialfrac[2]{f}{x | y}
                = \partialfrac*[2]{\left( \frac{cos(x y)}{x^2 + y^2} \right)}{x | y}$.
```

Si $f(x\,;y) = \dfrac{cos(xy)}{x^2+y^2}$ alors nous avons $\dfrac{\partial^2 f}{\partial x\,\partial y} = \dfrac{\partial^2}{\partial x\,\partial y}\left(\dfrac{cos(xy)}{x^2+y^2}\right)$.

Fiches techniques

`\partialfrac [1 Option] (2 Arguments)`
`\partialfrac* [1 Option] (2 Arguments)`

— `Option :` utilisée, cette option sera l'exposant total de dérivation mis en exposant de ∂.

— `Argument 1 :` la fonction à dériver partiellement.

— `Argument 2 :` les variables utilisées pour la dérivation partielle en utilisant la syntaxe suivante : par exemple, `x | y^3 | ...` indique de dériver suivant x une fois, puis suivant y trois fois... etc.

`\partialsub (2 Arguments)`
`\partialprime (2 Arguments)`

— `Argument 1 :` la fonction à dériver partiellement.

— `Argument 2 :` les variables utilisées pour la dérivation partielle en utilisant la syntaxe suivante : par exemple, `x | y^3 | ...` indique de dériver suivant x une fois, puis suivant y trois fois... etc.

9.6 Calcul intégral

9.6.1 L'opérateur crochet – 1ère version

Exemple d'utilisation 1

```
Par définition, $\displaystyle \int_{a}^{b} f(x) \dd{x} = \hook{F(x)}{a}{b}$ où
$\hook{F(x)}{a}{b} = F(b) - F(a)$.
```

Par définition, $\displaystyle \int_{a}^{b} f(x) \,\mathrm{d}x = \left[F(x)\right]_{a}^{b}$ où $\left[F(x)\right]_{a}^{b} = F(b) - F(a)$.

Exemple d'utilisation 2

Par défaut, les crochets s'étirent verticalement si besoin, mais si cela vous dérange, vous pouvez faire appel à la version étoilée de la macro comme dans l'exemple suivant

```
$\displaystyle \hook{\frac{x - 1}{5 + x^2}}{a}{b}
             = \hook*{\frac{x - 1}{5 + x^2}}{a}{b}$.
```

$\displaystyle \left[\frac{x-1}{5+x^2}\right]_{a}^{b} = [\frac{x-1}{5+x^2}]_{a}^{b}$.

Fiches techniques

`\hook (3 Arguments)`
`\hook* (3 Arguments)`

— `Argument 1 :` le contenu entre les crochets.

— `Argument 2 :` la borne inférieure affichée en indice.

— `Argument 3 :` la borne supérieure affichée en exposant.

9.6.2 L'opérateur crochet – 2nde version

Exemple d'utilisation 1

```
Vous pouvez utiliser $\vhook{F(x)}{a}{b}$ au lieu de $\hook{F(x)}{a}{b}$.
```

Vous pouvez utiliser $F(x)\big|_{a}^{b}$ au lieu de $\left[F(x)\right]_{a}^{b}$.

Exemple d'utilisation 2

Tout comme avec la première version de l'opérateur crochet, vous pouvez utiliser une version étoilée pour empêcher l'étirement verticalement du trait vertical. Voici un exemple.

```
$\displaystyle \vhook{\frac{x - 1}{5 + x^2}}{a}{b}
             = \vhook*{\frac{x - 1}{5 + x^2}}{a}{b}$.
```

$\displaystyle \frac{x-1}{5+x^2}\bigg|_a^b = \frac{x-1}{5+x^2}\Big|_a^b.$

Fiches techniques

`\vhook (3 Arguments)`
`\vhook* (3 Arguments)`

— `Argument 1 :` le contenu avant le trait vertical.

— `Argument 2 :` la borne inférieure affichée en indice.

— `Argument 3 :` la borne supérieure affichée en exposant.

9.6.3 Intégrales multiples

Le package réduit les espacements entres des symboles $\int$ successifs. Voici un exemple.

```
$\displaystyle
  \int \int \int F(x;y;z) \dd{x} \dd{y} \dd{z}
= \int_{a}^{b} \int_{c}^{d} \int_{e}^{f} F(x;y;z) \dd{x} \dd{y} \dd{z}$
```

$$\iiint F(x\,;y\,;z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \int_a^b\!\!\int_c^d\!\!\int_e^f F(x\,;y\,;z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z$$

Remarque. Par défaut, LATEX affiche $\displaystyle \int \int \int F(x\,;y\,;z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \int_a^b \int_c^d \int_e^f F(x\,;y\,;z)\,\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z$.

9.7 Tableaux de variation et de signe

Comment ça marche ?

Tout le boulot est fait par le package `tkz-tab` auquel on impose le choix d'une pointe de flèche plus visible. Nous vous demandons donc de vous reporter à la documentation de `tkz-tab` pour savoir comment s'y prendre.

Un exemple de tableaux de signes

```
\begin{tikzpicture}
\tkzTabInit{$x$ / 1 , $\cos(x)$ / 1}{$0$, $\frac{\pi}{2}$, $\pi$}
\tkzTabLine{, +, z, -, }
\end{tikzpicture}
```

x	0		$\frac{\pi}{2}$		π
$\cos(x)$		$+$	0	$-$	

Un exemple de tableaux de variation

