

Les extensions `amsfonts`, `amsmath`, `amssymb` et `mathrsfs` sont nécessaires.

Un tableau des symboles usuels se trouve en page 64.

7.1 Composition

Les recommandations suivantes proviennent de l'Inspection générale. ⁽¹⁾

Les minuscules qui correspondent à des variables, des inconnues, des indices, ... sont écrites en italique.

Néanmoins, sont écrits en romain les identificateurs de fonctions et constantes prédéfinies : d'une part, les noms des fonctions usuelles \sin , \cos , \ln , \exp , ... et, d'autre part, les constantes e ($= \exp(1)$), i (base des imaginaires purs) et le symbole d pour écrire un élément différentiel. On écrira donc :

$$e^{ix} = \cos x + i \sin x \quad \int_1^2 (f(x) - \ln x) dx$$

Les ensembles de nombres sont normalement écrits en gras (\mathbf{R} , etc.) dans un texte imprimé, les caractères ajourés (\mathbb{R} , etc.) étant en principe réservés à l'écriture au tableau.

Les noms des points sont écrits en majuscules et en romain (et non pas en italiques) ⁽²⁾. On écrira donc :

$$\overrightarrow{AB} \quad x_M \quad OM^2$$

Pour les ensembles de points en géométrie, on a intérêt à utiliser des italiques, voire des cursives : la courbe \mathcal{C} , la droite \mathcal{D} , ...

Pour écrire les majuscules en romain systématiquement, voici plusieurs possibilités, évitant d'utiliser les commandes locales `\text{}` ou `\mathrm{}` ⁽³⁾. Celles-ci portent donc sur *tout* le texte.

(1). D'après le *Lexique des règles typographiques en usage à l'Imprimerie nationale*.

(2). Ce sera le cas systématiquement dans la brochure, sauf si cette contrainte alourdit le code.

(3). Voir la section 7.2.3.1, page 49.

- on utilise l'option `upright` de l'extension ⁽⁴⁾ `fourier` ⁽⁵⁾ (cette extension propose une fonte alternative complète de la fonte « naturelle » `Computer Modern`);
- on saisit dans le préambule la commande suivante correspondant à A ⁽⁶⁾, en faisant de même pour les vingt-cinq autres lettres.

```
\DeclareMathSymbol{A}{%
\mathalpha}{operators}{'A}
```

Lorsqu'elles reçoivent un symbole, les lettres i et j doivent perdre leur point. On écrit \vec{i} et \vec{j} .

7.2 Environnement mathématique

7.2.1 Expressions « en ligne » ($\$ \dots \$$) ou « hors texte » ($\backslash [\dots \backslash]$)

La simple formule $E = mc^2$ s'insère correctement dans une ligne de texte tandis que l'équation

$$\frac{\pi}{4} = \int_0^1 \frac{1}{t^2+1} dt$$

a été écrite en passant à la ligne (et en centrant), ce qui la met en évidence.

Si l'on avait voulu que cette dernière équation soit simplement insérée dans la ligne courante, on aurait obtenu $\frac{\pi}{4} = \int_0^1 \frac{1}{t^2+1} dt$.

La première présentation est dite « en ligne ». Les formules sont encadrées aussi bien entre :

- `\(et \)`
- `\$ et \$`
- `\begin{math}` et `\end{math}`

Les trois écritures `\begin{math}4x=9\end{math}`, `\(4x=9\)` et `\$4x=9\$` donnent les mêmes résultats. En pratique, c'est la troisième (entre `\$`) qui est la plus souvent utilisée.

(4). Souvent utilisée sur la banque d'annales de l'APMEP, [125].

(5). Insérer ainsi `\usepackage[upright]{fourier}` dans le préambule.

(6). L'accent devant le second A est grave.

