

E

Exercices supplémentaires et solutions

Dans cette annexe, on propose des exercices supplémentaires à certains chapitres de l'ouvrage imprimé. À leur suite, on trouve les solutions des exercices du livre et des exercices supplémentaires.

Les exercices supplémentaires et exemples afférents sont numérotés avec le préfixe « E- » de façon à bien être distingués de ceux qui se trouvent dans l'ouvrage papier.

Chapitre 2

Exercices supplémentaires du chapitre 2	405
Solutions des exercices du chapitre 2	405

Chapitre 3

Exercices supplémentaires du chapitre 3	415
Solutions des exercices du chapitre 3	415

Chapitre 4

Exercices supplémentaires du chapitre 4	425
Solutions des exercices du chapitre 4	425

Chapitre 5

Exercices supplémentaires du chapitre 5	435
Solutions des exercices du chapitre 5	435

Chapitre 6

Solutions des exercices du chapitre 6	444
---	-----

Chapitre 7	
Solutions des exercices du chapitre 7	446
Chapitre 8	
Exercices supplémentaires du chapitre 8	452
Solutions des exercices du chapitre 8	452
Chapitre 9	
Exercices supplémentaires du chapitre 9	458
Solutions des exercices du chapitre 9	458
Chapitre 10	
Solutions des exercices du chapitre 10	465
Chapitre 11	
Exercices supplémentaires du chapitre 11	471
Solutions des exercices du chapitre 11	471
Chapitre 12	
Exercices supplémentaires du chapitre 12	476
Solutions des exercices du chapitre 12	476
Chapitre 13	
Solutions des exercices du chapitre 13	483
Chapitre 14	
Solutions des exercices du chapitre 14	485
Annexe C	
Solutions des exercices de l'annexe C	488

Chapitre 2

Exercices supplémentaires

Exercice E-2-8 Correction des erreurs

On demande dans cet exercice de créer puis de compiler un fichier `.tex` ayant le contenu suivant, et d'utiliser les outils de l'éditeur de texte orienté \LaTeX pour repérer et corriger les erreurs rencontrées (en recompilant après chaque problème réglé).

```

1 \documentclass[french]{article}
2 \usepackage[utf8]{inputenc}
3 \usepackage[T1]{fontenc}
4 \usepackage{lmodern}
5 \usepackage[a4paper]{geometry}
6 \usepackage{babel}
7 Bonjour tout le monde!
8 \begin{document}
9 \textbf{Les caprices de votre traitement de texte vous exaspèrent, vous
10 n'en pouvez plus de la lourdeur de votre éditeur d'équations, vos
11 documents sont d'une qualité typographique douteuse et vous
12 désespérez de voir votre mise en page évoluer au gré d'un changement
13 de machine ou d'un numéro de version\dots{}} Si vous vous
14 reconnaissez dans ce portrait, alors \LaTeX{} pourrait vous
15 intéresser.}
16 \begin{center}
17 \LaTeX{} est un logiciel de composition typographique
18 particulièrement bien adapté à la rédaction de textes techniques et
19 scientifiques de haute qualité: rapports, livres, transparents,
20 posters,~etc. Il est si efficace pour l'écriture d'équations
21 qu'aujourd'hui la quasi-totalité des livres et articles
22 mathématiques sont écrits en \LaTeX{}.
23 \end{tiny}
24 \textbf{La mise en page y est automatique, tout en étant entièrement
25 configurable. Les outils nécessaires à la génération d'une table des
26 matières, d'un index ou d'une bibliographie sont faciles à utiliser
27 et la numérotation des équations, figures, tableaux et références
28 est un jeu d'enfant.}}
29 \begin{flushleft}
30 Le même fichier source \LaTeX{} (par ailleurs extrêmement compact)
31 engendre aujourd'hui comme hier le même document (dvi, PostScript,
32 pdf, html,~etc.) que l'on utilise un environnement Unix, Linux,
33 Windows ou MacOS. Ajoutons à cela que \LaTeX{} est stable, libre et
34 gratuit.
35 \end{document}

```

Exercice E-2-9 Correction d'erreur délicate

Cet exercice est analogue au précédent : il faut lever l'erreur rencontrée.

```

1 \documentclass[french,12pt]{article}
2 \usepackage[utf8]{inputenc}
3 \usepackage[T1]{fontenc}
4 \usepackage[a4paper]{geometry}
5 \usepackage{lmodern}
6 \usepackage{babel}
7 \begin{document}
8 \textbf{\LaTeX} est un logiciel de composition typographique
9   particulièrement bien adapté à la rédaction de
10  textes techniques et scientifiques de haute qualité :
11  rapports, livres, transparents, posters,~etc.
12 \end{document}

```

Exercice E-2-10 Mise en forme d'une citation

a Obtenir la citation suivante.

E-1	<p>Jean DE LA BRUYÈRE disait : « L'esclave <i>n'a qu'un maître</i>; l'ambitieux <i>en a autant qu'il y a de gens utiles à sa fortune.</i> »</p>
-----	---

b La citation précédente n'est en fait pas typographiée de façon correcte car l'italique n'a pas lieu d'être. Utiliser le package `csquotes` pour améliorer sa composition.

Solutions des exercices

Solution de l'exercice 2-1 page 51

a Pour obtenir les tailles demandées, il suffit de créer puis de compiler comme d'habitude le fichier source suivant :

```

\documentclass[french,12pt]{article}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{lmodern}
\usepackage[a4paper]{geometry}
\usepackage{babel}
\begin{document}
\huge Un mathématicien est une machine à transformer le {\normalsize
  café} en théorèmes.
\end{document}

```

b En inspectant les menus de son éditeur de texte orienté \LaTeX , on trouve comment lancer une compilation \LaTeX sur le fichier `.tex` précédent. Le fichier visualisable produit, dit *DVI*, est d'extension `.dvi`. L'éditeur fournit un moyen simple de le visualiser.



On constate dans le menu Utils que la compilation \LaTeX se lance au moyen de la touche `F2` et que le visualisateur de fichiers `.dvi` se lance en pressant la touche `F3`.

L'utilisateur ayant à fréquemment contrôler la mise en page de son document pourra préférer recourir à la compilation \LaTeX et au format *DVI*, parfois plus rapides à mettre en œuvre que la compilation $\text{PDF}\LaTeX$ et le format *PDF*¹. Mais le format *DVI* souffre de certains inconvénients : il n'est d'une part pas autonome car il n'embarque ni les images incorporées dans le document, ni les fontes² ; il est d'autre part assez peu propice aux échanges car les visualisateurs *DVI* ne sont pas très répandus.

Cependant, dans certaines situations (en particulier si le document comprend des figures créées avec `PSTricks`, cf. chapitre 9), la compilation \LaTeX représente une alternative nécessaire³. Les fichiers *DVI* alors produits peuvent n'être utilisés que comme intermédiaires : il est en effet possible de les transformer en fichiers *PDF*, en leur faisant subir une première compilation `dvips` qui les transforme en fichiers *PS* (visualisables par exemple avec `Evince`), puis une seconde compilation `ps2pdf` qui les transforme en fichiers *PDF*⁴.



La compilation `dvips`, le visualisateur de fichiers *PS* et la compilation `ps2pdf` se lancent au moyen des touches `F4`, `F5` et `F8` respectivement. On notera que, pour éviter d'avoir à enchaîner ces compilations, on peut recourir à une « compilation rapide » : tout d'abord, on visite le menu `Options` → `Configurer TexMakerX` → bouton `Compil rapide` et on sélectionne le bouton radio `LaTeX + dvips + ps2pdf + Afficheur PDF` ; puis, à loisir, on lance cet enchaînement au moyen de l'unique raccourci clavier `F1`.

Cependant, selon les cas, le choix `LaTeX + dvips + ps2pdf + Afficheur PDF` peut présenter l'inconvénient qu'une nouvelle fenêtre de l'afficheur apparaisse à chaque compilation. Dans ce cas, on utilisera plutôt la compilation rapide « Utilisateur » qu'on définira ainsi :

```
latex %.tex|dvips -o %.ps %.dvi|ps2pdf %.ps
```

En conclusion, que l'on procède à l'une ou l'autre des compilations $\text{PDF}\LaTeX$ ou \LaTeX , il est toujours possible d'obtenir un fichier *PDF* qu'on privilégiera pour les échanges et pour la diffusion (sur son site Web personnel par exemple).

Solution de l'exercice 2-2 page 51

Les pictogrammes de la bicyclette et de la tasse de café sont fournies par le package `marvosym` et celui de l'étoile, par le package `pifont`. La police *Times* s'obtient grâce au package `mathptmx`.

1. Ceci du fait notamment de la légèreté des fichiers `.dvi` et de la rapidité des visualisateurs associés.
2. Un fichier `.dvi` a donc toutes les chances d'être illisible sur un ordinateur où \LaTeX n'est pas installé.
3. En fait, depuis l'apparition de $\text{X}\LaTeX$, ceci n'est plus vraiment le cas (cf. chapitre 15).
4. Ils peuvent aussi subir une transformation directe *DVI* vers *PDF* au moyen d'une compilation `dvipdfm`, moins fiable semble-t-il.

2-25

```
\usepackage{marvosym}\usepackage{pifont}\usepackage{mathptmx}
\Bicycle\ding{72}Passer pour un idiot aux yeux d'un imbécile est
une volupté de fin gourmet.\ding{72}\Coffeecup
```

Le corrigé précédent montre qu'il est possible de faire figurer plusieurs occurrences de la commande `\usepackage` sur une même ligne du préambule. On peut aller plus loin : pour charger plusieurs packages (sans option ou partageant la ou les mêmes options), le recours à une unique occurrence de la commande `\usepackage` est autorisé (bien que parfois moins lisible). On aurait ainsi pu saisir en préambule :

```
\usepackage{marvosym,pifont,mathptmx}
```

Solution de l'exercice 2-3 page 51

a Cet exercice met en évidence les particularités de \LaTeX en matière d'espaces horizontales et verticales vues à la section 2.8.1 page 42. Le fichier source compilé produit le document final suivant, très différent du corps du fichier source.

E-2

```
1-Science sans conscience n'est que ruine de l'âme. 2-Science sans conscience
n'est que ruine de l'âme.
3-Science sans conscience n'est que ruine de l'âme.
4-Science sans conscience n'est que ruine de l'âme. 5-Science sans conscience
n'est que ruine de l'âme. 6-Science sans conscience n'est que ruine de l'âme.
7-Science sans consciencen'est que ruine de l'âme.
```

En effet, dans le fichier `.tex` :

- La ligne 1 occupe deux lignes du document final et on constate le *retrait* figurant au début du paragraphe qu'elle constitue.
- La ligne 2 étant vide, le contenu de la ligne suivante débute un nouveau paragraphe, là aussi avec *retrait* ; on note dans cette ligne 3 que les nombreux espaces consécutifs reviennent à une unique espace dans le document final.
- Les lignes 4 et 5 sont toutes deux vides, ce qui revient à une unique ligne vide, si bien que la ligne 6 entame un nouveau paragraphe.
- Alors qu'on pourrait s'attendre à ce que la ligne 7 commence une nouvelle ligne dans le document final, il n'en est rien : les lignes 6 et 7 n'étant séparées dans le fichier source que par un *unique* retour chariot, elles ne sont séparées dans le document final que par une espace horizontale. De même pour les lignes 7 et 9, séparées par la ligne 8, certes vide mais en commentaire : elles aussi ne sont séparées dans le document final que par une espace horizontale.
- Le principe est le même pour les lignes 11 à 13 du fichier source qui ne produisent qu'une seule ligne dans le document final. Il est néanmoins important de noter qu'il n'y a pas d'espace dans le document final entre « conscience » et « n'est » à la ligne « 7-Science... » ; ceci est dû au caractère % en fin de ligne 12 qui absorbe l'espace que devrait provoquer le retour chariot qui suit.
- Le commentaire en fin de ligne 13 a été ignoré à la compilation.
- La ligne 15, en commentaire, n'apparaît pas dans le document final.

- b** Si le % de la ligne 15 est ôté, le contenu de cette ligne n'est plus un commentaire et est donc pris en compte à la compilation.
- c** Lorsque seulement une ligne ou une partie de ligne doit être commentée, on peut se contenter d'insérer manuellement le caractère spécial %.
- d** En revanche, si tout un bloc de texte doit être commenté ou décommenté, il convient d'utiliser les fonctionnalités de son éditeur de texte orienté L^AT_EX pour ne pas avoir à insérer ou ôter le symbole % au début d'un grand nombre de lignes.

Solution de l'exercice 2-4 page 52

- a** Puisqu'on utilise de la couleur, on charge le package xcolor pour pouvoir faire appel à la commande \color. Il faut noter que la méthode employée dans le corrigé suivant n'est pas celle à utiliser d'ordinaire pour créer un titre de document : nous verrons dans le chapitre suivant que L^AT_EX fournit des outils dédiés à la production de titres. Par ailleurs, l'environnement flushright est utilisé pour aligner à droite le texte, mais la phrase « Fils d'un général... » doit ici débiter un nouveau paragraphe, sans quoi l'alignement ne sera pas celui demandé.

2-26

```
\usepackage{xcolor}
\begin{center}
\color{blue} Une figure de la littérature française: Victor Hugo
\end{center}
\begin{flushright}
  Victor Hugo naquit à Besançon le 26 février 1802.

  Fils d'un général de Napoléon, il suivit d'abord son père
  dans le hasard des expéditions et des campagnes, en Italie,
  en Espagne, où il fut page du roi Joseph et élève au
  séminaire des nobles de Madrid.
\end{flushright}
```

On constate dans cet exemple que la commande \color, à portée semi-globale, voit son effet cesser avec la fin de l'environnement center.

En outre, le mot « figure » apparaît avec la ligature « fi » au lieu de « fi ». Effectivement, contrairement aux traitements de texte, L^AT_EX gère les ligatures esthétiques de la langue française : « ff, fi, fl, ffi, ffl » (au lieu de « ff, fi, fl, ffi, ffl »). Notons aussi que les codes sources -- et --- produisent « - » (à la place de « -- ») et « — » (à la place de « --- »). Il est également possible d'obtenir les ligatures orthographiques « æ, Æ, Œ » soit directement au clavier en cas de codage en UTF-8, soit au moyen des commandes \ae, \oe, \AE et \OE.

E-3

```
Le cœur a ses raisons que la raison } Le cœur a ses raisons que la raison ignore.
ignore.
```

- b** Le package xcolor met à disposition la commande \pagecolor qui permet de colorer le fond de page. Pour obtenir un fond en cyan, il suffit donc de débiter le corps de document par la commande \pagecolor{cyan}.

Solution de l'exercice 2-5 page 52

Nous avons vu que, lorsque leurs noms sont constitués de lettres, les commandes se terminent par un caractère autre qu'une lettre. À la première ligne de ce fichier source, l'espace qui suit immédiatement `\TeX` est utilisé par \LaTeX pour identifier la fin de la commande, et pas pour insérer une espace entre les mots « `\TeX` » et « `et` ». Du coup, cet espace paraît « avalé » lors de la compilation. Le phénomène est identique pour la commande `\LaTeX` de la première ligne. Il se produit en fait avec toutes les commandes \LaTeX qui, à la fois, n'ont pas d'argument et produisent du texte¹. Nous voyons aux lignes suivantes que le problème peut être résolu en insérant une paire d'accolades vide juste après le nom de la commande. Certains éditeurs de texte orientés \LaTeX , Emacs par exemple, assurent automatiquement cette insertion.

L'exercice 3-3 page 92 propose une méthode permettant de s'épargner la saisie parfois pénible de ces paires d'accolades.

Solution de l'exercice 2-6 page 52

a Il suffit de saisir `\fbox{vieillards}`. Le nom de cette commande est une abréviation de *framed box* (boîte encadrée).

b Si ce texte de plus d'une ligne est entièrement encadré, il dépasse dans la marge. Ceci est dû au fait que \LaTeX « encapsule » le texte à encadrer dans une boîte (cf. section C.3) qui ne gère pas les coupures de lignes.

Pour régler le problème, il suffit de procéder à une recherche sur le mot « encadrer » dans la FAQ². Parmi les résultats de la recherche, la question fréquemment posée qui nous intéresse est « Comment encadrer du texte ? ». La première réponse donnée lorsque nous cliquons sur le lien suggère de créer un tableau d'une seule cellule. Puisque nous ne savons pas encore construire de tableau, nous ignorons cette réponse qui n'est d'ailleurs pas la meilleure. La seconde solution propose de recourir à l'environnement `minipage` qui n'a pas encore été étudié ; mais, cette fois, le code source est fourni. En le copiant-collant, nous pouvons donc le tester et l'adapter :

E-4

```

\fbbox{
  \begin{minipage}{0.7\textwidth}
    Les vieillards aiment à donner de bons conseils, pour se
    consoler de n'être plus en âge de donner de mauvais exemples.
  \end{minipage}
}

```

Les vieillards aiment à donner de bons conseils, pour se consoler de n'être plus en âge de donner de mauvais exemples.

Nous avons là agi un peu à l'aveuglette puisque nous ne connaissons ni l'environnement `minipage` (qui permet de mettre en forme un paragraphe d'une largeur donnée,

1. Les commandes sans argument telles que `\slshape`, qui *agissent* sur le texte mais n'en *produisent* pas, ne sont donc pas concernées.

2. <http://www.grappa.univ-lille3.fr/FAQ-LaTeX>

cf. section C.3), ni la commande `\textwidth` (qui est la largeur allouée au texte dans la page). On n'hésitera pas à procéder ainsi avec \LaTeX : si le code testé fonctionne, on peut l'utiliser (en l'adaptant) même sans comprendre toutes les commandes qu'il contient. Avec le temps, bien sûr, il sera profitable de chercher à en percer les mystères... Dans le cas présent, nous pouvons toutefois deviner qu'avant d'encadrer le texte au moyen de la commande `\fbox`, nous l'insérons dans une « minipage » dont la largeur vaut 70% de la largeur normale de texte (`0.7\textwidth`).

Toutefois, cette solution souffre de quelques défauts. Les raisons des correctifs qui figurent dans le source suivant sont expliquées à l'annexe C :

```
\fbox{%
  \begin{minipage}{0.7\textwidth}
    Les vieillards aiment à donner de bons conseils, pour se
    consoler de n'être plus en âge de donner de mauvais exemples.
  \end{minipage}%
}
```

c La FAQ indique la commande `\shadowbox` fournie par le package `fancybox` :

E-5

```
\usepackage{fancybox}
\shadowbox{%
  \begin{minipage}{0.85\linewidth}
    Les vieillards aiment à donner de bons conseils, pour se consoler
    de n'être plus en âge de donner de mauvais exemples.
  \end{minipage}%
}
```

Les vieillards aiment à donner de bons conseils, pour se consoler de n'être plus en âge de donner de mauvais exemples.

Ici, la minipage a pour largeur 85% de `\linewidth`, largeur de la ligne courante. Celle-ci peut en effet être différente de `\textwidth`, par exemple à l'intérieur d'une liste à tirets (cf. section 3.6 page 70). On a d'ordinaire intérêt à se fonder sur la largeur de la ligne courante plutôt que sur la largeur normale du texte.

Solution de l'exercice 2-7 page 52

Une recherche sur le mot ¹ « listings », parmi les descriptions de packages (*Search the package descriptions*) sur le site du CTAN², conduit à plusieurs packages. Seuls trois d'entre eux, `listing`, `listings` et `tgrind`, semblent correspondre à ce que nous cherchons.

Le package `tgrind` est adapté à plain \TeX qui est un *format*³ plus ancien que \LaTeX et que nous n'étudierons pas. Si nous cliquons sur le lien de `listing` (sans « s »), la page d'information de ce package apparaît ; nous voyons que sa documentation (*Documentation on CTAN*) est disponible. Un clic sur le lien « *Package documentation* » provoque

1. Cette recherche doit porter sur des termes anglais.
2. <http://www.ctan.org/search.html#byDescription>
3. Dans cet ouvrage d'initiation, nous ne pouvons développer la question des formats.

le téléchargement de la documentation¹, fournie ici au format *PDF*. Nous constatons que ce document est très court et manifestement peu informatif.