```
\begin{tikzpicture}
    \tkzTabInit{$x$ / 1 , $f(x)$ / 1.5}{$-\infty$, $p$, $+\infty$}
    \tkzTabVar{+/ , -/ $f(p)$, +/ }
\end{tikzpicture}
```

x	$-\infty$		p		$+\infty$
$f(x)$		$\searrow$	$f(p)$	$\nearrow$	

Un exemple de tableaux de variation avec une dérivée

```
\begin{tikzpicture}
    \tkzTabInit{$x$ / 1 , $\cos(x)$ / 1, $\sin(x)$ / 1.5}{$0$, $\frac{\pi}{2}$, $\pi$}
    \tkzTabLine{, +, z, -, }
    \tkzTabVar{-/ 0, +/ 1, -/ 0}
\end{tikzpicture}
```

x	0		$\frac{\pi}{2}$		π
$\cos(x)$		$+$	0	$-$	
$\sin(x)$	0	$\nearrow$	1	$\searrow$	0

9.8 Comparaison asymptotique de suites et de fonctions

9.8.1 Les notations $\mathcal{O}$ et $\mathcal{o}$

Exemple d'utilisation 1

```
Vous pouvez utiliser les symboles $\bigO{}$ et $\smallO{}$ créés par Landau.
```

Vous pouvez utiliser les symboles $\mathcal{O}$ et $\mathcal{o}$ créés par Landau.

Exemple d'utilisation 2

```
Vous pouvez écrire $\bigO{x} \neq \smallO{x}$ et $e^{t + \smallO{t}} = e^{\bigO{t}}$.
```

Vous pouvez écrire $\mathcal{O}(x) \neq \mathcal{o}(x)$ et $e^{t+\mathcal{o}(t)} = e^{\mathcal{O}(t)}$.

Fiches techniques

`\bigO (1 Argument)`

`\smallO (1 Argument)`

— `Argument:` non vide, cet argument sera mis entre des parenthèses après $\mathcal{O}$ ou o.

9.8.2 La notation $\boldsymbol{\Omega}$

Exemple d'utilisation 1

```
Vous pouvez utiliser le symbole $\bigomega{}$ créé par Hardy et Littlewood.
```

Vous pouvez utiliser le symbole $\boldsymbol{\Omega}$ créé par Hardy et Littlewood.

Exemple d'utilisation 2

```
$f(n) = \bigomega{g(n)}$ signifie :
$\exists (m, n_0)$ tel que $n \geqslant n_0$ implique $f(n) \geqslant m g(n)$.
```

$f(n) = \boldsymbol{\Omega}(g(n))$ signifie : $\exists (m, n_0)$ tel que $n \geqslant n_0$ implique $f(n) \geqslant m g(n)$.

Fiche technique

`\bigomega (1 Argument)`

— `Argument:` non vide, cet argument sera mis entre des parenthèses après $\boldsymbol{\Omega}$.

9.8.3 La notation $\boldsymbol{\Theta}$

Exemple d'utilisation 1

```
Voici le dernier symbole $\bigtheta{}$ qui peut rendre service.
```

Voici le dernier symbole $\boldsymbol{\Theta}$ qui peut rendre service.

Exemple d'utilisation 2

```
$f(n) = \bigtheta{g(n)}$ signifie : $\exists (m, M, n_0)$ tel que $n \geqslant n_0$
implique $m g(n) \leqslant f(n) \leqslant M g(n)$.
```

$f(n) = \boldsymbol{\Theta}(g(n))$ signifie : $\exists (m, M, n_0)$ tel que $n \geqslant n_0$ implique $mg(n) \leqslant f(n) \leqslant Mg(n)$.

Fiche technique

`\bigtheta (1 Argument)`

— `Argument:` non vide, cet argument sera mis entre des parenthèses après $\boldsymbol{\Theta}$.

10 Géométrie

10.1 Points et lignes

10.1.1 Points

Exemple d'utilisation 1

```
$\pt{I}$ indique un point nommé "I".
```

I indique un point nommé "I".

Exemple d'utilisation 2

```
Une liste de points : $\pt*{I}{1}$, $\pt*{I}{2}$ \dots
```

Une liste de points : I_1, I_2 …

Exemple d'utilisation 3

```
Une droite $(\pts{AB})$ au lieu de $(\pt{A}\pt{B})$.
```

Une droite (AB) au lieu de (AB).

Fiches techniques

`\pt (1 Argument)`

— `Argument` : un texte donnant le nom d'un point.

`\pt* (2 Arguments)`

— `Argument 1` : un texte indiquant UP dans le nom UP_{down} d'un point.

— `Argument 2` : un texte indiquant *down* dans le nom UP_{down} d'un point.

`\pts (1 Argument)`

— `Argument` : un texte indiquant des noms de points non indicés.

10.1.2 Droites

L'opérateur `\parallel` utilise des obliques au lieu de barres verticales comme le montre l'exemple qui suit.

