

Mode mathématique

Jean-Michel Hufflen

Février 2006

1 Introduction au mode mathématique

1.1 Premier contact

L^AT_EX connaît deux modes principaux :

- le *mode texte*, qui est le seul que vous ayez pratiqué jusqu'à présent,
- le *mode mathématique*, que nous allons découvrir ci-après.

Ces deux modes présentent de nombreuses différences, ils n'emploient pour ainsi dire pas les mêmes commandes, même si quelques-unes — rares — sont utilisables dans les deux.

Le mode mathématique *simple* correspond aux formules à taper dans le courant du texte, par exemple, $a + b = x$, obtenu par `$a+b=x$`. Le mode mathématique *centré* correspond aux formules hors-texte, par exemple :

$$a + b = x$$

obtenu par `\[a+b=x\]` ou `$$a+b=x$$`. En fait, plusieurs écritures existent, nous les donnons ci-après :

- pour le mode mathématique simple, sont synonymes de `$a+b=x$` :
`\begin{math}`
`a+b`
`\end{math}`
et `\(a+b=x\)`,
- pour le mode mathématique centré, est synonyme de `\[a+b=x\]` ou `$$a+b=x$$` :
`\begin{displaymath}`
`a+b`
`\end{displaymath}`

Il faut veiller consciencieusement à ce que les symboles « \$ » soient appariés¹ car ce qui est accepté en mode texte ne l'est généralement pas en mode mathématique et *vice versa*. Les erreurs de balancement des « \$ » étant assez fréquentes, nous recommandons, pour simplifier leur mise en évidence, de réserver l'usage des « \$ » aux formules mathématiques simples, et d'utiliser `\[...\]` pour les formules mathématiques centrées. C'est une recommandation tout à fait personnelle, l'usage de `$$...$$` étant, il est vrai, assez répandu.

Lorsque l'on définit une commande, il est important de savoir si elle est pensée pour le mode texte, le mode mathématique, ou les deux. Considérons la commande `\mathafonction` suivante :

```
\newcommand{\mathafonction}{f : A \rightarrow B}
```

Vérifiez qu'elle ne peut s'employer qu'à l'intérieur d'une expression mathématique, c'est-à-dire encadrée par des « \$ ». À l'inverse, la variante `\textafonction` suivante :

```
\newcommand{\textafonction}{f : A \rightarrow B$}
```

ne fonctionnera qu'en mode texte. Si l'on souhaite définir une commande utilisable dans les deux modes, employer `\ensuremath`, qui permet le passage en mode mathématique si nous n'y sommes pas déjà :

```
\newcommand{\bothafonction}{\ensuremath{f : A \rightarrow B}}
```

Vérifiez que `\bothafonction` et `\bothafonction` sont tous deux acceptés.

¹Si l'on souhaite vraiment taper le signe « \$ », utiliser la commande `\$`.

1.2 Espacement — Fontes mathématiques

Les espaces tapées par l'utilisateur ne sont pas significatives en mode mathématique, qui gère lui-même l'espacement entre les divers symboles et signes. Reprenant la formule $\$a+b=x\$, vérifiez que le même résultat est produit par $\$ a + b = x \$$. L'important pour nous est que nous pouvons taper les sources des expressions mathématiques d'une façon qui en facilite la relecture, dans la mesure où nous pouvons insérer des caractères d'espacement et des fins de ligne à notre guise.$

Le comportement par défaut doit parfois être corrigé. Essayez de spécifier la loi d'Ohm par $\$U = R I\$, ce qui rapproche beaucoup trop les symboles « R » et « I ». Mais l'insertion d'une espace fine corrige ce défaut : $\[U = R\, I\]$, ce qui produit :$

$$U = RI$$

Voici les commandes d'espacement du mode mathématique :

$\,$	espace fine [*]	\quad	cadratin ^{* 2}
\thinspace	équivalent à « $\,$ » [*]	$\quad\quad$	double cadratin [*]
$\:$	espace moyenne	$\!$	léger resserrement vers la gauche
$\;$	espace large		

l'étoile « \star » signalant que la commande est également utilisable en mode texte. Ajoutons-y la commande déjà vue pour une espace inter-mots, « $_$ », d'un usage toutefois assez rare en mode mathématique.

Un exemple d'emploi du resserrement à gauche est donné par la supersposition de symboles. La commande \not du mode mathématique barre le symbole qui la suit. Ce qui donne un résultat satisfaisant dans un cas comme $\not\in$ (\notin), mais doit être corrigé dans le cas d'un symbole plus large comme \rightarrow : essayez d'abord $\not\rightarrow$ puis, pour voir la différence, $\!\rightarrow$ et finalement $\!\!\rightarrow$.

Exercice 1 Utilisez les commandes d'espacement et de resserrement pour donner un look acceptable à la formule $\int\int f(x,y) dx dy$, soit :

$$\iint f(x,y) dx dy$$

Vous avez pu remarquer dans les « dx » et « dy » de l'exercice précédent que le caractère « d » apparaît en style « romain », alors que le look de x et de y s'apparente à celui des caractères italiques. En fait, les fontes utilisées par le mode mathématique ne sont pas les mêmes qu'en mode texte ; en particulier, aucun *crénage* n'est effectué. Mettez ce point en évidence en comparant les effets respectifs de $\emph{efficace}$ et $\$efficace\$. Lorsque l'on utilise des identificateurs à plusieurs lettres, il est meilleur d'utiliser la commande \mathit — essayez $\mathit{efficace}\$ — plutôt que de jouer à outrance sur les commandes de resserrement. Plus généralement, il existe en mode mathématique des commandes analogues à la plupart des commandes de changement de look pour le mode texte :$$

Commande	Exemple	Effet et observations
\mathrm	efficace	Caractères en style « romain ».
\mathit	efficace	Caractères italiques.
\mathsf	efficace	Fonte sans empattements.
\mathhtt	efficace	Style « machine à écrire ».
\mathbf	efficace	Caractères gras ³ .
\mathcal	\mathcal{LMD}	Lettres capitales de calligraphie. Propre au mode mathématique.
\mathnormal	efficace	Fonte de base du mode mathématique.

Exercice 2 (Suite de l'exercice 1) Adapter la formule de cet exercice pour que les « d » apparaissent en style « romain ».

²Espacement dont la largeur — la *chasse*, comme disent les typographes — est celle de la lettre la plus large — « M » — de la fonte courante.

³Dans quelques ouvrages mathématiques, les caractères gras sont parfois employés pour dénoter les vecteurs : « \mathbf{v} » pour un vecteur noté « \vec{v} » dans vos cours.

À l'intérieur d'une formule mathématique, l'emploi des commandes de la forme « `\text{...}` » est tout à fait possible, auquel cas L^AT_EX réalise un basculement temporaire vers le mode texte. Par exemple : `\[\ln x = 0 \textbf{ d'où } x = 1 \]` produit :

$$\ln x = 0 \textbf{ d'où } x = 1$$

(mais ne *pas* oublier les espaces *avant* et *après* « d'où » dans le texte source⁴).

Exercice 3 Tentez de trouver par vous-même l'effet produit par :

```
\emph{Pensez-vous que  $\mathit{x}$ ,  $y = 0$  implique  $x = 0$ }
\textnormal{ ou }  $y = 0$  ?}
```

et vérifiez votre réponse. Quelle est l'influence de la commande `\mathit` ? Est-elle utile, ou non ?

Des *packages* donnent accès à des jeux supplémentaires de fontes :

- `amssymb`⁵ permet l'obtention de lettres :
 - en *black board* par la commande « `\mathbb{...}` » appliquée à des lettres capitales; tentez d'obtenir les notations pour les ensembles de nombres : $\mathbb{N}, \mathbb{Z}, \mathbb{D}, \mathbb{Q}, \mathbb{R}, \mathbb{C}, \mathbb{H}$,
 - en style gothique par la commande « `\mathfrak{...}` » ;
- `euscript` définit la commande `\EuScript`, applicable elle aussi à des lettres capitales, et permettant la production de lettres calligraphiques d'un *look* légèrement différent de celles obtenues avec la commande `\mathcal` : comparez `\EuScript{F}` (« \mathcal{F} ») et `\mathcal{F}` (« \mathcal{F} »)
- `mathrsfs` permet l'utilisation de belles lettres calligraphiques inclinées, au moyen de la commande `\mathscr`, elle aussi appliquée à des lettres capitales : $\mathscr{A}, \mathscr{B}, \mathscr{C}, \dots$

Exercice 4 Obtenir l'expression définissant un travail⁶ en fonction d'une puissance, soit $\mathfrak{W} = \mathcal{P}t$.

1.3 Lettres grecques

En mode mathématique, les lettres grecques sont accessibles par une commande coïncidant avec leur nom⁷, avec une capitale initiale pour une lettre grecque capitale, sauf lorsque leur dessin coïncide avec une lettre de l'alphabet latin, ce qui est plus fréquent dans les lettres capitales. Voici l'alphabet grec⁸ :

À taper	Résultat	À taper	Résultat
<code>\$A, \alpha\$</code>	A, α	<code>\$N, \nu\$</code>	N, ν
<code>\$B, \beta\$</code>	B, β	<code>\Xi, \xi\$</code>	Ξ, ξ
<code>\Gamma, \gamma\$</code>	Γ, γ	<code>\$O, o\$</code>	O, o
<code>\Delta, \delta\$</code>	Δ, δ	<code>\Pi, \pi\$</code>	Π, π
<code>\$E, \epsilon\$</code>	E, ϵ	<code>\$R, \rho\$</code>	R, ρ
<code>\$Z, \zeta\$</code>	Z, ζ	<code>\Sigma, \sigma\$</code>	Σ, σ
<code>\$H, \eta\$</code>	H, η	<code>\$T, \tau\$</code>	T, τ
<code>\Theta, \theta\$</code>	Θ, θ	<code>\$Y, \upsilon\$</code>	Y, υ
<code>\$I, \iota\$</code>	I, ι	<code>\Phi, \phi\$</code>	Φ, ϕ
<code>\$K, \kappa\$</code>	K, κ	<code>\$X, \chi\$</code>	X, χ
<code>\Lambda, \lambda\$</code>	Λ, λ	<code>\Psi, \psi\$</code>	Ψ, ψ
<code>\$M, \mu\$</code>	M, μ	<code>\Omega, \omega\$</code>	Ω, ω

⁴Ne pas oublier non plus de charger le *package* `inputenc` avec l'option `latin1`, pour que le caractère accentué « \hat{u} » soit traité correctement.