La seconde est dite « hors ligne ». Les formules sont encadrées aussi bien entre :

- `\[` et `\]`
- `$$` et `$$` ⁽⁷⁾
- `\begin{equation*}` et `\end{equation*}`

De plus, il ne faut pas faire de saut de ligne en mode mathématique sous peine d'arrêt de compilation, comme dans l'exemple suivant :

`$f(x) = x`

`g(x)=x^2$`

7.2.2 Commande `\displaystyle` et autres commandes de taille de police

La taille des caractères est gérée automatiquement en fonction de leurs emplacements (exposant, exposant d'exposant, ...) et du mode texte ou hors-texte.

Quatre tailles sont prédéfinies, hors-texte, texte, scripte et sous-scripte. La taille peut donc être forcée grâce aux macros liées, `\displaystyle`, `\textstyle`, `\scriptstyle` et `\scriptscriptstyle`.

Ces commandes amènent L^AT_EX à gérer les indices et les exposants comme si le texte mathématique est en mode « hors ligne ». Cela peut introduire une modification locale de l'interligne.

$$2^{2^{2^2}} \quad 2^{2^{2^2}}$$

`$2^{\{2^{\{2^2\}}}$`

`$2^{\{2^{\%`

`\scriptstyle {2^{\scriptstyle 2}}\}}}$`

Nous retrouverons `\displaystyle` pour les fractions, les sommes, les limites ou les intégrales.

Pour que les expressions mathématiques dans un texte soient toujours écrites en taille `\displaystyle` (et donc éviter de le préciser à chaque fois), on peut introduire la demande `\everymath{\displaystyle}` qui imposera la taille voulue à toute expression qui suivra cette demande. En particulier, cette commande peut (et doit ?) être placée dans le préambule pour affecter tout le document.

7.2.3 Texte et espace dans un environnement mathématique

7.2.3.1 Commande `\text{...}`

Un texte ordinaire ne peut pas être inséré tel quel dans un environnement mathématique.

(7). Même si la seconde possibilité est plus aisée à saisir, elle est, pour des raisons qui dépassent le cadre de cette brochure, nettement déconseillée. De plus, les espaces ne sont pas toujours correctement faites (pour les puristes!).

La saisie, par exemple, de `$x=1` et donc `y=2$` donne $x = 1$ et donc $y = 2$! L^AT_EX comprend qu'il y a un produit des facteurs e , t , d , ... et c : il le traduit alors comme tel.

Dans cet environnement, pour composer...

- un *symbole* en romain, on utilise `\mathrm{symbole}`;
- du *texte* ordinaire, on utilise `\text{texte}`.

`$x=1 \text{ et donc } y=2$`

`$$\mathrm{i}^2=-1$`

$x = 1$ et donc $y = 2$

$i^2 = -1$

Remarque 1. Si cela est possible, il vaut mieux saisir `$x=1$` et donc `$y=2$`.

Remarque 2. Pour le e et le i droit, il vaut mieux utiliser `\mathrm{e}` plutôt que `\text{e}`, car, si l'on est dans un environnement en italique, `\text` reprend la mise en forme et fera apparaître des lettres en italiques et non droites. Voyez aussi la différence entre point P_5 et point P_5 , obtenus respectivement avec `\textsf{point P_5}` et `\textsf{point P_5}`.

Remarque 3. `$x=1 \mathrm{ et donc } y=2$` donne $x = 1$ et donc $y = 2$.

7.2.3.2 Espace

Les espaces entre deux `$` sont ignorées : elles n'ont d'autre but que de clarifier le code source (surtout quand les formules deviennent complexes).

Par exemple, `$1+2=3$` donne le même résultat que `$1 + 2 = 3$`.

7.2.4 Mise en boîte

Il peut arriver qu'une expression mathématique, dans un texte, soit coupée en fin de ligne et continuée en début de ligne suivante. Pour éviter ce problème, on crée une boîte insécable *autour* de l'expression concernée grâce à la commande `\mbox` ⁽⁸⁾ :

`\mbox{$1+x+x^2+x^3+x^4$}`

7.3 Les commandes de base

7.3.1 Écriture d'un nombre et virgule

7.3.1.1 La commande `\np`

Dans les pays anglo-saxons, la virgule est un séparateur de milliers. Par conséquent, lorsque nous

(8). Voir les boîtes au chapitre 4, page 29.

tapons $\$3,14\$$, nous obtenons 3, 14. Pour éliminer cette espace derrière la virgule, on peut aussi bien écrire $\$3\{,\}14\$$ que $\$\nombre{3,14}\$$ ⁽⁹⁾. De même, pour avoir 1200, on peut aussi bien écrire $\$\nombre{1200}\$$ que $\$1\,200\$$ ⁽¹⁰⁾.