```
$(\pts{AB}) \parallel (\pts{CD})$ au lieu de $(\pts{AB}) \stdparallel (\pts{CD})$.
```

(AB) // (CD) au lieu de (AB) $\parallel$ (CD).

10.2 Vecteurs

10.2.1 Les écrire

Exemple d'utilisation 1

```
Voici un vecteur $\vect{ABCDEF}$ avec beaucoup de lettres et vous pouvez écrire
$\vect*{e}{rot}$ au lieu de $\vect{e_{rot}}$.
```

Voici un vecteur $\overrightarrow{ABCDEF}$ avec beaucoup de lettres et vous pouvez écrire $\overrightarrow{e}_{rot}$ au lieu de $\overrightarrow{e_{rot}}$.

Exemple d'utilisation 2

```
Vous pouvez écrire $\vect{i}$ and $\vect*{j}{2}$ sans point.
```

Vous pouvez écrire $\overrightarrow{\imath}$ and $\overrightarrow{\jmath}_2$ sans point.

Fiches techniques

`\vect (1 Argument)`

— `Argument:` un texte donnant le nom d'un vecteur.

`\vect* (2 Arguments)`

— `Argument 1:` un texte indiquant *up* dans le nom $\overrightarrow{up}_{down}$ d'un vecteur.

— `Argument 2:` un texte indiquant *down* dans le nom $\overrightarrow{up}_{down}$ d'un vecteur.

10.2.2 Norme

Exemple d'utilisation

```
Nous pouvons écrire $\norm{\vect{i}}$, $\displaystyle \norm{\frac{2}{7} \vect*{e}{k}}$,
ou $\displaystyle \norm*{\frac{2}{7} \vect*{e}{k}}$ avec de petites barres verticales.
```

Nous pouvons écrire $\|\overrightarrow{\imath}\|$, $\displaystyle \left\|\frac{2}{7}\overrightarrow{e}_k\right\|$, ou $\displaystyle \|\frac{2}{7}\overrightarrow{e}_k\|$ avec de petites barres verticales.

Remarque. Le code LaTeX vient directement de ce message : `https://tex.stackexchange.com/a/43009/6880`.

Fiches techniques

`\norm (1 Argument)`
`\norm* (1 Argument)`

— `Argument:` le vecteur sur lequel appliquer la norme.

10.2.3 Produit scalaire – Écriture minimaliste

Exemple d'utilisation - Version longue

```
En mathématique, il est usage d'écrire un produit scalaire avec un point via
$\dotprod{\dfrac{1}{2} \vect{i}}{\vect{j}}$ .
```

En mathématique, il est usage d'écrire un produit scalaire avec un point via $\dfrac{1}{2} \vec{\imath} \cdot \vec{\jmath}$.

Exemple d'utilisation - Version courte mais restrictive

Dans l'exemple suivant, le préfixe `v` est pour **v**-ector.

```
On peut aussi parfois juste taper $\vdotprod{i}{j}$ .
```

On peut aussi parfois juste taper $\vec{\imath} \cdot \vec{\jmath}$.

Fiches techniques

`\dotprod (2 Arguments)`

— `Argument 1 :` le premier vecteur qu'il faut taper via la macro `\vect`.

— `Argument 2 :` le second vecteur qu'il faut taper via la macro `\vect`.

`\vdotprod (2 Arguments)` où `v = v-ector`

— `Argument 1 :` le nom du premier vecteur sans utiliser la macro `\vect`.

— `Argument 2 :` le nom du second vecteur sans utiliser la macro `\vect`.

10.2.4 Produit scalaire – Écriture « physicienne »

Dans l'exemple suivant, le préfixe `a` est pour **a**-ngle, et `v` pour **v**-ector.

```
Les physiciens pourront utiliser
$\displaystyle \adotprod{\frac{1}{2} \vect{i}}{\vect{j}}$ ,
$\displaystyle \adotprod*{\frac{1}{2} \vect{i}}{\vect{j}}$ ,
$\displaystyle \vadotprod{i}{j}$
ou
$\displaystyle \vadotprod*{i}{j}$ .
```

Les physiciens pourront utiliser $\displaystyle \left\langle \frac{1}{2} \vec{\imath} \,\middle|\, \vec{\jmath} \right\rangle$, $\langle \frac{1}{2} \vec{\imath} \mid \vec{\jmath} \rangle$, $\langle \vec{\imath} \mid \vec{\jmath} \rangle$ ou $\langle \vec{\imath} \mid \vec{\jmath} \rangle$.

Fiches techniques

`\adotprod (2 Arguments)` où `a = a-ngle`
`\adotprod* (2 Arguments)`

— `Argument 1 :` le premier vecteur qu'il faut taper via la macro `\vect`.

— `Argument 2 :` le second vecteur qu'il faut taper via la macro `\vect`.

`\vadotprod (2 Arguments)` où `a = a-ngle` et `v = v-ector`

— `Argument 1 :` le nom du premier vecteur sans utiliser la macro `\vect`.

— `Argument 2 :` le nom du second vecteur sans utiliser la macro `\vect`.

10.2.5 Produit vectoriel

Exemple d'utilisation - Version longue

```
Un produit vectoriel peut s'écrire via $\crossprod{\dfrac{1}{2} \vect{i}}{\vect{j}}$ .
```

Un produit vectoriel peut s'écrire via $\dfrac{1}{2}\vec{\imath} \wedge \vec{\jmath}$.

Exemple d'utilisation - Version courte mais restrictive

```
Dans certain cas, un produit vectoriel s'écrit vite via $\vcrossprod{i}{j}$ .
```

Dans certain cas, un produit vectoriel s'écrit vite via $\vec{\imath} \wedge \vec{\jmath}$.

Fiche technique

`\crossprod (2 Arguments)`

— `Argument 1 :` le premier vecteur qu'il faut taper via la macro `\vect`.

— `Argument 2 :` le second vecteur qu'il faut taper via la macro `\vect`.

`\vcrossprod (2 Arguments)` où `v = v-ector`

— `Argument 1 :` le nom du premier vecteur sans utiliser la macro `\vect`.

— `Argument 2 :` le nom du second vecteur sans utiliser la macro `\vect`.

10.3 Coordonnées

Exemple d'utilisation 1

```
On peut choisir d'écrire
$\displaystyle \pt{I} \coord{\frac{1}{3} | -4 | 0}$
ou bien
$\displaystyle \pt{I} \coord*{\frac{1}{3} | -4 | 0}$ .
```

On peut choisir d'écrire $\mathrm{I}\left(\dfrac{1}{3}\,;-4\,;0\right)$ ou bien $\mathrm{I}(\dfrac{1}{3}\,;-4\,;0)$.

Exemple d'utilisation 2

Dans l'exemple suivant, le préfixe `v` est pour **v**-ertical.