⁵Ce *package* donne les signes utilisés par l'*American Mathematical Society*, d'où son nom.

⁶« *Work* » en anglais, d'où le symbole employé.

⁷Plus exactement, leur nom inspiré de la façon « classique » d'épeler les lettres du grec ancien. Ce qui ne correspond pas toujours à l'usage du grec moderne, loin s'en faut. Par exemple, la lettre « β » se prononce [v] en grec moderne et s'épelle « *vita* », tandis que le son [b] se note par « $\mu\pi$ » (par exemple, « $\mu\pi\alpha\rho$ » signifie « bar » et se prononce comme en français).

⁸Rappelons qu'il s'agit ici de l'utilisation de lettres grecques dans des formules mathématiques et non d'écriture dans la langue de *Μίκης Θεοδωράκης* (= Mikis Theodorakis). Dans ce dernier cas, utiliser l'option `greek` du *package* `babel`, dont la description sort du cadre de ces notes de travaux pratiques.

Nous pouvons remarquer que les commandes pour les lettres capitales grecques « spécifiques » les produisent en style « droit » : l'usage des commandes `\mathbf`, `\mathit`,... du tableau du § 1.2 — excepté `\mathcal` — permet de les obtenir dans tous les styles. Mentionnons aussi les commandes suivantes, permettant l'obtention de variantes inspirées pour la plupart de l'écriture cursive du grec :

`\varepsilon` `\vartheta` `\varpi` `\varrho` `\varsigma`⁹ `\varphi`

Il est maintenant temps de préciser l'étymologie de \LaTeX : ce programme est dérivé d'un programme appelé \TeX pour « *technology* » ($\tau\epsilon\chi$), les lettres « L » et « A » étant les premières du nom de famille de Leslie Lamport, créateur de \LaTeX .

1.4 Symboles

Au gré des exemples de cette introduction, nous avons déjà rencontré des commandes produisant des symboles mathématiques : `\in` pour l'appartenance à un ensemble, `\int` pour un signe d'intégration, `\rightarrow` (resp. `\longrightarrow`) pour une courte (resp. longue) flèche de gauche à droite. À titre d'exemples, nous développons ci-après la plupart des commandes pour les flèches horizontales ou en oblique :

<code>\leftarrow</code> ¹⁰	<code>\leftrightarrow</code>	<code>\longrightarrow</code>
<code>\Lleftarrow</code>	<code>\Lleftrightarrow</code>	<code>\Llongrightarrow</code>
<code>\rightarrow</code> ¹¹	<code>\longleftarrow</code>	<code>\longleftrightarrow</code>
<code>\Rrightarrow</code>	<code>\Llongleftarrow</code>	<code>\Llongleftrightarrow</code>
<code>\mapsto</code>	<code>\longmapsto</code>	<code>\nearrow</code>
<code>\nrightarrow</code>	<code>\searrow</code>	<code>\swarrow</code>

$\text{\TeX}nicCenter$ vous fournit une barre d'outils permettant de retrouver la plupart des commandes correspondantes. Il se peut que certains symboles nécessitent le chargement de *packages* spécifiques, auquel cas nous vous le signalerons dès que nous les rencontrerons.

Exercice 5 *En vous aidant de la barre d'outils de $\text{\TeX}nicCenter$, produire les texte et formules suivantes.*

La fonction Γ vérifie :

- $\forall z \in \mathbb{C}$ tel que $\Re(z) > 0$, $\Gamma(z + 1) = z\Gamma(z)$,
- $\forall n \in \mathbb{N}$, $\Gamma(n) = (n - 1)!$

Exercice 6 *Quelques variations sur les signes d'inclusion et d'égalité — nécessitant le chargement du package `amssymb`, sauf pour les deux premiers signes — :*

$A \subset B$ — $A \subseteq B$ — $A \not\subset B$ — $A \not\subseteq B$ — $A \subsetneq B$ — $A \subsetneqq B$

(le tiret long « — » étant produit par « --- » en mode texte).

Il peut être intéressant de *redéfinir* certaines commandes. Par exemple, essayez `x \le 0` et `x \ge 0` : vous obtiendrez des signes anglo-saxons « \leq » et « \geq ». Le *package* `amssymb` permet la production des signes « \leqq » et « \geqq » *via* les commandes `\leqslant` et `\geqslant`. Forcer les commandes `\le` et `\ge` à produire ces derniers symboles s'effectue comme suit :

```
\renewcommand{\le}{\leqslant}
\renewcommand{\ge}{\geqslant}
```

qu'il est préférable de placer dans le préambule, avant « `\begin{document}` », afin de grouper toutes les définitions et redéfinitions globales ; il est cependant possible d'effectuer ces opérations en cours de traitement d'un document. L'avantage de ce *modus operandi* : vous pouvez taper une formule telle que :

⁹En grec, cette forme « ς » s'emploie en fin de mot, alors que la forme « σ » s'emploie en début ou milieu de mot.

¹⁰Se note aussi `\gets`. Les notations « `\dots\arrow` » se rapportent plus à la *présentation* de la flèche qu'à sa *signification*, mais sont plus systématiques, donc plus faciles à retenir. Cette dualité entre représentation et signification pour la notation des symboles se retrouve dans le langage \MathML pour la notation de formules mathématiques sur le *Web*.

¹¹Se note aussi `\to`. Voir note précédente.

`\forall x \in \mathbb{R} \text{ tel que } x > 0, \; \ln x \le x - 1`

et l'utiliser aussi bien dans un document en français que dans un document en anglais, le *look* de `\le` s'ajustant au moyen de la redéfinition si besoin est. Bien comprendre toutefois que de telles redéfinitions peuvent rendre des symboles inaccessibles¹².

1.5 Indices et exposants

Les indices (resp. exposants) sont obtenus par le signe « `_` » (resp. « `^` ») suivis d'un argument : `u_{i}`, `\mathrm{e}^{x + 1}`. Les indices à l'intérieur d'indices — de même que les exposants à l'intérieur d'exposants — ne posent aucun problème ; essayez :

`\u_{i_{0}}` --- `\mathrm{e}^{-x^{2}}` --- `2^{2^{2}}`

Stricto sensu, les accolades ne sont pas toujours indispensables pour les indices et les exposants... à condition qu'un *seul* symbole — lettre, chiffre, signe ou commande — soit concerné. Essayez :

`\u_0` --- `f_{\infty}` --- `f_{-\infty}` --- `f^{2006}`

Nous préconisons l'emploi systématique d'accolades, peut-être quelque peu fastidieuses à taper, les notations obtenues sont plus homogènes... et cela permet d'éviter des erreurs.

En cas d'emploi conjoint d'indices et d'exposants, ils sont alignés verticalement : essayez `\u_{1}^{1}`. Si l'on souhaite « décaler » la position de l'exposant, utiliser une paire supplémentaire d'accolades formant bloc : « `\u_{i + 1}^{2}` » donne « u_{i+1}^2 ». Un procédé analogue permet, bien sûr, le « décalage » de la position de l'indice. Dans cas — emploi conjoint d'indices et d'exposants — nous conseillons d'ailleurs d'adopter toujours le même ordre et personnellement, nous tapons toujours les indices d'abord, les exposants ensuite. Cela n'a aucune incidence sur le traitement par \LaTeX , mais permet de trouver plus facilement d'éventuelles erreurs, causés pour la plupart par la présence de multiples indices — `\u_{0}_{1}` — ou exposants — `f^{0}^{1}`.

Exercice 7 *Ce sont les spécifications d'indices que l'on emploie pour les notations de limites (par la commande `\lim`). Trouvez la notation pour la formule mathématique centrée suivante :*

$$\lim_{n \rightarrow -\infty} e^x = 0$$

Les spécifications d'indices et exposants sont également employées pour les opérations « couvrant » un intervalle, c'est-à-dire les sommes (`\sum`), les produits (`\prod`) et les intégrales (`\int`). Donner en mode mathématique centré les trois formules suivantes :

$$u_n = \sum_{i=0}^{n-1} i$$

$$n! = \prod_{i=1}^n i$$

$$\Gamma(z) = \int_0^{+\infty} t^{z-1} e^{-t} dt$$

1.6 Fractions — racines carrées

Les fractions s'indiquent par la commande `\frac`, dont les deux arguments sont le numérateur et le dénominateur. Elle produit de jolies fractions en mode mathématique centré : essayez `\frac{1}{2}`.

Le signe « radical » est donné par la commande `\sqrt`¹³ : « `\sqrt{2}` » produit « $\sqrt{2}$ ». Un argument optionnel permet de spécifier les racines cubiques, n^{es} , etc. : « `\sqrt[3]{10}` » produit « $\sqrt[3]{10}$ »¹⁴.

¹²Dans ce cas-ci, le symbole « \leq » resterait accessible par la commande `\leq`, synonyme de `\le`. En outre, il est toujours possible d'accéder à un symbole dont on connaît le codage, la description de ces fonctionnalités sortant toutefois du cadre de ce manuel introductif.

¹³« *Square Root* » en anglais, d'où le nom de cette commande.

¹⁴Ce qui peut sembler étrange, car cette expression ne dénote pas vraiment une racine carrée. Mais rappelons-nous que nous spécifions plutôt des *présentations*, comme nous l'avons déjà mentionné dans la note 10 en bas de la page 4.

Exercice 8 Donner en mode mathématique centré les formules suivantes :

$$e^z = \sum_{n \geq 0} \frac{z^n}{n!}$$

$$n! = n^n e^{-n} \sqrt{2\pi n} (1 + \mathcal{O}(n^{-1}))$$

2 Ajustements

2.1 Styles

L'une des différences entre les modes mathématique simple et centré est que le premier tente par défaut de ne pas prendre plus de place que la hauteur d'une ligne courante, limitation qu'ignore le second. Essayez `\frac{x}{y}` et `\[\frac{x}{y}\]` pour bien voir cette différence concernant la taille. En réalité, elle ne s'applique qu'au premier niveau pour le mode centré, tapez :

$$\backslash[\frac{x}{y} + \frac{\frac{a}{b}}{\frac{c}{d}}\backslash]$$

pour vous en convaincre. Voici à présent quels sont les styles qu'utilise le mode mathématique de L^AT_EX, désignés par les commandes correspondantes :

`\displaystyle` activé à l'entrée en mode mathématique centré ;

`\textstyle` activé à l'entrée en mode mathématique simple ;

`\scriptstyle` utilisé pour les indices et les exposants, au premier niveau ;

`\scriptscriptstyle` utilisé pour les indices d'indices, indices d'exposants, etc.

