

# Documentation

Le package `Preamble.sty`<sup>1</sup> ou `HTMLPreamble.sty`<sup>2</sup> doit être chargé pour pouvoir utiliser les autres qui sont donnés ci-dessous.

Les fichiers `.sty` doivent être placés dans le même répertoire que le fichier `.tex` qui est utilisé.

Pour charger un package (par exemple `NomDuPackage.sty`), il faut utiliser la commande `\usepackage{nomdupackage}` avant `\begin{document}`.

En utilisant `Preamble.sty` ou `HTMLPreamble.sty`, les packages suivant seront chargés :

- `\usepackage[utf8]{inputenc}`
- `\usepackage[french]{babel}`
- `\usepackage[T1]{fontenc}`
- `\usepackage{amsmath, amsfonts, amssymb}`
- `\usepackage{stmaryrd}`
- `\usepackage{adjustbox}` (pour `HTMLPreamble.sty`)
- `\usepackage{xcolor}` (pour `Preamble.sty`)

Il est nécessaire que `cm-super` soit installé (disponible sur [CTAN](#)) pour pouvoir utiliser `Preamble.sty`. Pour ne pas avoir à installer `cm-super`, il est possible de commenter les lignes `\usepackage{sffont}` et `\renewcommand{\sfdefault}{cmssp}` du fichier `Preamble.sty` en mettant % au début de chacune de ces lignes (numéro 10 et 11).

Lors de l'utilisation de BEAMER (avec une police sans-sérif), il est possible d'utiliser les commandes avec les polices sans-serif, sauf pour les lettres grecques ( $\Omega$ ,  $\phi$ ,  $\varphi$ , ...), la redéfinition du  $\ell$  en mathématiques, les alphabets `\mathcal` et `\mathbb` ainsi que les symboles.

Il est possibles de changer les polices de caractères/symboles en important des packages après `\usepackage{preamble}`. Il peut être nécessaire de placer l'importation avant d'importer les autres modules décrit ci-dessous.

Il n'est pas possible d'utiliser en simultanée le package `Dsfont.sty` disponible sur [CTAN](#) et `Dsft.sty` décrit ci-dessous. De plus, la commande `\l` ne sera pas modifiée si un package définissant `\mathbb{1}` est importé. Il est alors possible de redéfinir la commande en utilisant `\newcommand\l[1]{\mathbb{1}_{\#1}}` (si `Dsft.sty` n'est pas importé) ou `\renewcommand\l[1]{\mathbb{1}_{\#1}}`.

Il est possible de redéfinir le  $\ell$  à sa version d'origine ( $\ell$ ) avec :

```
\mathcode`l="8000
\begingroup
\makeatletter
```

---

1. Pour utiliser avec BEAMER

2. Pour les documents autres que BEAMER

```

\lccode`~=`\l
\DeclareMathSymbol{\lsb@l}{\mathalpha}{letters}{`l}
\lowercase{\gdef~{\lsb@l}}%
\endgroup
\makeatother

```

Pour utiliser des commandes avec des parenthèses automatiques (comme pour sup), il est possible de faire<sup>3</sup> :

```

\let\oldsup\sup
\renewcommand{\sup}[1]{\oldsup\l#1\r}

```

\l et \r sont définis dans Preamble.sty et HTMLPreamble.sty.

La commande `\sup{\left\{x\in\mathbb{Q}\;;\;\sqrt{x^2}<2\right\}}` donne donc  $\sup(\{x \in \mathbb{Q} \mid x^2 < 2\})$ .

L'ensembles des titres des sections (et sous-sections si le fichier n'est pas déjà dans une section) sont des liens qui pointent vers les fichiers en ligne pour un téléchargement direct.

---

3. Cette commande est déjà définie dans `Usuelles.sty`

# Table des matières

0	Documentation	1
1	Flashcards.py et Htmlcards.py	3
2	Preamble.sty et HTMLPreamble.sty	5
3	AL.sty	6
4	Analyse.sty	7
5	Arithmetique.sty	8
6	BigOperators.sty	9
7	Complexes.sty	10
8	Dsft.sty	11
9	Equivalents.sty	12
10	Matrices.sty	13
11	Polynomes.sty	16
12	Probas.sty	17
13	Sffont.sty	18
14	Structures.sty	19
15	Tables.sty	20
16	Trigo.sty	21
17	Usuelles.sty	22

# 1 Flashcards.py et Htmlcards.py

Les fichiers `Flashcards.py` et `Htmlcards.py` permettent d'exporter facilement des flashcards en `.pdf` et `.svg` (pour affichage dans le navigateur).

Pour pouvoir créer des fiches de révision, il faut mettre un fichier `.txt` (décrit plus bas) dans un dossier `input` et mettre les fichiers `.sty` nécessaires dans un dossier `output`.