```
Pour les vecteurs, on peut préférer $\vect{i} \vcoord{3 | -4 | 0}$, voire
$\vect{i} \vcoord*{3 | -4 | 0}$ , à la place de $\vect{i} \coord{3 | -4 | 0}$
afin de bien différencier les coordonnées de points de celles de vecteurs.
```

Pour les vecteurs, on peut préférer $\vec{\imath}\begin{pmatrix}3\\-4\\0\end{pmatrix}$, voire $\vec{\imath}\begin{bmatrix}3\\-4\\0\end{bmatrix}$, à la place de $\vec{\imath}\,(3\,;-4\,;0)$ afin de bien différencier les coordonnées de points de celles de vecteurs.

Fiches techniques

`\coord (1 Argument)`
`\coord* (1 Argument)`

— `Argument` : l'argument est une suite de "morceaux" séparés par des barres `|` , chaque morceau étant une coordonnée. Il peut n'y avoir qu'un seul morceau.

`\vcoord (1 Argument)` où `v = v-ertical`
`\vcoord* (1 Argument)` pour des crochets à la place de parenthèses

— `Argument` : l'argument est une suite de "morceaux" séparés par des barres `|` , chaque morceau étant une coordonnée. Il peut n'y avoir qu'un seul morceau.

10.4 Nommer un repère

Exemple d'utilisation 1 – La méthode basique

Commençons par la manière la plus basique d'écrire un repère *(nous verrons d'autres méthodes qui peuvent être plus efficaces)*.

```
Dans le plan, trois points $\pt{O}$, $\pt{I}$ and $\pt{J}$ non alignés définissent
un repère cartésien $\axes{\pt{O} | \pt{I} | \pt{J}}$.
```

Dans le plan, trois points O, I and J non alignés définissent un repère cartésien $(\mathrm{O}\,;\mathrm{I},\mathrm{J})$.

Exemple d'utilisation 2 – La méthode basique en version étoilée

Dans l'exemple ci-dessous, on voit que la version étoilée produit des petites parenthèses.

```
$\displaystyle \axes{\pt{O} | \frac{7}{3} \vect{i} | \vect{j}}$
ou
$\displaystyle \axes*{\pt{O} | \frac{7}{3} \vect{i} | \vect{j}}$
```

$\displaystyle \left(\mathrm{O}\,;\frac{7}{3}\vec{\imath},\vec{\jmath}\right)$ ou $\displaystyle (\mathrm{O}\,;\frac{7}{3}\vec{\imath},\vec{\jmath})$

Exemple d'utilisation 3 – La méthode basique en dimension quelconque

Il faut au minimum deux "morceaux" séparés par des barres `|`, cas de la dimension 1, mais il n'y a pas de maximum, cas d'une dimension quelconque $n > 0$.

```
$\axes{\pt{O} | \vect*{i}{1} | \vect*{i}{2} | \vect*{i}{3} | \dots |
 \vect*{i}{9} | \vect*{i}{10} | \vect*{i}{11} | \vect*{i}{12}}$
```

$(\mathrm{O}\,;\vec{\imath}_1, \vec{\imath}_2, \vec{\imath}_3, \dots, \vec{\imath}_9, \vec{\imath}_{10}, \vec{\imath}_{11}, \vec{\imath}_{12})$

Exemple d'utilisation 4 – Repère affine

Dans l'exemple suivant, le préfixe `p` est pour **p**-oint.

```
$\paxes{O | I | J | K}$ évite de taper $\axes{\pt{O} | \pt{I} | \pt{J} | \pt{K}}$.
```

$(\mathrm{O}\,;\mathrm{I}, \mathrm{J}, \mathrm{K})$ évite de taper $(\mathrm{O}\,;\mathrm{I}, \mathrm{J}, \mathrm{K})$.

Exemple d'utilisation 5 – Repère vectoriel (méthode 1)

Dans l'exemple suivant, le préfixe `v` est pour **v**-ecteur.

```
$\vaxes{\pt{O} | i | j}$ est un raccourci de $\axes{\pt{O} | \vect{i} | \vect{j}}$.
```

$(\mathrm{O}\,;\vec{\imath}, \vec{\jmath})$ est un raccourci de $(\mathrm{O}\,;\vec{\imath}, \vec{\jmath})$.

Exemple d'utilisation 6 – Repère vectoriel (méthode 2)

Dans l'exemple suivant, le préfixe `pv` permet de combiner à la fois les fonctionnalités proposés par les préfixes `p` et `v`.