On peut aussi utiliser le package `numprint` avec l'option `np` ⁽¹¹⁾. Alors `\np{3,4567}` écrira correctement le nombre 3,4567. Et `\np{6e-12}` donne $6 \cdot 10^{-12}$.

Il permet d'afficher des valeurs avec une unité, sans risque de coupure au passage à la ligne.

`\np[kg]{91}` donne 91 kg.

7.3.1.2 La commande `\DecimalMathComma`

Une autre piste, consistant à faire comprendre à \LaTeX une fois pour toutes que la virgule est bien notre séparateur usuel, il *suffit tout simplement* d'écrire dans le préambule, et avec l'option `français` du package `babel`, l'instruction :

`\DecimalMathComma`

7.3.2 Flèches

7.3.2.1 Flèches « droites »

Principes des commandes générant des flèches.

- toutes les commandes finissent par `arrow` (flèche) ;
- le préfixe obligatoire `left` (gauche), `right` (droite), `up` (haut) et `down` (bas) indique la direction ;
- le préfixe facultatif `long` donne une version longue ;
- la première lettre de la commande mise en majuscule rend la flèche double ;
- on peut mettre des flèches aux deux extrémités en collant les deux mots `left` et `right`.
- on obtient la négation du symbole fléché avec la préfixe `n`, sauf si c'est la « version longue ».

\rightarrow \leftarrow \longrightarrow \longleftarrow \longleftrightarrow \Uparrow \nrightarrow

`\rightarrow` `\Leftarrow`
`\longrightarrow` `\Lrightarrow`
`\leftrightharrow` `\Uparrow`
`\nRightarrow`

7.3.2.2 « équivaut à » et « implique »

Les macros `\Lrightarrow` (\Leftrightarrow) et `\iff` (\iff) semblent identiques mais différent en réalité sur la quantité de blanc qui les entoure. Le symbole correct à utiliser est bien `\iff`. C'est une question de lisibilité de la formule ⁽¹²⁾.

$$A \iff B \qquad A \iff B$$

(9). Avec `\usepackage[français]{babel}`.

(10). Le `\`, correspond à une espace fine.

(11). Donc avec `\usepackage[np]{numprint}`.

(12). De plus, le second symbole est plus rapide à saisir !

`\Lrightarrow B` `A \iff B`

De même, c'est `\implies` (\implies) et non `\Rightarrow` (\Rightarrow) ni `\Longrightarrow` (\implies) qu'il faut utiliser dans une relation d'implication.

$$A \implies B \qquad A \Rightarrow B \qquad A \implies B$$

`\implies B` `A \Rightarrow B`

`\Longrightarrow B`

Pour les négations \nrightarrow et \nLeftarrow des symboles \implies et \iff , on utilise la commande `\centernot\implies` et le package `centernot` ⁽¹³⁾.

$$A \nrightarrow B \qquad A \nLeftarrow B$$

`\centernot\implies B`

`\centernot\iff B`

7.3.2.3 Flèches « obliques »

Les flèches « obliques », très utiles dans les tableaux de variation, sont construites avec la même idée d'orientation. Il suffit de penser aux points cardinaux : `n` pour le nord, `w` pour l'ouest, `s` pour le sud et `e` pour l'est. Ainsi une flèche dirigée en haut à droite sera orientée « nord-est ».



`\nearrow` `\searrow`
`\nwarrow` `\swarrow`

7.3.2.4 Flèches « arrondies »

Elles peuvent servir pour indiquer le sens de rotation, notamment en trigonométrie.