En pratique, c'est surtout la première de ces commandes qui est utilisée pour forcer un affichage plus « aéré » en mode simple ou dans des sous-formules du mode centré. Comparez $\sum_{i=0}^{+\infty} \frac{1}{2^i}$ et $\sum_{i=0}^{+\infty} \frac{1}{2^i}$, obtenus respectivement par :

– `\sum_{i = 0}^{+\infty} \frac{1}{2^{i}}`

– `\displaystyle \sum_{i = 0}^{+\infty} \frac{1}{2^{i}}`.

Vérifiez aussi que le passage de cette formule en mode mathématique centré rend inutile l'emploi de la commande `\displaystyle`.

Exercice 9 Utilisez :

– la commande `\textstyle` pour obtenir le look suivant en mode mathématique centré :

$$\sum_{i \geq 0} 2^{-i}$$

– et la commande `\displaystyle` pour obtenir $\lim_{x \rightarrow 1} \frac{\ln x}{x - 1} = 1$ en mode mathématique simple.

Exercice 10 Le package `amsmath` fournit la commande `\cfrac`, d'emploi analogue à la commande `\frac`, mais produisant des fractions en `displaystyle`. L'utiliser pour obtenir la fraction continue suivante :

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

« [l] » pour « left » et « [r] » pour « right ». Quel argument optionnel employer pour réécrire notre fraction continue précédente comme suit ?

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

Les *fantômes* permettent la pose d'espacements correspondant à un fragment de texte absent. Si l'on s'attache à la *largeur* d'un fragment de texte, on parle de fantôme *horizontal*. Un tel fantôme a donc une largeur, mais pas de hauteur. L'exemple suivant — à essayer, il s'agit de deux alinéas consécutifs — vous montre comment réaliser un alignement à l'aide d'un fantôme horizontal :

`\exp \mathrm{i}\pi = \cos \pi + \mathrm{i} \sin \pi`
`\hphantom{\exp \mathrm{i}\pi} = -1`

(Au-delà de cet exemple, nous verrons au § 3.3 des environnements de formatage de plusieurs formules consécutives.)

2.4 Ajustements verticaux

Nous l'avons déjà vu : les points de suspension s'expriment par la commande `\ldots` en mode texte. Cette commande peut aussi s'employer en mode mathématique, mais si son résultat est satisfaisant dans un cas comme « x_1, \dots, x_n », il l'est moins dans « $x_1 + \dots + x_n$ ». Dans ce cas, utiliser des points de suspension centrés verticalement — de telle sorte que les points apparaissent dans le prolongement horizontal des « + » — au moyen de la commande `\cdots` du mode mathématique. Ce mode fournit aussi la commande `\vdots` (resp. `\ddots`) pour des points de suspensions alignés verticalement (resp. en diagonale).

Exercice 13 (Suite de l'exercice 12) Utilisez les commandes adéquates pour les points de suspension en vue d'obtenir les effets suivants :

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}}$$

$$a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \frac{1}{\ddots}}}}$$

Signalons que le *package* `amsmath` fournit une commande générique `\dots` qui, dans la plupart des cas, réussit à s'adapter à la « bonne » présentation.

Nous avons vu au § 2.3 l'utilisation de fantômes horizontaux. De façon analogue, il existe des fantômes *verticaux*, qui ont une hauteur, mais pas de largeur. En pratique, les fantômes verticaux sont utilisés pour corriger une présentation telle que « $\sqrt{a} + \sqrt{b}$ », dans laquelle les barres horizontales des radicaux ne sont pas à la même hauteur. La commande `\mathstrut` — c'est un fantôme vertical dont la hauteur est celle d'une parenthèse ouvrante¹⁷ — permet de corriger ce problème ; comme vous pouvez l'essayer, « `\[\sqrt{a\mathstrut} + \sqrt{b\mathstrut}\]` » donne :

$$\sqrt{a} + \sqrt{b}$$

Exercice 14 La commande `\cfrac` du *package* `amsmath` pose un *strut* : vérifiez-le en écrivant de plusieurs façons la fraction suivante :

$$\frac{b}{\frac{a}{b}}$$

¹⁷Ce fantôme vertical est appelé « *strut* » dans la terminologie anglo-saxonne, d'où le nom de cette commande.

2.5 Ajustements de caractères

Comme nous l'avons vu au § 1.2, l'utilisation de commandes `\textrm`, `\textit`, `\textbf`, ... crée un basculement vers le mode texte. L'effet n'est pas toujours très heureux, car la taille ne s'ajuste pas automatiquement. La commande `\text` du *package* `amsmath` permet d'écrire du texte dans la même taille que les caractères mathématiques qui eussent pu être écrits à cet endroit.

```
\[\underbrace{u_{0} + u_{2} + \cdots}_{\text{termes de rang pair}}\]
```

Cette commande `\text` est devenue, de ce fait, une commande passe-partout pour l'insertion de textes à l'intérieur de formules mathématiques.

2.6 Définitions de noms d'opérations

L^AT_EX connaît deux notations possibles pour les fonctions :

- l'utilisation d'*opérateurs*, alors écrits en position *préfixée*,
- l'utilisation d'*opérations binaires* et de *relations*, écrites en position *infixée*.

L'opération *modulo* nous offre une bonne mise en évidence de cette différence¹⁸ :

$$\begin{array}{ll} \$\gcd(p,q) = \gcd(p,p \bmod q)$ & \gcd(p,q) = \gcd(p,p \bmod q) \\ \$r \equiv 0 \pmod{2\pi}$ & r \equiv 0 \pmod{2\pi} \end{array}$$

En ce qui concerne les opérateurs, ils se créent à l'aide de la commande `\mathop`, l'argument indiquant comment composer le nom de cet opérateur :

```
$\mathop{\mathrm{sh}} z = \frac{\exp z - \exp (-z)}{2}$
```

voyez bien que « *sinh* » a le même *look* que « *exp* ». Nous pouvons définir comme suit une commande `\sh` pour la fonction sinus hyperbolique¹⁹ :

```
\newcommand{\sh}{\mathop{\mathrm{sh}}\nolimits}
```

la mention de la commande `\nolimits` étant préférable, car par défaut, la notation d'un opérateur présente des limites (*cf.* § 2.2). Le *package* `amsmath` permet d'exprimer de telles définitions sous une forme plus contractée :

```
\DeclareMathOperator{\sh}{\mathrm{sh}}
```

Voici la variante à utiliser pour un opérateur présentant des limites²⁰ :

```
\DeclareMathOperator*{\twohollowsigns}{\Diamond\Box}
```

Exercice 15 Introduisez l'opérateur `\erf`, sans limites, et tapez sa définition :

$$\operatorname{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-\infty}^x e^{-t^2} dt$$

puis la formule simple $\forall x \in \mathbb{R}, \operatorname{erf}(x) < 1$.

Exercice 16 Définir les opérateurs mathématiques `\divergence`, `\grad`, `\laplaceopv` et `\rot`, de telle sorte que la formule :

```
\[\overrightarrow{F} = -\grad U \quad \quad \quad \rot(\grad f) = \\ \overrightarrow{0} \quad \quad \quad \laplaceopv \overrightarrow{v} = \\ \grad(\divergence \overrightarrow{v}) - \rot(\rot \overrightarrow{v})\]
```

¹⁸Il est aisé de se souvenir du nom de ces deux commandes : « `\bmod` » est pour une opération binaire, tandis que « `\pmod` » se met entre parenthèses à la fin d'une formule.

¹⁹En réalité, cette opérateur est prédéfini dans L^AT_EX, mais noté par la commande `\sinh`, qui produit une notation à l'anglo-saxonne (« *sinh* »).

²⁰Si vous voulez essayer la définition suivante, chargez le *package* `amssymb` pour que la commande `\Diamond` soit définie.

apparaisse comme suit :

$$\vec{F} = -\overrightarrow{\text{grad}}U \quad \overrightarrow{\text{rot}}(\overrightarrow{\text{grad}}f) = \vec{0} \quad \overrightarrow{\Delta} \vec{v} = \overrightarrow{\text{grad}}(\text{div } \vec{v}) - \overrightarrow{\text{rot}}(\overrightarrow{\text{rot}} \vec{v})$$

Modifiez ensuite ces définitions par l'ajout de struts, pour que toutes les flèches soient alignés horizontalement.

Quelle condition doit satisfaire la définition de la commande `\rot`, pour que nous puissions écrire la sous-expression précédente « $\overrightarrow{\text{rot}}(\overrightarrow{\text{rot}} \vec{v})$ » sous la forme « $\overrightarrow{\text{rot}}^2 \vec{v}$ » ?

Il n'y a pas de possibilité de définitions « abrégées » pour les opérations binaires et les relations, les commandes respectives étant `\mathbin` et `\mathrel`. Voici quelques exemples utilisant la commande `\newcommand` :

```
\newcommand{\pgcd}{\mathbin{\underline{pgcd}}}
\newcommand{\reversehollowsigns}{\mathrel{\Box\Diamond}}
\newcommand{\reversehollowsignsplus}{%
\mathrel{\mathop{\Box\Diamond}\limits_{-\infty}^{+\infty}}}
```

Par défaut, les opérations binaires et les relations ne présentent pas de limites. Nous avons montré comment donner des limites dans le dernier exemple, cette possibilité n'étant que très peu employée en pratique²¹.

Exercice 17 Sur le modèle de la commande `\pgcd` précédente, définir une opération binaire `\ppcm`, telle que « `\$a \ppcm b\$` » produise « $a \overline{ppcm} b$ ».

2.7 Délimiteurs

Nous pouvons remarquer l'utilisation, dans des formules mathématiques, de *délimiteurs*. Cette notion nous fait bien sûr penser aux parenthèses, mais les crochets droits (« [» et «] »), les accolades (« { » et « } »), produits respectivement par « `\{` » et « `\}` », les barres verticales simples (« | ») et doubles (« || », au moyen de « `\|`²² »), les chevrons gauche et droit (« < » et « > »), produits respectivement par « `\langle` » et « `\rangle` ») rentrent également dans cette catégorie.