```
$\pvaxes{O | i | j}$ donne rapidement $\axes{\pt{O} | \vect{i} | \vect{j}}$.
```

$(\mathrm{O}\,;\vec{\imath}, \vec{\jmath})$ donne rapidement $(\mathrm{O}\,;\vec{\imath}, \vec{\jmath})$.

Fiches techniques

`\axes (1 Argument)`
`\axes* (1 Argument)`

— `Argument`: l'argument est une suite de "morceaux" séparés par des barres `|`.
  - Le premier morceau est l'origine du repère.
  - Les morceaux suivants sont des points ou des vecteurs qui "définissent" chaque axe.

`\paxes (1 Argument)` où `p = p-oint`

— `Argument`: l'argument est une suite de "morceaux" séparés par des barres `|`.
  - Le premier morceau est le nom de l'origine du repère sur laquelle la macro-commande `\pt` sera automatiquement appliquée.
  - Viennent ensuite les noms des points "définissant" chaque axe. Pour chacun de ces points la macro-commande `\pt` sera automatiquement appliquée.

`\vaxes (1 Argument)` où `v = v-ector`

— `Argument`: l'argument est une suite de "morceaux" séparés par des barres `|`.
  - Le premier morceau est l'origine du repère.

- Viennent ensuite les noms des vecteurs "définissant" chaque axe. Pour chacun de ces vecteurs la macro-commande `\vect` sera automatiquement appliquée.

`\pvaxes (3 Arguments)` où `pv = p + v`

— `Argument :` l'argument est une suite de "morceaux" séparés par des barres `|`.
  - Le premier morceau est le nom de l'origine du repère sur laquelle la macro-commande `\pt` sera automatiquement appliquée.
  - Viennent ensuite les noms des vecteurs "définissant" chaque axe. Pour chacun de ces vecteurs la macro-commande `\vect` sera automatiquement appliquée.

10.5 Arcs circulaires

Exemple d'utilisation 1

```
Voici un arc $\arc{ABCDEF}$ utilisant beaucoup de lettres, et vous pouvez écrire
$\arc*{A}{rot}$ à la place de $\arc{A_{rot}}$.
```

Voici un arc $\overparen{ABCDEF}$ utilisant beaucoup de lettres, et vous pouvez écrire $\overparen{A}_{rot}$ à la place de $\overparen{A_{rot}}$.

Exemple d'utilisation 2

```
$\arc{i}$ et $\arc*{j}{2}$ n'affiche pas de point sous l'arc.
```

$\overparen{\imath}$ et $\overparen{\jmath}_2$ n'affiche pas de point sous l'arc.

Fiches techniques

`\arc (1 Argument)`

— `Argument :` un texte donnant le nom d'un arc circulaire.

`\arc* (2 Arguments)`

— `Argument 1 :` un texte indiquant *up* dans le nom $\overparen{up}_{down}$ d'un arc circulaire.

— `Argument 2 :` un texte indiquant *down* dans le nom $\overparen{up}_{down}$ d'un arc circulaire.

10.6 Angles

10.6.1 Angles géométriques intérieurs

Exemple d'utilisation 1

```
Voici un angle géométrique intérieur $\anglein{ABCDEF}$ avec un long nom, et vous
pouvez écrire $\anglein*{A}{rot}$ au lieu de $\anglein{A_{rot}}$.
```

Voici un angle géométrique intérieur $\widehat{ABCDEF}$ avec un long nom, et vous pouvez écrire $\widehat{A}_{rot}$ au lieu de $\widehat{A_{rot}}$.

Exemple d'utilisation 2

```
Vous pouvez aussi écrire $\anglein{i}$ and $\anglein*{j}{2}$ sans point.
```

Vous pouvez aussi écrire $\widehat{\imath}$ and $\widehat{\jmath}_2$ sans point.

Fiches techniques

`\anglein (1 Argument)`

— `Argument`: un texte donnant le nom d'un angle intérieur.

`\anglein* (2 Arguments)`

— `Argument 1`: un texte indiquant *up* dans le nom $\widehat{up}_{down}$ d'un angle intérieur.

— `Argument 2`: un texte indiquant *down* dans le nom $\widehat{up}_{down}$ d'un angle intérieur.

10.6.2 Angles orientés de vecteurs

Sans chapeau - Version longue

```
En mathématique, il est d'usage de noter les angles orientés via
$\displaystyle \angleorient{\frac{1}{2} \vect{i}}{\vect{j}}$
ou
$\displaystyle \angleorient*{\frac{1}{2} \vect{i}}{\vect{j}}$ .
```

En mathématique, il est d'usage de noter les angles orientés via $\left(\dfrac{1}{2}\vec{\imath}\,;\vec{\jmath}\right)$ ou $(\dfrac{1}{2}\vec{\imath}\,;\vec{\jmath})$.

Sans chapeau - Version courte mais restrictive

Dans l'exemple suivant, le préfixe `v` est pour **v**-ector.

```
Avec des noms de vecteurs utilisant juste des lettres, on peut juste taper
$\displaystyle \vangleorient{i}{j}$
ou
$\displaystyle \vangleorient*{i}{j}$ (la seconde écriture n'apporte rien de nouveau).
```

Avec des noms de vecteurs utilisant juste des lettres, on peut juste taper $(\vec{\imath}\,;\vec{\jmath})$ ou $\left(\vec{\imath}\,;\vec{\jmath}\right)$ (la seconde écriture n'apporte rien de nouveau).

Avec un chapeau

Dans l'exemple suivant, le préfixe `h` est pour **h**-at, et `v` pour **v**-ector.