`\circlearrowleft` `\circlearrowright`
`\curvearrowleft` `\curvearrowright`

7.3.2.5 Flèches des fonctions

Le « : » pour une application s'obtient avec `\colon`. Cette commande règle automatiquement l'espace d'une part et d'autre des deux points ; elle est donc à préférer à « : ».

`\to` `\mapsto` `\longmapsto`

$$f: [0; \pi[\to [0; 1]$$

$$g: x \mapsto 3x + 4$$

`f \colon [0\,; \pi[\to [4\,; 7]`

`g \colon x \mapsto 3x+4`

(13). `\not\implies` donne \nrightarrow .

7.3.3 Indices et exposants

Le syntaxe est : `^{exposant}` et `_{indice}`.

Si *exposant* et *indice* ne sont formés que d'un caractère, les accolades sont inutiles.

$$x^2 \quad x^{15} \quad x^{3a} \quad x_5$$

$$\$x^2\$ \quad \$x^{15}\$ \quad \$x^{3a}\$ \quad \$x_5\$$$

Si $U_n = 2^n$, alors $U_{n+1} = 2^{n+1}$.

Si $U_n = 2^n$, alors $U_{n+1} = 2^{n+1}$.

$$U_{n+1} \neq U_n + 1$$

$$\$U_{n+1} \neq U_n + 1\$$$

$$F_n = 2^{2^n}$$

$$\$F_n = 2^{2^n}\$$$

$$x_1^2 + x_2^2$$

$$\$x_1^2 + x_2^2\$$$

(La saisie de `$x_1^2+x_2^2$` donne $x_1^2 + x_2^2$.)

Au passage, les accents circonflexes $\hat{}$ et \wedge sont respectivement obtenus avec `\circonflexe` ou avec `\wedge` (en mode mathématique pour celui-ci). On peut s'en servir pour écrire une ligne de saisie sur une calculatrice ⁽¹⁴⁾.

7.3.4 Fractions

La syntaxe est : `\frac{num}{den}`. La taille du symbole est automatiquement adaptée au contexte (pour garder le même interlignage).

$$\$\frac{2}{3}\$ \text{ donne donc } \frac{2}{3}.$$

Lorsque *num* et *den* ne comporte chacun qu'un caractère, on peut écrire rapidement `\frac num den` pour obtenir le même résultat.

On peut imposer la taille des caractères avec la commande `\displaystyle\frac{a}{b}` ou `\dfrac{a}{b}` ou ⁽¹⁵⁾ pour agrandir leur taille ou la commande `\tfrac{a}{b}` ou `\scriptstyle\frac{a}{b}` pour la diminuer, comme l'indiquent les deux tableaux ci-dessous :

$\frac{a}{b}$	<code>\frac{a}{b}</code>	$\frac{a}{b}$	<code>\[\frac{a}{b}\]</code>
$\tfrac{a}{b}$	<code>\tfrac{a}{b}</code>	$\frac{a}{b}$	<code>\dfrac{a}{b}</code>
$\tfrac{a}{b}$	<code>\[\tfrac{a}{b}\]</code>	$\frac{a}{b}$	<code>\[\dfrac{a}{b}\]</code>

$$\frac{1}{3} + \frac{1}{6} = \frac{1}{2}$$

$$\$\dfrac{1}{3} + \dfrac{1}{6} = \dfrac{1}{2}\$$$

$$t = 1,09^{\frac{1}{9}} - 1 \quad \$t = 1,09^{\tfrac{1}{9}} - 1\$$$

(14). Voir aussi sur ce point page 89.

(15). La première est un raccourci de la seconde.

On peut écrire des fractions dans des fractions :

$$\frac{1}{1 + \frac{1}{x^2}} \quad \backslash\frac{1}{1+\dfrac{1}{x^2}}$$

Des espaces peuvent être placées pour agrandir le trait de fraction :

$$\frac{1}{x+2} \quad \$\dfrac{1}{\sim x+2}\$$$

Enfin, on peut utiliser une grande barre oblique :

$$\$3/4\$ \text{ donne } 3/4 \text{ et } \$3\Big/4\$ \text{ donne } 3/4.$$

Voyez aussi le paragraphe 10.3.2, page 88, une macro pour écrire autrement des fractions.