## 1.1 Flashcards.py

Pour exporter la fiche `fiche.txt`, il faut soit lancer le fichier python et entrer le nom du fichier (`fiche`), soit utiliser la commande `python Flashcards.py --file=fiche` (ou `python3`), à laquelle on peut rajouter les paramètres optionnels `--n=nombre` (avec le nombre d'exemplaires), `--dest=dossier` (avec le dossier où il faut mettre le `.pdf` produit) et `--open=True/False` (pour ouvrir le dossier où le `.pdf` est produit).

## 1.2 Htmlcards.py

Pour exporter la fiche `fiche.txt`, il faut soit lancer le fichier python et entrer le nom du fichier (`fiche`), soit utiliser la commande `python Flashcards.py --file=fiche` (ou `python3`), à laquelle on peut rajouter les paramètres optionnels `--dest=dossier` (avec le dossier où il faut mettre le `.pdf` produit) et `--open=True/False` (pour ouvrir le dossier où le `.pdf` est produit).

## 1.3 Les fiches .txt

Pour faire des fiches, il faut créer un fichier `.txt` de la forme

TITRE

Shuffle questions : True/False

Q/R & R/Q : True/False

PACKAGES & COMMANDES SUPPLÉMENTAIRES

QUESTION;;RÉPONSE

...

QUESTION;;RÉPONSE

Le titre doit être de la forme `Thème -- Chapitre` ou `Chapitre`. On peut aussi spécifier un titre raccourci pour le nom du fichier avec `Titre_raccourci!!titleTitre classique` où le titre raccourci ne peut pas contenir d'espaces ou de caractères spéciaux, et titre classique étant de la forme des deux premiers.

La ligne 2 indique si on peut ou non mettre un ordre aléatoire pour les questions.

La ligne 3 indique si on peut échanger l'ordre des questions et des réponses pour les fiches. Avec cette option à `True`, il est possible de forcer une question à être avant la réponse en mettant `!!fst` devant la/les ligne(s) concernée(s).

Les packages et commandes supplémentaires (voir [CTAN](#)) doivent être placées sur une seule ligne.

Il faut cependant s'assurer que le fichier ne contienne pas de lignes avec uniquement des espaces à la fin.

S'il y a une erreur lors de la compilation  $\text{\LaTeX}$ , le programme python affichera le message d'erreur affiché par  $\text{\LaTeX}$ .

Exemple de fiches : <https://github.com/rfoxinter/revisions/tree/main/input>.

## 1.4 Visionner les flashcards en svg

Pour pouvoir visionner les flashcards exportées en svg, il faut disposer d'un serveur web (comme [github](#) avec [github pages](#)) sur lequel on met le dossier généré par `Htmlcards.py` (on suppose que l'url est `https://example.fr/dossier`).

Il faut alors convertir l'url du dossier en base 64 (cette conversion peut se faire sur le site <https://www.base64encode.org/>, avec la fonction `btoa` de JavaScript ou avec la fonction Python `base64.b64encode`). Dans l'exemple, en exécutant le code Python suivant

```
import base64
url = "https://example.fr/dossier"
print(base64.b64encode(url.encode()).decode())
```

on obtient `aHR0cHM6Ly9leGFtcGxlLmZyL2Rvc3NpZXI=`.

Il faut alors aller sur le site <https://rfoxinter.github.io/revisions/flashcards/> en rajoutant à la fin de l'url `?file=nom_du_dossier` où le nom du dossier correspond à celui en base 64.

Dans l'exemple, on obtient l'url suivante :

```
https://rfoxinter.github.io/revisions/flashcards/
?file=aHR0cHM6Ly9leGFtcGxlLmZyL2Rvc3NpZXI=
```

## 2 Preamble.sty et HTMLPreamble.sty

Commande	Résultat
<code>\l</code> <sup>4</sup>	(
<code>\r</code> <sup>5</sup>	)
<code>\llb</code> <sup>6</sup>	[[
<code>\rrb</code> <sup>7</sup>	]]
<code>\oldfrac{a}{b}</code> <sup>8</sup>	$\frac{a}{b}$
<code>\frac{a}{b}</code> <sup>9</sup>	$\frac{a}{b}$
<code>\l</code> <sup>10</sup>	$\ell$
<code>\oldvec{x}</code> <sup>11</sup>	$\vec{x}$
<code>\vec{AB}</code>	$\overrightarrow{AB}$
<code>\overrightarrow{AB}</code> <sup>12</sup>	$\overrightarrow{AB}$

- 
4. Correspond à la commande usuelle `\left(`
  5. Correspond à la commande usuelle `\right)`
  6. Correspond à la commande usuelle `\left\llbracket`
  7. Correspond à la commande usuelle `\right\rrbracket`
  8. Correspond à la commande usuelle `\frac`
  9. Correspond à la commande usuelle `\dfrac`
  10. Correspond à la commande usuelle `\ell`
  11. Correspond à la commande usuelle `\vec`
  12. Le résultat est le même qu'avec `\vec`

### 3 AL.sty

Le package `Matrices.sty` sera importé automatiquement avec `AL.sty`.