L'adjonction à ces délimiteurs des commandes `\left` et `\right` poursuit deux buts :

- préciser quels sont les délimiteurs ouvrant et fermant lorsque c'est nécessaire :
 - un cas très simple est l'utilisation d'un symbole qui est à la fois un délimiteur ouvrant et fermant, par exemple pour une expression de valeur absolue — `\left| x + 1 \right|` — ou de la norme d'un vecteur — `\left\| \overrightarrow{F} \right\|` —,
 - d'autre cas consistent en des utilisations à l'encontre des conventions par défaut ; par exemple, tapez les formules suivantes et comparez-les :

$$\left[-\pi, \pi\right] \quad \left[-\pi, \pi\right]$$

le défaut dans la notation de l'intervalle ouvert est corrigé par `\left] -\pi, \pi \right[`, la même technique s'employant pour un intervalle semi-ouvert : `\left] -\pi, \pi \right]` ;

- ajuster la taille des délimiteurs à celle de la formule qu'ils encadrent : par exemple, comparez les effets respectifs de `\left(\frac{1}{2}\right)` et de `\left(\left(\frac{1}{2}\right)\right)`.

Exercice 18 (Suite de l'exercice 8) Vous pouvez à présent écrire la formule de Stirling in extenso, avec des parenthèses bien dimensionnées au texte qu'elles encadrent :

$$n! = \left(\frac{n}{e}\right)^n \sqrt{2\pi n} \left(1 + \frac{1}{12n} + \frac{1}{128n^2} - \frac{139}{51\,840n^3} + \mathcal{O}\left(\frac{1}{n^4}\right)\right)$$

(Utilisez la commande `\nombre`, fournie par l'option `french` du package `babel`, pour obtenir le bon espacement entre les tranches des mille et des unités : `\nombre{51480}`.)

²¹Elle est, de plus, assez lourde. Comme seuls les opérateurs peuvent présenter des limites, cela oblige à utiliser conjointement les commandes `\mathrel` (ou `\mathbin`) et `\mathop`. De plus, Il faut en outre définir une commande à plusieurs arguments si les limites à poser dépendent de l'utilisation du symbole.

²²Commande utilisable uniquement en mode mathématique.

Il est possible de forcer encore l'agrandissement de délimiteurs. Il existe plusieurs solutions qui sont toutes plus ou moins du bricolage, nous vous donnons celle qui, à notre avis, se ré-utilise le mieux d'un texte à l'autre. Considérons la formule suivante :

$$1 + \left((1 + x) \left(1 + \frac{y}{z} \right) \right) \quad (1)$$

et supposons que nous souhaitions agrandir les parenthèses les plus englobantes, alors qu'elles sont d'une taille identique à celles qui délimitent la sous-expression :

$$1 + \frac{y}{z} \quad (2)$$

Ce qui s'explique puisque la hauteur de l'expression globale (1) n'a pas varié par rapport à la hauteur de la sous-expression (2). La solution est l'utilisation d'une *boîte verticale*²³ dont on spécifie les dimensions par rapport à la distance entre deux lignes consécutives dans le document, dimension donnée par la commande `\baselineskip`²⁴. En pratique, un coefficient un peu plus grand que 1 suffit à obtenir un résultat satisfaisant, comme vous pouvez l'essayer vous-mêmes :

```
\[1 + \left( \vbox to 1.1\baselineskip{} (1 + x)
\left( 1 + \frac{y}{z} \right) \right)\]
```

(la dimension de la boîte verticale est donnée après le mot-clé « to » et son contenu est vide, d'où l'emploi d'un argument vide « {} ».)

Les commandes `\left` et `\right` doivent être strictement appariées. Il est possible de leur adjoindre un ou plusieurs délimiteurs *médians*, introduits par la commande `\middle`²⁵. À titre d'exemple, voici une notation parfois utilisée pour le produit scalaire :

$$\langle \vec{u} | \vec{v} \rangle$$

Exercice 19 Vérifiez l'ajustement de la taille des délimiteurs médians en tapant :

```
\[\left( \frac{1}{2} \middle| \frac{3}{4} \middle| \frac{4}{6} \right)\]
```

ce qui donne²⁶ :

$$\left(\frac{1}{2} \middle| \frac{3}{4} \middle| \frac{4}{6} \right)$$

Si un délimiteur — ouvrant ou fermant — est utilisé seul, l'autre doit être spécifié par un point. Par exemple :

```
\[\left\{ ... \right.\]
```

début une formule par une accolade ouvrante, sans accolade fermante correspondante. Des applications de ce *modus operandi* seront données par la suite au § 4.5.

Exercice 20 Utiliser la commande `\middle` pour ajuster la taille de la barre oblique de division dans l'exemple du birapport suivant :

$$\frac{\overline{AC}}{\overline{AD}} \middle/ \frac{\overline{BC}}{\overline{BD}}$$

²³Cette notion de *boîte* est fondamentale pour \LaTeX : pendant qu'il effectue les opérations de formatage, les caractères sont assimilés à de simples boîtes. À ce moment, ce sont les *dimensions* de la boîte qui sont manipulées sans que l'on sache où ira l'encre.

²⁴Pour des raisons très techniques dans le détail desquelles nous n'entrerons pas ici, il vaut mieux ne pas chercher à modifier le contenu de cette commande `\baselineskip`. Mais c'est parfaitement possible de l'utiliser en lecture.

²⁵Pour des raisons techniques que nous détaillerons pas, cette commande `\middle` n'est pas présente dans toutes les implantations de \LaTeX . En particulier, il est recommandé d'utiliser une version récente.

²⁶... sous réserve de la disponibilité de la commande `\middle` (voir la note précédente).

3 Environnements avancés

3.1 Définitions par cas

Le *package* `amsmath` fournit l'environnement `cases` pour les définitions par cas. Voici un exemple d'utilisation à travers la définition de la suite de Syracuse :

```
\[u_{n + 1} = \begin{cases}
\frac{u_n}{2} & \text{si } u_n \text{ est pair,} \\
3 u_n + 1 & \text{sinon.}
\end{cases}\]
```

Le caractère `&` est spécial et utilisé pour des taquets de tabulation²⁷. « `\` » est une forme abrégée de la commande `\newline` de passage à la ligne suivante. Bien que ce ne soit pas une obligation, nous recommandons d'aller systématiquement à la ligne après une telle commande dans le texte source, et d'y aligner verticalement les caractères « `&` », comme nous l'avons fait dans le texte source de l'exemple précédent. Cela permet une meilleure lisibilité du texte source de la formule, avantage non négligeable s'il faut la retoucher ou y chercher une erreur.

3.2 Équations numérotées

L'environnement `equation` est prédéfini en L^AT_EX et produit une formule à la fois centrée et numérotée :

$$J(x) = \frac{1}{2\pi} \int_0^{2\pi} \cos(x \sin \theta) d\theta \quad (3)$$

obtenue par :

```
\begin{equation}
J(x) = \frac{1}{2\pi} \int_0^{2\pi} \cos(x \sin \theta) \, \mathrm{d}\theta
\label{pretty-equation}
\end{equation}
```

Nous vous avons montré comment associer un identificateur à une telle équation au moyen de la commande `\label`, déjà rencontrée en association avec des directives de sectionnement. Dans un cas comme celui-ci, bien poser ladite commande `\label` avant « `\end{equation}` ». De façon tout à fait analogue, la citation de cette équation s'effectue par « `(\ref{pretty-equation})` », qui dans le cas présent produit « (3) », par le biais du fichier auxiliaire.

Exercice 21 *La présentation d'une telle équation et de son numéro est contrôlée par les options suivantes de la commande `documentclass` :*

`leqno` `reqno` `fleqn`

Essayez-les pour découvrir leur influence. Quelle est l'option correspondant au comportement par défaut²⁸ ? Les documents français présentent souvent les numéros d'équations à gauche : à quelle option cela correspond-il ?

3.3 Environnements du *package* `amsmath`

Les fonctionnalités que nous allons détailler à présent appartiennent au *package* `amsmath`. La première d'entre elles est la commande `\tag` qui permet de *forcer* la valeur affichée pour repérer une équation. À titre d'exemple, reprenez l'équation (3) et spécifiez-la comme suit :

²⁷Nous avons déjà vu qu'il faut taper `\&` pour l'obtenir lui-même dans un texte.

²⁸Plus exactement, le comportement par défaut dans les classes standard comme `article` et `book`. Ce n'est pas le même si l'on utilise les classes de documents de l'*American Mathematical Society* — par exemple, `amsart` et `amsbook` —, autrement dit, cette option a son utilité.

```
\begin{equation}
J(x) = \dots \tag{J} \label{pretty-equation-j}
\end{equation}
```

Vérifiez que vous obtenez bel et bien :

$$J(x) = \frac{1}{2\pi} \int_0^{2\pi} \cos(x \sin \theta) d\theta \quad (\text{J})$$

Puis vérifiez également que la commande `\ref{pretty-equation-j}` vous restitue l'identification de cette équation.

Exercice 22 Utiliser judicieusement la commande `\tag` pour ajouter le signe « prime » à l'étiquette `pretty-equation-j` précédente et obtenir :

$$J'(x) = -\frac{1}{2\pi} \int_0^{2\pi} \sin(x \sin \theta) \sin \theta d\theta \quad (\text{J}')$$

Le premier des environnements fournis par le *package* `amsmath` est `equation*`, assez proche en fait de l'environnement `displaymath` du mode mathématique centré²⁹ : l'équation est centrée, mais non numérotée.

Exercice 23 Utiliser l'environnement `equation*` pour :

$$J''(x) = -\frac{1}{2\pi} \int_0^{2\pi} \cos(x \sin \theta) \sin^2 \theta d\theta$$

Les autres environnements permettent la frappe aisée d'équations s'étalant sur plusieurs lignes, ou de *groupes* d'équations. Comme les environnements `equation` et `equation*` précédents, ils vont par deux, le nom suivi du caractère « * » signifiant que la ou les équations de cet environnement ne sont pas numérotées.

Les environnements `multline` et `multline*` permettent de fixer les endroits où intercaler des sauts de ligne au moyen de la commande « `\` ». La première ligne est serrée à gauche, la dernière serrée à droite, tandis que les autres lignes sont centrées.