7.3.5 Radicaux

La racine carrée de x est saisie par `\sqrt{x}`; la racine n -ième de x est produite par `\sqrt[n]{x}`.

$$\sqrt{49} = 7 \quad \sqrt{1 + \sqrt{2}} \quad \sqrt{2x+1}$$

$$\sqrt{a} + \sqrt{b} \neq \sqrt{a+b} \quad \sqrt[3]{64} = 4 \quad \sqrt[6]{2} = 2^{\frac{1}{6}}$$

$$\sqrt{2013 \times \sqrt{2014 \times 2016 + 1} + 1}$$

$$\$\sqrt{49}=7\$ \quad \$\sqrt{1+\sqrt{2}}\$$$

$$\$\sqrt{2\, , x+1}\$$$

$$\$(\sqrt{a} + \sqrt{b}) \neq \sqrt{a+b}\$$$

$$\$\sqrt[3]{64}=4\$$$

$$\$\sqrt[6]{2}=2^{\frac{1}{6}}\$$$

$$\$\sqrt{2\, , 013\ \times\ \sqrt{2\, , 014\ \times\ 2016\ +\ 1}\ +\ 1}\$$$

$$\$\times\ 2\, , 016\ +\ 1\ +\ 1)\$$$

$$\sqrt{1 - \frac{3}{x^2}} \text{ se code } \$\sqrt{1-\dfrac{3}{x^2}}\$.$$

Le symbole seul $\sqrt{}$ est obtenu avec `\surd`.

7.3.6 Fonctions usuelles

arg cos exp lim ln log max min sin tan

$$\$\arg\$ \quad \$\cos\$ \quad \dots \quad \$\tan\$$$

$$\sin(2x) = 2 \cos x \sin x \quad \cos(2x) = \cos^2 x - 1$$

$$\$\sin(2x) = 2 \ \cos x \ \sin x\$$$

$$\$\cos(2x) = \cos^2 x - 1\$$$

7.3.7 Dérivées

On utilise la commande `\prime` ou `'`.

$$(u \cdot v)' = u' \cdot v + u \cdot v'$$

$$\$(u \ \cdot\ v)' = u' \ \cdot\ v + u \ \cdot\ v'\$$$

$f'(x) = (x^2)'$ donne :

$$f'(x) = (x^2)' = 2x$$

Pour la dérivée seconde de f , on utilise deux apostrophes et non pas le guillemet anglais : `f''` donne f'' tandis que `f"` donne f'' .

7.3.8 Limites, intégrales, sommes et produits

7.3.8.1 Symboles et utilisation

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

En mode « en ligne » :

$\sum_{k=1}^n \frac{1}{k^2}$ $\lim_{n \rightarrow +\infty} u_n$
 $\int_1^2 \frac{1}{t^2} dt$ $\prod_{k=1}^n f(k)$

`\sum_{k=1}^n \frac{1}{k^2}`
`\int_1^2 \frac{1}{t^2} dt`
`\lim_{n \to +\infty} u_n`
`\prod_{k=1}^n f(k)`

En mode « hors ligne » :

$$\sum_{k=1}^n \frac{1}{k^2} \quad \int_1^2 \frac{1}{t^2} dt \quad \lim_{n \rightarrow +\infty} u_n \quad \prod_{k=1}^n f(k)$$

`\[\sum_{k=1}^n \frac{1}{k^2}\]` ...

Avec la commande `\displaystyle` :

La somme $\sum_{k=1}^n \frac{1}{k(k+1)}$ est égale à $\frac{n}{n+1}$.

La somme `\displaystyle{\sum_{k=1}^n \frac{1}{k(k+1)}}` est ...