Commande	Résultat
<code>\oldvect</code>	$\text{Vect}$
<code>\vect{E}</code>	$\text{Vect}(E)$
<code>\al{E}\{\}</code>	$\mathcal{L}(E)$
<code>\al{E}\{F\}</code>	$\mathcal{L}(E, F)$
<code>\oplus</code> <sup>13</sup>	$\oplus$
<code>\matgl{n}\{\mathbb{K}\}</code> <sup>14</sup>	$\text{GL}_n(\mathbb{K})$
<code>\gl{E}</code>	$\text{GL}(E)$
<code>\olddim</code> <sup>15</sup>	$\dim$
<code>\dim{E}</code>	$\dim(E)$
<code>\oldrg</code>	$\text{rg}$
<code>\rg{u}</code>	$\text{rg}(u)$
<code>\oldtr</code>	$\text{tr}$
<code>\tr{u}</code>	$\text{tr}(u)$
<code>\oldmat</code>	$\text{Mat}$
<code>\almat{u}\{\mathcal{B}\}\{\}</code> <sup>16</sup>	$\text{Mat}_{\mathcal{B}}(u)$
<code>\almat{u}\{\mathcal{B}\}\{\mathcal{C}\}</code>	$\text{Mat}_{\mathcal{B}, \mathcal{C}}(u)$
<code>\lc</code>	[
<code>\rc</code>	]
<code>\oldsl</code>	$\text{SL}$
<code>\sl{E}</code>	$\text{SL}(E)$
<code>\sl{n}\{\mathbb{K}\}</code>	$\text{SL}_n(\mathbb{K})$
<code>\orth{n}</code>	$\text{O}(n)$
<code>\orth{n}\{\mathbb{R}\}</code>	$\text{O}_n(\mathbb{R})$
<code>\so{n}</code>	$\text{SO}(n)$
<code>\so{n}\{\mathbb{R}\}</code>	$\text{SO}_n(\mathbb{R})$

---

13. Le `\oplus` utilisé est celui de `stmaryrd`  
 Pour récupérer celui de  $\text{\LaTeX}$ , il est possible d'utiliser la commande `\let\oldoplus\oplus` avant `\usepackage{al}` puis de faire `\let\oplus\oldoplus` après importation  
 Comparaison  $\text{\LaTeX}$  - `stmaryrd` avec le plus normal  $\oplus \oplus +$

14. Le `\matgl` de `AL.sty` correspond à la commande `\gl` de `Matrices.sty` qui a été renommé

15. Correspond à la commande usuelle `\dim`

16. Ne pas confondre cette commande avec `\mat` de `Matrices.sty`

## 4 Analyse.sty

Le package `BigOperators.sty` sera importé automatiquement avec `Analyse.sty`.

Commande	Résultat
<code>\oldd</code> <sup>17</sup>	$d$
<code>\der{f(x)}</code>	$\frac{d}{dx}(f(x))$
<code>\der[n]{f(x)}</code>	$\frac{d^n}{dx^n}(f(x))$
<code>\der[] [t]{f(t)}</code>	$\frac{d}{dt}(f(t))$
<code>\der[n] [t]{f(t)}</code>	$\frac{d^n}{dt^n}(f(t))$
<code>\pder{f(x)}</code> <sup>18</sup>	$\frac{\partial}{\partial x}(f(x))$
<code>\oldint</code> <sup>19</sup>	$\int$
<code>\int{f}</code>	$\int (f)$
<code>\int [t]{f(t)}</code>	$\int (f(t)) dt$
<code>\int [t] [{[a,b]}]{f(t)}</code> <sup>20</sup>	$\int_{[a,b]} (f(t)) dt$
<code>\int [t] [a] [b]{f(t)}</code>	$\int_a^b (f(t)) dt$
<code>\int [] [a] [b]{f'}</code>	$\int_a^b (f')$
<code>\eval [{[a,b]}]{f(t)}</code>	$[f(t)]_{[a,b]}$
<code>\eval [a] [b]{f(t)}</code>	$[f(t)]_a^b$
<code>\serie{a_n}</code>	$\sum a_n$

---

17.  $d$  de dérivation

18. On peut appliquer les mêmes arguments optionnels que pour `\der`

19. Correspond à la commande usuelle `\int`

20. L'argument `[a,b]` doit être mis entre accolades pour être traité correctement par  $\text{\LaTeX}$



## 5 Arithmetique.sty

Commande	Résultat
<code>\olddiv</code> <sup>21</sup>	$\div$
<code>\div</code> <sup>22</sup>	$ $
<code>\cgr{a}{b}{n}</code>	$a \equiv b [n]$
<code>\oldphi</code> <sup>23</sup>	$\phi$
<code>\phi</code> <sup>24</sup>	$\varphi$