```
Si vous préférez les angles orientés avec un chapeau, tapez les alors via
$\displaystyle \hangleorient{\frac{1}{2} \vect{i}}{\vect{j}}$ ,
$\displaystyle \hangleorient*{\frac{1}{2} \vect{i}}{\vect{j}}$ ,
$\displaystyle \hvangleorient{i}{j}$
ou
$\displaystyle \hvangleorient*{i}{j}$ (la dernière écriture n'apportant rien de neuf).
```

Si vous préférez les angles orientés avec un chapeau, tapez les alors via $\displaystyle \widehat{\left(\frac{1}{2} \vec{\imath} ; \vec{\jmath} \right)}$, $\displaystyle \widehat{(\frac{1}{2} \vec{\imath} ; \vec{\jmath})}$, $\widehat{(\vec{\imath} ; \vec{\jmath})}$ ou $\widehat{\left(\vec{\imath} ; \vec{\jmath} \right)}$ (la dernière écriture n'apportant rien de neuf).

Fiche technique

`\angleorient (2 Arguments)`
`\hangleorient (2 Arguments)` où `h = h-at`

— `Argument 1:` le premier vecteur qu'il faut taper via la macro `\vect`.

— `Argument 2:` le second vecteur qu'il faut taper via la macro `\vect`.

`\vangleorient (2 Arguments)` où `v = v-ector`
`\hvangleorient (2 Arguments)` où `h = h-at` et `v = v-ector`

— `Argument 1:` le nom du premier vecteur sans utiliser la macro `\vect`.

— `Argument 2:` le nom du second vecteur sans utiliser la macro `\vect`.

11 Arithmétiques

11.1 Opérateurs de base

Pour des raisons d'expressivité des codes LaTeX, les opérateurs binaires `\divides`, `\notdivides` et `\modulo` ont été ajoutés comme alias de `\mid`, `\nmid` et `\bmod` respectivement qui sont proposés par le package `amssymb`.

```
$10 \divides 150$ et $10 \notdivides 154$ rend mieux que $10 | 150$ et $10 \not| 154$.

Facile alors d'écrire que $a \equiv b \modulo p \iff p \divides (a - b)$.
```

$10 \mid 150$ et $10 \nmid 154$ rend mieux que $10 | 150$ et $10 \not| 154$.
Facile alors d'écrire que $a \equiv b \bmod p \iff p \mid (a - b)$.

11.2 Fractions continuées

11.2.1 Fractions continuées standard

Exemple d'utilisation

Dans l'exemple suivant, la notation en ligne semble être due à Alfred Pringsheim. La notation à gauche utilise toujours le maximum d'espace pour améliorer la lisibilité.

```
$ \contfrac{u_0 | u_1 | u_2 | \dots | u_n}
= \contfrac*{u_0 | u_1 | u_2 | \dots | u_n}$
```

$u_0 + \cfrac{1}{u_1 + \cfrac{1}{u_2 + \cfrac{1}{\dots + \cfrac{1}{u_n}}}} = u_0 + \frac{1\,\rvert}{\lvert\, u_1} + \frac{1\,\rvert}{\lvert\, u_2} + \frac{1\,\rvert}{\lvert\, \dots} + \frac{1\,\rvert}{\lvert\, u_n}$

Fiches techniques

`\contfrac (1 Argument)`
`\contfrac* (1 Argument)`

— `Argument` : tous les éléments de la fraction continuée séparés par des `|`.

11.2.2 Fractions continuées généralisées

Exemple d'utilisation

Voici comment écrire une fraction continuée généralisée.

```
$\displaystyle \contfracgene{a | b | c | d | e | f | \dots | y | z}
            = \contfracgene*{a | b | c | d | e | f | \dots | y | z}$
```

$a + \cfrac{b}{c + \cfrac{d}{e + \cfrac{f}{\dots + \cfrac{y}{z}}}} = a + \frac{b\,\rvert}{\lvert\, c} + \frac{d\,\rvert}{\lvert\, e} + \frac{f\,\rvert}{\lvert\, \dots} + \frac{y\,\rvert}{\lvert\, z}$

Fiches techniques

`\contfracgene (1 Argument)`
`\contfracgene* (1 Argument)`

— `Argument` : tous les éléments de la fraction continuée généralisée séparés par des `|`.

11.2.3 Comme une fraction continuée isolée

Exemple d'utilisation

La raison d'être de la macro ci-dessous vient juste de son usage en interne.

```
Les fous (?\,) adorent vraiment écrire des choses comme $\singlecontfrac{a}{b}$.
```

Les fous (?) adorent vraiment écrire des choses comme $\frac{a\,\rvert}{\lvert\, b}$.

Fiche technique

`\singlecontfrac (2 Arguments)`

— `Argument 1 :` le pseudo numérateur.

— `Argument 2 :` le pseudo dénominateur.

11.2.4 L'opérateur $\mathcal{K}$

Exemple d'utilisation 1

La notation suivante est proche de celle qu'utilisait Carl Friedrich Gauss.