7.3.8.2 Placement des « limites »

Les styles `\displaystyle`, `\scriptstyle`, ... déterminent, entre autres, la taille des symboles, leur espacement et la place des indices et exposants. On contrôle indépendamment le placement des « limites » avec `\limits` et `\nolimits`.

Comparons ainsi :

$\sum_a^b \int_a^b \sum_a^b \int_a^b \sum_a^b \int_a^b \sum_a^b \int_a^b$
`\sum_a^b` `\int_a^b`
`\displaystyle \sum_a^b`
`\displaystyle \int_a^b`
`\displaystyle \sum\limits_a^b`
`\displaystyle \int\limits_a^b`
`\sum\limits_a^b` `\int\limits_a^b`

En mode texte :

$\sum_{i=1}^n f(i)$ $\sum_{i=1}^n f(i)$ $\sum_{i=1}^n f(i)$

`\sum_{i=1}^n f(i)`
`\displaystyle \sum_{i=1}^n f(i)`
`\displaystyle \sum\limits_{i=1}^n f(i)`

Cela s'applique aussi pour le produit `\prod`, \prod .

7.3.9 Vecteurs

7.3.9.1 Commandes `\vec` et `\overrightarrow`

La syntaxe est `\overrightarrow{vect}` ⁽¹⁶⁾ ou bien `\vec{vect}`.

$$\overrightarrow{AB} \quad \vec{u}$$

`\overrightarrow{\mathrm{AB}}` `\vec{u}`

Lorsque l'écriture du vecteur a deux lettres, il vaut mieux utiliser la première syntaxe pour éviter un \overrightarrow{AB} !

7.3.9.2 Cas des lettres *i* et *j*

Lorsqu'elles reçoivent un symbole, les lettres *i* et *j* doivent perdre leur point. On écrit :

\vec{i} `\vec{\imath}` \vec{j} `\vec{\jmath}`

7.3.9.3 Norme

L'écriture de la norme d'un complexe est obtenue avec la double commande `\lVert ... \rVert` ou bien avec la double commande `\left| ... \right|` :

$$\|\overrightarrow{AM}\| = 1 \quad \left\| \overrightarrow{AM} \right\| = 1$$

`\lVert \overrightarrow{AM} \rVert = 1`
`\left| \overrightarrow{AM} \right| = 1`

La seconde utilise en fait des délimiteurs ⁽¹⁷⁾ .

7.3.10 Complexes

7.3.10.1 Réels et imaginaires

On pourra écrire `\text{Re } z` pour $\text{Re } z$.

Dans l'écriture anglo-saxonne, $\Re z$ et $\Im z$ sont donnés respectivement par `\Re z` et `\Im z`.

7.3.10.2 Conjugué

Le conjugué \bar{z} de z s'obtient avec la commande ⁽¹⁸⁾ `\overline{z}`.

$$\overline{1+4i} \quad \overline{1+4\mathrm{i}}$$

7.3.10.3 Module

L'écriture du module d'un nombre complexe est obtenue avec la double commande `\lvert ... \rvert` ou bien avec la double commande `\left| ... \right|`.

$$|1-2i| \quad \left| 1-2i \right|$$

(16). Littéralement : « flèche au-dessus orientée à droite ».

(17). Voir la section 7.4.

(18). Celle-ci est à préférer à `\bar{z}`, qui donne aussi \bar{z} , surtout pour écrire le conjugué d'un nombre complexe donné. En effet, `\bar{1+4i}` donne $1+4i$, ce qui n'est pas très heureux.

`\lvert 1 - 2\, \mathrm{i} \rvert`
`\left\lvert 1 - 2\, \mathrm{i} \right\vert`

La seconde utilise en fait des délimiteurs ⁽¹⁹⁾.

Certains utilisateurs préfèrent l'une ou l'autre des deux doubles commandes `| ... |` ou `\vert ... \vert`; ceci est déconseillé. En effet, ces commandes sont à utiliser dans le cadre d'opérateur (alors que la commande préconisée est bien utilisée comme délimiteur) ⁽²⁰⁾.