Nous allons donc plutôt nous intéresser au package `listings` dont on peut lire la documentation en procédant comme nous l'avons fait pour `listing`. Il faut se rendre à l'évidence : le document est en anglais ! Mais, même sans maîtriser cette langue, il est possible de parfaitement comprendre les grandes lignes de l'utilisation du package en survolant le texte. Par exemple, aux pages 3 et 4, on trouve de quoi constituer par simples copier-coller l'exemple minimal suivant.

E-6	<pre> \usepackage{listings} \lstset{language=Pascal} % Insert Pascal examples here. \begin{lstlisting} for i:=maxint to 0 do begin { do nothing } end; Write('Case insensitive '); WritE('Pascal keywords. '); \end{lstlisting} </pre>	<pre> for i:=maxint to 0 do begin { do nothing } end; Write('Case insensitive '); WritE('Pascal keywords. '); </pre>
-----	--	--

Nous constatons que, sans que nous ayons eu à recourir à quelque commande que ce soit, le package `listings` a assuré la mise en forme (notamment la mise en relief des mots clés) du listing informatique de ce programme en langage Pascal.

Ce package, extrêmement souple et puissant, pourra être très utile aux informaticiens, étudiants et enseignants en informatique ou en mathématiques. Ceux-ci pourront, sans effort, obtenir des listings impeccablement présentés pour de nombreux langages informatiques (cf. page 12 de la documentation). Il leur sera même possible de le configurer de façon qu'il reconnaisse de nouveaux langages, par exemple celui qu'ils sont en train de développer... L'usage de `listings` est détaillé au chapitre 11.

Solution de l'exercice E-2-8 page 405

La compilation annonce parfois un très grand nombre d'erreurs, ce qui a de quoi décourager. En fait, certaines erreurs en provoquent une multitude d'autres, et il arrive qu'en solutionnant la première signalée par \LaTeX , on règle toutes les suivantes. C'est pourquoi on a suggéré de recompiler après chaque problème résolu.

Il y a ici six erreurs successives :

1. Le package `geomatry`, ligne 5, n'existe pas : il faut corriger en remplaçant `geomatry` par `geometry`.
2. La ligne 7 comporte du texte avant le début du document. Non seulement ce texte ne sera pas pris en compte mais en plus il perturbe \LaTeX !
3. La commande `\textbf`, ligne 9, est inconnue de \LaTeX : en rectifiant avec `\textbf`, cette erreur est corrigée.

1. Cette documentation peut également être consultée au moyen des programmes `texdoctk` et `texdoc`, comme expliqué page 50.

4. L'environnement `center`, ouvert par `\begin{center}` à la ligne 16, est fermé par `\end{tiny}` à la ligne 23 : remplacer `tiny` par `center` (ou *vice versa*) suffit.
5. L'accolade fermante en fin de ligne 28 n'est pas précédée d'une accolade ouvrante : il suffit de la supprimer.
6. L'environnement `flushleft`, ouvert par `\begin{flushleft}` à la ligne 29, n'est pas fermé par une commande `\end{flushleft}`. Il suffit donc d'ajouter cette commande après `\begin{flushleft}` et, bien sûr, avant `\end{document}`.

Notons au passage la commande `\dots` permettant d'obtenir des points de suspension. Maintenant que tous les problèmes ont été réglés, il ne faut pas oublier de tirer toute la quintessence du texte ci-dessus, œuvre d'Éric THIÉMARD.

Solution de l'exercice E-2-9 page 406

Les erreurs précédentes étaient assez faciles à lever, les indications fournies par \LaTeX étant somme toute assez claires. Ici, le message est un peu plus abscons. Un examen attentif de celui-ci et du code source montre que la commande `\textbf` ouverte par une accolade ouvrante à la ligne 8 n'a pas été fermée par une accolade fermante. Il suffit donc d'ajouter cette accolade, où *on veut* après `\textbf{` (et, bien sûr, avant `\end{document}`). Rappelons à cette occasion que ce type d'erreur peut être évité en recourant aux fonctionnalités d'un éditeur de texte orienté \LaTeX . Apprendre à bien maîtriser un tel outil est essentiel dans l'usage de \LaTeX et permet, à court terme, un gain significatif en temps et en efficacité.

Solution de l'exercice E-2-10 page 406

a Pour gagner en efficacité, on utilise la commande `\emph` de façon que, si toute cette phrase devait elle-même figurer en italique, la citation apparaisse en *romain*. On prend soin de faire figurer les guillemets eux aussi en italique.

E-1 `Jean \bnc{de la Bruyère} disait: \emph{\og \emph{L'esclave} n'a qu'un maître; \emph{l'ambitieux} en a autant qu'il y a de gens utiles à sa fortune.\fg{}}`

b Les auteurs de travaux littéraires, souvent amenés à insérer des citations, pourront utiliser avec profit le package `csquotes` : celui-ci permet d'éviter ce type d'erreurs en déléguant à \LaTeX les questions de conformité typographique. L'exemple suivant, où le soulignement emphatique indésirable de la première question a été occulté, donne un aperçu de son usage : tout d'abord, on charge ce package après le package `babel` et avec l'option `babel` pour que les règles typographiques appliquées soient celles de la langue courante. Ensuite, la commande `\textquote` fournie par `csquotes` permet de mettre en forme les citations au fil du texte, en les ponctuant de façon *ad hoc*. Les citations de second rang sont composées en emboîtant les commandes `\textquote`.

E-7	<pre>\usepackage[babel]{csquotes} Jean \bsc{de la Bruyère} disait: \textquote{L'esclave n'a qu'un maître; l'ambitieux en a autant qu'il y a de gens utiles à sa fortune.}. Il s'amuse également à affirmer: \textquote{\textquote{Diseurs de bons mots, mauvais caractère}: je le dirais, s'il n'avait été dit. Ceux qui nuisent à la réputation ou à la fortune des autres, plutôt que de perdre un bon mot, méritent une peine infamante; cela n'a pas été dit, et je l'ose dire.}}</pre>	<p>Jean DE LA BRUYÈRE disait : « L'esclave n'a qu'un maître ; l'ambitieux en a autant qu'il y a de gens utiles à sa fortune. ».</p> <p>Il s'amuse également à affirmer : « "Diseurs de bons mots, mauvais caractère" : je le dirais, s'il n'avait été dit. Ceux qui nuisent à la réputation ou à la fortune des autres, plutôt que de perdre un bon mot, méritent une peine infamante ; cela n'a pas été dit, et je l'ose dire. »</p>
-----	---	---

Dans cet exemple, on pourra remplacer les deux premières commandes `\textquote` par la commande `\blockquote`¹ qui, si la citation ne dépasse pas trois² lignes, la compose elle aussi au fil du texte mais qui, dans le cas contraire, la présente automatiquement sous forme de bloc.

1. Et pas la troisième : il n'est pas possible d'emboîter deux commandes `\blockquote`.
2. Ce seuil est bien sûr paramétrable.

Chapitre 3

Exercices supplémentaires

Exercice E-3-4 Hyperliens

Cet exercice utilise le fichier source finalement obtenu à l'exercice 3-1 page 91.

- a** Charger le package `hyperref` et, en créant une version *DVI* et une version *PDF* du document (cf. exercice 2-1 page 51), tester avec les visualisateurs respectifs tous les liens hypertexte générés.
- b** Effectuer une recherche sur le mot « *URL* » dans la documentation du package `hyperref` pour apprendre à créer des hyperliens vers des adresses Web, puis faire en sorte que l'*URL* de l'organisme dont relève l'auteur Agathe ZEBLUES soit un hyperlien.
- c** Le package `nameref` (chargé automatiquement par `hyperref`) fournit la commande éponyme `\nameref` permettant de faire référence à une unité de sectionnement par son intitulé plutôt que par son numéro ou sa page. Tirer parti de ce package pour obtenir une version plus pérenne de l'exemple 3-14 page 66.

Exercice E-3-5 Macro personnelle avec argument

En ne faisant apparaître, dans le fichier source, qu'une seule fois la structure :

```
Ils étaient ... dans le nid
Et le petit dit : « Poussez-vous, poussez-vous ! »
Et l'un d'eux tomba du nid.
```

faire en sorte que, après compilation, apparaisse dans la page le texte suivant :

E-8	<pre>Ils étaient cinq dans le nid Et le petit dit : « Poussez-vous, poussez-vous ! » Et l'un d'eux tomba du nid. Ils étaient quatre dans le nid Et le petit dit : « Poussez-vous, poussez-vous ! » Et l'un d'eux tomba du nid. Ils étaient trois dans le nid Et le petit dit : « Poussez-vous, poussez-vous ! » Et l'un d'eux tomba du nid. Ils étaient deux dans le nid Et le petit dit : « Poussez-vous, poussez-vous ! » Et l'un d'eux tomba du nid. Et le petit dit : « Aaaaah ! »</pre>
-----	--

Exercice E-3-6 Séparation fond-forme

Le source suivant est un exemple typique de ce que peut produire un débutant ! Comment rendre le source plus correct ?

Dans son ouvrage `\textit{L'art poétique}`,
Nicolas `\textsc{Boileau}`...

Exercice E-3-7 Ajustement de l'interlignage

Chercher dans les différentes documentations \LaTeX comment obtenir le document suivant (de largeur de texte 200pt) où la première phrase est en interligne double, la deuxième en interligne de facteur 1,25, la troisième à nouveau en interligne double et la dernière en interligne simple.

E-9

\LaTeX présente des avantages (il en a beaucoup) et des inconvénients (il en a peu).

\LaTeX présente des avantages (il en a beaucoup) et des inconvénients (il en a peu).

\LaTeX présente des avantages (il en a beaucoup) et des inconvénients (il en a peu).

\LaTeX présente des avantages (il en a beaucoup) et des inconvénients (il en a peu).

Solutions des exercices

Solution de l'exercice 3-1 page 91

a Il suffit de créer le document suivant.

3-36

```
\tableofcontents
\section{AAAA}
\subsection{aaaa}
\subsection{aaaa}
\subsection{aaaa}
\subsection{aaaa}
\section{BBBB}
\section{CCCC}
\subsection{cccc}
\subsection{cccc}
```

b Pour créer le titre demandé, il suffit de placer, en préambule :

```
\title{Document de test des outils fondamentaux de \LaTeX{}}
\author{Agathe \bsc{Zablues}\thanks{Association GUTenberg
(http://www.gutenberg.eu.org)}}
```

et de faire figurer la commande `\maketitle` juste après `\begin{document}`. En particulier, l'absence de commande `\date` assure que le titre comportera la date de la compilation. Cependant, les *URL* ne doivent en général pas être insérées telles quelles et une méthode pour les composer correctement est indiquée à l'exercice E-3-4 page 415.

c Une recherche dans la *FAQ* L^AT_EX francophone sur le mot « résumé » conduit à la question « Comment mettre un résumé et un abstract dans un document ? », dont la réponse montre que l'environnement `abstract` a été prévu pour créer des résumés. Il suffit donc d'insérer, juste après la commande `\maketitle` :

```
\begin{abstract}
  Ce document sert de test aux outils fondamentaux de \LaTeX{}.
\end{abstract}
```

Par ailleurs, nous avons vu page 64 comment redéfinir la commande `\contentsname` pour modifier l'intitulé de la table des matières. Pour que l'intitulé « Résumé » soit remplacé par « En bref », il suffit de procéder de façon analogue et donc d'insérer, en préambule, la ligne :

```
\addto\captionsfrench{\def\abstractname{En bref}}
```

d La commande permettant d'afficher en caractères latins majuscules la valeur d'un compteur est `\Alph{<compteur>}` et le compteur attaché à la commande `\section` est `section`. Avec ce que nous connaissons en plus du package `sectsty` (cf. page 61), il suffit donc de saisir, en préambule, outre les packages requis :

```
\renewcommand\thesection{\Alph{section}}
\sectionfont{\itshape\color{red}}
```

e Il suffit « d'étiqueter » la troisième section en ajoutant, immédiatement après la commande `\section{CCCC}`, la commande `\label{sec-CCCC}`¹, puis d'ajouter, au début de la deuxième section :

La `section~\ref{sec-CCCC}` page~`\pageref{sec-CCCC}` est passionnante!

ce qui compose en effet automatiquement « La section C page 1 est passionnante ! ».

f Pour insérer la note de bas de page, il suffit de remplacer, dans le résumé :

... outils fondamentaux...

par :

... outils fondamentaux\footnote{Bien que fondamentaux, ces outils sont faciles à mettre en œuvre.}...

1. Le préfixe « sec- » n'est pas obligatoire mais peut aider l'utilisateur à se remémorer que c'est à une section que se rapporte cette clé.

Nous remarquons que l'insertion de cette note a provoqué un changement de page juste avant la section « CCCC », si bien que la référence est restée sur la page 1 alors que l'objet référencé (la section « CCCC ») est passé à la page 2. À la première compilation, L^AT_EX n'a pas encore pris la mesure de ce changement et compose encore « La section C page 1 est passionnante ! » ; après la seconde compilation, toutes les références croisées sont à jour et il affiche cette fois ce qu'on attend de lui, à savoir : « La section C page 2 est passionnante ! »

g Les références croisées à la solution de la question e page précédente ne produisent pas un résultat très heureux : le numéro de page est précisé dans tous les cas, même lorsque la page référencée est justement celle qu'on est en train de lire (cas où cette précision n'est pas utile)¹. Le package `varioref` améliore la situation car l'insertion de :

```
La section\href{sec-CCCC} est passionnante!
```

affiche seulement « La section C est passionnante ! » (ou « La section C de la présente page est passionnante ! ») lorsque la page référencée et celle qui est lue sont les mêmes, et précise la page sinon : « La section C page ... est passionnante ! »²

h Un examen rapide de la documentation du package `datetime`³ permet de découvrir, parmi les commandes proposées, la commande `\currenttime` qui compose l'horaire courant ou, plus précisément, l'horaire de compilation. Nous savons par ailleurs que, pour personnaliser les titres courants, le plus simple est de recourir au package `fancyhdr` et d'en utiliser le style `fancy` qui fournit la commande `\lfoot` pour le pied de page gauche. Pour répondre à la question, il suffit donc d'insérer en préambule, outre les packages requis :

```
1 | \pagestyle{fancy}
2 | \lfoot{\currenttime}
```

Mais nous notons alors que le séparateur heures/minutes composé est « : » et non, comme c'est l'usage en France, « h » (entouré d'espaces inter-mot *espaces insécables*). En replongeant dans la documentation du package `datetime`, nous constatons que ce dernier a été bien conçu : il offre en effet la commande `\timeseparator` permettant de personnaliser ce séparateur. Ainsi, il nous suffit d'insérer en préambule, avant la ligne 2 du listing précédent, la ligne :

```
\renewcommand{\timeseparator}{~h~}
```

Pour que la date dans le titre du document soit suivie de l'heure et de la minute de compilation, il faut l'explicitier au moyen de la commande :

```
\date{le \today{} à \currenttime}
```

Pour résumer, le fichier complet que nous devrions obtenir est le suivant.

1. C'était le cas ici avant l'insertion de la note de bas de page.
 2. Après l'insertion de la note de bas de page, on obtient en l'occurrence : « La section C page précédente est passionnante ! »
 3. Ce package est à charger après le package `babel` pour que certaines traductions soient assurées.

```

\documentclass[french]{article}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{lmodern}
\usepackage{xcolor,sectsty,fancyhdr}
\usepackage[a4paper]{geometry}
\usepackage{babel}
\usepackage{datetime,varioref}
% Commandes de personnalisation
\renewcommand\thesection{\Alph{section}}
\sectionfont{\itshape\color{red}}
\renewcommand{\timeseparator}{~h~}
\pagestyle{fancy}
\lfoot{\currenttime}
\addtocaptionsfrench{\def\abstractname{En bref}}
% Titre
\title{Document de test des outils fondamentaux de \LaTeX{}}
\author{Agathe \bsc{Zelues}\thanks{Association GUTenberg
(http://www.gutenberg.eu.org)}}
\date{le \today{} à \currenttime}
% Début du document
\begin{document}
\maketitle
\begin{abstract}
Ce document sert de test aux outils fondamentaux\footnote{Bien que
fondamentaux, ces outils sont faciles à mettre en œuvre.} de
\LaTeX{}.
\end{abstract}
\tableofcontents
\section{AAAA}
\subsection{aaaa}
\subsection{aaaa}
\subsection{aaaa}
\subsection{aaaa}
\section{BBBB}
La section\vref{sec-CCCC} est passionnante!
\section{CCCC}\label{sec-CCCC}
\subsection{cccc}
\subsection{cccc}
\end{document}

```

Les commentaires qui se trouvent dans ce fichier source sont bien sûr facultatifs mais ils en facilitent la (re)lecture. Nous conseillons au lecteur de procéder de même pour ses propres documents.

Solution de l'exercice 3-2 page 92

a La classe `report`, qui est censée contenir des chapitres, prépare en interne le compteur `chapter` des chapitres en l'initialisant à zéro, en attendant qu'il soit incrémenté de 1 dès la première commande `\chapter` rencontrée. Par ailleurs, avec cette classe, les sections sont par défaut numérotées avec un chiffre arabe précédé du numéro

du chapitre courant, les deux étant séparés d'un point. Puisque aucune commande `\chapter` ne figure dans le fichier source, le chapitre sous-jacent est de numéro zéro, si bien que, par exemple, la section intitulée AAAA est numérotée « 0.1 » et ses sous-sections sont numérotées « 0.1.1 » et « 0.1.2 ».

b Il suffit d'utiliser les fonctionnalités de l'éditeur de texte pour remplacer d'un seul coup toutes les commandes `\section` par des commandes `\chapter`, puis toutes les commandes `\subsection` par des commandes `\section`. Dans un cas comme celui-là, on prendra garde à procéder dans cet ordre sans quoi, à la fin de la première étape, les `\subsection` et les `\section` seront mélangées.

Solution de l'exercice 3-3 page 92

a Il suffit de définir une macro sans argument dont la valeur est le texte en question et d'insérer quatre fois cette commande dans le corps du document :

E-10

```
\newcommand{\MonTexte}{\textbf{Les} caprices de votre
traitement de texte vous exaspèrent, vous n'en pouvez
plus de la lourdeur de votre éditeur d'équations, vos
documents sont d'une qualité typographique douteuse
et vous désespérez de voir votre mise en page évoluer
au gré d'un changement de machine ou d'un numéro de
version\dots{}}

```

```
\MonTexte\MonTexte\MonTexte\MonTexte
```

b Le package `xspace` fournit la commande `\xspace` qu'on peut ajouter à la fin de la définition de macros personnelles générant du texte. Alors, lorsque ces macros ne seront pas immédiatement suivies, dans le fichier source, d'un signe de ponctuation ou d'un changement de ligne, le texte qu'elles produiront sera automatiquement suivi d'une espace. Donc, au lieu de coder `\TeX{}` et `\LaTeX{}` comme à l'exercice 2-5 page 52, nous pouvons définir des macros personnelles comme suit :

E-11

```
\usepackage{xspace}
\newcommand{\tex}{\TeX\xspace}
\newcommand{\latex}{\LaTeX\xspace}
\tex et \latex et \tex, \latex.

```

```
\TeX et \LaTeX et \TeX, \LaTeX.

```

De même :

E-12

```
\usepackage{xspace}
\newcommand{\adn}{acide
désoxyribonucléique\xspace}
L'\adn (souvent abrégé en ADN)...
L'\adn est par exemple...
Pour fabriquer un brin d'\adn,
il suffit...

```

```
L'acide désoxyribonucléique (sou-
vent abrégé en ADN)... L'acide désoxy-
ribonucléique est par exemple... Pour
fabriquer un brin d'acide désoxyribo-
nucléique, il suffit...

```

Notons que le package `xspace` permet également de ne pas avoir à saisir les paires d'accolades vides après les commandes du package `frenchb` : `\ier`, `\ier`, `\iere`, `\iers`, `\ieres`, `\ieme`, `\iemes` et `\fg` (cf. section 2.6 page 37).