```
$\displaystyle
  \contfracope_{k=1}^{n} (b_k:c_k)
= \cfrac{b_1}{\contfracgene{c_1 | b_2 | c_2 | b_3 | \dots | b_n | c_n}}$
```

$$\mathop{\mathcal{K}}_{k=1}^{n} (b_k : c_k) = \cfrac{b_1}{c_1 + \cfrac{b_2}{c_2 + \cfrac{b_3}{\dots + \cfrac{b_n}{c_n}}}}$$

Remarque. La lettre $\mathcal{K}$ vient de "kettenbruch" qui signifie "fraction continuée" en allemand.

Exemple d'utilisation 2

```
$\displaystyle
  u_0 + \contfracope_{k=1}^{n} (1:u_k)
= \contfrac{u_0 | u_1 | u_2 | \dots | u_n}$
```

$$u_0 + \mathop{\mathcal{K}}_{k=1}^{n} (1 : u_k) = u_0 + \cfrac{1}{u_1 + \cfrac{1}{u_2 + \cfrac{1}{\dots + \cfrac{1}{u_n}}}}$$

12 Algèbre

12.1 Polynômes, séries formelles et compagnie

12.1.1 Polynômes et fractions polynômiales

Exemple d'utilisation 1 : Polynômes

```
$\polyset{\RR}{X}$ est l'ensemble des polynômes à coefficients réels en la variable
$X$, et $\polyset{\RR}{X | Y | Z}$ est l'ensemble des polynômes à coefficients réels
en les variables $X$ , $Y$ et $Z$.
```

$\mathbb{R}\left[X\right]$ est l'ensemble des polynômes à coefficients réels en la variable X, et $\mathbb{R}\left[X\,;Y\,;Z\right]$ est l'ensemble des polynômes à coefficients réels en les variables X , Y et Z.

Exemple d'utilisation 2 : Fractions polynômiales

```
$\polyfracset{\QQ}{T}$ et $\polyfracset{\QQ}{S_1 | S_2 | \dots | S_k}$ permettent
d'indiquer des ensemble de fractions polynomiales à coefficients rationnels.
```

$\mathbb{Q}\left(T\right)$ et $\mathbb{Q}\left(S_1\,;S_2\,;\dots;S_k\right)$ permettent d'indiquer des ensemble de fractions polynomiales à coefficients rationnels.

12.1.2 Séries formelles et leurs corps de fractions

Exemple d'utilisation 1 : Séries formelles

```
$\serieset{\CC}{X}$ et $\serieset{\CC}{T | O | P}$ permettent de travailler avec des
séries formelles à coefficients complexes.
```

$\mathbb{C}\left[\left[X\right]\right]$ et $\mathbb{C}\left[\left[T\,;O\,;P\right]\right]$ permettent de travailler avec des séries formelles à coefficients complexes.

Exemple d'utilisation 2 : Corps des fractions de séries formelles

```
$\seriefracset{\ZZ}{X}$ et $\seriefracset{\ZZ}{Z | T | O | P}$ permettent de travailler
avec des fractions de séries formelles à coefficients entiers.
```

$\mathbb{Z}\left(\left(X\right)\right)$ et $\mathbb{Z}\left(\left(Z\,;T\,;O\,;P\right)\right)$ permettent de travailler avec des fractions de séries formelles à coefficients entiers.

12.1.3 Polynômes de Laurent et séries formelles de Laurent

Exemple d'utilisation 1 : Polynômes de Laurent

```
$\polylaurentset{\RR}{X} = \polyset{\RR}{X | X^{-1}}$ est l'ensemble des polynômes
réels de Laurent en $X$. On propose de généraliser comme suit (notation non standard) :
$\polylaurentset{\RR}{X_1 | X_2} = \polyset{\RR}{X_1 | X_1^{-1} | X_2 | X_2^{-1}}$
```

$\mathbb{R}\{X\} = \mathbb{R}[X\,;X^{-1}]$ est l'ensemble des polynômes réels de Laurent en X. On propose de généraliser comme suit (notation non standard) : $\mathbb{R}\{X_1\,;X_2\} = \mathbb{R}[X_1\,;X_1^{-1}\,;X_2\,;X_2^{-1}]$

Exemple d'utilisation 2 : Séries formelles de Laurent

```
$\serielaurentset{\QQ}{X} = \serieset{\QQ}{X | X^{-1}}$ est l'ensemble des séries
formelles rationnelles de Laurent en $X$. On généralise via, notation non standard,
$\serielaurentset{\QQ}{X_1 | X_2} = \serieset{\QQ}{X_1 | X_1^{-1} | X_2 | X_2^{-1}}$
```

$\mathbb{Q}\{\{X\}\} = \mathbb{Q}[[X\,;X^{-1}]]$ est l'ensemble des séries formelles rationnelles de Laurent en X. On généralise via, notation non standard, $\mathbb{Q}\{\{X_1\,;X_2\}\} = \mathbb{Q}[[X_1\,;X_1^{-1}\,;X_2\,;X_2^{-1}]]$

12.1.4 Toutes les fiches techniques

`\polyset` (2 Arguments)
`\polyfracset` (2 Arguments)
`\serieset` (2 Arguments)
`\seriefracset` (2 Arguments)
`\polylaurentset` (2 Arguments)
`\serielaurentset` (2 Arguments)

— `Argument 1`: l'ensemble auquel les coefficients appartiennent.

— `Argument 2`: cet argument est une suite de "morceaux" séparés par des barres `|`, chaque morceau étant une variable formelle.

12.2 Matrices

Comment ça marche ?

Tout le boulot est fait par le package `nicematrix` auquel on impose l'option `transparent`. Veuillez vous reporter à la documentation de `nicematrix` pour savoir comment s'y prendre.

Exemple 1 tiré de la documentation de `nicematrix`