---

21. Correspond à la commande usuelle `\div`

22. Correspond à la commande usuelle `\mid`

23. Correspond à la commande usuelle `\phi`

24. Correspond à la commande usuelle `\varphi`

## 6 BigOperators.sty

Commande	Résultat
<code>\oldsum</code> <sup>25</sup>	$\sum$
<code>\sum{n=0}{+\infty}{u_n}</code>	$\sum_{n=0}^{+\infty} (u_n)$
<code>\oldprod</code> <sup>26</sup>	$\prod$
<code>\prod{n=0}{+\infty}{u_n}</code>	$\prod_{n=0}^{+\infty} (u_n)$
<code>\oldcap</code> <sup>27</sup>	$\cap$
<code>\bigcap{n=0}{+\infty}{A_n}</code>	$\bigcap_{n=0}^{+\infty} (A_n)$
<code>\oldcup</code> <sup>28</sup>	$\cup$
<code>\bigcup{n=0}{+\infty}{A_n}</code>	$\bigcup_{n=0}^{+\infty} (A_n)$
<code>\olduplus</code> <sup>29</sup>	$\uplus$
<code>\biguplus{n=0}{+\infty}{A_n}</code>	$\biguplus_{n=0}^{+\infty} (A_n)$
<code>\bigoplus{n=0}{+\infty}{E_n}</code>	$\bigoplus_{n=0}^{+\infty} (E_n)$

---

25. Correspond à la commande usuelle `\sum`

26. Correspond à la commande usuelle `\prod`

27. Correspond à la commande usuelle `\bigcap`

28. Correspond à la commande usuelle `\bigcup`

29. Correspond à la commande usuelle `\biguplus`

## 7 Complexes.sty

Commande	Résultat
<code>\oldbar{z}</code> <sup>30</sup>	$\bar{z}$
<code>\bar{z}</code> <sup>31</sup>	$\bar{z}$
<code>\e</code> <sup>32</sup>	$e$
<code>\i</code> <sup>33</sup>	$i$
<code>\j</code> <sup>34</sup>	$j$
<code>\oldIm</code> <sup>35</sup>	$\Im$
<code>\Im</code>	$\mathrm{Im}$
<code>\pIm{x}</code>	$\mathrm{Im}(x)$
<code>\oldRe</code> <sup>36</sup>	$\Re$
<code>\Re</code>	$\mathrm{Re}$
<code>\pRe{x}</code>	$\mathrm{Re}(x)$

### 7.1 Dessiner des arcs

Il est aussi possible de faire des arcs en important le package `Complexes.sty` avec l'option `arc`, en utilisant `\usepackage[arc]{complexes}`.

Cette option importe automatiquement le package `graphics` (disponible sur [CTAN](#)).

Commande	Résultat
<code>\arc{AB}</code>	$\widehat{AB}$
<code>\arc{ABCDEFGFGH}</code>	$\widehat{ABCDEFGFGH}$

---

30. Correspond à la commande usuelle `\bar`

31. Se comporte comme `\overline`

32.  $e$  de la fonction exponentielle

33.  $i$  complexe

L'ancienne commande `\i` s'obtient avec `\ii`

34.  $j = e^{\frac{2i\pi}{3}}$

L'ancienne commande `\j` s'obtient avec `\jj`

35. Correspond à la commande usuelle `\Im`

36. Correspond à la commande usuelle `\Re`

## 8 Dsft.sty

Ce package remplace le `\mathds{1}` du package `Dsfonts.sty` disponible sur [CTAN](#) et introduit quelques symboles.

Commande	Résultat
<code>\mathds{1}</code>	$\mathbb{1}$
<code>\1{E}(x)</code>	$\mathbb{1}_E(x)$
<code>\square</code>	$\square$
<code>\star</code>	$\star$
<code>\triangle</code>	$\triangle$

### 8.1 dsrom12.pfb et dsrom12.tfm

Pour utiliser ce package, il faut copier les fichiers `dsrom12.pfb` et `dsrom12.tfm` dans les dossiers où ils sont actuellement avec `dsfonts` (et éventuellement créer une copie des anciens fichiers).