Solution de l'exercice E-3-4 page 415

a Nous constatons que le package `hyperref` offre une fonctionnalité extrêmement pratique : les entrées de la table des matières, de même que les objets référencés et les notes de bas de page, sont devenus des hyperliens.

b Pour afficher une *URL* et faire d'elle un hyperlien, `hyperref` fournit la commande :

```
\url{<adresse de page Web>}
```

Pour répondre à la question posée, il suffit donc de saisir

```
\author{Agathe \bsc{Zablues}\thanks{Association GUTenberg
(\url{http://www.gutenberg.eu.org})}}
```

Il peut être plus élégant de ne pas faire figurer explicitement l'*URL* mais d'en faire la « cible » d'un ou de quelques mots.

```
\href{<adresse de page Web>}{<(quelques) mot(s)>}
```

Ainsi, nous aurions pu saisir plutôt :

```
\author{Agathe \bsc{Zablues}
\thanks{\href{http://www.gutenberg.eu.org}{Association GUTenberg}}}
```

Ceci fait de « Association GUTenberg » un hyperlien : lorsqu'on clique dessus, le navigateur Web par défaut est lancé directement à la page <http://www.gutenberg.eu.org>. Notons qu'il est possible, dans la syntaxe ci-dessus, de remplacer *<adresse de page Web>* par *<adresse de document PDF>* où le fichier *PDF* en question peut se trouver aussi bien sur un site distant que sur l'ordinateur de celui qui lit le document. Si le fichier `document-passionnant.pdf` se situe dans le répertoire parent du document actuellement lu, le source `.tex` peut ainsi contenir :

Il faut lire le `\href{../document-passionnant.pdf}{document passionnant}`.

et un clic sur l'hyperlien « document passionnant » lance le visualisateur de fichiers *PDF* et y ouvre automatiquement le fichier `document-passionnant.pdf`.

Signalons également la syntaxe :

```
\href{mailto:<adresse courriel>}{<texte>}
```

qui fait de *<texte>* un hyperlien permettant d'ouvrir une fenêtre de composition de message du courriel où l'*<adresse courriel>* est déjà remplie. Par exemple, notre commande `\author` pourrait être complétée en :

```
\author{\href{mailto:Agathe.Zeblues@fai.fr}{Agathe \bsc{Zeblues}}
\thanks{\href{http://www.gutenberg.eu.org}{Association GUTenberg}}}
```

Pour clore ce sujet, indiquons la possibilité de lancer n'importe quel *<programme>* au moyen de la syntaxe :

```
\href{run:<programme>}{<texte>}
```

La commande suivante permet ainsi, en cliquant sur l'hyperlien « Lancer TexMakerX », de lancer TexMakerX directement ouvert sur un certain fichier.tex :

```
\href{run:texmakerx fichier.tex}{Lancer TexMakerX}
```

Il est important de noter ici un problème de sécurité, indépendant de \LaTeX , inhérent aux liens hypertexte : ceux-ci peuvent exécuter un programme arbitraire. On prendra donc soin, avant de cliquer sur un hyperlien d'un fichier *PDF* dont on n'est pas l'auteur, de vérifier l'action qui lui est liée : celle-ci est affichée, souvent sous forme d'info-bulle, lorsque le pointeur de la souris est positionné sur le lien ¹.

c Dans l'exemple 3-14 page 66, saisir manuellement l'intitulé de la section en référence croisée peut conduire à un document non consistant : si on modifie l'intitulé dans la commande de sectionnement, on risque d'oublier de le faire également dans les références. La commande `\nameref` permet d'éviter ce problème :

E-13

```
\usepackage{babel}\usepackage{varioref}\usepackage{hyperref}
...
... La section \og \nameref{parite}\fg{}\vpageref{parite}...
...
\subsubsection{Parité}\label{parite}
...
```

En compilant le code ci-dessus, nous observons par ailleurs que l'intitulé composé par `\nameref` est lui aussi un hyperlien.

Solution de l'exercice E-3-5 page 415

Il suffit de définir une macro reproduisant le texte statique et pourvue d'un paramètre dont la valeur est le nombre variable (ici d'oiseaux dans le nid !).

Notons au passage, dans le corrigé suivant, comment l'environnement `verse` permet de mettre en page un poème : les vers sont séparés par des doubles *contre-obliques* et les strophes introduites par la commande `\par` (ou par une ligne vide), c'est-à-dire par des changements de paragraphe. Ces derniers peuvent d'ailleurs figurer dans la définition même de la commande. Au prix d'une certaine complexité, le package `poemscol` permet un contrôle bien plus fin de la composition de poèmes et même de recueils de poésie.

1. Certains visualisateurs *PDF* demandent confirmation avant l'exécution d'un programme.

```

E-8 \newcommand{\tutu}[1]{%
    Ils étaient #1 dans le nid\\
    Et le petit dit: \og Poussez-vous, poussez-vous!\fg{}\\
    Et l'un d'eux tomba du nid.\par
}
\begin{verse}
\tutu{cinq}
\tutu{quatre}
\tutu{trois}
\tutu{deux}
\par
Et le petit dit: \og Aaaaah!\fg{}
\end{verse}

```

Solution de l'exercice E-3-6 page 416

La présence de deux commandes d'indications typographiques doit éveiller les soupçons. Il faut penser en termes sémantiques plutôt qu'en termes typographiques.

Utiliser la commande `\emph`, au lieu de `\textit`, relèverait déjà d'une meilleure habitude, d'abord parce que ce serait un premier pas vers une séparation fond-forme : on indiquerait ainsi non pas qu'on veut composer en italique (ordre typographique), mais qu'on veut mettre en évidence (ordre sémantique). Ensuite, si `\emph` compose par défaut en italique, rien n'empêche de modifier ce comportement et c'est d'ailleurs ce que font certains packages ; par exemple, `ulem` redéfinit la commande `\emph` de façon qu'elle mette en évidence en soulignant au moyen d'un trait¹.

Puisque le terme mis en évidence est un titre de livre, nous pouvons pousser plus loin la séparation fond-forme, en prévoyant une commande `\TitreLivre` (par exemple) qui fera le travail.

```
\newcommand\TitreLivre[1]{\emph{#1}}
```

D'autre part, la composition du nom « Nicolas BOILEAU » est dangereuse puisque rien n'interdit les divisions sur le prénom ou le nom, ce que l'on cherche à éviter normalement. Là aussi, il est préférable de prévoir une commande `\Nom` pour écrire les noms complets de personnes. Le source deviendrait alors :

```
Dans l'ouvrage \TitreLivre{L'art poétique},
\Nom{Nicolas}{Boileau} ...
```

où la commande `\Nom` est définie par l'utilisateur, par exemple par :

```
\newcommand\Nom[2]{\mbox{#1} \bsc{#2}}
```

Si notre document comporte une bibliographie et l'ouvrage en question y est cité, nous verrons au chapitre 12 page 277 une façon beaucoup plus efficace de faire figurer son titre et le nom de son ou ses auteurs.

1. Rappelons que souligner ainsi est en général déconseillé.

Solution de l'exercice E-3-7 page 416

Nous savons qu'un document de largeur de texte 200pt s'obtient en chargeant le package `geometry` avec l'option `textwidth=200pt`. En ce qui concerne l'interlignage, ici encore, la FAQ francophone fournit très rapidement la solution. Elle indique en effet, à la question « Comment modifier l'interligne d'un document ? », que le package `setspace` est notre ami¹. Pour obtenir (semi-)globalement des interlignes un et demi, double et d'un certain $\langle \text{facteur} \rangle$, celui-ci propose les commandes respectivement :

```
\onehalfspacing
\doublespacing
\setstretch{baselinestretch}{\langle \text{facteur} \rangle}
```

Pour un changement local, il fournit les environnements correspondants, respectivement `onehalfspace`, `doublespace` et `spacing`, ce dernier s'utilisant avec la syntaxe :

```
\begin{spacing}{\langle \text{facteur} \rangle}
...
\end{spacing}
```

Pour revenir semi-globalement ou localement à un interligne normal, on dispose respectivement de la commande `\singlespacing` ou de l'environnement `singlespace`.

Dans le corps de document suivant, il est nécessaire de changer de paragraphe avant de repasser en interligne simple ligne 10, sans quoi c'est *tout* le paragraphe contenant la commande `\singlespacing` (donc les deux dernières phrases, lignes 8 à 12) qui en subit l'effet. Une telle précaution n'est pas nécessaire entre les lignes 2 et 3 car l'environnement `spacing` provoque déjà un changement de paragraphe.

```
E-9 1 \usepackage[a4paper, textwidth=200pt]{geometry}
2 \usepackage{setspace}
3 \doublespacing % Par défaut, double interligne
4 \LaTeX{} présente des avantages (il en a beaucoup) et
5 des inconvénients (il en a peu).
6 \begin{spacing}{1.25} % Début interligne 1,25
7 \LaTeX{} présente des avantages (il en a beaucoup) et
8 des inconvénients (il en a peu).
9 \end{spacing} % Fin interligne 1,25
10 % À nouveau, double interligne
11 \LaTeX{} présente des avantages (il en a beaucoup) et
12 des inconvénients (il en a peu).\par
13 \singlespacing % Désormais, interligne simple
14 \LaTeX{} présente des avantages (il en a beaucoup) et
15 des inconvénients (il en a peu).
```

1. Ceci est l'expression de prédilection des informaticiens pour indiquer un outil adéquat.

Chapitre 4

Exercices supplémentaires

Exercice E-4-8 Alignements dans les matrices

Obtenir la matrice suivante :

$$\boxed{\text{E-14}} \left[\begin{array}{cc} 17 & 4 \\ 5 & -2 \end{array} \right]$$

puis consulter la documentation du package `mathtools` pour obtenir celles-ci :

$$\boxed{\text{E-15}} \left[\Delta = \begin{bmatrix} 17 & 4 \\ 5 & -2 \end{bmatrix} = \begin{Bmatrix} 17 & 4 \\ 5 & -2 \end{Bmatrix} \right]$$

Exercice E-4-9 Matrices bordées

a La commande `\bordermatrix`, étudiée page 102, ne correspond pas aux usages en vigueur en France où les vecteurs sont indiqués à droite plutôt qu'à gauche de la matrice. Rechercher, dans les archives du forum `fctt`, une contribution d'un des auteurs du présent ouvrage permettant d'obtenir :

$$\boxed{\text{E-16}} \left[\begin{array}{cc} f(e_1) & f(e_2) \\ \left(\begin{array}{cc} 1 & 2 \\ 0 & 4 \end{array} \right) & \begin{array}{l} e_1 \\ e_2 \end{array} \end{array} \right]$$

b Quelle commande permet d'aérer les lignes de cette matrice ?

Exercice E-4-10 Polynômes et dérivées

a Chercher sur le *CTAN* la façon la plus efficace de composer les polynômes :

$$\boxed{\text{E-17}} \left[\begin{array}{l} 5 - 3x + 2x^2, 5 - 3t + 2t^2, 5 - 3x^2 + 2x^4, 5x^3 - 3x^4 + 2x^5, 5x^2 - 3x + 2, \\ 5 - 3x^{-1} + 2x^{-2}. \end{array} \right]$$

b Même question pour les dérivées :

E-18

$$\frac{df}{dx} \quad \left(\frac{df}{dx}\right)_{x_0} \quad \frac{d^3f}{dx^3} \quad \left(\frac{d^3f}{dx^3}\right)_{x_0}$$

$$\frac{\partial p}{\partial V} \quad \left(\frac{\partial p}{\partial V}\right)_T \quad \frac{\partial^3 p}{\partial V^3} = \frac{\partial^3 p}{\partial V^3} \quad \left(\frac{\partial^3 p}{\partial V^3}\right)_T$$

$$\frac{\partial^2 f}{\partial x \partial y} \quad \left(\frac{\partial^2 f}{\partial x \partial y}\right)_z \quad \frac{\partial^3 f}{\partial x^2 \partial y} \quad \left(\frac{\partial^3 f}{\partial x^2 \partial y}\right)_z$$

Solutions des exercices

Solution de l'exercice 4-1 page 126

a Ce texte ne pose pas de difficulté particulière.

4-66

```
\usepackage{mathtools,amssymb}
```

Pour tout réel x , on appelle partie entière de x et on note $E(x)$ l'entier relatif n tel que $n \leq x < n+1$.

```
\begin{enumerate}
```

```
\item Prouver que, pour tout réel  $x$ ,
```

```
\[E(x+1)=E(x)+1\]
```

```
\item On pose
```

```
\begin{equation}\label{part-ent}
```

```
A(x)=E(x+\frac{1}{2})
```

```
\end{equation}
```

Vérifier que l'ensemble de définition de la fonction définie par $\eqref{part-ent}$ est \mathbf{R} et que

```
\[\forall x \in \mathbf{R}, \quad A(x)=E(2x)-E(x)\]
```

```
\end{enumerate}
```

On aura noté, dans ce corrigé, l'usage de la commande `\eqref`.

b Par défaut, les parenthèses n'adaptent pas leur taille à ce qu'elles délimitent. On recourt aux commandes `\left` et `\right` pour régler le problème :

E-19

```
\[A(x)=E\left(x+\frac{1}{2}\right)\]
```

$$A(x) = E\left(x + \frac{1}{2}\right)$$

c Lorsqu'on utilise souvent des notations susceptibles de modifications ultérieures, on a intérêt à créer des commandes personnelles. Ici, on déclare une macro à argument, de préférence en préambule du document :

E-20

```
\newcommand\entiere[1]{E(#1)}
```

```
\[A(x)=\entiere{x+\frac{1}{2}}\]
```

$$A(x) = E\left(x + \frac{1}{2}\right)$$

Ainsi, la modification de notation (et, d'une pierre deux coups, l'adaptation de la taille des symboles) n'a à être effectuée qu'à un seul endroit du fichier source :

E-21

```
\newcommand\entiere[1]{\left\lfloor#1\right\rfloor}
```

```
\[A(x)=\entiere{x+\frac{1}{2}}\]
```

$$A(x) = \left\lfloor x + \frac{1}{2} \right\rfloor$$

Solution de l'exercice 4-2 page 126

Si on omet les accolades lors de la mise en indice (ou exposant) d'un groupe d'objets, seul le premier de ces objets est affecté :

E-22

```
\$u_n + u_{n+1}\neq u_{n}+u_{n+1}\$
```

$$u_n + u_{n+1} + 1 \neq u_n + u_{n+1}$$

De manière générale avec L^AT_EX, si l'action d'une commande, trop longue ou trop courte, n'est pas celle attendue, il y a moyen d'en délimiter la portée en créant un « groupe » au moyen d'une paire d'accolades.

Solution de l'exercice 4-3 page 126

a Dans le corrigé suivant, on note la définition (à placer de préférence en préambule du document) d'un nouvel opérateur et l'utilisation de la commande `\vv` du package `esvect` pour obtenir la flèche du vecteur :

4-67

```
\usepackage{mathtools,amssymb}\usepackage[e]{esvect}
\DeclareMathOperator{\grad}{grad}
\newcommand{\vecgrad}{\vv{\grad}}
```

On note $\vecgrad f$ le gradient de f . Prouver que

```
\[\vecgrad f=\begin{pmatrix}\frac{\partial f}{\partial x_1}\\\
\vdots\end{pmatrix}\frac{\partial f}{\partial x_n}\end{pmatrix}\]
```

b Il suffit de procéder à une recherche sur le mot « encadrer » dans l'index de la FAQ. Le premier résultat, « Comment encadrer des formules ? », est la question qui nous

intéresse ; la première réponse donnée lorsqu'on clique sur le lien propose la commande `\boxed` qui convient ici. Pour obtenir un mode mathématique hors texte, cette question propose la commande `$$...$$` mais il est important de noter que, comme nous l'avons déjà signalé, l'usage de celle-ci est à proscrire absolument avec \LaTeX .

Solution de l'exercice 4-4 page 127

Il fallait ici remarquer que les indices « eff » et « m » étaient à composer en *romain*. Dans le corrigé suivant, c'est avec la commande `\text` et non avec `\mathrm` qu'ils ont été obtenus car ils ont ici un rôle *textuel* : « eff » et « m » sont les abréviations des mots « efficace » et « moyenne ». Plus généralement, lorsque les indices et exposants ont un rôle descriptif, notamment lorsque ce sont des abréviations de substantifs, ils doivent être saisis en tant que texte, donc au moyen de la macro `\text`.

On a par contre utilisé la commande `\mathrm` pour obtenir « dt » plutôt que « *dt* », car « d » est ici un objet mathématique qu'on a choisi de composer en *romain* (comme c'est l'usage en France) :

4-68

```
\usepackage{mathtools,amssymb}
```

On appelle facteur de forme le rapport

```
\[F=\frac{s_{\text{eff}}}{s_{\text{m}}}\]
=\frac{\sqrt{\frac{1}{T}\int_{t_0}^{t_0+T}s^2(t)\mathrm{d}t}}{\frac{1}{T}\int_{t_0}^{t_0+T}s(t)\mathrm{d}t}
```

et on peut écrire

```
\[s(t)=c_0+\sum_{n=1}^{+\infty}c_n\cos(n\omega t-\varphi_n)\] où
$c_0=\frac{a_0}{2}$, $c_n=\sqrt{a_n^2+b_n^2}$ pour $n\geqslant 1$,
$\tan\varphi_n=\frac{b_n}{a_n}$ et
```

```
\[\begin{split}
a_n&=\frac{2}{T}\int_{t_0}^{t_0+T}s(t)\cos n\omega t\mathrm{d}t\\
b_n&=\frac{2}{T}\int_{t_0}^{t_0+T}s(t)\sin n\omega t\mathrm{d}t
\end{split}\]
```

Remarquons qu'il pourrait être plus judicieux de considérer « d » comme l'opérateur différentiel et donc de définir, au moyen de `\DeclareMathOperator{\dd}{d}`, l'opérateur `\dd` à utiliser en lieu et place de `\mathrm{d}`.

Solution de l'exercice 4-5 page 127

On utilise ici bien sûr les packages `ntheorem` et `thmtools`. Examinons comment sont composés ces deux « théorèmes ». Tout d'abord, leurs intitulés apparaissent de la même façon (en gras italique), ce qui nécessite que la commande responsable de cette caractéristique soit placée avant toutes les autres déclarations. Ensuite :

- Le rappel a les caractéristiques des théorèmes par défaut (style `plain`), mais il est non numéroté ; on le déclare donc avec le style `nonumberplain`. Son séparateur (un point) est stipulé au moyen de l'option `postheadhook`. Enfin, la marque en forme de carré plein (commande `\blacksquare` du package `amssymb` par exemple) indiquant la fin du rappel est spécifiée au moyen de l'option

prefoothook ; mais, afin qu'elle figure systématiquement en fin de ligne, elle est précédée de la commande `\hspace*{\stretch{1}}` (cf. page 43).

- Les exercices sont manifestement de style plain (pas de saut de ligne, ni d'inversion titre-numéro) mais présentent quelques caractéristiques particulières, insérées juste avant leur déclaration : corps composé en *romain* et numérotation en lettres grecques minuscules.

Par ailleurs, on a recouru à la commande `\mathbbm` du package `bbm` pour composer l'indicatrice sous forme d'un « 1 » ajouré. Alors, par souci d'homogénéité, on a utilisé cette même commande (et non la commande `\mathbb` du package `amssymb`) pour l'ensemble des entiers naturels.

Le texte demandé s'obtient donc comme suit :

4-69

```
\usepackage{amssymb,ntheorem,thmtools,bbm}
\theoremheaderfont{\bfseries\itshape}
\declaretheorem[title=Rappel,style=nonumberplain,postheadhook=.,
  prefoothook=\hspace*{\stretch{1}}$\blacksquare$]{rap}
\theorembodyfont{\normalfont}
\theoremnumbering{greek}
\declaretheorem[title=Exercice,style=plain]{exo}

\begin{rap}[nilpotence]
  Un élément  $x$  d'un anneau commutatif  $A$  est dit nilpotent s'il
  existe un entier  $n \in \mathbbm{N}$  tel que  $x^n = 0_A$ .
\end{rap}

\begin{exo}\label{ex-toto} Soit  $A$  un anneau commutatif non nul.
  \begin{enumerate}
    \item\label{qu-div} Prouver que tout élément nilpotent est un
      diviseur de zéro de  $A$ .
    \item\label{qu-inv} Établir que si  $x \in A$  est nilpotent, alors
       $x + \mathbbm{1}_A$  est inversible dans  $A$ .
  \end{enumerate}
\end{exo}

\begin{exo}\label{ex-titi}
  Démontrer que l'ensemble des éléments nilpotents d'un anneau
  commutatif  $A$  est un idéal.
\end{exo}

Dans l'exercice~\ref{ex-toto}, on veillera bien à traiter la
question~\ref{qu-div} \emph{avant} la question~\ref{qu-inv}.
L'exercice~\ref{ex-titi} est en bonus.
```

On note dans cet exemple que les théorèmes, comme tous les objets numérotés sous \LaTeX , peuvent faire l'objet de références croisées.