Remarque.

La valeur absolue d'un nombre est obtenue de même avec la double commande `\lvert ... \rvert` ou bien avec la double commande `\left\lvert ... \right\lvert`. Certains utilisent l'une ou l'autre des deux doubles commandes `| ... |` ou `\vert ... \vert`; ceci est déconseillé. En effet, ces commandes sont à utiliser dans le cadre d'opérateur (alors que la commande préconisée est utilisée comme délimiteur) ⁽²¹⁾.

7.3.10.4 Argument

On utilise la commande `\arg`.

L'écriture $\arg z$ est ainsi obtenue par `\arg z`.

7.3.11 Matrices

7.3.11.1 Matrices usuelles

L'utilisateur dispose de deux moyens pour écrire une matrice.

La première utilise l'extension `amsmath`, qui permet de définir des matrices avec des encadrements différents (entre parenthèses, crochets, etc.).

- `{matrix}` : matrice sans délimiteur ;
- `{pmatrix}` : matrice entre parenthèses (...);
- `{vmatrix}` : matrice entre barres [...];
- `{Vmatrix}` : matrice entre doubles barres ||...||;
- `{bmatrix}` : matrice entre crochets [...];
- `{Bmatrix}` : matrice entre accolades {...}.

$$\begin{array}{ccc} a & b & \begin{pmatrix} a & b \\ c & d \end{pmatrix} \\ c & d & \begin{vmatrix} a & b \\ c & d \end{vmatrix} \\ \left\| \begin{array}{cc} a & b \\ c & d \end{array} \right\| & \left[\begin{array}{cc} a & b \\ c & d \end{array} \right] & \left\{ \begin{array}{cc} a & b \\ c & d \end{array} \right\} \end{array}$$

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

(19). Voir la section 7.4.

(20). La gestion des espaces est différente!

(21). La gestion des espaces est différente!

La seconde utilise les délimiteurs (voir la section 7.4) et l'extension `array` qui travaille sur les tableaux mathématiques (voir la section 8.17, page 76).

De plus, les coordonnées vectorielles tout comme les coefficients binomiaux peuvent être affichés comme une matrice à 1 colonne ⁽²²⁾ :

$$\begin{pmatrix} n \\ p \end{pmatrix} \quad \text{\begin{pmatrix} n \\ p \end{pmatrix}}$$

Par ailleurs, la transposée tA de la matrice A se code `\t A`.

7.3.11.2 Matrices « bordées »

On peut aussi écrire une matrice associée. Voici par exemple celle qui correspond au graphe probabiliste donné page 134.

$$M = \begin{array}{c} A \\ B \\ C \end{array} \begin{pmatrix} 0,6 & 0,1 & 0,3 \\ 0,2 & 0,6 & 0,2 \\ 0,35 & 0,05 & 0,6 \end{pmatrix}$$

```
$M=\bordermatrix{
&A&B&C\cr
A&0,6 & 0,1 & 0,3\cr
B&0,2 & 0,6 & 0,2\cr
C&0,35 & 0,05 & 0,6\cr
}$
```

7.3.11.3 Espacement vertical

Si l'on écrit des nombres fractionnaires, le résultat n'est pas forcément heureux :

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{5} \end{pmatrix}$$

Voilà deux méthodes pour améliorer la présentation.

1. La hauteur des lignes est gérée par la commande `\arraystretch`, qui gère un facteur d'espace-ment (1 par défaut). On le redéfinit.