## 9 Equivalents.sty

Commande	Résultat
$\backslash o\{x\}$	$o(x)$
$\backslash o[x\to 0]\{x\}$	$o_{x\rightarrow 0}(x)$
$\backslash O\{x\}$	$O(x)$
$\backslash O[x\to 0]\{x\}$	$O_{x\rightarrow 0}(x)$
$\backslash Th\{x\}$	$\Theta(x)$
$\backslash Th[x\to 0]\{x\}$	$\Theta_{x\rightarrow 0}(x)$
$\backslash Om\{x\}$	$\Omega(x)$
$\backslash Om[x\to 0]\{x\}$	$\Omega_{x\rightarrow 0}(x)$
$\backslash eq\{u_n\}\{v_n\}$	$u_n \sim v_n$
$\backslash eq[n\to+\infty]\{u_n\}\{v_n\}$	$u_n \underset{n\rightarrow+\infty}{\sim} v_n$
$\backslash eg\{u_n\}\{v_n+\backslash o\{v_n\}\}$	$u_n = v_n + o(v_n)$
$\backslash eg[n\to+\infty]\{u_n\}\{v_n+\backslash o\{v_n\}\}$	$u_n \underset{n\rightarrow+\infty}{=} v_n + o(v_n)$

## 10 Matrices.sty

Commande	Résultat
<code>\mat{n}{p}{\mathbb{K}}</code>	$\mathcal{M}_{n,p}(\mathbb{K})$
<code>\mat{n}{}{\mathbb{K}}</code>	$\mathcal{M}_n(\mathbb{K})$
<code>\sym{n}{\mathbb{K}}</code>	$\mathcal{S}_n(\mathbb{K})$
<code>\ant{n}{\mathbb{K}}</code>	$\mathcal{A}_n(\mathbb{K})$
<code>\diag{n}{\mathbb{K}}</code>	$\mathcal{D}_n(\mathbb{K})$
<code>\ts{n}{\mathbb{K}}</code>	$\mathcal{T}_n^+(\mathbb{K})$
<code>\ti{n}{\mathbb{K}}</code>	$\mathcal{T}_n^-(\mathbb{K})$
<code>\olddet</code> <sup>37</sup>	det
<code>\det{M}</code>	$\det(M)$
<code>\det[\mathcal{B}]{\mathcal{B}'}</code>	$\det_{\mathcal{B}}(\mathcal{B}')$
<code>\gl{n}{\mathbb{K}}</code> <sup>38</sup>	$\mathrm{GL}_n(\mathbb{K})$
<code>\oldcom</code>	Com
<code>\com{M}</code>	$\mathrm{Com}(M)$
<code>\mdots</code>	$\cdots$
<code>\ddots</code>	$\ddots$
<code>\idots</code>	$\cdot\cdot$
<code>\vdots</code>	$\vdots$
<code>\xdots</code>	$\ddots$
<code>\plusdots</code>	$\ddots$
<code>\tmatrix({1\&amp;0\\0\&amp;1\\})</code> <sup>39</sup>	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

### 10.1 Ne pas importer les commandes avec ·

Il est possible de ne pas modifier les commandes usuelles en utilisant l'option `nodots` de ce package. Il faut alors importer le package avec `\usepackage[nodots]{matrices}`.

### 10.2 Modifier les séparations entre les ·

Les commandes avec des points tel que  $\ddots$  ont des définitions qui dépendent de la taille de la police (celle pour L<sup>A</sup>T<sub>E</sub>X est adaptée pour 12pt, et celle de BEAMER pour 17pt). Pour avoir des points alignés correctement, il est possible de modifier la valeur de `\dotsep` en utilisant `\setlength{\dotsep}{taille_en_pt}`.

Par exemple, avec 2pt, on obtient :  $\ddots$ .

Il est également possible de faire de même avec la hauteur des  $\cdots$  en modifiant la longueur `\dotlift`. De même avec `\matmin` pour `minimum width` et `minimum width`, ou encore `\matsep` pour `row sep` et `column sep`.

<sup>37</sup>. Correspond à la commande usuelle `\det`

<sup>38</sup>. Si `AL.sty` est chargé, cette commande est remplacée et il faut utiliser `\matgl{n}{\mathbb{K}}` pour obtenir ce résultat

<sup>39</sup>. Les caractères `\&` sont utilisés au lieu du `&` utilisé habituellement avec TikZ pour des raisons de compatibilité avec BEAMER

### 10.3 La commande `\tmatrix`

`\tmatrix` est composé de deux arguments optionnels (les éléments à ajouter à la matrice *TikZ* et les éléments de mise en page de la matrice) ainsi que de trois arguments (le délimiteur d'ouverture, le contenu de la matrice et le délimiteur de fermeture).