Solution de l'exercice 4-6 page 127

Dans le corrigé ci-dessous, de nombreuses commandes ont été définies pour se faciliter la tâche mais aussi, surtout, pour séparer fond et forme. L'environnement `split` a été utilisé pour aligner les flèches qui servent à définir la fonction Γ . Comme

elles ne nécessitaient pas d'alignement particulier, les deux propriétés de cette fonction ont été groupées dans un environnement gather. La commande `\setminus` a été employée pour symboliser la différence de deux ensembles. Enfin, on a recouru aux options `fleqn` et `tbtags` du package `mathtools` pour aligner à gauche les formules et ne pas centrer verticalement le numéro de la dernière équation :

4-70

```
\usepackage[fleqn,tbtags]{mathtools}\usepackage{amssymb,mathrsfs}
\newcommand\compl{\mathbb{C}}\newcommand\dplan{\mathscr{C}}
\newcommand\rela{\mathbb{Z}}\newcommand\natu{\mathbb{N}}
\newcommand\expo{\mathrm{e}}\DeclareMathOperator{\dd}{d}
```

On note \dplan l'ensemble $\{z \in \compl, \operatorname{Re}(z) > 0\}$ et on considère la fonction

```
\[
\begin{split}
& \Gamma \colon \dplan \rightarrow \compl \\
& z \mapsto \int_0^{+\infty} \exp^{-t} t^{z-1} \dd t
\end{split}
\]
\begin{enumerate}
\item Prouver que
\begin{gather}
\forall z \in \dplan, \quad \Gamma(z+1) = z \Gamma(z) \\
\forall n \in \natu^*, \quad \Gamma(n) = (n-1)!
\end{gather}
\item Démontrer que la fonction  $\Gamma$  s'étend à
 $\compl \setminus \rela^-$  et que
\[\forall n \in \rela^-, \quad \lim_{x \rightarrow n} (x-n) \Gamma(x) = \frac{(-1)^n}{(-n)!}\]
\]
On notera que, pour  $n$  négatif,
\begin{equation}
\begin{split}
\Gamma(z) &= \frac{\Gamma(z+1)}{z} \\
&= \frac{\Gamma(z+2)}{z(z+1)} \\
&= \cdots \\
&= \frac{\Gamma(z-n+1)}{z(z+1)\cdots(z-n-1)(z-n)}
\end{split}
\end{equation}
\end{enumerate}
```

Si le symbole de la partie réelle ne convient pas, on peut le modifier :

```
\DeclareMathOperator{\reelle}{Re}
\newcommand\preelle[1]{\operatorname{Re}\left(\#1\right)}
```

Solution de l'exercice 4-7 page 128

a L'environnement `subequations` permet la sous-numérotation d'équations.

Attention

L'environnement `subequations` doit nécessairement contenir des environnements produisant des équations numérotées.

Voici comment a été obtenu le texte précédent :

4-71

```
\usepackage{mathtools,amssymb}
\begin{equation}
e^{i\pi}+1=0
\end{equation}
\begin{subequations}
\begin{align}
a&=1\\
b&=2\\
c&=3
\end{align}
\end{subequations}
```

b Pour supprimer la numérotation d'une équation, il suffit de la faire suivre de la commande `\notag` (*avant* la commande `\\`). Pour remplacer un numéro d'équation par quoi que ce soit d'autre, on emploie la commande à argument `\tag` :

```
\begin{subequations}
\begin{align}
a&=1\\
b&=2\notag\\
c&=3\tag{coucou!}
\end{align}
\end{subequations}
```

On peut étiqueter de telles équations et y faire référence comme c'est le cas pour celles « normalement » numérotées.

c Pour numéroter les équations principales en *romain*, on insère en préambule :

```
\renewcommand\theequation{\Roman{equation}}
```

Pour que la sous-numérotation ait la forme voulue, on procède de façon analogue à l'intérieur de l'environnement `subequations`, la commande `\theparentequation` désignant le numéro de la numérotation principale :

```
\renewcommand\theequation{\theparentequation-\arabic{equation}}
```

Pour transformer les parenthèses en crochets, le package `mathtools` propose les commandes suivantes (qu'on placera de préférence en préambule du document) :

```
\newtagform{brackets}{[ ]}
\usetagform{brackets}
```

Solution de l'exercice E-4-8 page 425

Il suffit de coder :

E-14

```
\usepackage{mathtools}
\[\begin{bmatrix}17 & 4\sqrt{5} & -2\end{bmatrix}\]
```

Nous verrons au chapitre 5 que l'environnement `tabular`, responsable de la création d'un tableau, nécessite de spécifier l'alignement horizontal (à gauche, centré ou à droite) de chaque colonne. Au contraire, les environnements du package `amsmath` qui génèrent des matrices, eux, ne permettent pas une telle précision et les colonnes des matrices ainsi créées sont centrées. Le package `mathtools` permet de s'affranchir de cette limitation, comme suit (l et r comme *left* et *right*) :

E-15

```
\usepackage{mathtools}
\[\Delta=\begin{bmatrix*}[r]17 & 4\sqrt{5} & -2\end{bmatrix*}
=\begin{bmatrix*}[l]17 & 4\sqrt{5} & -2\end{bmatrix*}\]
```

Solution de l'exercice E-4-9 page 425

a Cet exercice, assez difficile, met le lecteur en situation réelle lorsqu'il cherche dans les archives de `ftt` une réponse à une question qu'il se pose : formuler sa requête de façon suffisamment précise pour ne pas être noyé sous une multitude de questions connexes et ne pas être envoyé sur de fausses pistes. Par exemple, une requête portant sur les mots « matrice droite » n'est probablement pas assez précise. Affiner la recherche, en la faisant porter sur « matrice ligne à droite », conduit assez rapidement à la réponse permettant de composer :

E-16

```
\usepackage{blkarray}
\[\begin{blockarray}{ccl}
f(e_1) & f(e_2) & \\
\begin{block}{(cc)l}
1 & 2 & e_1 \\
0 & 4 & e_2
\end{block}
\end{blockarray}\]
```

b On peut ici redéfinir la commande `\arraystretch` qui sera vue au chapitre 5 :

```
\renewcommand*\arraystretch{1.5}
```

Solution de l'exercice E-4-10 page 425

a Une recherche sur le mot « polynomial », parmi les descriptions de packages sur le site du `CTAN` (cf. rubrique « Recherche de packages » page 50), conduit à une liste de quatre packages concernant les polynômes : deux d'entre eux (en particulier le

package `pst-func` étudié dans le présent ouvrage) pour les dessiner, un (`polynom`) pour les manipuler (par exemple les diviser, les factoriser, etc.), et le dernier (`polynomial`) est celui qu'on cherche puisqu'il permet de les *composer* (*typeset*).

La documentation du package `polynomial`, illustrée de nombreux exemples, permet d'obtenir facilement le texte demandé : la commande `\polynomial` prend pour argument obligatoire la liste des coefficients (séparés par des virgules) qui, par défaut :

- sont consécutifs ;
- sont donnés dans l'ordre croissant, le premier d'entre eux étant celui du monôme de degré 0.

À l'aide de différentes options, il est possible de modifier la variable par défaut (`var`), le pas de la suite des coefficients (`step`), le degré du premier monôme (`start`), l'ordre des coefficients (`reciprocal` et `falling`), etc. :

E-17

```
\usepackage{polynomial}
$\polynomial{5,-3,2}$, $\polynomial[var=t]{5,-3,2}$,
$\polynomial[step=2]{5,-3,2}$, $\polynomial[start=3]{5,-3,2}$,
$\polynomial[reciprocal]{5,-3,2}$, $\polynomial[falling]{5,-3,2}$.
```

On notera, à la lecture de la documentation de ce package, que les options peuvent être stipulées en argument de la commande :

- `\polynomial` pour, sauf contre ordre, être appliquées de façon locale à tout ce qui suit jusqu'à la fin de l'environnement mathématique où elles apparaissent ;
- `\polynomialstyle` pour être appliquées de façon semi-globale au reste du document. Il est possible de spécifier cette option tant à l'intérieur qu'à l'extérieur des modes mathématiques.

Au moyen de `\polynomialstyle{default}`, il est à tout moment possible de revenir aux réglages par défaut.

Ce package permet également de composer des fractions rationnelles grâce à la commande `\polynomialfrac`.

b Sachant que *dérivée* se dit *derivative* en anglais, une démarche analogue à la précédente conduit au package `esdiff` qui offre les commandes `\diff` pour les dérivées ordinaires et `\diffp` pour les dérivées partielles. Ces commandes ont deux arguments obligatoires : le premier comprend le nom de la fonction à dériver et le deuxième la ou les variables par rapport auxquelles la dérivation est effectuée. S'il est commun à la fonction et à la variable (unique), l'ordre de dérivation peut être spécifié en argument optionnel de ces deux commandes. Le point où a lieu la dérivée peut être précisé en troisième argument obligatoire des variantes étoilées des commandes `\diff` et `\diffp`. Dans le cas de dérivation partielle par rapport à plusieurs variables, chacune des variables doit, dans le deuxième argument obligatoire, figurer entre paire d'accolades, son ordre de dérivation étant stipulé en exposant ; l'ordre total de dérivation n'a pas besoin d'être spécifié dans le premier argument : il est automatiquement calculé ! On prendra soin de charger le package `esdiff` avec l'option `text`, sans quoi l'homogénéité de l'espace interligne pourrait être rompue (cf. page 98).

Le texte demandé peut donc être efficacement obtenu comme suit :

E-18

```

\usepackage[text]{esdiff}

\[\diff{f}{x}\quad\diff*{f}{x}{x_0}\quad\diff[3]{f}{x}
\quad\diff*[3]{f}{x}{x_0}\]
\[\diffp{p}{V}\quad\diffp*{p}{V}{T}\quad\diffp[3]{p}{V}=\diffp{p}{V^3}
\quad\diffp*{p}{V^3}{T}\]
\[\diffp{f}{x}{y}\quad\diffp*{f}{x}{y}{z}\quad\diffp{f}{x^2}{y}
\quad\diffp*{f}{x^2}{y}{z}\]

```

Chapitre 5

Exercices supplémentaires

Exercice E-5-3 Fusionner verticalement et horizontalement

E-23	Quadriques		
	propre	à centre	ellipsoïde $x^2 + y^2 + z^2 = 1$
			hyperboloïde $x^2 + y^2 - z^2 = 1$ $x^2 - y^2 - z^2 = 1$
	paraboloïde	elliptique $x^2 + y^2 = z$	
		hyperbolique $x^2 - y^2 = z$	
	impropre	cône $x^2 - y^2 - z^2 = 0$	
		cylindre $x^2 + y^2 = 1$	
		deux plans sécants $x^2 + y^2 = 0$	
		deux plans parallèles $x^2 = 1$	
		deux plans confondus $x^2 = 0$	

Le tableau ci-dessus possède trois colonnes. Placer un texte verticalement dans un tableau ne devrait plus poser de problème. En revanche, l'entrée « impropre » en bas à gauche du tableau est plus compliquée puisqu'elle fusionne à la fois des lignes et des colonnes. Comment obtenir ce tableau ?

Exercice E-5-4 Tableaux emboîtés

De nombreux moyens permettent d'obtenir le tableau de l'exemple E-24 page suivante. Dans cet exercice, il est demandé de le réaliser en emboîtant des tableaux et tenant notamment compte des caractéristiques suivantes :

- les chaînes d'ADN et de peptides doivent eux-mêmes être des tableaux ;
- la première colonne doit avoir une largeur de trois centimètres ;
- les bases mutées sont placées dans un cadre grâce à la commande `\fbox` (on aurait pu se servir de tableaux à la place de cette commande ce qui aurait donné des tableaux dans des tableaux dans des tableaux ! Un tel emboîtement à trois niveaux n'est pas utile ici, mais serait parfaitement valide pour \LaTeX).

E-24	Situation initiale	ADN	—	<table border="1"><tr><td>AAG</td><td>AAG</td><td>AAG</td><td>AAG</td></tr></table>	AAG	AAG	AAG	AAG	—	
		AAG	AAG	AAG	AAG					
Peptide	—	<table border="1"><tr><td>Lys</td><td>Lys</td><td>Lys</td><td>Lys</td></tr></table>	Lys	Lys	Lys	Lys	—			
Lys	Lys	Lys	Lys							
Mutation ponctuelle	Substitution	ADN	—	<table border="1"><tr><td><table border="1"><tr><td>G</td></tr></table>AG</td><td>AAG</td><td>AAG</td><td>AAG</td></tr></table>	<table border="1"><tr><td>G</td></tr></table> AG	G	AAG	AAG	AAG	—
		<table border="1"><tr><td>G</td></tr></table> AG	G	AAG	AAG	AAG				
	G									
	Peptide	—	<table border="1"><tr><td>Glu</td><td>Lys</td><td>Lys</td><td>Lys</td></tr></table>	Glu	Lys	Lys	Lys	—		
	Glu	Lys	Lys	Lys						
Synonyme	ADN	—	<table border="1"><tr><td>AA</td><td><table border="1"><tr><td>A</td></tr></table></td><td>AAG</td><td>AAG</td><td>AAG</td></tr></table>	AA	<table border="1"><tr><td>A</td></tr></table>	A	AAG	AAG	AAG	—
	AA	<table border="1"><tr><td>A</td></tr></table>	A	AAG	AAG	AAG				
A										
Peptide	—	<table border="1"><tr><td>Lys</td><td>Lys</td><td>Lys</td><td>Lys</td></tr></table>	Lys	Lys	Lys	Lys	—			
Lys	Lys	Lys	Lys							
Terme	ADN	—	<table border="1"><tr><td>AAG</td><td>AAG</td><td><table border="1"><tr><td>A</td></tr></table><table border="1"><tr><td>T</td></tr></table>G</td><td>AAG</td></tr></table>	AAG	AAG	<table border="1"><tr><td>A</td></tr></table> <table border="1"><tr><td>T</td></tr></table> G	A	T	AAG	—
	AAG	AAG	<table border="1"><tr><td>A</td></tr></table> <table border="1"><tr><td>T</td></tr></table> G	A	T	AAG				
A										
T										
Peptide	—	<table border="1"><tr><td>Lys</td><td>Lys</td></tr></table>	Lys	Lys	—					
Lys	Lys									

Exercice E-5-5 Tableaux en couleurs

Le but de cet exercice est de construire le plus simplement possible le tableau suivant qui compose les nombres premiers normalement et les nombres composés en blanc sur fond rouge.

E-25			2	3	4	5	6	7	8	9
	10	11	12	13	14	15	16	17	18	19
	20	21	22	23	24	25	26	27	28	29
	30	31	32	33	34	35	36	37	38	39
	40	41	42	43	44	45	46	47	48	49
	50	51	52	53	54	55	56	57	58	59
	60	61	62	63	64	65	66	67	68	69
	70	71	72	73	74	75	76	77	78	79
	80	81	82	83	84	85	86	87	88	89
	90	91	92	93	94	95	96	97	98	99

Il serait possible, pour ce faire, de demander à \LaTeX d'effectuer tous les calculs pour vérifier qu'un nombre est premier ou non. On pourrait même penser construire une macro, par exemple `\premier{100}`, qui ferait directement et automatiquement le travail de composition du tableau, mais cela demanderait des compétences en programmation \LaTeX qui sortent du cadre de ce livre. Si des lecteurs courageux veulent s'y essayer, nous recommandons les packages `calc` et `ifthen`.

Ici, on demande uniquement de se servir astucieusement du motif du tableau pour éviter autant que faire se peut les commandes dans le corps du tableau. Pour simplifier la saisie, on pourra prévoir des commandes `\PR{<nb>}` et `\C{<nb>}` pour indiquer qu'un nombre est premier ou pas (C comme « composé ») :

Exercice E-5-6 Numérotation automatique des lignes

Le package `xcolor` avec l'option `table` permet de colorier une ligne sur deux de façon différente. En interne, il utilise le compteur `\rownum`. Cet exercice propose d'employer ce compteur pour réaliser une numérotation automatique des lignes.

Comment obtenir le résultat suivant, sans taper directement le numéro des lignes, mais en se servant d'une macro `\numline` qui fera automatiquement le travail ? La difficulté est que la numérotation ne doit pas débiter immédiatement et on prévoira un compteur `firstline` indiquant la première ligne à numérotter :

Satellites d'Uranus				
	Nom	Période (j)	Diamètre	Magnitude
1	Miranda	1,413	300	16,5
2	Ariel	2,520	800	14,4
3	Umbriel	4,144	600	15,3
4	Titania	8,706	1000	14,0
5	Obéron	13,463	1000	14,2

Finalement, on demande d'obtenir ce résultat avec exactement le même corps de tableau que celui de l'exemple 5-31 page 151. Tout le travail doit être fait dans le motif du tableau et à l'aide de la macro `\numline`.

Solutions des exercices

Solution de l'exercice 5-1 page 156

Le code va utiliser le package `array` pour la création du nouveau spécificateur de colonne M composant l'entrée en mode mathématique et centré dans la cellule.

Il utilise également le package `graphicx` pour construire des textes penchés grâce à la commande `\rotatebox` (cf. chapitre 8). Le code pour obtenir un texte penché étant relativement long et comme il doit être répété quatre fois, il est souhaitable de définir une macro dédiée à cette tâche.

Une approche directe ne va pas donner le résultat escompté. Le problème est qu'un texte penché occupe une largeur importante et que c'est cette dimension qui va déterminer ici la largeur de la colonne, comme on le voit dans l'exemple E-27.

Une solution est de placer le texte dans une boîte de largeur nulle à l'intérieur de la commande `\rotatebox`. De cette façon, le texte, même penché, aura une largeur faible qui n'aura pas d'influence sur la largeur des colonnes.

E-27

```
\usepackage{graphicx}
\usepackage{array}
\newcommand\penche[1]{%
  \multicolumn{1}{c}{%
    \rotatebox{45}{#1}}%
}
\newcolumntype{M}{%
  >{$}c<{$}}
\begin{tabular}{|l|*{4}{M|}}
  % Cf. ci-dessous
\end{tabular}
```

	Photon	Électron	Proton	Neutron
Spin	1	1/2	1/2	1/2
Charge	0	-1	+1	0
Masse	0	0.511	938	940

Cette solution a néanmoins une conséquence malheureuse. Le texte penché a une largeur faible... mais aussi une hauteur faible. Il risque donc de déborder sur le matériel placé au-dessus du tableau.

Pour annuler cet effet, il faut redonner à la première ligne du tableau la hauteur qu'elle devrait avoir sans l'utilisation de ces boîtes de largeur nulle. Le principe va être de récupérer la largeur de l'entrée penchée la plus importante (en l'occurrence celle du mot « Électron ») et de l'utiliser dans un strut. Le plus simple, pour cela, est d'employer la commande `\settowidth` du package `calc` :

5-36

```
\usepackage{graphicx, array, calc}
\newcommand\penche[1]{%
  \multicolumn{1}{c}{\rotatebox{45}{\makebox[0pt][l]{#1}}}%
}
\newcolumntype{M}{>{$}c<{$}}
\newlength{\hauteur}
\settowidth{\hauteur}{\rotatebox{45}{Électron}}
\begin{tabular}{|l|*{4}{M|}}
  \multicolumn{1}{c}{\rule{0pt}{\hauteur}} &
  \penche{Photon} & \penche{Électron} &
  \penche{Proton} & \penche{Neutron} \\ \hline
  Spin & 1 & 1/2 & 1/2 & 1/2 \\ \hline
  Charge & 0 & -1 & +1 & 0 \\ \hline
  Masse & 0 & 0.511 & 938 & 940 \\ \hline
\end{tabular}
```

On aurait pu faire un peu mieux en récupérant la hauteur de chaque entrée penchée et en ajoutant un strut de cette hauteur. De cette façon, on n'aurait même pas à se demander quelle est l'entrée la plus large. Cela ne change pas fondamentalement la façon de gérer ce problème, la macro `\penche` se code maintenant :

```
\newcommand\penche[1]{%
  \multicolumn{1}{c}{%
    \settowidth{\hauteur}{\rotatebox{45}{\makebox{45}{#1}}}%
    \rule{0pt}{\hauteur}\rotatebox{45}{\makebox[0pt][l]{#1}}%
  }}

```

Solution de l'exercice 5-2 page 156

L'idée est de tout réaliser avec un seul tableau d'une seule colonne. La première ligne contient le titre et la deuxième ligne est l'ensemble du tabulary. Cette approche se complique puisqu'ici on a des *filets* verticaux qui doivent se raccorder précisément à des *filets* horizontaux.