Exercice 24 Obtenir l'équation suivante sur deux lignes en utilisant l'environnement `multline*` :

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \frac{x^{11}}{11!} + \frac{x^{13}}{13!} - \frac{x^{15}}{15!} + \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} + (-1)^{n+1} \frac{x^{2(n+1)+1}}{(2(n+1)+1)!} + \dots$$

L'environnement `split` s'utilise à l'intérieur d'un environnement pour équations — pour l'instant, vous ne connaissez que les environnements `equation` et `equation*` — et permet un alignement vertical indiqué par le signe « & ». Tout ce qui est spécifié dans un environnement `split` est compris comme *une seule* équation. À titre d'exemple, essayez les lignes suivantes :

```
\begin{equation}
\begin{split}
(a + b)^2 &= a^2 + 2ab + b^2 \\
(a - b)^2 &= a^2 - 2ab + b^2 \\
(a + b) \cdot (a - b) &= a^2 - b^2
\end{split}
\end{equation}
```

²⁹En réalité, le positionnement par rapport au texte qui l'entoure est légèrement différent.

Exercice 25 Tapez le texte suivant avec l'équation sur deux lignes.

Soit un polynôme de la forme $x^3 + px + q$, avec $p, q \in \mathbb{R}$ et $4p^3 + 27q^2 > 0$. Il admet une racine réelle et deux racines complexes données par :

$$x = \sqrt[3]{-\frac{q}{2} + \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}} + \sqrt[3]{-\frac{q}{2} - \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3}} \quad (4)$$

Les environnements `gather` et `gather*` construisent des équations centrées globalement, sans point d'alignement vertical.

Exercice 26 Utilisez l'environnement `gather*` pour produire :

$$\begin{aligned} x &= \sqrt[3]{u+v} + \sqrt[3]{u-v} \\ u &= -\frac{q}{2} \\ v &= \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3} \end{aligned}$$

Intercaler une ou plusieurs lignes de textes à l'intérieur d'un environnement du *package* `amsmath` s'effectue à l'aide de la commande `\intertext`. Comme exemple, reprenez l'environnement de l'exercice 26 précédent sous la forme :

```
x = ... \\
\intertext{avec :}
u = ...
```

- Les environnements `align` et `align*` sont d'un emploi similaires à l'environnement `split` mais :
- on peut utiliser un nombre quelconque de points d'alignement vertical, toujours signalés par le signe « & » ;
 - ils ne nécessitent pas d'être placés à l'intérieur d'un environnement `equation` ou `equation*` ;
 - dans le cas de l'environnement `align`, toutes les équations sont numérotées, sauf celles qui se terminent par la commande `\notag`.

Exercice 27 (Suite de l'exercice 26) Réécrire les équations de l'exercice précédent avec l'environnement `align` en alignant verticalement les signes d'égalité. Utiliser la commande `\notag` de façon à numéroté seulement deux équations, comme dans le résultat qui vous est montré ci-après :

$$x = \sqrt[3]{u+v} + \sqrt[3]{u-v} \quad (5)$$

avec :

$$\begin{aligned} u &= -\frac{q}{2} \\ v &= \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3} \end{aligned} \quad (6)$$

Les formatages que réalisent les environnements `aligned` et `gathered` sont analogues à ceux de `align*` et `gather*`, mais ces environnements s'emploient à l'intérieur d'environnements d'équations.

Exercice 28 (Suite de l'exercice 27) Les équations de cet exercice ont été réécrites en utilisant un environnement `gather`, à l'intérieur duquel se niche un environnement `aligned` et deux emplois de la commande `\intertext`. Quelle est la bonne combinaison qui permet d'obtenir le texte suivant ?

$$x^3 + px + q = 0 \quad (7)$$

tels que $4p^3 + 27q^2 > 0$

$$x = \sqrt[3]{u+v} + \sqrt[3]{u-v} \quad (8)$$

avec :

$$\begin{aligned} u &= -\frac{q}{2} \\ v &= \sqrt{\left(\frac{q}{2}\right)^2 + \left(\frac{p}{3}\right)^3} \end{aligned} \quad (9)$$

Exercice 29 Tapez le texte suivant, à titre d'exercice récapitulatif. Veillez à obtenir les mêmes numéros d'équations, en utilisant les commandes `\label`, `\tag` et `\notag` de façon judicieuse. Pour vous synchroniser avec le premier de nos numéros, insérez au préalable la commande suivante :

`\setcounter{equation}{9}`

Bien qu'aucune formule algébrique ne permette de calculer une primitive de la fonction $x \mapsto e^{-x^2}$ pour $x \in \mathbb{R}$, il est cependant possible de calculer l'intégrale généralisée suivante :

$$\begin{aligned} \int_{-\infty}^{+\infty} e^{-x^2} dx &= \sqrt{\left(\int_{-\infty}^{+\infty} e^{-x^2} dx\right)^2} \\ &= \sqrt{\int_{-\infty}^{+\infty} e^{-x^2} dx \int_{-\infty}^{+\infty} e^{-y^2} dy} \end{aligned} \quad (10)$$

$$= \sqrt{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} e^{-(x^2+y^2)} dx dy} \quad (10')$$

$$= \sqrt{\int_0^{2\pi} \int_0^{+\infty} \rho e^{-\rho^2} d\rho d\theta} \quad (11)$$

$$= \sqrt{\int_0^{2\pi} \left[\frac{e^{-\rho^2}}{2}\right]_0^{+\infty} d\theta} \quad (12)$$

$$= \sqrt{\int_0^{2\pi} \frac{1}{2} d\theta}$$

$$\begin{aligned} &= \sqrt{\left[\frac{\theta}{2}\right]_0^{2\pi}} \\ &= \sqrt{\pi} \end{aligned} \quad (13)$$

en notant que :

- en (10), nous utilisons le fait que les variables d'intégration sont muettes, le théorème des mesures produits nous permet ensuite d'écrire (10');
- en (11), nous considérons les nombres réels x et y comme les parties réelle et imaginaire d'un nombre complexe, puis nous passons en coordonnées polaires; les équivalences étant :

$$x + iy = \rho e^{i\theta} \quad \text{d'où} \quad \begin{cases} x^2 + y^2 = \rho^2 \\ dx dy = \rho d\rho d\theta \end{cases}$$

- en (12), nous nous permettons un léger abus de langage en utilisant le symbole « $+\infty$ » dans la notation « $[\dots]$ » des images d'une primitive;
- le résultat — trouvé pour la première fois par le mathématicien Carl-Friedrich Gauss (1777–1855) — est donné en (13).

3.4 Environnements fournis par L^AT_EX

Si l'on n'utilise pas le *package* `amsmath`, il existe toutefois deux environnements fournis par L^AT_EX pour la frappe d'ensemble d'équations : `eqnarray` et `eqnarray*`. Ils sont moins puissants que les environnements du *package* `amsmath` décrits au § 3.3 — en particulier, ils ne reconnaissent pas la commande `\tag` permettant la redéfinition d'étiquettes —, ils peuvent néanmoins être utilisés dans les cas « simples ». À la différence des environnements du § 3.3 qui n'utilisent qu'un seul taquet de tabulation, d'ordinaire placé avant le signe central de l'équation, ils en emploient deux, de part et d'autre de ce signe. Voici un exemple d'utilisation de l'environnement `eqnarray` :

```
\begin{eqnarray}
\divergence \overrightarrow{v} & = & \frac{\partial v_x}{\partial x} +
\frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} \\
\label{divergence-def} & \\
\Delta f & = & \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} \\
\Delta f & = & \divergence (\operatorname{grad} f) \nonumber
\end{eqnarray}
```

Essayez cet environnement, vérifiez que vous pouvez référencer la première équation³⁰ au moyen de l'identificateur `divergence-def`. Notez également la commande `\nonumber` pour désactiver la numérotation d'une équation. Remarquez également que les équations obtenues apparaissent toutes en style « `\displaystyle` ». L'environnement `eqnarray*` est d'un emploi similaire, mais les équations qu'il produit ne sont pas numérotées.

Exercice 30 Utilisez l'environnement `eqnarray*` pour la frappe des équations suivantes — en ce qui concerne la notation des fonctions argument de sinus hyperbolique, de cosinus hyperbolique et de tangente hyperbolique³¹, nous vous conseillons de définir des commandes `\argsinh`, `\argcosh`, `\argtanh` à l'aide de la commande `\DeclareMathOperator` du *package* `amsmath` — :

$$\begin{aligned} \operatorname{argsh} x &= \ln(x + \sqrt{x^2 + 1}) \\ \operatorname{argch} x &= \ln(x + \sqrt{x^2 - 1}) \\ \operatorname{argth} x &= \frac{1}{2} \ln \left(\frac{1+x}{1-x} \right) \end{aligned}$$

et l'environnement `eqnarray` pour ce second groupe d'équations — sans numérotation de la définition de l'opérateur de gradient³² — :

$$\operatorname{div} \overrightarrow{v} = \overrightarrow{\nabla} \cdot \overrightarrow{v} \tag{14}$$

$$\operatorname{grad} f = \overrightarrow{\nabla}(f)$$

$$\operatorname{rot} \overrightarrow{v} = \overrightarrow{\nabla} \wedge \overrightarrow{v} \tag{15}$$

$$\Delta = \overrightarrow{\nabla}^2 \tag{16}$$

(les signes « ∇ » et « \wedge » étant respectivement obtenus par les commandes `\nabla` et `\wedge`). N'oubliez pas d'aligner horizontalement les flèches des vecteurs en utilisant si besoin est la commande `\mathstrut`.

³⁰Cette équation définit l'opérateur de divergence — utilisé en Physique — d'un champ de vecteurs en un point : c'est le flux local du champ autour de ce point. L'opérateur défini dans l'équation suivante — *laplacien* — apparaît dans l'écriture d'équations aux dérivées partielles jouant un rôle fondamental en Physique. Des exemples en sont la théorie du potentiel, utilisée en Électricité, et l'équation des cordes vibrantes.

³¹En ce qui concerne cette famille de fonctions trigonométriques, seules sont prédéfinies en L^AT_EX les fonctions arc sinus, arc cosinus et arc tangente, ainsi que la fonction argument d'un nombre complexe : ce sont les commandes `\arcsin`, `\arccos`, `\arctan` et `\arg`.