Commande	Résultat
<code>\mtxvline{params}{n}</code>	Crée une ligne verticale après la colonne <b>n</b> (ou left/right pour les extrémités) avec les paramètres <i>TikZ</i> <b>params</b>
<code>\mtxhline{params}{n}</code>	Crée une ligne horizontale après la ligne <b>n</b> (ou top/bottom pour les extrémités) avec les paramètres <i>TikZ</i> <b>params</b>
<code>\mtxvpartial{params}{n}{a}{b}</code>	Crée une ligne verticale après la colonne <b>n</b> (ou left/right pour les extrémités), la ligne ayant pour extrémités la fin de la ligne <b>a</b> et <b>b</b> (ou top/bottom) avec les paramètres <i>TikZ</i> <b>params</b>
<code>\mtxhpartial{params}{n}{a}{b}</code>	Crée une ligne horizontale après la ligne <b>n</b> (ou top/bottom pour les extrémités), la ligne ayant pour extrémités la fin de la ligne <b>a</b> et <b>b</b> (ou left/right) avec les paramètres <i>TikZ</i> <b>params</b>
<code>\mtxbox{params}{x}{y}</code>	Crée une boîte autour de la case de coordonnées <b>x</b> et <b>y</b> (l'indexation commence à 1) avec les paramètres <i>TikZ</i> <b>params</b>

### 10.4 Exemples avec `\tmatrix`

$\det(M) = \begin{vmatrix} a & b \\ c & d \end{vmatrix}$  est produit par `\det{M}=\tmatrix|{a\&b\\c\&d\\}|$`.

$I_{n,p,r} = \left( \begin{array}{c|c} I_r & 0_{r,p-r} \\ \hline 0_{n-r,r} & 0_{n-r,p-r} \end{array} \right)$  est produit par

```
$I_{n,p,r}=\tmatrix
[\mtxvline{line width = 0.05em}{1}\mtxhline{line width = 0.05em}{1}]
[minimum height = 5ex, row sep = 1ex, minimum width = 5ex,
column sep = 1ex,]
({I_r\&0_{r,p-r}\\0_{n-r,r}\&0_{n-r,p-r}\\})$
```

$$\left[ \begin{array}{cccc} \boxed{A_1} & 0 & 0 & 0 \\ 0 & \boxed{A_2} & \ddots & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \boxed{A_n} \end{array} \right] \text{ est produit par }$$

`\tmatrix`

```
[\mtxbox{red, dashed}{1}{1}\mtxbox{teal, dotted,
  ultra thick}{2}{2}\mtxbox{}{4}{4}]
[minimum height = 5ex, minimum width = 5ex, row sep = 10pt,
  inner sep = 5pt, column sep = 10pt,]
{[ ]} % Le crochet est entouré de deux paires d'accolades
{A_1\&0\&0\&0\&A_2\&\ddots\&0\&0\&\ddots\&\ddots\&0\&
  0\&0\&0\&A_n\&}\{ \}
```



## 11 Polynomes.sty

Commande	Résultat
<code>\pol{K}{X}</code>	$\mathbb{K}[X]$
<code>\fr{K}{X}</code>	$\mathbb{K}(X)$
<code>\olddeg</code> <sup>40</sup>	$\deg$
<code>\deg{P}</code>	$\deg(P)$
<code>\oldval</code>	$\text{val}$
<code>\val{P}</code>	$\text{val}(P)$
<code>\oldcar</code>	$\text{car}$
<code>\car{\mathbb{K}}</code>	$\text{car}(\mathbb{K})$

---

40. Correspond à la commande usuelle `\deg`

## 12 Probas.sty

Commande	Résultat
<code>\p{A}</code>	$\mathbb{P}(A)$
<code>\p[B]{A}</code>	$\mathbb{P}_B(A)$
<code>\oldOmega</code> <sup>41</sup>	$\Omega$
<code>\Omega</code> <sup>42</sup>	$\Omega$
<code>\sq</code> <sup>43</sup>	$ $
<code>\bor</code> <sup>44</sup>	$\mathcal{B}$
<code>\esp{X}</code>	$\mathbb{E}(X)$
<code>\var{X}</code>	$\mathbb{V}(X)$
<code>\ect{X}</code>	$\sigma(X)$
<code>\oldcov</code>	$\text{cov}$
<code>\cov{X}{Y}</code>	$\text{cov}(X, Y)$
<code>\indep</code> <sup>45</sup>	$\perp$
<code>\unif{n}</code>	$\mathcal{U}(n)$
<code>\bin{p}</code>	$\mathcal{B}(p)$
<code>\bin[n]{p}</code>	$\mathcal{B}(n, p)$
<code>\geom{p}</code>	$\mathcal{G}(p)$
<code>\pasc{r}{p}</code>	$\mathcal{P}(r, p)$
<code>\nbin{r}{p}</code>	$\mathcal{J}(r, p)$
<code>\hypg{N}{n}{q}</code>	$\mathcal{H}(N, n, q)$
<code>\poiss{\lambda}</code>	$\mathcal{P}(\lambda)$

---

41. Correspond à la commande usuelle `\Omega`

42. Correspond à la commande usuelle `\varOmega`

43. Doit être utilisé entre `\left` et `\right`, ou dans la commande `\p : \mathbb{P}\left(A \middle| \bigcap_{k=1}^n (B_i)\right)`

44. Correspond à la commande usuelle `\mathcal{B}`

45. Ce symbole est obtenu avec la commande `\perp\!\!\!\perp`

## 13 Sffont.sty

Ce package définit une nouvelle police `cmssp` qui correspond à `cmss` en 10pt. Pour l'utiliser, il faut utiliser `\fontfamily{cmssp}\fontsize{Xpt}{\baselineskip}\selectfont`.