Le tableau externe est défini par un environnement tabular basique. La seule précaution à prendre est de supprimer l'espace intercolonne à gauche et à droite avec les mots clés @{} pour que les raccordements de *filets* se fassent correctement.

Le tableau interne sera donc composé avec un environnement tabulary ne comportant pas de *filets* verticaux à gauche et à droite. De plus, la largeur ne doit pas être tout à fait \linewidth puisque les *filets* verticaux du tableau externe diminuent un peu cette largeur disponible. La largeur doit donc être égale à \linewidth diminuée de deux fois l'épaisseur d'un *filet*. Ce calcul est effectué en se servant des fonctionnalités du package calc (cf. section C.2.4 page 383).

La solution proposée est donc :

5-37

```
\usepackage{tabulary}
\usepackage{calc}
\newlength{\largeur}

\setlength{\largeur}{\linewidth-\arrayrulewidth*2}
\begin{tabular}{|@{}c@{}}
  \hline
  Instructions du microprocesseur 68000 \\ \hline
  \begin{tabulary}{\largeur}{1|L|J}
    % etc.
  \end{tabulary}
\end{tabular}
```

On peut noter le motif 1|L|J du tabulary, au lieu du motif L|L|J qui avait été utilisé dans l'exemple exemple 5-24 page 146. Cela permet de ne pas tenir compte du paramètre \tymmin pour obtenir une première colonne correctement composée.

Solution de l'exercice E-5-3 page 435

Il est ici nécessaire de charger le package multirow, pour la fusion verticale, et le package graphicx, pour les textes verticaux.

La solution montre qu'il est possible de mettre une commande \multirow dans l'argument d'une commande \multicolumn. En revanche, l'inverse est interdit.

On notera la suite des commandes \multicolumn à la fin du code. Elles servent uniquement à supprimer le *filet* vertical entre les deux premières colonnes, celui-ci étant indiqué dans le motif du tableau. Il est clair que l'on aurait pu faire l'inverse : ne pas indiquer ce *filet* dans le motif du tableau et utiliser \multicolumn pour les entrées qui devront le faire apparaître. Pour ce tableau, cela revient à peu près au même en terme de saisie.

La présence de la redéfinition \renewcommand*\arraystretch{1.3} peut sembler étrange. En réalité, sans cette augmentation de la hauteur des lignes du tableau, on se

heurterait à un problème : le mot « paraboloïde » composé verticalement couvrirait plus que l'équivalent de quatre lignes de tableau.

E-23

```
\usepackage{multirow, graphicx}
\renewcommand*\arraystretch{1.3}
\begin{tabular}{|l|l|l|}
\hline
\multicolumn{3}{|c|}{\textbf{Quadriques}} \\ \hline
\multirow{8}{*}{\rotatebox{90}{propre}} &
\multirow{4}{*}{\rotatebox{90}{à centre}} &
ellipsoïde \\
& &  $x^2+y^2+z^2=1$  \\ \cline{3-3}
& & hyperboloïde \\
& &  $x^2+y^2-z^2=1$  \\ \quad \quad & &  $x^2-y^2-z^2=1$  \\ \cline{2-3}
& \multirow{4}{*}{\rotatebox{90}{paraboloïde}} &
elliptique \\
& &  $x^2+y^2=z$  \\ \cline{3-3}
& & hyperbolique \\
& &  $x^2-y^2=z$  \\ \hline
\multicolumn{2}{|c|}{\multirow{10}{*}{\rotatebox{90}{impropre}}}}
& cône \\
\multicolumn{2}{|c|}{}} &  $x^2-y^2-z^2=0$  \\ \cline{3-3}
\multicolumn{2}{|c|}{}} & cylindre \\
\multicolumn{2}{|c|}{}} &  $x^2+y^2=1$  \\ \cline{3-3}
\multicolumn{2}{|c|}{}} & deux plans sécants \\
\multicolumn{2}{|c|}{}} &  $x^2+y^2=0$  \\ \cline{3-3}
\multicolumn{2}{|c|}{}} & deux plans parallèles \\
\multicolumn{2}{|c|}{}} &  $x^2=1$  \\ \cline{3-3}
\multicolumn{2}{|c|}{}} & deux plans confondus \\
\multicolumn{2}{|c|}{}} &  $x^2=0$  \\ \hline
\end{tabular}
```

Solution de l'exercice E-5-4 page 435

En raison de sa complexité, un tel tableau nécessite une analyse avant de commencer à le coder au clavier.

Le tableau complet comporte trois colonnes et huit lignes. Les textes de la première colonne seront composés avec la commande `\multirow`. Ceux de la deuxième colonne seront composés *au fer à gauche* et les contenus de la dernière colonne seront, comme l'exercice le demande, des tableaux. Tous ces tableaux internes ont la même largeur, mis à part celui de la dernière ligne. La troisième colonne sera donc du type `l` pour que tous les tableaux internes soient composés *au fer à gauche*. Le spécificateur de la première colonne importe peu, puisque tous les textes de cette colonne seront composés avec des commandes `\multirow`. Le tableau externe a donc pour motif :

```
|l|ll|
```


Solution de l'exercice E-5-6 page 437

La première difficulté consiste à utiliser le compteur `\rownum` correctement. En effet, il s'agit d'un compteur \TeX et non d'un compteur \LaTeX . Pour obtenir le nombre contenu dans ce compteur, il ne faut plus utiliser la syntaxe \LaTeX `\value{\rownum}` mais la syntaxe \TeX `\number\rownum`.

Pour le motif du tableau, il n'y a rien de bien compliqué. Il suffit juste d'indiquer un `>\numline` avant la première colonne.

Tout le travail est donc effectué par cette macro `\numline`. En premier lieu, elle va calculer la différence entre la valeur du compteur (\TeX) `\rownum` et celle du compteur (\LaTeX) `firstline`. Le résultat sera sauvegardé dans un compteur `numdisplay`, et si celui-ci est strictement positif, il sera effectivement affiché. Le test utilise le package `ifthen` et sa commande `\ifthenelse`. Les calculs s'appuient sur les facilités offertes par le package `calc`. Ces deux packages sont présentés à l'annexe C.

Cela donne finalement le code :

E-26

```
\usepackage[table]{xcolor}
\usepackage{ifthen, calc}
\newcounter{firstline}
\newcounter{numdisplay}
\newcommand\numline{%
  \setcounter{numdisplay}
    {\number\rownum-\value{firstline}+1}%
  \ifthenelse{\value{numdisplay}>0}
    {\thenumdisplay\quad}
    {}%
}

\setcounter{firstline}{3}
\rowcolors{1}{}{}
\begin{tabular}{|>\numline}lrrr|}
  \hline
  \multicolumn{4}{|c|}{Satellites d'Uranus} \\ \hline
  \multicolumn{1}{|c|}{Nom} & \multicolumn{1}{c|}{Période (j)} & & \\
  \multicolumn{1}{c|}{Diamètre} & \multicolumn{1}{c|}{Magnitude} & & \\
  \hline
  Miranda & 1,413 & 300 & 16,5 \\
  Ariel & 2,520 & 800 & 14,4 \\
  Umbriel & 4,144 & 600 & 15,3 \\
  Titania & 8,706 & 1000 & 14,0 \\
  Obéron & 13,463 & 1000 & 14,2 \\ \hline
\end{tabular}
```

Il faut bien penser à placer une commande `\rowcolors` minimale avant le tableau pour pouvoir se servir du compteur `\rownum`.

Chapitre 6

Solutions des exercices

Solution de l'exercice 6-1 page 179

Les deux premières grandeurs ne posent pas de problèmes. Par contre, si la dernière est saisie sans précaution (`\SI{\sqrt{5}}{\metre}`), la compilation échoue.

Cela tient au fait que, pour permettre l'analyse nécessaire à la mise en forme des nombres, le premier argument de la commande `\SI`, comme celui de la commande `\num`, ne doit rien contenir d'autre que ce qui est indiqué page 158 ; en particulier, ces arguments ne peuvent *a priori* pas contenir de code \LaTeX (ici `\sqrt`). Toutefois, pour de telles situations, `siunitx` a prévu une option permettant de débrayer l'analyse : `parse-numbers=false`.

C'est donc au moyen de `\SI[parse-numbers=false]{\sqrt{5}}{\metre}` que doit être saisie la dernière grandeur.

Solution de l'exercice 6-2 page 179

Il suffit de recourir à l'option `retain-explicit-plus`.

Solution de l'exercice 6-3 page 179

L'option `detect-italic` seule ne suffit pas à pour composer les nombres en italique parce que :

1. d'une part, les commandes `\num`, `\ang` et `\SI` (et assimilées) composent par défaut les nombres en mode mathématique ;
2. d'autre part, le mode mathématique laisse en *romain* les chiffres et signes de ponctuation (cf. point 2 page 96) ;

Par ailleurs, la documentation de `siunitx` nous apprend que l'option `detect-mode`, automatiquement chargée par `detect-all`, compose les nombres des commandes `\num`, `\ang` et `\SI` dans le mode (*texte* ou *mathématique*) courant. Donc, lorsqu'une commande telle que `\num{136}` est utilisée, on note que :

1. Si l'option `detect-mode` est activée, le nombre 136 est composé en mode texte (tandis que la commande `\num{136}` le composerait en mode mathématique).
2. Si, de plus, la forme par défaut du mode texte est l'italique, le nombre 136 a beau être composé en mode texte, tant que ne sont pas chargées les options `detect-weight`, `detect-italic` et `detect-family`, seuls les attributs autres que la *graisse*, l'italique et la famille de fonte sont pris en compte.
3. Si, de plus, l'option `detect-italic` est activée, alors la commande `\num{136}` va, cette fois-ci, composer le nombre 136 en mode texte italique.

Pour illustrer la prise en compte par l'option `detect-mode` des attributs des modes texte et mathématique (autres que la *graisse*, l'italique et la famille de fonte), on recourt dans l'exemple suivant au package `eco` qui compose en chiffres *elzéviens* tous les nombres du mode texte, et normalement tous ceux du mode mathématique :

E-28

```
\usepackage{eco}
```

```
\itshape
```

Par défaut, texte en italique et chiffres elzéviens: 0123456789. \par

Sans option: $\SI{136}{\pascal}$ et $\$ \SI{136}{\pascal} \$$. \par

Option `\verb|detect-italic|`:

```
{\sisetup{detect-italic}\SI{136}{\pascal} et  $\$ \SI{136}{\pascal} \$$ }. \par
```

Option `\verb|detect-mode|`:

```
{\sisetup{detect-mode}\SI{136}{\pascal} et  $\$ \SI{136}{\pascal} \$$ }. \par
```

Option `\verb|detect-italic|` et `\verb|detect-mode|`:

```
{\sisetup{detect-italic,detect-mode}
  \SI{136}{\pascal} et  $\$ \SI{136}{\pascal} \$$ }. \par
```

Par défaut, texte en italique et chiffres elzéviens : 0123456789.

Sans option : 136 Pa et 136 Pa.

Option `detect-italic` : 136 Pa et 136 Pa.

Option `detect-mode` : 136 Pa et 136 Pa.

Option `detect-italic` et `detect-mode` : 136 Pa et 136 Pa.

Chapitre 7

Solutions des exercices

Solution de l'exercice 7-1 page 202

Évidemment, les trois tableaux seront déclarés avec l'environnement `table`. Le problème, ici, est dû à la valeur par défaut de `topnumber` qui vaut 2 et qui indique que le nombre maximal de flottants en haut de page est de 2.

Une approche directe consisterait à placer la commande

```
\setcounter{topnumber}{3}
```

(ou plus) juste avant de composer le texte, mais cela ne fonctionnerait pas car c'est la valeur *avant* que la page ne débute qui est prise en compte. Comme il n'y a pas de page avant la première page, la seule possibilité consiste à placer cette affectation dans le préambule du document. Le code complet de l'exemple est donné au listing 7.1 page ci-contre.

En réalité, il est préférable d'indiquer une seule fois le placement des flottants dans le préambule. Le package `floatrow`, par exemple, permet de spécifier que tous les flottants de type `table` devront se comporter comme s'ils avaient un spécificateur de placement `t`. Évidemment, cette facilité est à éviter si l'on veut obtenir des flottants de type `table` qui ne flotteront pas toujours de la même façon. Dans ce cas, il faut se demander si cette présentation est vraiment logique !

Solution de l'exercice 7-2 page 202

a Le code qui a permis cette présentation est le suivant :

7-20

```
\usepackage{floatrow}
\DeclareNewFloatType{serie}{name=Série}
\floatsetup[serie]{style=ruled, capposition=beside}
\captionsetup[serie]{format=hang}

\begin{serie}

$$\sum_{i=0}^{\infty} (-1)^i \frac{x^{2n}}{(2n)!}$$

\caption{Cosinus. Cette série peut être obtenue à partir
de la série fondamentale donnant  $e^x$ .}
\label{ser-cos}
\end{serie}
```

Un nom d'environnement tel que `Série` est théoriquement permis, mais il n'est pas recommandé d'utiliser des lettres accentuées dans les noms des environnements (et c'est totalement interdit pour les noms de commandes). Pour obtenir le texte « Série » au niveau de la légende, on a donc eu recours à la clé `name`.

7-19

```
\setcounter{topnumber}{3}
```

Les tableaux~\ref{x2},

```
\begin{table}[t]
\centering
$\begin{array}{|*{6}{c}|}
\hline
x & -2 & -1 & 0 & 1 & 2 \\ \hline
f(x) & 4 & 1 & 0 & 1 & 4 \\ \hline
\end{array}$
\caption{Valeur de $f(x)=x^2$}
\label{x2}
```

```
\end{table}
```

et~\ref{x3}

```
\begin{table}[t]
\centering
$\begin{array}{|*{6}{c}|}
\hline
x & -2 & -1 & 0 & 1 & 2 \\ \hline
f(x) & -8 & -1 & 0 & 1 & 8 \\ \hline
\end{array}$
\caption{Valeur de $f(x)=x^3$}
\label{x3}
```

```
\end{table}
```

et~\ref{x4}

```
\begin{table}[t]
\centering
$\begin{array}{|*{6}{c}|}
\hline
x & -2 & -1 & 0 & 1 & 2 \\ \hline
f(x) & 16 & 1 & 0 & 1 & 16 \\ \hline
\end{array}$
\caption{Valeur de $f(x)=x^4$}
\label{x4}
```

```
\end{table}
```

vont permettre de construire les premières représentations graphiques.

Listing 7.1 : Trois flottants en haut de page (code)

On notera l'utilisation de l'argument optionnel dans les commandes `\floatsetup` et `\captionsetup`, permettant de définir cette présentation uniquement pour les flottants de type `serie`.

b Dans la solution de cette question, on notera que la clé servant à composer la légende de l'autre côté a été indiquée dans la commande `\floatsetup`, alors que la clé servant à signaler une largeur de quatre centimètres pour la légende a été spécifiée dans la commande `\thisfloatsetup`.

Cette utilisation de `\thisfloatsetup` n'est évidemment pas obligatoire : on aurait tout aussi bien pu mentionner l'ordre dans la commande `\floatsetup`. Cela dit, il peut être judicieux de ne modifier une largeur de légende qu'au cas par cas.

7-21

```
\usepackage{floatrow}
\DeclareNewFloatType{serie}{name=Série}
\floatsetup[serie]{style=ruled,capposition=beside,
  capbesideposition=outside}
\captionsetup[serie]{format=hang}
\thisfloatsetup{capbesidewidth=4cm}
\begin{serie}
  $\displaystyle \sum_{i=0}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}$
  \caption{Cosinus. Cette série peut être obtenue à partir
    de la série fondamentale donnant $e^x$}
  \label{ser-cos}
\end{serie}
```

Solution de l'exercice 7-3 page 203

Il n'y a pas de difficulté particulière : il suffit d'utiliser directement les fonctionnalités des packages `subfig` et `floatrow` et tout se passe bien.

7-22

```
\usepackage{floatrow}
\usepackage{subfig}
\DeclareNewFloatType{serie}{name=Série}
\floatsetup[serie]{style=ruled}
\captionsetup[serie]{format=hang}
\begin{serie}
  \subfloat[$\sin(x)$]{$\displaystyle \sum_{i=0}^{\infty} (-1)^n
    \frac{x^{2n+1}}{(2n+1)!}$}\hfil
  \subfloat[$\cos(x)$]{$\displaystyle \sum_{i=0}^{\infty} (-1)^n
    \frac{x^{2n}}{(2n)!}$}
  \caption{Sinus et cosinus. Ces séries peuvent s'obtenir
    à partir de la série fondamentale donnant $e^x$}
  \label{ser-cossin}
\end{serie}
```

Pour le placement des deux sous-flottants au tiers et aux deux tiers de la page, on s'est servi des commandes d'espacement qui seront présentées à l'annexe C. Le

flottant, dans son ensemble, est centré grâce à deux commandes `\hfil` posées, en interne, à gauche et à droite de la structure. Ainsi, en utilisant cette même commande entre les deux sous-flottants, on obtient l'effet désiré. Il aurait été possible d'employer des longueurs fixes, mais c'est une mauvaise idée qu'il faut éviter !

Comme dans les autres exercices, aucune commande `\label` n'a été utilisée ici. L'unique raison est que ces commandes ne servent pas pour répondre aux questions posées. Il va de soi qu'il est toujours possible d'affecter une étiquette à un flottant et aux sous-flottants.

Solution de l'exercice 7-4 page 203

a Si l'on veut une solution automatique, on peut redéfinir la commande `\section` (`\sectionori` pour « section originale ») :

```
\let\sectionori\section
\renewcommand\section{\suppressfloats \sectionori}
```

Si l'on avait utilisé `\clearpage` au lieu de `\suppressfloats`, il y aurait eu systématiquement un saut de page avant chaque section. La solution proposée ne provoquera pas de saut de page sauf s'il est nécessaire de produire une page flottante pour placer les flottants en attente.

Une autre solution consiste à utiliser le package `placeins` qu'il suffit de charger avec l'option `section` :

```
\usepackage[section]{placeins}
```

b Il suffit d'utiliser la commande `\afterpage` du package `afterpage`, ainsi :

```
\afterpage{\clearpage\input{tableau.tex}}
```

La présence de la commande `\clearpage` est une précaution. Si des flottants sont en attente, ils sont traités avant que le grand tableau ne soit composé.

Si l'on veut un résultat s'apparentant à une page flottante, c'est-à-dire un tableau sur plusieurs pages sans texte directement à la suite, il suffit d'ajouter une commande `\newpage` après la commande `\input`.

Solution de l'exercice 7-5 page 203

Il n'y a malheureusement pas de solution automatique. L'astuce consiste à composer d'abord le texte sans la figure et à voir où se situe la coupure de ligne au niveau de laquelle on veut faire débiter l'objet habillé. Dans le cas présent, le mot « complexe » est situé à la fin de la deuxième ligne avec la division « compl-exe ».

Avec le package `wrapfig`, on tape donc les deux premières lignes avec cette division avant de placer l'environnement `wrapfigure` de façon habituelle.