```
$\begin{pmatrix}
    1      & \cdots & \cdots & 1      \\
    0      & \ddots &        & \vdots \\
    \vdots & \ddots & \ddots & \vdots \\
    0      & \cdots & 0      & 1
\end{pmatrix}$
```

$$\begin{pmatrix} 1 & \cdots & \cdots & 1 \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1 \end{pmatrix}$$

Exemple 2 tiré de la documentation de `nicematrix`

```
$\begin{pNiceMatrix}[name=mymatrix]
    1 & 2 & 3 \\
    4 & 5 & 6 \\
    7 & 8 & 9
\end{pNiceMatrix}$

\tikz[remember picture,overlay]
\draw (mymatrix-2-2) circle (2mm) ;
```

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$$

Exemple 3 tiré de la documentation de `nicematrix`

```
$\left(
    \begin{NiceArray}{CCCC:C}
        1  & 2  & 3  & 4  & 5  \\
        6  & 7  & 8  & 9  & 10 \\
        11 & 12 & 13 & 14 &15
    \end{NiceArray}
\right)$
```

$$\left(\begin{array}{cccc:c} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \end{array}\right)$$

13 Historique

Nous ne donnons ici qu'un très bref historique de `lymath` côté utilisateur principalement. Tous les changements sont disponibles uniquement en anglais dans le dossier `change-log` : voir le code source de `lymath` sur `github`.

2019-09-27 Nouvelle version mineure `0.5.0-beta`.

- Ajout des macros `\dsum` et `\dprod` qui sont vis à vis de `\sum` et `\prod` des équivalents de `\dfrac` pour `\frac`.
- En arithmétique, ajout des opérateurs `\divides`, `\notdivides` et `\modulo`.
- En géométrie, une nouvelle macro et un opérateur modifié.
  - `\pts` permet d'indiquer plusieurs points.
  - `\parallel` utilise des obliques pour symboliser le parallélisme au lieu de barres verticales.
- En logique, il y a les nouveautés suivantes.
  - La version doublement étoilée `\eqdef**` donne une deuxième écriture symbolique d'un symbole égal de type définition *(cette notation vient du langage B)*.
  - Ajout de `\liesimp` comme alias de `\Longleftarrow`.
  - Les macros `\vimplies`, `\viff` et `\vliesimp` sont des versions verticales de `\implies`, `\iff` et `\liesimp`.
  - Comme pour les égalités, il existe les macros `\impliestest`, `\iffhyp` ... etc.

2019-09-06 Nouvelle version mineure `0.4.0-beta`.

- Dans « *Logique et fondements* », différents types de signes d'inéquation et de non égalité pour des cas de test, d'hypothèse faite et de condition à vérifier.
- Intégration du package `tkz-tab` pour rédiger des tableaux de variations et de signes.
- Intégration du package `nicematrix` pour écrire des matrices.

2019-07-23 Nouvelle version mineure `0.3.0-beta`.

- Une nouvelle section « *Logique et fondements* » a été ajoutée.
  - Trois types de signes = décorés sont proposés : voir les macros `\eqdef` , `\eqid` et `\eqtest`.
  - Via la macro `\explain`, il devient facile d'expliquer des étapes de raisonnement ou des calculs.
- Pour les ensembles, la macro `\fieldset` a été renommé `\algeset` et la macro `\PP` permet d'indiquer l'ensemble des nombres premiers.
- En géométrie, il y a quelques nouveautés.
  - La macro `\hangleorient` permet l'écriture d'angles orientés avec un chapeau en plus.
  - Les macros `\vangleorient` et `\vhangleorient` évite d'avoir à utiliser `\vect` lorsque l'on a juste des vecturs simples nommés et non coefficientés.
  - De même pour les macros `\vdotprod`, `\vadotprod` et `\vcroosprod`.
- Ajout de `\lymathsubsep` qui définit le séparateur des arguments de second niveau.

2019-02-21 Nouvelle version mineure `0.2.0-beta`.

- L'usage de `//` pour les macros-commandes avec un nombre quelconque d'arguments a été remplacé par celui de `|`.
- En géométrie, il y a diverses nouveautés.
  - Ajout de l'écriture de coordonnées, de produits scalaires et de produits vectoriels.
  - `\axis` a été correctement traduit en `\axes`.
  - Les macros `\gpaxis` et `\gpvaxis` deviennent `\paxes` et `\pvaxes` pour être cohérent avec `\pt` qui a remplacé l'ancien `\gpt`.

- En analyse, ajout de la macro commande étoilée `\derpow*` pour la gestion automatique des primes d'une dérivée.
- Une nouvelle section "algèbre" propose des macros pour écrire des ensembles de polynômes, de fractions polynomiales, de séries formelles, de fractions de séries formelles, et aussi de polynômes et de séries formelles de Laurent.
- Redéfinition de `\frac` et `\dfrac` pour obtenir des traits de fraction un peu plus longs.
- Ajout de `\lymathsep` qui définit le séparateur d'arguments.

2017-11-01 Nouvelle version mineure `0.1.0-beta` : pour les ensembles, les fonctions et la géométrie, il y a eu des changements et l'ajout de nouveaux outils.

2017-10-21 Historique court de `lymath` ajouté au présent document.

2017-10-18 Nouvelle version "patchée" `0.0.2-beta` : de nouveaux outils pour le calcul différentiel.

2017-10-06 Nouvelle version "patchée" `0.0.1-beta` : de nouveaux outils pour l'arithmétique, la géométrie, le calcul intégral et le calcul différentiel.

2017-10-02 Première version `0.0.0-beta` du package.