Exemples avec `cmssp` :

**Exemple avec une police de taille 25pt en gras.**

*Exemple avec une police de taille 17pt en italique.*

***Exemple avec une police de taille 12pt en gras italique.***

Les mêmes exemples avec `cmss` :

**Exemple avec une police de taille 25pt en gras.**

*Exemple avec une police de taille 17pt en italique.*

***Exemple avec une police de taille 12pt en gras italique.***

## 14 Structures.sty

Commande	Résultat
<code>\oldhom</code>	$\text{Hom}$
<code>\hom{E}</code>	$\text{Hom}(E)$
<code>\oldaut</code>	$\text{Aut}$
<code>\aut{E}</code>	$\text{Aut}(E)$
<code>\oldker</code> <sup>46</sup>	$\ker$
<code>\ker{f}</code>	$\ker(f)$
<code>\oldim</code>	$\text{im}$
<code>\im{f}</code>	$\text{im}(f)$
<code>\la</code> <sup>47</sup>	$\langle$
<code>\ra</code> <sup>48</sup>	$\rangle$
<code>\oldord</code>	$\text{ord}$
<code>\ord{x}</code>	$\text{ord}(x)$

---

46. Correspond à la commande usuelle `\ker`

47. Correspond à la commande usuelle `\left\langle`

48. Correspond à la commande usuelle `\right\rangle`

## 15 Tables.sty

Ce package sert à mettre en forme des tables an latex grâce à TikZ.

Pour insérer une table, il faut appeler `\setrowcol[width][height]{ncols}{nrows}` avec le nombre de colonnes et de lignes de la table, puis rentrer la table TikZ, les arguments optionnels étant la largeur de la table et sa hauteur.

Une table a une largeur par défaut de 10cm et une hauteur de 6,5cm (est est réinitialisée à chaque appel de `\setrowcol`).

Il est possible d'utiliser `[ampersand replacement=\&]` puis `\&` pour la matrice lorsque `&` est déjà défini par .

Il est possible de récupérer la valeur de la largeur et de la hauteur avec `\tblw` et `\tblh`.

Par exemple, la table

	0	$\frac{\pi}{6}$	$\frac{\pi}{4}$	$\frac{\pi}{3}$	$\frac{\pi}{2}$
sin	0	$\frac{1}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{3}}{2}$	1
cos	1	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{1}{2}$	0
tan	0	$\frac{1}{\sqrt{3}}$	1	$\sqrt{3}$	—
cot	—	$\sqrt{3}$	1	$\frac{1}{\sqrt{3}}$	0

est produite avec le code suivant

```
\LARGE
\setcolrow{6}{5}
\begin{tikzpicture}
  \matrix[table] {
    & $\oldfrac{\pi}{6}$ & $\oldfrac{\pi}{4}$ & $\oldfrac{\pi}{3}$ & $\oldfrac{\pi}{2}$ \\
    sin & 0 & $\frac{1}{2}$ & $\frac{\sqrt{2}}{2}$ & $\frac{\sqrt{3}}{2}$ & 1 \\
    cos & 1 & $\frac{\sqrt{3}}{2}$ & $\frac{\sqrt{2}}{2}$ & $\frac{1}{2}$ & 0 \\
    tan & 0 & $\frac{1}{\sqrt{3}}$ & 1 & $\sqrt{3}$ & — \\
    cot & — & $\sqrt{3}$ & 1 & $\frac{1}{\sqrt{3}}$ & 0
  };
  \draw [line width=0.5mm] (-\tblw/3,-\tblh/2) -- (-\tblw/3,\tblh/2);
  \draw [line width=0.5mm] (-\tblw/2,3*\tblh/10) --
    (\tblw/2,3*\tblh/10);
  \draw [line width=0.5mm] (-\tblw/2,-\tblh/2) rectangle
    (\tblw/2,\tblh/2);
\end{tikzpicture}
```