L'utilisation du package `picins` pour ce type de présentation pose un peu plus de problèmes puisque, en interne, la commande `\parpic` commence un nouveau paragraphe. Avec ce package et cette commande, on aurait alors eu un *retrait* au niveau du « exe » de « complexe » et un espacement vertical anormal entre la deuxième et la troisième ligne. Le premier problème se résout immédiatement en plaçant une

commande `\noindent` mais le deuxième problème est un peu plus délicat à régler. Il faudrait compenser cet espacement non voulu avec un `\vspace` négatif approprié.

En utilisant le package `wrapfig`, cela donne :

7-23

```
\usepackage{wrapfig}
\usepackage{graphicx}
```

Le phage T4 est un virus à ADN infectant *Escherichia Coli*.

La structure du phage T4 est plus compl-

```
\begin{wrapfigure}{1}{4cm}
```

```
\centering
```

```
\includegraphics{T4}
```

```
\caption{Phage T4}
```

```
\end{wrapfigure}
```

exe que celle

d'autres virus et l'assemblage du virus à l'intérieur de la cellule-hôte est un processus encore mal connu. Une quarantaine de gènes vont participer à l'élaboration des différents composants de ce bactériophage.

En raison de la procédure en deux temps qui est utilisée, la moindre modification sur les deux premières lignes (une faute de frappe oubliée par exemple) peut obliger à reprendre tout le travail.

Solution de l'exercice 7-6 page 204

L'exercice précédent proposait de placer un objet habillé par du texte après la deuxième ligne du paragraphe. Le début de la solution est identique et consiste à composer le paragraphe sans image pour savoir à quel niveau se situe la coupure de deuxième ligne. Dans le cas présent, elle a lieu après le mot « fonction ».

Ce même exercice signalait également les difficultés auxquelles on pouvait s'attendre en cas d'utilisation du package `picins` pour obtenir une telle présentation. Voyons ces difficultés une à une.

Comme on l'a signalé, la commande `\parpic` fait que le texte qui suit est considéré comme un nouveau paragraphe. Il devrait donc y avoir un *retrait* que l'on supprime grâce à la commande `\noindent`. Il y a également un espacement vertical inhabituel entre la deuxième ligne et la suite du texte. Celui-ci sera compensé par une commande `\vspace` avec une longueur négative.

Le texte à la suite des deux premières lignes étant un nouveau paragraphe, il est logique de penser que la fin de la deuxième ligne est également une fin de paragraphe. Cela a une conséquence malheureuse ici : il y a un ressort d'extension infinie qui fait que cette ligne n'est plus *justifiée*. Pour retrouver cette *justification*, même pour la dernière ligne d'un paragraphe, on emploiera la commande `\linebreak`.

Il reste à décaler l'image. Au niveau horizontal, cela se fait simplement en utilisant les paramètres optionnels entre parenthèses de la commande `\parpic`. Malheureusement, si l'on décale l'image vers la droite, il va y avoir un espace important entre la droite du texte et la gauche de l'image. Une solution consiste à tricher sur la largeur de l'image en disant qu'elle est moins large que ce décalage. Le problème est ana-

logue avec la hauteur : comme le paragraphe a été décalé vers le haut à cause de la commande `\vspace`, il va falloir tricher également sur la hauteur de l'image.

Pour ne pas avoir à effectuer trop de recherches par essais-erreurs, on peut automatiser les calculs. Le principe va être de placer l'image dans une boîte avant de s'en servir et d'utiliser la hauteur et la largeur de cette boîte pour calculer les fausses dimensions. Cette façon de faire utilise des techniques présentées à l'annexe C. Le lecteur ayant utilisé des dimensions trouvées par essais-erreurs sera largement pardonné eu égard à la difficulté de l'exercice !

Nous voici au bout de tout ce à quoi il fallait songer et le résultat s'obtient ainsi :

7-24

```

\usepackage{picins}
\usepackage{graphicx}
\usepackage{calc}
\newsavebox{\tempbox}
\newlength{\largeur}
\newlength{\hauteur}

\begin{lrbox}{\tempbox}
\includegraphics{carre}
\end{lrbox}
Avec le tableau de valeurs réalisé auparavant, on peut
tracer la courbe représentative de la fonction \linebreak
\setlength{\largeur}{\wd\tempbox-1cm}%
\setlength{\hauteur}{\ht\tempbox+15pt}%
\parpic(\largeur,\hauteur)(0mm,\ht\tempbox)[r]{\usebox{\tempbox}}
\vspace{-15pt}\noindent
 $f(x)=x^2$ . On se servira également du fait qu'il s'agit
d'une fonction paire pour tracer la partie dans les  $x$ 
négatifs. La représentation graphique de  $f$  est indiquée
ci-contre. On peut la comparer aux représentations des
autres fonctions puissances ( $f(x)=x^3$  et  $f(x)=x^4$ )
montrées plus loin et pour lesquelles la construction des
courbes utilise des tableaux de valeurs de même type
réalisés précédemment.

```

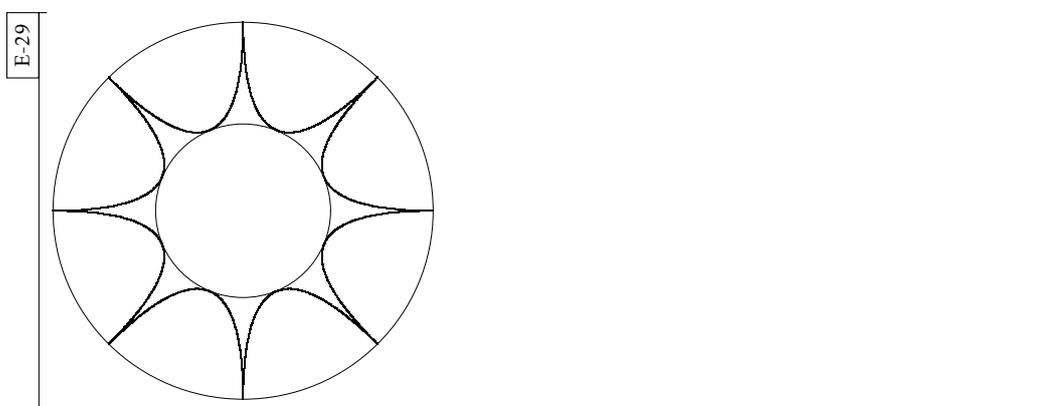
Il existe d'autres façons de procéder pour obtenir le résultat souhaité. Par exemple, le package `shapepar`, non présenté dans cet ouvrage, permet de composer des paragraphes avec des formes libres. Pour l'exemple 7-24 de la page précédente, il faudrait composer un paragraphe avec une augmentation de la marge droite pour certaines lignes, en l'occurrence de la troisième à la neuvième ligne.

Chapitre 8

Exercices supplémentaires

Exercice E-8-4 Frise L^AT_EXienne

Pour le tracé suivant, on a utilisé les packages `epic` et `eepic` :

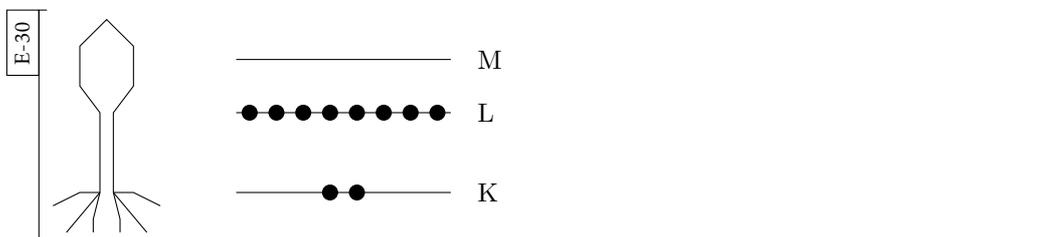


Il s'agit de réaliser la rosace à huit branches ci-dessus. Avec uniquement les packages `epic` et `eepic`, il faut s'attendre à devoir réaliser quelques calculs.

Exercice E-8-5 Alignement de graphiques

Supposons que l'on dispose de plusieurs fichiers graphiques et que l'on veuille afficher ces dessins de façon à ce qu'ils soient verticalement centrés. Dans cet exercice, nous prendrons les fichiers `T4.eps`, dont la *boîte englobante* vaut `72 640 112 720`, et `neon1.eps`, dont la *boîte englobante* vaut `72 666 172 724`.

On veut donc obtenir le résultat suivant :



où les deux images sont séparées d'exactly 1 cm.

Il existe de multiples façons de réaliser une telle présentation. L'exercice demande d'essayer d'en imaginer plusieurs.

Exercice E-8-6 Table des matières en couleurs

Supposons que l'on réalise un document de classe book. Comment faire pour obtenir une table des matières composée entièrement en vert, alors que le reste du document sera composé normalement en noir ?

E-31

Table des matières

I	Le nombre et la grandeur	3
1	Sur la nature du raisonnement mathématique	5
2	La grandeur mathématique et l'expérience	7

Le premier titre provient d'une commande `\part` et les deux suivants de commandes `\chapter`. Pour les plus curieux de nos lecteurs, il s'agit du début de *La Science et l'Hypothèse* de Poincaré.

Solutions des exercices**Solution de l'exercice 8-1 page 216**

On pourrait théoriquement s'en sortir en gérant le placement avec des boîtes mais, outre que cela n'est pas vu dans ce chapitre (les boîtes sont présentées à la section C.3 page 384), les manipulations seront complexes alors que l'environnement `picture` permet de réaliser cela bien plus efficacement :

8-16

```
\setlength{\unitlength}{5mm}
\begin{picture}(16,4)
  \put(0,1){*} \put(1,4){*} \put(6,4){*} \put(7,1.5){*}
  \put(8,1){*} \put(10.5,0){*} \put(16,1){*}
\end{picture}
```

Remarquons que les espaces ne sont pas pris en compte par les commandes graphiques à l'intérieur de l'environnement `picture`. On aura la même propriété avec les commandes graphiques de `PSTricks`. C'est bien pratique puisque cela permet d'organiser comme on le souhaite ces sources. Ainsi, dans l'exemple, cela a permis d'aligner verticalement les différentes macros `\put` ce qui est bien plus lisible que si toutes ces commandes étaient collées les unes aux autres.

Dans cet exemple, on a indiqué l'unité de longueur avant l'environnement. C'est obligatoire si l'on veut que les coordonnées (16,4) de la *boîte englobante* soient exprimées dans cette unité. Il faut toutefois être conscient que l'unité a été ainsi modifiée pour le reste du document ou jusqu'au prochain changement d'unité.

Solution de l'exercice 8-2 page 216

La commande `\graphicspath` permet de spécifier les noms de fichiers graphiques sans devoir écrire le chemin complet. Dans le cas présent, cela donnerait :

```
\graphicspath{{. ./}{figures/}{/usr/local/images/}}
```

Voici les points souvent sujets à erreur :

- chaque chemin est entre accolades, même s'il n'y en qu'un seul ;
- chaque chemin se termine par une barre oblique.

Même sous Windows, il faut utiliser la barre oblique et non la barre oblique inverse.

Notons que ce système d'exploitation a tendance à créer des noms de répertoire contenant des espaces. C'est une très mauvaise idée et il vaut mieux l'éviter. Si le mal est fait, par exemple si un des répertoires concernés est Documents and Settings, on peut essayer d'encadrer le nom complet du chemin par des *double quotes* ("), mais ce remède ne fonctionne pas avec toutes les distribution L^AT_EX.

Solution de l'exercice 8-3 page 216

La commande `\scalebox` avec les deux facteurs d'échelle à -1 permet d'obtenir le résultat. Il faut toutefois se méfier des décalages :

E-32

```
\usepackage{graphicx}
```

Quels sont les entiers n tels que n^2 divise 2^{n+1} ?

```
\par\scalebox{-1}{Seuls 1 et 3 sont solutions.}
```

Quels sont les entiers n tels que n^2 divise $2^n + 1$?

Seuls 1 et 3 sont solutions.

En effet, avec la rotation d'un demi-tour, la profondeur est devenue la hauteur et *vice versa*. Or, les lignes normales ont une hauteur de 0.7\baselineskip et une profondeur de 0.3\baselineskip . Le résultat ci-dessus montre une ligne « à l'envers » trop éloignée de la ligne normale.

Une solution « propre » consiste donc à effectuer un décalage vertical négatif correspondant à la différence entre hauteur et profondeur (0.4\baselineskip) avant de composer la ligne retournée :

8-17

```
\usepackage{graphicx}
\newcommand\solution[1]{%
  \par\vspace{-0.4\baselineskip}
  \scalebox{-1}{#1}%
}
```

Quels sont les entiers n tels que n^2 divise 2^{n+1} ?

```
\solution{Seuls 1 et 3 sont solutions.}
```

Le package `graphicx` propose également la commande `\rotatebox` qui permet d'obtenir ce type de présentation.

Solution de l'exercice E-8-4 page 452

Les deux cercles ne posent pas de problème : ils sont obtenus avec la commande `\circle` à l'intérieur d'un `\put`. La première difficulté consiste à trouver le bon rapport entre les rayons de ces deux cercles. En réalité, les huit arches sont des courbes de Bézier tracé avec `\qbezier` dont le point de contrôle médian est le centre des cercles. Il est alors très difficile de calculer le rayon du cercle qui sera tangent à toutes ces arches. L'auteur confesse qu'il a procédé par essai-erreur pour trouver les deux diamètres de 10 et 4,6.

En revanche, les points de contrôle des arches ont été calculés. Comme on l'a déjà dit précédemment, le point médian est le centre des cercles. Pour les points extrêmes, il faut réaliser quelques calculs trigonométriques (ou avec Pythagore) pour trouver certaines coordonnées. On a $5 \times \sin(45) \approx 3,54$. Les valeurs que l'on trouve sont donc 0, 5, 10 et le résultat des deux opérations $5 - 3,54$ et $5 + 3,54$. Muni de ces valeurs, on peut exprimer les coordonnées des huit points régulièrement répartis sur le cercle externe qui sont les bases des arches. Incidemment, le code permet de voir que les courbes de Bézier définies par `epic` ont bien la propriété d'invariance par rotation.

Il reste à définir une unité de longueur qui donne un bon résultat. En réalité, le choix de l'unité de longueur sert aussi à travailler avec des nombres plus simples à manipuler. En pratique, ce choix s'est fait pour avoir un cercle de diamètre 10 et non l'inverse. Dans l'exemple, on a pris 5mm :

E-29

```
\usepackage{epic, eepic}
\setlength{\unitlength}{5mm}
\begin{picture}(10,10)
  \put(5,5){\circle{10}} \put(5,5){\circle{4.6}}
  \qbezier(1.46,1.46)(5,5)(5,0)
  \qbezier(5,0)(5,5)(8.54,1.46)
  \qbezier(8.54,1.46)(5,5)(10,5)
  \qbezier(10,5)(5,5)(8.54,8.54)
  \qbezier(8.54,8.54)(5,5)(5,10)
  \qbezier(5,10)(5,5)(1.46,8.54)
  \qbezier(1.46,8.54)(5,5)(0,5)
  \qbezier(0,5)(5,5)(1.46,1.46)
\end{picture}
```

On a le droit de trouver cela bien compliqué pour un dessin finalement assez basique. Cela illustre bien que le package `epic`, même épaulé par `eepic`, ne permet pas d'obtenir facilement des dessins techniques complexes. Ce même exemple, traité par `PSTricks` donnerait un source bien plus simple et ne nécessiterait aucun calcul de la part de l'utilisateur.

Solution de l'exercice E-8-5 page 452

On peut tricher sur les dimensions de la *boîte englobante* de `neon1.eps` afin de la rendre aussi haute que celle de `T4.eps`. Cela donne :

E-30

```
\usepackage{graphicx}
\includegraphics{T4}\hspace{1cm}%
\includegraphics[bb=72 655 172 724]{neon1}
```

La boîte englobante 72 655 172 724 reprend les coordonnées indiquées dans le fichier, sauf en ce qui concerne la deuxième, c'est-à-dire l'ordonnée du bas de l'image. Pour obtenir la valeur 655, on a effectué le raisonnement suivant :

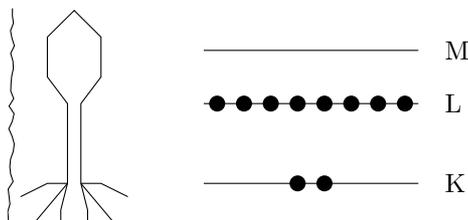
- La hauteur de T4.eps vaut $720 - 640 = 80$, celle de neon1.eps $724 - 666 = 58$.
- La différence de hauteur entre ces deux images vaut donc $80 - 58 = 22$.
- Finalement, on doit surélever la seconde image de la moitié de cette hauteur, c'est-à-dire de 11 bp.

Avec la méthode utilisée, on a donc indiqué une ordonnée de 11 bp inférieure à l'ordonnée véritable.

Une autre méthode est de recourir à la commande `\raisebox` :

E-33

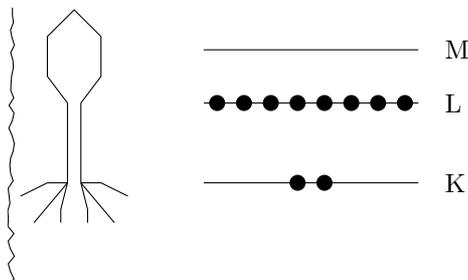
```
\usepackage{graphicx}
\includegraphics{T4}%
\hspace{1cm}%
\raisebox{11bp}
{\includegraphics{neon1}}
```



Ces deux méthodes, utiles pour des cas spéciaux, restent ici assez complexes puisqu'elles nécessitent de connaître les coordonnées exactes des boîtes englobantes. Une meilleure approche consiste à utiliser l'environnement `minipage` (cf. annexe C) :

E-34

```
\usepackage{graphicx}
\usepackage{calc}
\begin{minipage}{112bp-72bp}
\includegraphics{T4}%
\end{minipage}\hspace{1cm}%
\begin{minipage}{172bp-72bp}
\includegraphics{neon1}%
\end{minipage}
```



On notera l'utilisation du package `calc` (cf. annexe C) qui évite tout calcul. En réalité, cette solution, bien que plus simple, nécessite encore une fois de connaître les dimensions des boîtes englobantes (pour les $112\text{bp}-72\text{bp}$ et $172\text{bp}-72\text{bp}$). On peut automatiser entièrement la présentation en plaçant au préalable les deux images dans des boîtes pour récupérer leurs largeurs comme l'exemple E-35 page ci-contre le propose. On aurait d'ailleurs tout intérêt à définir une macro qui ferait le travail :

```
\newcommand*\ImageCentree[1]{\sbox{\tmpb}{\includegraphics{#1}}%
\begin{minipage}{\wd\tmpb}
\usebox{\tmpb}%
\end{minipage}%
}
```

E-35

```
\usepackage{graphicx}
\usepackage{calc}
\newsavebox{\tmpb}

\sbox{\tmpb}{\includegraphics{T4}}%
\begin{minipage}{\wd\tmpb}
  \usebox{\tmpb}%
\end{minipage}\hspace{1cm}%
\sbox{\tmpb}
  {\includegraphics{neon1}}%
\begin{minipage}{\wd\tmpb}
  \usebox{\tmpb}%
\end{minipage}
\end{pre>

```

Solution de l'exercice E-8-6 page 453

Le résultat s'obtient simplement en utilisant la macro `\tableofcontents` dans un groupe où la couleur verte a été activée par l'instruction `\color{green}`. Le placement dans un groupe est nécessaire si on ne veut pas que le document entier soit composé en vert !

E-31

```
\usepackage{xcolor}
{\color{green}\tableofcontents}
\part{Le nombre et la grandeur}
\chapter{Sur la nature du raisonnement mathématique}
La possibilité même du raisonnement mathématique semble une
contradiction insoluble.
% etc.
\chapter{La grandeur mathématique et l'expérience}
Si l'on veut savoir ce que les mathématiciens entendent par un
continu, ce n'est pas à la géométrie qu'il faut le demander.
% etc.
```

On aurait pu également proposer la solution :

```
\textcolor{green}{\tableofcontents}
```

mais, en pratique, cela échouera pour des raisons techniques trop délicates à expliquer ici : la compilation se passera sans erreur, mais deux pages vides seront produites en début de document.