³²En Physique, il indique la direction de la plus grande variation d'un potentiel et l'intensité de cette variation.

4 Matrices et tableaux

4.1 Environnements du *package* amsmath pour les matrices

Voici un exemple très simple de matrice :

```

\[\begin{matrix}
\cos \varphi & - \sin \varphi \\
\sin \varphi & \cos \varphi
\end{matrix}\]

```

qui produit :

$$\begin{pmatrix} \cos \varphi & - \sin \varphi \\ \sin \varphi & \cos \varphi \end{pmatrix}$$

c'est-à-dire une matrice sans délimiteurs. Des matrices délimitées par :

des parenthèses s'obtiennent par l'environnement `pmatrix`,
des crochets droits `bmatrix`,
des barres verticales `vmatrix`,
des doubles barres verticales `Vmatrix`.

Exercice 31 Quel environnement utiliser pour donner la spécification du vecteur nabla ($\vec{\nabla}$), soit :

$$\begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix}$$

Exercice 32 Il existe deux façons de définir l'opérateur rotationnel³³ ($\vec{\text{rot}}$) : par un déterminant ou un produit vectoriel (cf. exercice 30). Tapez d'abord la première formule :

$$\vec{\text{rot}} \vec{v} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ v_x & v_y & v_z \end{vmatrix}$$

puis la seconde :

$$\vec{\text{rot}} \vec{v} = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \wedge \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix}$$

Exercice 33 Écrire la matrice unité de dimension quelconque, soit :

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

Nous rappelons que les commandes de production de points de suspension sont `\ldots` pour des points de suspension posés sur une ligne horizontale, ainsi que `\vdots` (resp. `\ddots`) pour des points de suspension alignés verticalement (resp. en diagonale).

³³C'est un opérateur qui transforme un champ de vecteurs en un autre champ de vecteurs. En Physique, il est utilisé pour exprimer la tendance qu'a un champ de vecteurs à tourner autour d'un point.

Exercice 34 Nous pouvons utiliser les commandes de changements de taille pour les contenus identiques des matrices creuses. Attention, toutefois : ces commandes ne sont pas directement accessibles en mode mathématique. Solution : utiliser une commande telle que `\texttrm`, ou la commande `\text` du package `amsmath`.

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

(Dans cette matrice, nous avons utilisé la commande `\LARGE` de changement de taille pour les zéros.)

Exercice 35 La frappe de l'équation suivante — le calcul récursif de la valeur d'un déterminant — nécessite un passage en mode texte pour changer de taille, puis à nouveau un passage en mode mathématique à l'intérieur de la commande de changement de taille :

$$\begin{vmatrix} a_{1,1} & \dots & \\ \vdots & (a_{i,j})_{n \setminus 1, n \setminus 1} & \\ a_{n,1} & & \end{vmatrix} = \sum_{k=1}^n (-1)^{k+1} a_{k,1} |a_{i,j}|_{n \setminus k, n \setminus 1}$$

où « $(a_{i,j})_{n,n}$ » (resp. « $|a_{i,j}|_{n,n}$ ») désigne une matrice (resp. un déterminant) de n lignes et n colonnes dont le terme général est $a_{i,j}$, et « $(a_{i,j})_{n \setminus k_1, n \setminus k_2}$ » désigne la sous-matrice extraite de la matrice précédente en supprimant les k_1^e ligne et k_2^e colonne.

Exercice 36 Vérifiez qu'il est possible d'emboîter les environnements de matrices en construisant la formule suivante :

$$\begin{pmatrix} \begin{matrix} \cos \varphi & -\sin \varphi & \sin \varphi \\ \sin \varphi & \cos \varphi & -\sin \varphi \\ -\sin \varphi & \sin \varphi & \cos \varphi \end{matrix} & A_{1,2} & A_{1,3} \\ & \begin{matrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{matrix} & A_{2,3} \\ & A_{3,2} & \begin{matrix} \sin \psi & \cos \psi & -\sin \psi \\ -\sin \psi & \sin \psi & \cos \psi \\ \cos \psi & -\sin \psi & \sin \psi \end{matrix} \end{pmatrix}$$

4.2 Définitions de tableaux

Les tableaux, obtenus au moyen de l'environnement `array` du mode mathématique, sont une généralisation des matrices dans laquelle le formatage est entièrement à la charge de l'utilisateur. Considérons l'exemple suivant :

```
\[\begin{array}{cccccccccccccc}
0 & 1 & 10 & 11 & 100 & 101 & 110 & 111 & 1000 & 1001 & 1010 & 1011 & 1100 &
1101 & 1110 \\\
0 & 1 & 2 & 10 & 11 & 12 & 20 & 21 & 22 & 100 & 101 & 102 & 110 & 111 & 112 \\\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14
\end{array}\]
```

Il s'agit d'un tableau de quinze colonnes, toutes centrées, ce qui est indiqué dans l'argument de cet environnement, après « `\begin{array}` ». Vous pouvez remplacer tout ou partie des « `c` » par des « `l` » ou des « `r` » pour obtenir des colonnes dont le contenu est serré à gauche ou à droite³⁴. Il est également

³⁴Ce sont les mêmes conventions que pour l'argument optionnel de la commande `\cfrac` (cf. exercice 12), avec autant de directives que de colonnes.

possible d'utiliser des *facteurs de répétition* : par exemple, « `\begin{array}{*{15}{c}}` » au lieu de juxtaposer quinze fois le caractère « c ». En ce qui concerne le contenu du tableau, nous reconnaissons les conventions déjà utilisées précédemment : « & » pour la place des taquets de tabulation, « \\ » pour un passage à la ligne suivante.

Exercice 37 Écrivez le tableau suivant³⁵ — utilisez un facteur de répétition pour les cinq premières colonnes, la dernière étant centrée — :

0	1	2	3	4	...
\emptyset	$\{\emptyset\}$	$\{\emptyset, \{\emptyset\}\}$	$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$	$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}\}$...

La construction de « vrais » tableaux, composés de cases délimitées par des lignes horizontales et verticales, est possible. Les lignes verticales sont spécifiées par le caractère « | » dans le format, et les lignes horizontales par la commande `\hline`, à insérer avant ou après une ligne du tableau. En reprenant notre premier tableau, cela revient à le modifier comme suit :

```

\[\begin{array}{*{15}{|c|}|}
\hline
0 & 1 & \dots \\
\hline
0 & 1 & \dots \\
\hline
0 & 1 & \dots \\
\hline
\end{array}\]

```

une double ligne horizontale (resp. verticale) étant produite par la répétition de la commande `\hline` (resp. du signe « | » dans le format).

Exercice 38 (Suite de l'exercice 37) Reprendre le tableau de cet exercice de façon à obtenir :

0	1	2	3	4	...
\emptyset	$\{\emptyset\}$	$\{\emptyset, \{\emptyset\}\}$	$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$	$\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}\}$...

On peut utiliser à tout moment la commande `\\` de passage à la ligne suivante, mais elle inhibe le tracé des lignes verticales appartenant aux colonnes situées à droite de la ligne courante. Vous pouvez vous en convaincre en essayant :

```

\[\begin{array}{*{4}{|c|}|}
\hline
0 & 1 & 2 & 3 \\
\hline
0 & 1 & 2 \\
\hline
0 & 1 \\
\hline
0 \\
\hline
\end{array}\]

```

Pour limiter le tracé d'une ligne horizontale aux colonnes comprises entre la i^e et la j^e , utiliser la commande `\cline{i-j}`.

Exercice 39 Retouchez le tableau précédent en utilisant la commande `\cline` pour obtenir le tableau suivant :

0	1	2	3
0	1	2	
0	1		
0			

³⁵Il s'agit de la définition des nombres entiers naturels donnée par Georg Cantor et utilisée en Théorie des Ensembles : l'entier naturel zéro est l'ensemble vide, l'entier naturel non nul n est l'ensemble $\{0, \dots, n - 1\}$.

Exercice 42 Sur le modèle du tableau précédent, construire le tableau ci-après (il peut être intéressant d'utiliser un facteur de répétition) :

$$\begin{array}{l} \cos x = \frac{e^{ix} + e^{-ix}}{2} \\ \sin x = \frac{e^{ix} - e^{-ix}}{2i} \\ \tan x = \frac{\sin x}{\cos x} \end{array} \quad \begin{array}{l} \cos\left(x + \frac{\pi}{2}\right) = -\sin x \\ \sin\left(x + \frac{\pi}{2}\right) = \cos x \\ \tan\left(x + \frac{\pi}{2}\right) = -\frac{1}{\tan x} \end{array} \quad \begin{array}{l} \cos(-x) = \cos x \\ \sin(-x) = -\sin x \\ \cos^2 x + \sin^2 x = 1 \end{array}$$

L'environnement `array` admet un argument optionnel indiquant le placement vertical de ce tableau. Admettons que les tableaux suivants représentent des matrices³⁷ et considérons le produit de matrices suivant :

$$\begin{array}{|c|c|c|} \hline 1 & -3 & 4 \\ \hline -2 & 2 & -5 \\ \hline \end{array} \begin{array}{|c|} \hline 1 \\ \hline 1 \\ \hline 0 \\ \hline \end{array} \tag{17}$$

Rappelons que les arguments optionnels se placent, entre crochets droits, avant les arguments obligatoires. En l'occurrence, cet argument optionnel de l'environnement `array` se place avant les directives de formatage du tableau — l'argument obligatoire de cet environnement — comme nous pouvons le voir dans la manière dont cette formule a été obtenue :

```
\begin{equation}
\begin{array}[t]{*{4}{|r|}} ... \end{array}\; \begin{array}[t]{|c|} ...
\end{array} \label{matrix-product}
\end{equation}
```

Si l'argument optionnel utilisé est « `[t]` » (resp. « `[b]` ») — pour « *top* » (resp. « *bottom* ») — c'est le haut (resp. le bas) du tableau qui est pris en compte pour l'alignement.

4.3 Mini-tableaux

Le package `amsmath` fournit une commande `\substack` utilisable lorsque les indices ou les exposants nécessitent une écriture sur plusieurs lignes. À titre d'exemple, nous avons utilisé :

```
..._{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}}
```

pour obtenir :

$$(a_{i,j})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}} \quad \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}} a_{i,i}$$

Comme le montre le deuxième exemple, le placement de ces indices ou exposants tient compte du fait que l'opérateur qui les porte présente des limites ou non.