## 16 Trigo.sty

Commande	Résultat
<code>\oldcos</code> <sup>49</sup>	$\cos$
<code>\cos{x}</code>	$\cos(x)$
<code>\cos[n]{x}</code>	$\cos^n(x)$
<code>\oldsin</code> <sup>50</sup>	$\sin$
<code>\sin{x}</code>	$\sin(x)$
<code>\sin[n]{x}</code>	$\sin^n(x)$
<code>\oldtan</code> <sup>51</sup>	$\tan$
<code>\tan{x}</code>	$\tan(x)$
<code>\tan[n]{x}</code>	$\tan^n(x)$
<code>\oldcot</code> <sup>52</sup>	$\cot$
<code>\cot{x}</code>	$\cot(x)$
<code>\cot[n]{x}</code>	$\cot^n(x)$
<code>\acos{x}</code>	$\arccos(x)$
<code>\acos[n]{x}</code>	$\arccos^n(x)$
<code>\asin{x}</code>	$\arcsin(x)$
<code>\asin[n]{x}</code>	$\arcsin^n(x)$
<code>\atan{x}</code>	$\arctan(x)$
<code>\atan[n]{x}</code>	$\arctan^n(x)$
<code>\oldch</code>	$\operatorname{ch}$
<code>\ch{x}</code>	$\operatorname{ch}(x)$
<code>\ch[n]{x}</code>	$\operatorname{ch}^n(x)$
<code>\oldsh</code>	$\operatorname{sh}$
<code>\sh{x}</code>	$\operatorname{sh}(x)$
<code>\sh[n]{x}</code>	$\operatorname{sh}^n(x)$
<code>\oldth</code>	$\operatorname{th}$
<code>\th{x}</code>	$\operatorname{th}(x)$
<code>\th[n]{x}</code>	$\operatorname{th}^n(x)$
<code>\oldach</code>	$\operatorname{argch}$
<code>\ach{x}</code>	$\operatorname{argch}(x)$
<code>\ach[n]{x}</code>	$\operatorname{argch}^n(x)$
<code>\oldash</code>	$\operatorname{argsh}$
<code>\ash{x}</code>	$\operatorname{argsh}(x)$
<code>\ash[n]{x}</code>	$\operatorname{argsh}^n(x)$
<code>\oldath</code>	$\operatorname{argth}$
<code>\ath{x}</code>	$\operatorname{argth}(x)$
<code>\ath[n]{x}</code>	$\operatorname{argth}^n(x)$

---

49. Correspond à la commande usuelle `\cos`

50. Correspond à la commande usuelle `\sin`

51. Correspond à la commande usuelle `\tan`

52. Correspond à la commande usuelle `\cot`

## 17 Usuelles.sty

Commande	Résultat
<code>\oldmin</code> <sup>53</sup>	$\min$
<code>\min{\llb0,n\rrb}</code>	$\min(\llbracket 0, n \rrbracket)$
<code>\min[\mathbb{N}^*]{\llb0,n\rrb}</code>	$\min_{\mathbb{N}^*}(\llbracket 0, n \rrbracket)$
<code>\oldmax</code> <sup>54</sup>	$\max$
<code>\max{\llb0,n\rrb}</code>	$\max(\llbracket 0, n \rrbracket)$
<code>\max[\mathbb{Z}_-]{\llb0,n\rrb}</code>	$\max_{\mathbb{Z}_-}(\llbracket 0, n \rrbracket)$
<code>\oldlim</code> <sup>55</sup>	$\lim$
<code>\lim{u_n}</code>	$\lim(u_n)$
<code>\lim[x\to+\infty]{f(x)}</code>	$\lim_{x \rightarrow +\infty}(f(x))$
<code>\limi{u_n}</code>	$\liminf(u_n)$
<code>\limi[x\to+\infty]{f(x)}</code>	$\liminf_{x \rightarrow +\infty}(f(x))$
<code>\lims{u_n}</code>	$\limsup(u_n)$
<code>\lims[x\to+\infty]{f(x)}</code>	$\limsup_{x \rightarrow +\infty}(f(x))$
<code>\oldexp</code> <sup>56</sup>	$\exp$
<code>\exp{x}</code>	$\exp(x)$
<code>\exp[n]{x}</code>	$\exp^n(x)$
<code>\oldln</code> <sup>57</sup>	$\ln$
<code>\ln{x}</code>	$\ln(x)$
<code>\ln[n]{x}</code>	$\ln^n(x)$
<code>\oldinf</code> <sup>58</sup>	$\inf$
<code>\inf{\varnothing}</code>	$\inf(\emptyset)$
<code>\inf[\bar{\mathbb{R}}]{\{u_n\}}</code>	$\inf_{\mathbb{R}}(\{u_n\})$
<code>\oldsup</code> <sup>59</sup>	$\sup$
<code>\sup{\varnothing}</code>	$\sup(\emptyset)$
<code>\sup[\bar{\mathbb{R}}]{\{u_n\}}</code>	$\sup_{\mathbb{R}}(\{u_n\})$

53. Correspond à la commande usuelle `\min`

54. Correspond à la commande usuelle `\max`

55. Correspond à la commande usuelle `\lim`

56. Correspond à la commande usuelle `\exp`

57. Correspond à la commande usuelle `\ln`

58. Correspond à la commande usuelle `\inf`

59. Correspond à la commande usuelle `\sup`