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{5} \end{pmatrix}$$

```
\renewcommand{\arraystretch}{2.2}
\begin{pmatrix}
\dfrac{1}{2} & \dfrac{1}{3} \\
\dfrac{1}{4} & \dfrac{1}{5}
\end{pmatrix}
```

(22). Voir aussi le paragraphe 8.17.4, page 77.

ou encore

```
{\renewcommand{\arraystretch}{2.2}
\begin{pmatrix}
\dfrac{1}{2} & \dfrac{1}{3} \\
\dfrac{1}{4} & \dfrac{1}{5}
\end{pmatrix}
```

2. On augmente l'interligne.

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{5} \end{pmatrix}$$

```
\begin{pmatrix}
\dfrac{1}{2} & \dfrac{1}{3} \\
\dfrac{1}{4} & \dfrac{1}{5}
\end{pmatrix}
```

7.3.12 Systèmes

On utilise l'environnement `cases`. Il produit une accolade verticale à gauche (par défaut) et l'intérieur (c'est-à-dire les équations) se gère comme dans une matrice avec des `&` et des `\\`.

$$|x| = \begin{cases} -x & \text{si } x \text{ est négatif} \\ x & \text{si } x \text{ est positif (ou nul)} \end{cases}$$

```
\lvert x \rvert =
\begin{cases}
-x & \text{\text{si } $x$ est négatif} \\
x & \text{\text{si } $x$ est positif (ou nul)}
\end{cases}
```

$$\begin{cases} 2x + 13y = 43 \\ 3x - y = 3 \end{cases}$$

```
\begin{cases}
2x+13y=43 \\
3x-y=3
\end{cases}
```

$$\begin{cases} x = -2 + t \\ y = 3 - t \\ z = 4 + 5t \end{cases} \quad (t \in \mathbb{R})$$

```
\begin{cases}
\tilde{x} = -2 + t \\
\tilde{y} = 3 - t \\
\tilde{z} = 4 + 5t
\end{cases}
(t \in \mathbb{R})
```

D'autres exemples sont donnés au paragraphe 8.17.2, page 76, via des tableaux (mathématiques).

7.3.13 Parallèles et perpendiculaires

$D \perp D'$ se code `\perp D'` ou `\bot D'` ⁽²³⁾.

Pour écrire le fait que les droites D et D' sont parallèles, il y a plusieurs possibilités ⁽²⁴⁾, au goût du lecteur :

```
D // D'           $D // D'$
D /\ D'           $D \ /\ D'$
D // D'           $D \ /!\ / D'$
D || D'           $D \parallel D'$
```

7.3.14 Ensemble de nombres entiers

L'ensemble des entiers de 1 à n , noté souvent avec une paire de doubles crochets $\llbracket 1;n \rrbracket$, peut se coder `\llbracket 1,;n \rrbracket` ⁽²⁵⁾.

7.3.15 Probabilités

```
A \cup B           $A \cup B$
A \cap B = \emptyset   $A \cap B = \varnothing$
```

```
\complement A     $\complement A$
\overline{A}       $\overline{A}$
```

```
\mathrm{P}(A)=0,15   $\mathrm{P}(A)=0,15$
\mathrm{P}_B(A)=0,5  $\mathrm{P}_B(A)=0,5$
```

```
A \setminus B       $A \setminus B$
p(A \cap B) = p(A) \times p_A(B)
                    $p(A \cap B) = p(A) \times p_A(B)$
```

```
\binom{n}{p}       $\displaystyle \binom{n}{p}$
```

```
p(B) = p(A) \times p_A(B) + p(\overline{A}) \times p_{\overline{A}}(B)
$p(B) = p(A) \times p_A(B) + p(\overline{A}) \times p_{\overline{A}}(B)$
```

7.3.16 Arithmétique

```
a \equiv b \pmod n   $a \equiv b \pmod n$
a \equiv b (mod n)   $a \equiv b \pmod n$
a \equiv b (n)       $a \equiv b \pmod n$
a \equiv b \bmod n   $a \equiv b \bmod n$
```

(23). Seul `\perp` des deux est un symbole relationnel.
(24). Petit jeu avec les espaces !
(25). C'est l'une des nombreuses possibilités. Celle-ci joue sur l'espace négative entre les deux paires de crochets. Il y a aussi, par exemple, le package `mathabx` qui donne des crochets plus marqués avec la saisie de `\lbrack 1,;n \rbrack`. Néanmoins l'appel de ce package transforme certains signes mathématiques, comme \square , `\square`.