Exercice 43 Obtenir :

$$\lim_{\substack{x \rightarrow x_0 \\ x \neq x_0}} \frac{f(x) - f(x_0)}{x - x_0} \quad \sum_{\substack{i=1 \\ j=1}}^{i=n \\ j=n} a_{i,j}^2$$

³⁷... ce qui n'est pas hérétique : nous avons déjà personnellement vu ce genre de notation dans des ouvrages de Programmation linéaire. Pas d'inquiétude, toutefois : nous reviendrons à un *look* plus proche de ce que vous connaissez à l'exercice 48.

Comme le montrent les exemples précédents, les lignes d'un mini-tableau obtenu par la commande `\substack` sont centrées. Il existe un environnement d'utilisation plus générale, `subarray`, également fourni par le *package* `amsmath`, dont l'argument — « `{c}` », « `{l}` » ou « `{r}` » — indique comment formater verticalement les diverses lignes. Bien noter, toutefois, que tous ces mini-tableaux ne comportent qu'une seule colonne.

Exercice 44 Utiliser l'environnement `subarray` pour obtenir :

$$\sum_{\substack{0 \leq i \leq n \\ 0 \leq j \leq n \\ 0 \leq k \leq \binom{j}{i}^2}} i + j - k \qquad \sum_{\substack{i=n \\ j=n \\ k=\binom{j}{i}^2 \\ i=0 \\ j=0 \\ k=0}} i + j - k$$

4.4 Ajustement de cases

Quittons très provisoirement les tableaux et intéressons-nous aux formules encadrées, comme celle-ci :

$$\boxed{\lim_{x \rightarrow 1} \frac{\ln x}{x - 1} = 1}$$

que nous avons obtenue par `\[\boxed{\lim_{x \rightarrow 1} \frac{\ln x}{x - 1} = 1}\]`, la commande `\boxed` appartenant au *package* `amsmath`. En réalité, cette commande `\boxed` y est définie comme suit³⁸ :

```
\newcommand{\boxed}[1]{\text{\fbox{$\displaystyle#1$}}}
```

la commande `\fbox` encadrant son argument — sans le formater sur plusieurs lignes — en mode texte. Cette commande utilise deux longueurs :

`\fboxrule` l'épaisseur de la ligne tracée autour de la boîte ;

`\fboxsep` la distance entre un coin de la boîte et son contenu ;

ces dimensions pouvant être modifiées par la commande `\setlength` à tout moment³⁹. Vous pouvez essayer quelques variations sur la séquence de commandes suivante :

```
\setlength{\fboxrule}{2mm}
\setlength{\fboxsep}{3mm}
\fbox{Vive le mode mathématique de \LaTeX !}
```

Voici à présent le lien entre la commande `\boxed` et les tableaux du mode mathématique de \LaTeX . S'il arrive que le contenu d'une case télescope l'encadrement, la solution la plus simple, à notre avis, est d'utiliser un encadrement dont la ligne a une épaisseur nulle. Ne pas oublier que le second argument de la commande `\setlength` est une dimension et qu'il doit posséder une unité, même lorsqu'il s'agit d'une dimension nulle :

```
\setlength{\fboxrule}{0mm}
```

Quant à la dimension donnée par `\fboxsep`, sa valeur par défaut est de 3 points⁴⁰. En pratique, il est rarement nécessaire de s'en éloigner beaucoup.

³⁸À une légère nuance près, comme pourrait le constater tout lecteur qui consulterait le texte source du *package* `amsmath`. Mais l'exposé de cette nuance sort du cadre de ce modeste support de travaux pratiques.

³⁹Ce qui n'est pas en contradiction avec le § 1.7 du document précédent *Premier contact avec \LaTeX* : ce sont les longueurs que nous y ajustions qui ne peuvent pas être modifiées après « `\begin{document}` », d'où la contrainte de placer toutes les commandes `\setlength` correspondantes dans le préambule.

⁴⁰C'est-à-dire approximativement 1,054 mm.

Exercice 45 (Suite de l'exercice 41) Utiliser de tels encadrements à épaisseur nulle pour « aérer » les cases contenant des signes de racines carrées dans la présentation du tableau de l'exercice 41. Comparez la réalisation précédente avec celle que voici :

$\Delta = b^2 - 4ac < 0$	$\Delta = 0$	$\Delta > 0$	
Pas de racine	$x = \frac{b}{2a}$	$x = \frac{-b - \sqrt{\Delta}}{2a}$	$x = \frac{-b + \sqrt{\Delta}}{2a}$
Signe de a	Signe de a sauf à la racine	Signe de a pour les valeurs extérieures aux racines	

4.5 Packages spécialisés pour les tableaux

Le package `multirow` est utilisé lorsque du texte doit prendre place sur plusieurs lignes d'une même colonne. Considérons l'exemple suivant :

$$\left(\begin{array}{l} \text{Presque la matrice identité,} \\ \text{mais pas tout à fait.} \end{array} \quad \begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{array} \right)$$

obtenu par :

```
\left( \begin{array}{*{4}{c}}
\multirow{3}{50mm}{Presque ...} & 1 & 0 & 0 \\
& 0 & 1 & 0 \\
& 0 & 0 & -1
\end{array} \right)
```

La commande `\multirow`, fournie par ce package, admet trois arguments : le premier donne le nombre de lignes à utiliser pour formater le texte, le deuxième la largeur à utiliser pour « caser » le texte, le troisième le texte lui-même. Remarquons qu'il ne s'agit pas d'un formatage en alinéa : les diverses lignes de ce texte sont toutes serrées à gauche.

Le package `array` accroît les possibilités de spécifications de tableaux. La fonctionnalité la plus utilisée est l'insertion de commandes avant ou après le contenu de chaque colonne.

	$\Gamma(z)$
$\Re(z) > 0$	$\int_1^{+\infty} t^{z-1} e^{-t} dt$
$z \in \mathbb{C} \setminus \mathbb{Z}^-$	$\sum_{n \leq 0} \frac{(-1)^n}{n!} \frac{1}{z+n} + \int_1^{+\infty} t^{z-1} e^{-t} dt$

Spécifier ce tableau par :

```
\begin{array}{|l|c|}
\hline
& \Gamma(z) \\
\hline
\Re(z) > 0 & \int_1^{+\infty} \dots \mathrm{d}t \\
z \in \dots & \sum_{n \leq 0} \dots \\
\hline
\end{array}
```

provoquerait l'affichage des opérateurs de somme et d'intégration en mode `\textstyle` (cf. § 2.1). Mais plutôt que de répéter la commande `\displaystyle` pour chaque ligne de la seconde colonne, nous pouvons prévoir son insertion dans le format : `{|l|>{\displaystyle}c|}`, « >{...} » désignant un fragment à insérer avant le contenu de la colonne correspondante. Symétriquement, « <{...} », à placer après une directive concernant une colonne, désigne un fragment à insérer après son contenu. Une utilisation fréquente de ces deux possibilités est donnée par le tableau suivant, repris de l'exemple qui introduisait les

tableaux au § 4.2, le format est « `{|>{\$}r<{\$}*{15}{|c|}` », les divers contenus de la première colonne étant dès lors donnés en mode texte :

Base 2	0	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110
Base 3	0	1	2	10	11	12	20	21	22	100	101	102	110	111	112
Base 10	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Un moyen mnémotechnique pour ne pas confondre les signes `<` et `>` dans cette utilisation : si l'on interprète $x > y$ en disant que x vient *après* y , alors « `>{...}` » désigne quelque chose à insérer *après* la séparation entre colonnes, et le contenu indiqué par « `<{...}` » s'insère *avant* la séparation suivante.

Exercice 46 Construire la matrice suivante en n'utilisant que deux occurrences au plus de la commande `\varphi`, permettant d'obtenir la lettre grecque φ .

$$\begin{pmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{pmatrix}$$

(Utiliser les commandes `\left` et `\right` pour les délimiteurs.)

Le package `delarray` incorpore les fonctionnalités du package `array` et permet en sus d'incorporer d'éventuels délimiteurs aux instructions de formatage des colonnes. On pourra ainsi écrire :

```
\begin{array}({cc})...\end{array}
```

au lieu d'utiliser explicitement les commandes `\left` et `\right`. Nous pouvons observer avec ce package la même limitation qu'avec ces commandes `\left` et `\right` : l'impossibilité d'utiliser un délimiteur ouvrant sans un délimiteur fermant et *vice versa*. Mais là encore, il est possible de spécifier un délimiteur vide par un point.

Exercice 47 Vérifier que le package `delarray` offre les fonctionnalités du package `array` en supprimant la mention de ce dernier et en utilisant le package `delarray` pour formater à nouveau la matrice de l'exercice 46 avec la même contrainte de style : deux occurrences au plus de la commande `\varphi`.

Exercice 48 Utiliser le package `delarray` pour reformuler et compléter le produit de matrices déjà vu dans l'équation (17) :

$$\begin{pmatrix} 1 & -3 & 4 \\ -2 & 2 & -5 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}$$

5 Retour au mode texte

5.1 Environnement tabular

Les tableaux du mode texte s'obtiennent par l'environnement `tabular`, dont l'utilisation suit les règles décrites au § 4.2.

Exercice 49 Obtenir le tableau suivant en mode texte :

	Mathématiques pures	Mathématiques appliquées
Unités obligatoires	Topologie générale Géométrie différentielle Fonctions d'une variable complexe	Analyse numérique Probabilités Programmation linéaire
Unités optionnelles	Théorie des ensembles Théorie des catégories Calculabilité et décidabilité	Mécanique Mathématiques pour la Physique Accélération de la convergence

L'environnement `tabular` permet en sus le formatage des éléments d'une colonne sur plusieurs lignes consécutives organisées en mode paragraphe : pour cela, on indique comme directive de formatage de la colonne « `p{width}` », où `width` est la largeur de la colonne concernée.

Exercice 50 (Suite de l'exercice 49) *Modifier le tableau précédent comme suit :*

	Mathématiques pures	Mathématiques appliquées
Unités obligatoires	Topologie générale — Géométrie différentielle — Fonctions d'une variable complexe	Analyse numérique — Probabilités — Programmation linéaire
Unités optionnelles	Théorie des ensembles — Théorie des catégories — Calculabilité et décidabilité	Mécanique — Mathématiques pour la Physique — Accélération de la convergence

Dans notre réalisation, nous avons utilisé la directive `p{55mm}` pour les deux colonnes de droite⁴¹. Quant au tiret long (« — »), rappelons qu'il est obtenu par « --- ».