Attention

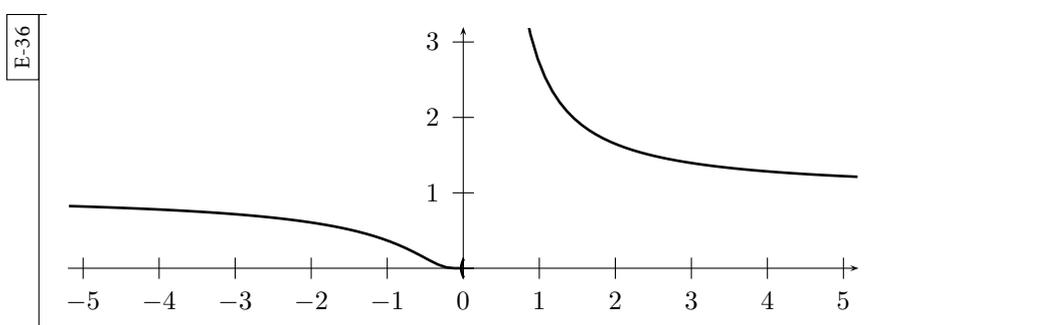
Il est important de bien refermer les accolades après la commande `\tableofcontents` ; sinon, il y aura une erreur à la compilation et le document entier sera composé en vert.

Chapitre 9

Exercices supplémentaires

Exercice E-9-3 Représentations graphiques de fonctions

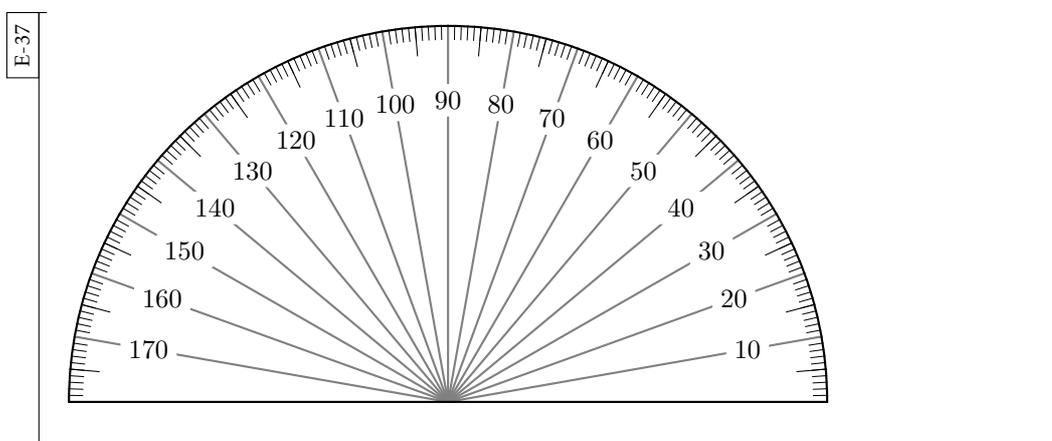
Le but de cet exercice est de tracer la courbe représentative de la fonction $e^{1/x}$.



Les axes sont tracés avec une épaisseur de ligne de 0.4pt tandis que la courbe est tracée avec une épaisseur de ligne de 1pt. On remarquera que la courbe s'arrête au niveau des frontières du repère, tant au niveau des abscisses qu'au niveau des ordonnées. Enfin, on a indiqué que le point d'abscisse nul ne fait pas partie de la courbe en plaçant une parenthèse à la fin de la partie gauche de cette courbe. Les abscisses se situent entre -5.2 et 5.2 et les ordonnées entre 0 et 3.2.

Exercice E-9-4 Un rapporteur

Voici le dessin d'un rapporteur :



Dans cet exercice, qui montre les possibilités offertes par le package `multido` utilisé en association avec `pstricks`, on demande de réaliser ce dessin le plus simplement possible. En particulier, on ne devra même pas envisager d'écrire dix-sept fois une commande pour écrire les angles, et encore moins d'écrire cent quatre-vingt fois (environ) une commande permettant de dessiner les petites graduations.

Pour la petite histoire, ce rapporteur est le résultat d'une sorte de concours sauvage qui a eu lieu sur Internet. Les résultats des participants peuvent être admirés sur la page <http://melusine.eu.org/syracuse/pstricks/rapporteurs>.

De façon générale, le site <http://melusine.eu.org/syracuse> est une mine d'or pour un grand nombre de thèmes et pour PSTricks en particulier.

Exercice E-9-5 Arbres

Comment obtenir l'arbre de probabilités suivant ?



Solutions des exercices

Solution de l'exercice 9-1 page 237

a Cette première question ne pose pas de problème si l'on a pensé à utiliser les coordonnées polaires. Dans la solution, on les active avec la commande `\SpecialCoor` mais, en pratique, les auteurs recommandent de toujours charger `pstricks-add` qui appelle automatiquement cette commande.

9-37

```
\usepackage{pstricks}
\SpecialCoor

\begin{pspicture}(0,0)(2,3)
  \psline(2,0)(0,0)(2;50)
  \psarc{->}(0,0){1}{0}{50}
\end{pspicture}
```

b Ici, le recours aux coordonnées polaires n'est plus de mise puisqu'on ne connaît ni la longueur des segments, ni l'angle entre eux. Le tracé des segments est immédiat, le problème se situe au niveau du tracé de l'arc de cercle représentant l'angle.

Cette question se résout facilement grâce à la possibilité de spécifier un angle par les coordonnées des vecteurs qui le définissent (cf. page 218). On a tendance à oublier trop facilement cette syntaxe !

9-38

```
\usepackage{pstricks}
\SpecialCoor

\begin{pspicture}(0,0)(2,3)
  \psline(2,0)(0,0)(2,3)
  \psarc{->}(0,0){1}{0}{(2,3)}
\end{pspicture}
```

c Il suffit d'ajouter une commande `\rput` au code précédent.

9-39

```
\usepackage{pstricks}
\SpecialCoor

\begin{pspicture}(0,0)(5,4)
  \rput(2,1){%
    \psline(2,0)(0,0)(2,3)
    \psarc{->}(0,0){1}{0}{(2,3)}
  }
\end{pspicture}
```

Solution de l'exercice 9-2 page 238

a La disposition des cercles au niveau des sommets d'un pentagone régulier ne devrait maintenant plus poser de problème. On l'obtient en utilisant des coordonnées polaires et avec l'instruction `\degrees[5]`.

Ici, on doit relier les cinq cercles par dix traits. Le premier point auquel il faut penser est d'utiliser des nœuds, en l'occurrence de type `\Cnode`, en ayant fixé le rayon auparavant. Ensuite, on pourrait relier les nœuds en saisissant dix commandes `\ncline` successives. La solution propose un peu mieux en automatisant cette construction.

9-40

```
\usepackage{pstricks-add}
\usepackage{multido}
\usepackage{ifthen}

\begin{pspicture}(-2.1,-2.1)(2.1,2.1)
  \degrees[5]
  \psset{radius=1mm,linewidth=0.4pt}%
  \multido{\ia=0+1}{5}{\Cnode(2;\ia){A\ia}}%
  \multido{\ia=0+1}{5}{%
    \multido{\ib=0+1}{5}{%
      \ifthenelse{\ia=\ib}{\ncline{A\ia}{A\ib}}%
    }%
  }%
\end{pspicture}
```

Le fait d'automatiser n'est pas nécessairement un avantage dans ce cas particulier. Après tout, recopier 10 fois une instruction presque identique est chose facile avec n'importe quel éditeur. Le plus grand avantage est que si on décide après coup que le polygone devrait comporter non pas cinq branches mais sept, l'adaptation sera très simple. On peut même se servir de ce code comme une base pour concevoir une commande personnelle où le nombre de sommets serait donné en paramètre. C'est d'ailleurs ce qui va avoir lieu dans la question b.

La première commande `\multido` devient maintenant assez classique. On remarquera juste que le nombre `\ia` peut également servir pour les noms de nœuds. Ici, on aura donc les nœuds A_0, \dots, A_4 .

Les deux commandes `\multido` suivantes sont emboîtées. Cela ne pose aucun problème. L'idée est de relier tous les nœuds A_i et A_j où i et j vont de 0 à 4.

L'utilisation de la commande `\ifthenelse` (cf. annexe C) permet de ne pas tracer un trait d'un nœud vers lui-même, ce qui se traduirait par un cercle barré. On aurait également pu interdire de tracer deux fois le même arc (un de A_i vers A_j et un autre de A_j vers A_i) mais dans l'exemple, cela ne pose pas de problème ; ceal deviendrait nécessaire si les nœuds étaient reliés par des arcs courbes par exemple.

b À partir de l'étude de la question précédente, la définition d'une telle commande est presque immédiate : il suffit de remplacer les occurrences des « 5 » par l'argument de la commande :

```

1 \newcommand\complet[1]{%
2   {%
3   \degrees[#1]
4   \psset{radius=1mm,linewidth=0.4pt}%
5   \multido{\ia=0+1}{#1}{\Cnode(2;\ia){A\ia}}%
6   \multido{\ia=0+1}{#1}{%
7     \multido{\ib=0+1}{#1}{%
8       \ifthenelse{\ia>\ib}{\ncline{A\ia}{A\ib}}{}}}%
9   }%
10 }
```

On notera la précaution qui consiste à tout placer dans un groupe (lignes 2 et 9). En effet, il ne faut pas que les initialisations dues aux commandes `\degrees` et `\psset` perturbent le reste d'un schéma qui utiliserait la commande `\complet`. Dans l'exemple, le groupe a été délimité par une paire d'accolade (lignes 2 et 9) mais \TeX prévoit deux autres couples de délimiteurs pour spécifier les groupes : `\bgroup... \egroup` et `\begingroup... \endgroup`. L'étude comparée de ces différents types de groupe sort très largement du cadre de cet ouvrage.

Une amélioration consisterait à paramétrer le rayon du graphe, soit *via* un argument supplémentaire au niveau de la commande, soit en utilisant une longueur qui conserverait cette dimension.

c La méthode de résolution est très proche de celle de la question a. Afin de pouvoir automatiser entièrement le tracé aboutissant aux nœuds externes, on a utilisé une astuce consistant à déclarer sept nœuds au lieu de cinq, les nœuds A_5 et A_6 étant identiques aux nœuds A_0 et A_1 . Cela permet d'accéder aux deux voisins sans être ennuyé par les bornes des indices. Cette astuce a été utilisée dans la troisième commande `\multido` du source suivant. Remarquons également la présence de trois initialisations élémentaires pour incrémenter les compteurs `\ia`, `\ib` et `\ic` en parallèle.

```

\usepackage{pstricks-add}
\usepackage{multido}
\usepackage{ifthen}

\begin{pspicture}(-2.1,-2.1)(2.1,2.1)
\degrees[5]
\psset{radius=1mm,linewidth=0.4pt}%
\multido{\ia=0+1}{7}{\Cnode(1;\ia){A\ia}}%
\multido{\ia=0+1}{7}{\Cnode(2;\ia){B\ia}}%
\multido{\ia=0+1,\ib=1+1,\ic=2+1}{5}{%
  \ncline{B\ib}{B\ia}
  \ncline{B\ib}{B\ic}
  \ncline{B\ib}{A\ib}
}%
\multido{\ia=0+1}{5}{%
  \multido{\ib=0+1}{5}{%
    \ifthenelse{\ia>\ib}{\ncline{A\ia}{A\ib}}{}}%
  }%
}%
\end{pspicture}

```

Solution de l'exercice E-9-3 page 458

Il n'y a aucun problème particulier pour les axes, l'instruction suivante donnant exactement le résultat voulu :

```
\psaxes[linewidth=0.4pt]{->}(0,0)(-5.2,0)(5.2,3.2)
```

Le tracé du graphe de la fonction pose plus de problème. Une première (mauvaise) idée serait d'écrire quelque chose comme :

```
\psplot{-5.2}{5.2}{2.71828 1 x div exp}
```

Passons sur le fait que la syntaxe utilise la notation polonaise inverse alors que l'on pourrait spécifier la formule sous la forme infixée traditionnelle :

```
\psplot[algebraic=true]{-5.2}{5.2}{2.71828^(1/x)}
```

En revanche, cette instruction poserait un problème sérieux. Au pire, on n'a pas de chance et la variable x va recevoir la valeur zéro, ce qui ne provoquera rien de spécial au moment de la compilation \LaTeX mais un flot d'injures lors d'une tentative de visualisation du fichier *PS*. Au mieux, on a de la chance (c'est le cas ici) et x va prendre une valeur très petite, en l'occurrence environ 0.106 si l'on a gardé la valeur par défaut de 50 pour le nombre de points d'interpolation. Or, avec une telle valeur de x , la fonction vaut environ 12370, ce qui donnera finalement une ligne quasiment verticale proche de l'axe des ordonnées.

On n'a guère le choix : la fonction n'est pas définie en zéro et il faudra tracer la courbe en deux parties. De plus, on a justement demandé de noter une des extrémités de la courbe avec une parenthèse. On arrive alors au code fonctionnel suivant :

```
\psset{algebraic=true,linewidth=1pt}
\psplot[arrows=-]{-5.2}{-0.01}{2.71828^(1/x)}
\psplot{0.8597}{5.2}{2.71828^(1/x)}
```

Il reste juste à comprendre d'où vient la valeur spéciale 0.8597. Il s'agit de la valeur (approchée) pour laquelle on a $e^{1/x} = 3.2$, c'est-à-dire la valeur de x pour laquelle la courbe de la fonction s'étendra exactement jusqu'à la frontière du repère. Cette méthode est peu satisfaisante car elle oblige l'utilisateur à résoudre une équation et à effectuer des calculs pour trouver les bornes. Avec cette fonction, on a eu de la chance : elle est monotone et la résolution de l'équation n'est pas vraiment difficile.

Une bien meilleure méthode, qui ne nécessite (presque) aucun calcul, consiste à utiliser l'environnement `psclip`. Le seul « calcul » que l'on a fait (de tête) a été de s'assurer que $e^{1/0.5} > 3.2$. On obtient finalement la solution :

E-36

```
\usepackage{pstricks-add}
\usepackage{multido}

\begin{pspicture}(-5.2,-0.7)(5.2,3.2)
\psaxes[linewidth=0.4pt]{->}(0,0)(-5.2,0)(5.2,3.2)
\begin{psclip}{\psframe[linestyle=none](-5.2,-0.2)(5.2,3.2)}
\psset{algebraic=true,linewidth=1pt}
\psplot[arrows=-]{-5.2}{-0.01}{2.71828^(1/x)}
\psplot{0.5}{5.2}{2.71828^(1/x)}
\end{psclip}
\end{pspicture}
```

Solution de l'exercice E-9-4 page 458

Voici le code, les explications suivent :

E-37

```
\usepackage{pstricks}
\SpecialCoor
\usepackage{multido}

\begin{pspicture}(-5,-0.5)(5,5)
\multido{\na=1+1}{179}{\psline[linewidth=.2pt](4.8;\na)(5;\na)}
\multido{\na=5+10}{18}{\psline[linewidth=.4pt](4.6;\na)(5;\na)}
\multido{\na=10+10}{17}{%
\psline[linestyle=gray](0;\na)(5;\na)
\rput*(4;\na){\na}}
\pswedge{5}{0}{180}
\end{pspicture}
```

Comme dans l'exercice précédent, le recours aux coordonnées polaires simplifie largement le travail.

Il est évidemment hors de question de tracer les graduations individuellement. On utilise pour cela la commande `\multido` du package `multido` (cf. annexe C). Il y a plusieurs écueils à éviter. Le premier est de ne pas se tromper sur le nombre des différents types de traits ! Le deuxième est de ne pas chercher à tracer les plus petits

traits en évitant les multiples de cinq. Enfin, dans PSTricks, un élément graphique tracé après un autre le recouvrira. L'ordre de tracé est donc important. On effectue ainsi, dans l'ordre, le tracé de :

- 179 petits traits (de 1 à 179) ;
- 18 traits moyens (de 5 à 175 de 5 en 5) ;
- 17 grands traits avec la mesure de l'angle (de 10 à 170 de 10 en 10) ;
- la frontière du rapporteur.

Pour obtenir les mesures des angles se réservant de la place au niveau des grands traits, on a utilisé la commande étoilée `\rput*`.

Solution de l'exercice E-9-5 page 459

La structure générale de l'arbre ne pose pas de problème. Notons juste que la racine est un nœud vide. Cela donne donc le squelette :

```
\pstree{\Tp}{%
  \pstree{\TR{$M$}}{%
    \TR{$P$}
    \TR{$\overline{P}$}
  }
  \pstree{\TR{$\overline{M}$}}{%
    \TR{$P$}
    \TR{$\overline{P}$}
  }
}
```

Il reste à faire grandir l'arbre de gauche à droite (et non de haut en bas) et à placer les différentes étiquettes sur les arcs. La présentation de gauche à droite est obtenue avec la valeur R affectée à la clé `treemode`. Les étiquettes sont composées grâce aux commandes `\naput` ou `\nbput`, selon qu'elles sont placées au-dessus ou au-dessous de l'arc. Ces commandes doivent figurer immédiatement après la commande de nœud `\T<type>` où l'arc aboutit. Enfin, la composition de l'étiquette parallèlement à l'arc est obtenue grâce à l'affectation de la valeur `:U` au mot clé `nrot`.

E-38

```
\usepackage{pst-tree}
\pstree[treemode=R,nrot=:U]{\Tp}{%
  \pstree{\TR{$M$}\naput{$0.001$}}{%
    \TR{$P$}\naput{$0.98$}
    \TR{$\overline{P}$}\nbput{$0.02$}
  }
  \pstree{\TR{$\overline{M}$}\nbput{$0.999$}}{%
    \TR{$P$}\naput{$0.03$}
    \TR{$\overline{P}$}\nbput{$0.97$}
  }
}
```

Chapitre 10

Solutions des exercices

Solution de l'exercice 10-1 page 257

L'affichage « à l'envers » sera obtenu en utilisant les indications de couches au niveau des commandes `\item` par ordre décroissant. Voici un code source complet permettant d'obtenir le résultat :

```
\documentclass[french]{beamer}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{lmodern}
\usepackage{babel}
\usetheme{Luebeck}
\title{Les très basses températures}
\author{Denis Bitouzé \and Jean-Côme Charpentier}
\begin{document}
\begin{frame}{Différentes méthodes}
\begin{enumerate}
\item<5-> Mélange frigorifique
\item<4-> Expansion des gaz
\item<3-> Hélium liquide
\item<2-> Magnétisme
\item<1-> Température inférieure à  $10^{-6}$  Kelvin
\end{enumerate}
\end{frame}
\end{document}
```

Contrairement à ce qui se passait avec l'affichage de haut en bas où on disposait du spécificateur automatique `<+>`, ici il faut indiquer explicitement les couches à afficher.

En toute rigueur, les commandes `\title` et `\author` semblent inutiles. Il est cependant préférable de toujours indiquer ces informations puisque certains thèmes les utilisent automatiquement. Ce n'est pas le cas de Luebeck, choisi pour cet exemple, mais d'autres thèmes, tels que Madrid, affichent ces informations sur chaque cadre.

Remarquons que, pour composer les grandeurs de façon conforme, il est préférable de recourir au package `siunitx`. On chargera donc ce package et, plutôt que Température inférieure à 10^{-6} Kelvin, on saisira :

Température inférieure à `\SI{e-6}{\kelvin}`

Solution de l'exercice 10-2 page 257

a En fait, tous les thèmes de couleurs intérieurs et extérieurs ne proposent que des variations autour du bleu. Plusieurs thèmes de couleurs généraux proposent également du blanc et du gris et seuls les thèmes crane et wolferine fournissent une couleur supplémentaire : jaune-orange pour crane et jaune sur l'intérieur et bleu sur l'extérieur pour wolferine.

b Une possibilité est de spécifier les couleurs demandées pour *tous* les éléments. C'est idiot pour trois raisons :

- La plupart des éléments sont en fait des sous-éléments. Par exemple, les textes mathématiques (`\setbeamercolor{math text}` et apparentés dans le listing ci-dessous) sont des sous-éléments des textes normaux.
- Le code est très répétitif. En informatique, c'est toujours douteux et, en l'occurrence, il existe des façons plus efficaces d'écrire le code présenté ci-dessous.
- Le résultat est absolument illisible : tout est composé ton sur ton !