La commande `\multirow` du *package* de même nom peut s'employer à l'intérieur d'un environnement `tabular`.

Exercice 51 (Suite des exercices 49 & 50) *Reprendre encore une fois le tableau précédent et le modifier de façon à obtenir :*

		Mathématiques pures	Mathématiques appliquées
Unités obligatoires	Choisir 5 unités (3 + 2) en tout	Topologie générale — Géométrie différentielle — Fonctions d'une variable complexe	Analyse numérique — Probabilités — Programmation linéaire
Unités optionnelles		Théorie des ensembles — Théorie des catégories — Calculabilité et décidabilité	Mécanique — Mathématiques pour la Physique — Accélération de la convergence

Pour ce faire, nous avons utilisé la directive « `\multirow{5}{15mm}{Choisir ...}` » pour la colonne ajoutée et ramené à 48mm la largeur des deux colonnes de droite. Vous pouvez essayer d'autres valeurs (de 4 à 7) pour le premier argument de la commande `\multirow` et observer ce qui se produit. Remarquer aussi que de trop petites valeurs pour ce premier argument provoquent un message d'avertissement de *L^AT_EX*. Enfin, noter également qu'une commande `\hline` des exemples précédents doit être modifiée en deux commandes `\cline`.

5.2 Autres extensions du *package* array

Les extensions du *package* `array` qui concernent le mode texte et l'environnement `tabular` sont principalement les directives suivantes, utilisables dans un format :

`m{<width>}` — où `<width>` est la largeur de la colonne — est analogue à la directive « `p` » (cf. § 5.1), mais — comme nous l'avons vu au § 5.1 — cette dernière commence le texte sur la ligne qu'utilisent les directives « `c` », « `l` » ou « `r` », alors que la directive « `m` » permet de centrer verticalement du texte autour de cette ligne de base ;

`b{<width>}` est elle aussi analogue à la ligne de base, mais c'est la dernière ligne du texte qui est alignée avec la ligne de base ;

`!{<decl>}` est analogue à la directive « `@` » (cf. § 4.2) mais alors que cette dernière remplace l'espace entre deux colonnes consécutives par l'argument de « `@` », la directive « `!` » divise cet espace en deux parties et insère au milieu le contenu de son argument.

Nous mettons en évidence les utilisations comparées des directives « `p` », « `m` » et « `b` » au moyen de l'exemple suivant :

⁴¹On remarquera que leur contenu est bel et bien formaté en mode paragraphe : à preuve, quelques mots sont coupés.

Ligne de base	Ce texte illustre le formatage vertical à partir de la ligne de base.	Cette phrase illustre le formatage vertical centré sur la ligne de base.	Et encore une belle phrase pour illustrer le formatage vertical dont la dernière ligne est la ligne de base.	N'est-ce pas ?
---------------	---	--	--	----------------

les directives de format étant « `{|l|p{32mm}|m{32mm}|b{32mm}|l|}` ». Quant à la différence entre les directives « @ » et « ! », voici comment nous pouvons l'illustrer :

— Roméo Juliette
 — Sailor — Lula Steve — Julie
 XIII—Jones Mickey —Mallory

le texte source *in extenso* de ce tableau divisé en trois séries de deux colonnes étant :

```

\begin{tabular}{r@{---}lrlr!{---}l}
  & & Roméo & Juliette & \\
  & & Sailor & \multicolumn{1}{!{---}l}{Lula} & Steve & Julie \\
XIII & Jones & Mickey & \multicolumn{1}{@{---}l}{Mallory} & & \\
\end{tabular}

```

La troisième ligne de la première série confirme ce que nous avons déjà annoncé au § 4.2 : l'espacement entre colonnes est totalement remplacé lors d'une utilisation de la directive « @ ». La deuxième ligne de la troisième série fait apparaître le partage de l'espacement de part et d'autre du tiret long. Le partage de cet espacement apparaît davantage si l'on compare les première et deuxième lignes de la deuxième série de colonnes : l'espacement entre le tiret long et « Lula » est le même que l'espacement entre le début de la colonne (indiqué par le début du tiret long) et « Juliette ». Enfin, la troisième ligne de cette deuxième série montre également où commence la colonne de droite, l'espacement après le tiret long étant supprimé, conformément au comportement de la directive « @ ». Pour finir, nous pouvons aussi remarquer, en ce qui concerne cette troisième ligne, que le tiret long correspondant à la directive « !{---} » n'y apparaît pas : c'est parce que nous avons interrompu la mention des colonnes avant d'arriver à celle qui précédait cette directive ; ce point étant à rapprocher de la remarque qui suit l'exercice 38, à propos de la disparition de lignes verticales.

Exercice 52 (Suite de l'exercice 50) *Considérons à nouveau le tableau de cet exercice, mais légèrement modifié comme suit — les éléments de la première colonne sont serrés à gauche et quelques encadrements ont disparu — :*

	Mathématiques pures	Mathématiques appliquées
Unités obligatoires :	Topologie générale — Géométrie différentielle — Fonctions d'une variable complexe	Analyse numérique — Probabilités — Programmation linéaire
Unités optionnelles :	Théorie des ensembles — Théorie des catégories — Calculabilité et décidabilité	Mécanique — Mathématiques pour la Physique — Accélération de la convergence

Commencer sa formulation par « `\begin{tabular}{l{2}{p{58mm}|}}` » le place en début d'alinéa, après le renforcement à gauche présent au début de tout alinéa⁴². Ce renforcement peut être supprimé au moyen de la commande `\noindent`, mais si nous tapons :*

`\noindent \begin{tabular}{l*{2}{p{58mm}|}}`

⁴²... sauf le premier alinéa de chaque section, si le *package* `indentfirst` n'est pas utilisé : voir le § 1.6 du document *Premier contact avec L^AT_EX* à ce sujet.

il subsiste un léger espacement avant la première ligne verticale du tableau. Cet espacement — qui n'existe pas si la première colonne est délimitée à gauche par un trait vertical, comme elle l'était pour l'exercice 50 — peut être rapproché de l'espacement entre le tiret long et le mot « Lula » dans le mot précédent. L'exercice consiste à découvrir le moyen de supprimer cet espacement : vous pouvez remarquer qu'il n'apparaît pas dans notre réalisation.

Les directives « < » et « > » du *package* `array`, vues au § 4.5, peuvent s'appliquer pour les environnements `tabular` : en pratique, on les utilise souvent lorsque les contenus d'une colonne doivent adopter une présentation particulière (caractères gras, italiques, *etc.*) ou doivent être donnés en mode mathématique.

Exercice 53 Utiliser autant que possibles les directives « < » et « > » du *package* `array` pour spécifier le tableau suivant de la manière la plus légère qui soit :

Nom du corps	Notation	Radioactif?	Observations
Hydrogène	${}^1_1\text{H}$	<i>non</i>	
Deutérium	${}^2_1\text{H}$	<i>oui</i>	Composant de l'eau lourde, parfois noté D.
Tritium	${}^3_1\text{H}$	<i>oui</i>	Parfois noté T.
Carbone	${}^{12}_6\text{C}$	<i>non</i>	
Carbone 14	${}^{14}_6\text{C}$	<i>oui</i>	Utilisé pour la datation des fossiles.
Fluor	${}^{19}_9\text{F}$	<i>non</i>	
Fer	${}^{56}_{26}\text{Fe}$	<i>non</i>	
Sélénium	${}^{79}_{34}\text{Se}$	<i>non</i>	
Uranium	${}^{238}_{92}\text{U}$	<i>oui</i>	À l'origine de la découverte de la radio-activité.
Plutonium	${}^{244}_{94}\text{Pu}$	<i>oui</i>	Très dangereux.
Rutherfordium	${}^{261}_{104}\text{Rf}$	<i>oui</i>	Découvert en 1964.

5.3 Aller plus loin

\LaTeX possède une vaste panoplie de possibilités de spécifications de tableaux, dont nous avons tenté de vous présenter l'essentiel. Il est par exemple possible de définir de nouveaux types de colonnes (par le *package* `array`), de nouvelles combinaisons entre lignes horizontales et verticales (*package* `hhline`), ou de spécifier des tableaux s'étalant sur plusieurs pages consécutives (*packages* `longtable` ou `supertab`). L'exposé de ces possibilités sort clairement du cadre de ce support, mais ce qu'il est important de mentionner, c'est qu'étant donné une fonctionnalité précise pour un tableau particulier, il est rare de ne pas trouver dans la documentation des *packages* de \LaTeX un moyen de la réaliser.

Pour terminer

Bruno Aebischer a écrit en 2005 une première version de l'introduction au mode mathématique de \LaTeX , version dont je me suis inspiré au départ, merci à lui d'avoir « initialisé » le mouvement. Merci également à Anne-Marie Aebischer et François Pétiard, pour leur relecture très attentive des parties successives de ce document.

Table des matières

1	Introduction au mode mathématique	1
1.1	Premier contact	1
1.2	Espacement — Fontes mathématiques	2
1.3	Lettres grecques	3
1.4	Symboles	4
1.5	Indices et exposants	5
1.6	Fractions — racines carrées	5

2	Ajustements	6
2.1	Styles	6
2.2	Ajustements de limites	7
2.3	Ajustements horizontaux	7
2.4	Ajustements verticaux	8
2.5	Ajustements de caractères	9
2.6	Définitions de noms d'opérations	9
2.7	Délimiteurs	10
3	Environnements avancés	12
3.1	Définitions par cas	12
3.2	Équations numérotées	12
3.3	Environnements du <i>package</i> <code>amsmath</code>	12
3.4	Environnements fournis par L ^A T _E X	16
4	Matrices et tableaux	17
4.1	Environnements du <i>package</i> <code>amsmath</code> pour les matrices	17
4.2	Définitions de tableaux	18
4.3	Mini-tableaux	21
4.4	Ajustement de cases	22
4.5	<i>Packages</i> spécialisés pour les tableaux	23
5	Retour au mode texte	24
5.1	Environnement <code>tabular</code>	24
5.2	Autres extensions du <i>package</i> <code>array</code>	25
5.3	Aller plus loin	27