En revanche, cela donne la liste exhaustive des éléments sur lesquels on peut jouer au niveau de la couleur. Il reste juste à comprendre un peu d'anglais pour voir à quels éléments se rapportent les différents mots-clés utilisés. Voici un exemple réel :

```

\documentclass[french]{beamer}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{lmodern}
\usepackage{babel}
\title{Présentation de Beamer}
\author{Denis Bitouzé \ \ Jean-Côme Charpentier}
\date{Septembre 2010}
\usetheme{PaloAlto}

\setbeamercolor{normal text}{fg=red,bg=white}
\setbeamercolor{alerted text}{fg=red, bg=white}
\setbeamercolor{example text}{fg=red, bg=white}

\setbeamercolor{structure}{fg=red, bg=red!25}
\setbeamercolor{local structure}{fg=red, bg=red!25}

\setbeamercolor{background canvas}{fg=red, bg=white}
\setbeamercolor{background}{fg=red, bg=white}

\setbeamercolor{palette primary}{fg=red, bg=red!25}
\setbeamercolor{palette secondary}{fg=red, bg=red!25}
\setbeamercolor{palette tertiary}{fg=red, bg=red!25}
\setbeamercolor{palette quaternary}{fg=red, bg=red!25}

\setbeamercolor{palette sidebar primary}{fg=red, bg=red!25}
\setbeamercolor{palette sidebar secondary}{fg=red, bg=red!25}
\setbeamercolor{palette sidebar tertiary}{fg=red, bg=red!25}
\setbeamercolor{palette sidebar quaternary}{fg=red, bg=red!25}

```

```

\setbeamercolor{math text}{fg=red, bg=red!25}
\setbeamercolor{math text inlined}{fg=red, bg=red!25}
\setbeamercolor{math text displayed}{fg=red, bg=red!25}
\setbeamercolor{normal text in math text}{fg=red, bg=red!25}

\setbeamercolor{titlelike}{fg=red, bg=red!25}
\setbeamercolor{title}{fg=red, bg=red!25}
\setbeamercolor{title in head/foot}{fg=red, bg=red!25}
\setbeamercolor{title in sidebar}{fg=red, bg=red!25}
\setbeamercolor{subtitle}{fg=red, bg=red!25}

\setbeamercolor{author}{fg=red, bg=red!25}
\setbeamercolor{author in head/foot}{fg=red, bg=red!25}
\setbeamercolor{author in sidebar}{fg=red, bg=red!25}
\setbeamercolor{institute}{fg=red, bg=red!25}
\setbeamercolor{institute in head/foot}{fg=red, bg=red!25}
\setbeamercolor{institute in sidebar}{fg=red, bg=red!25}
\setbeamercolor{date}{fg=red, bg=red!25}
\setbeamercolor{date in head/foot}{fg=red, bg=red!25}
\setbeamercolor{date in sidebar}{fg=red, bg=red!25}
\setbeamercolor{titlegraphic}{fg=red, bg=red!25}

\setbeamercolor{part name}{fg=red, bg=red!25}
\setbeamercolor{part title}{fg=red, bg=red!25}
\setbeamercolor{section name}{fg=red, bg=red!25}
\setbeamercolor{section title}{fg=red, bg=red!25}
\setbeamercolor{section in toc}{fg=red, bg=red!25}
\setbeamercolor{section in toc shaded}{fg=red, bg=red!25}
\setbeamercolor{section in head/foot}{fg=red, bg=red!25}
\setbeamercolor{section in sidebar}{fg=red, bg=red!25}
\setbeamercolor{section in sidebar shaded}{fg=red, bg=red!25}
\setbeamercolor{section number projected}{fg=red, bg=red!25}
\setbeamercolor{subsection name}{fg=red, bg=red!25}
\setbeamercolor{subsection title}{fg=red, bg=red!25}
\setbeamercolor{subsection in toc}{fg=red, bg=red!25}
\setbeamercolor{subsection in toc shaded}{fg=red, bg=red!25}
\setbeamercolor{subsection in head/foot}{fg=red, bg=red!25}
\setbeamercolor{subsection in sidebar}{fg=red, bg=red!25}
\setbeamercolor{subsection in sidebar shaded}{fg=red, bg=red!25}
\setbeamercolor{subsection number projected}{fg=red, bg=red!25}
\setbeamercolor{subsubsection in toc}{fg=red, bg=red!25}
\setbeamercolor{subsubsection in toc shaded}{fg=red, bg=red!25}
\setbeamercolor{subsubsection in head/foot}{fg=red, bg=red!25}
\setbeamercolor{subsubsection in sidebar}{fg=red, bg=red!25}
\setbeamercolor{subsubsection in sidebar shaded}{fg=red, bg=red!25}
\setbeamercolor{subsubsection number projected}{fg=red, bg=red!25}

```

```

\setbeamercolor{headline}{fg=red, bg=red!25}
\setbeamercolor{footline}{fg=red, bg=red!25}

\setbeamercolor{sidebar}{fg=red, bg=red!25}
\setbeamercolor{sidebar left}{fg=red, bg=red!25}
\setbeamercolor{sidebar right}{fg=red, bg=red!25}

\setbeamercolor{logo}{fg=red, bg=red!25}

\setbeamercolor{frametitle}{fg=red, bg=red!25}
\setbeamercolor{framesubtitle}{fg=red, bg=red!25}
\setbeamercolor{frametitle right}{fg=red, bg=red!25}

\setbeamercolor{caption}{fg=red, bg=red!25}
\setbeamercolor{caption name}{fg=red, bg=red!25}

\setbeamercolor{button}{fg=red, bg=red!25}
\setbeamercolor{button border}{fg=red, bg=red!25}
\setbeamercolor{navigation symbols}{fg=red, bg=red!25}
\setbeamercolor{navigation symbols dimmed}{fg=red, bg=red!25}
\setbeamercolor{mini frame}{fg=red, bg=red!25}

\setbeamercolor{block body}{fg=red, bg=red!25}
\setbeamercolor{block body alerted}{fg=red, bg=red!25}
\setbeamercolor{block body example}{fg=red, bg=red!25}
\setbeamercolor{block title}{fg=red, bg=red!25}
\setbeamercolor{block title}{fg=red, bg=red!25}
\setbeamercolor{block title alerted}{fg=red, bg=red!25}
\setbeamercolor{block title example}{fg=red, bg=red!25}

\setbeamercolor{item}{fg=red, bg=red!25}
\setbeamercolor{subitem}{fg=red, bg=red!25}
\setbeamercolor{subsubitem}{fg=red, bg=red!25}
\setbeamercolor{item projected}{fg=red, bg=red!25}
\setbeamercolor{subitem projected}{fg=red, bg=red!25}
\setbeamercolor{subsubitem projected}{fg=red, bg=red!25}
\setbeamercolor{enumerate item}{fg=red, bg=red!25}
\setbeamercolor{enumerate subitem}{fg=red, bg=red!25}
\setbeamercolor{enumerate subsubitem}{fg=red, bg=red!25}
\setbeamercolor{itemize item}{fg=red, bg=red!25}
\setbeamercolor{itemize subitem}{fg=red, bg=red!25}
\setbeamercolor{itemize subsubitem}{fg=red, bg=red!25}
\setbeamercolor{itemize/enumerate body}{fg=red, bg=red!25}
\setbeamercolor{itemize/enumerate subbody}{fg=red, bg=red!25}
\setbeamercolor{itemize/enumerate subsubbody}{fg=red, bg=red!25}
\setbeamercolor{description item}{fg=red, bg=red!25}

```

```

\setbeamercolor{bibliography item}{fg=red, bg=red!25}
\setbeamercolor{bibliography entry author}{fg=red, bg=red!25}
\setbeamercolor{bibliography entry title}{fg=red, bg=red!25}
\setbeamercolor{bibliography entry location}{fg=red, bg=red!25}
\setbeamercolor{bibliography entry note}{fg=red, bg=red!25}

\setbeamercolor{separation line}{fg=red, bg=red!25}
\setbeamercolor{upper separation line head}{fg=red, bg=red!25}
\setbeamercolor{middle separation line head}{fg=red, bg=red!25}
\setbeamercolor{lower separation line head}{fg=red, bg=red!25}
\setbeamercolor{upper separation line foot}{fg=red, bg=red!25}
\setbeamercolor{middle separation line foot}{fg=red, bg=red!25}
\setbeamercolor{lower separation line foot}{fg=red, bg=red!25}

\setbeamercolor{abstract}{fg=red, bg=red!25}
\setbeamercolor{abstract title}{fg=red, bg=red!25}

\setbeamercolor{verse}{fg=red, bg=red!25}
\setbeamercolor{quotation}{fg=red, bg=red!25}
\setbeamercolor{quote}{fg=red, bg=red!25}

\setbeamercolor{page number in head/foot}{fg=red, bg=red!25}

\setbeamercolor{qed symbol}{fg=red, bg=red!25}

\mode<presentation>
\begin{document}
\section{Un théorème célèbre}
\subsection{Fermat}
\begin{frame}{Théorème de Fermat}
Texte, math  $x^n+y^n=z^n$  et math hors texte :
\[x^n+y^n=z^n\]
\begin{theorem}
Il n'existe pas de nombres entiers non nuls  $x$ ,  $y$  et  $z$  tels que
\[x^n+y^n=z^n\]
dès que  $n$  est un entier strictement supérieur à  $2$ .
\end{theorem}
\begin{proof}
Trop longue pour tenir dans un cadre.
\end{proof}
\end{frame}
\subsection{Successeurs de Fermat}
\begin{frame}
...
\end{frame}
\end{document}

```

Solution de l'exercice 10-3 page 257

a Le package `fourier` indique que la fonte de texte par défaut sera Utopia (et initialise également les fontes mathématiques qui vont avec). Or Utopia est une fonte avec empattement. Donc Utopia est bien chargée mais, par défaut, elle n'est jamais utilisée dans une présentation. Si on veut vraiment Utopia, il va falloir changer les thèmes de fontes. Par exemple, demander dans le préambule un

```
\usefonttheme{serif}
```

b Les listings sont des environnements de type *verbatim* : les caractères sont lus de façon très particulière pour que les mécanismes de \TeX concernant les caractères spéciaux soient désactivés. Pour permettre à un cadre de contenir de tels matériels, il faut indiquer l'option `fragile` :

```
\begin{frame}[fragile]{Squelette d'un source}  
...
```

Le code ne présente pas d'autre erreur.

Chapitre 11

Exercices supplémentaires

Exercice E-11-4 Composition spéciale

On est parfois amené à introduire des méta-informations dans un code : un texte qu'il ne faudra pas taper tel quel mais qui indique la signification de ce qu'il faut taper. Par exemple, dans le listing ci-dessous, *nom fonction* signale qu'il faut saisir un nom de fonction et non pas le texte `nom fonction` lui-même. Pour bien distinguer une méta-information du texte brut, nous avons employé ici deux symboles pour encadrer, en l'occurrence `\texttriangle` et `\textriangle`, et le texte a été composé en italique. Voici le résultat :

```
E-39 int <nom fonction>(int n){
      if (n > 1) return n*<nom fonction>(n-1);
      else return 1;
    }
```

Comment obtenir cette forme sachant que le corps de l'exemple a été saisi ainsi :

```
\begin{lstlisting}[language=C]
int |nom fonction|(int n){
    if (n > 1) return n*|nom fonction|(n-1);
    else return 1;
}
\end{lstlisting}
```

Exercice E-11-5 Numéros de lignes personnalisés

Les souhaits des utilisateurs concernant leur mise en page sont une source inépuisable de défis. Celui-ci a été posé sur le forum `fctt` (cf. page 49). Le but était d'arriver à ce type de présentation :

```
E-40 1 int factorielle(int n){
      2     if (n > 1) return n*factorielle(n-1);
      3     else return 1;
      4 }
```

La réponse n'est pas forcément simple à mettre en œuvre. Elle demande un peu de travail sur les boîtes et donc une lecture préalable de l'annexe C également disponible au format *PDF* sur le site du livre.

La bande grise de gauche a une largeur de 12pt et elle est à 1 em des numéros. La ligne grise de droite fait 0,4pt d'épaisseur et elle est à 0,5 em des numéros.

Solutions des exercices

Solution de l'exercice 11-1 page 276

Si on ajoute `special` à la liste des mots clés grâce au paramètre `morekeywords`, cela ne fonctionnera pas bien pour deux raisons. Tout d'abord l'esperluette ne sera pas composée en gras (elle ne fera pas partie du mot clé) et ensuite le mot `special` isolé sera pris comme un mot clé ce qui ne doit pas être le cas.

En fait, on dispose (au moins) de deux solutions pour régler le problème. Tout d'abord, on peut demander à ce que ce soit le mot `&special` qui soit ajouté à la liste des mots clés. Cela ne fonctionne pas directement car un mot clé ne doit être composé que de lettres. Dans ce cas, `alsoletter` permet de régler définitivement le problème et on aboutit à la solution :

11-19

```
\usepackage{listings}
\usepackage{luximono}
\lstset{language=C, basicstyle=\ttfamily,
  alsoletter={&}, morekeywords={&special}}

\begin{lstlisting}
&special rec_non_term
int factorielle(int n){
  if (n > 1) return n*factorielle(n-1);
  else return 1;
}
\end{lstlisting}
```

En réalité, cette solution n'est pas totalement satisfaisante car la chaîne de caractère `&if`, par exemple, ne sera plus reconnue comme une esperluette suivie du mot clé `if`. Si on veut que l'esperluette soit associée à `special` sans qu'elle devienne pour autant une lettre, il vaut mieux passer par les directives et leur délimitation. Dans le cas présent, on va tirer parti du fait que le langage C a des directives et on ne va donc pas ajouter un mot clé mais une directive en se servant du paramètre `delim` pour gérer l'esperluette. Le `\lstset` initial devient :

```
\lstset{language=C, basicstyle=\ttfamily,
  moredirectives={special}, moredelim=*[directive]\&}
```

On objectera que cette solution fait que `&if` sera effectivement traité correctement (caractère `&` suivi du mot clé `if`) mais que, soudainement, `&include` va devenir une directive du langage... Si on utilise à la fois `&if` et `&include`, la solution risque de devenir pénible à mettre en œuvre !

On peut s'en sortir en utilisant le concept de classe d'offset proposé par `listings`. Cette notion n'a pas été abordée dans le livre et nous n'en dirons pas plus, laissant le lecteur découvrir cette notion et son application possible dans le problème actuel. L'idée de base est de définir une nouvelle classe de directives qui va contenir uniquement le mot `special` et qui aura l'esperluette comme délimiteur. Les autres directives ne seront pas influencées par cette classe.

Solution de l'exercice 11-2 page 276

En fait, on joue sur les mots lorsqu'on dit que les mots clés *restent* en noirs. Ils ne restent pas, il *deviennent* noirs, tous les autres composants étant gris. Cela donne immédiatement la réponse : la couleur de base est le gris et celle des mots clés est le noir. À part cette inversion de couleur, il n'y a rien de spécial dans le code.

11-20

```
\usepackage{listings}
\usepackage{xcolor}
\lstset{basicstyle=\ttfamily\color{gray},
keywordstyle=\color{black}}

\begin{lstlisting}[language=C]
int factorielle(int n){
    if (n > 1) return n*factorielle(n-1);
    else return 1;
}
\end{lstlisting}
```

Solution de l'exercice 11-3 page 276

Il n'est pas possible (ou alors difficilement) d'avoir un cadre uniquement sur une partie d'un listing. La première idée pour obtenir le résultat voulu est qu'il ne s'agit pas d'un listing mais de *deux* listings !

Par défaut, les listings ont un espacement vertical avant et après mesurant la moitié d'un interlignage. Sans précaution, deux listings à la suite seront donc séparés par une ligne vide. Pour éviter cela, il faut que le paramètre `belowskip` du listing encadré soit à zéro ainsi que le paramètre `aboveskip` du listing non encadré.

Le reste du code sert uniquement à obtenir le cadre ombré tel que spécifié :

11-21

```
\usepackage{listings}
\usepackage{xcolor}

\begin{lstlisting}[rulesepcolor=\color{black},
frame=shadowbox, belowskip=0pt]
Lignes avec un cadre
et une ombre autour du cadre.
\end{lstlisting}
\begin{lstlisting}[aboveskip=0pt]
Lignes sans cadre
Composition avec listings
\end{lstlisting}
```

En pratique, si cette façon de composer devait revenir souvent, on aurait tout intérêt à utiliser les styles de listing. Par exemple :

```
\lstdefinestyle{cadre}{rulesepcolor=\color{black},
frame=shadowbox, belowskip=0pt}
```

pour demander ensuite `\begin{listing}[style=cadre]`.

Solution de l'exercice E-11-4 page 471

On notera au préalable que les deux symboles `\textlangle` et `\textrangle` nécessitent le chargement du package `textcomp` pour être accessibles.

Le corps de l'exemple montre que c'est le caractère `|` qui va déclencher tout le mécanisme. La première idée est de faire de ce caractère un délimiteur (paramètre `delim`) mais, s'il est possible de donner un style pour le matériel qui va être délimité, ici en italique, voire même en italique en commençant par le caractère `\textlangle`, il n'est pas possible de donner un code de fin.

La seule catégorie qui permet d'avoir un code de début et de fin est l'échappement. Faire de `|` un caractère d'échappement s'obtient simplement avec la syntaxe `escapeinside=||`. Concernant les crochets et l'italique, il faut noter que les crochets eux-mêmes ne sont pas en italique. Les codes de début et de fin sont donc :

```
escapebegin=\textlangle\itshape, escapeend=\upshape\textrangle
```

On notera l'obligation de mettre un `\upshape` afin d'éviter d'avoir le crochets fermant en italique. C'est une commande d'emploi assez rare, vous êtes pardonné si vous n'y pas pensé à l'utiliser.

On a alors quasiment l'exemple complet. En fait, un désavantage de l'échappement c'est qu'il interrompt l'alignement vertical des caractères. C'est d'autant plus dommage qu'on utilise une fonte à chasse fixe. Pour garder l'alignement, il suffit de demander que la largeur de base soit exactement celle correspondant aux caractères. Avec la fonte Latin Modern, cela correspond à `0.5em`. Le code complet est alors :

E-39

```
\usepackage{listings}
\usepackage{textcomp}
\lstset{basicstyle=\ttfamily, basewidth=0.5em, escapeinside=||,
escapebegin=\textlangle\itshape, escapeend=\upshape\textrangle}
\begin{lstlisting}[language=C]
int |nom fonction|(int n){
  if (n > 1) return n*|nom fonction|(n-1);
  else return 1;
}
\end{lstlisting}
```

Si la fonte à chasse fixe de Latin Modern ne convient pas (par exemple parce que le gras n'est pas très marqué), on pourra préférer la fonte Luximono obtenue simplement en ajoutant `\usepackage{luximono}` dans le préambule du document ¹.

Cependant, les caractères `\textlangle` et `\textrangle` n'existent pas avec cette fonte et donc le package `textcomp` va émettre une erreur au moment de la compilation. Il suffit alors d'utiliser `\langle` et `\rangle` à la place de, respectivement, `\textlangle` et `\textrangle`.

1. En fait, il faut aussi récupérer les fichiers `.pfb` de cette fonte qui ne font pas directement partie de la \TeX Live en raison de problème de licence. Le plus simple pour ce faire est de recourir au script `getnonfreefonts` (cf. <http://tug.org/fonts/getnonfreefonts>).

Solution de l'exercice E-11-5 page 471

L'idée générale est de tout gérer avec le paramètre `numberstyle`. Le problème est qu'il va falloir procéder à des opérations avant et après avoir affiché le numéro de ligne, ce qui est délicat à spécifier de façon directe après ce paramètre. Le plus simple est d'indiquer une macro à un paramètre, le paramètre étant le numéro de ligne.

Dans le code proposé, on a utilisé une boîte pour permettre à du matériel de dépasser sur la gauche du listing. On a également utilisé des commandes PSTricks. On pourrait obtenir un résultat similaire sans utiliser cet outil mais il facilite quand même un peu les choses dans le cas présent.

E-40

```

\usepackage{listings}
\usepackage{pstricks}
\newcommand*\bord[1]{%
  \makebox[0pt][r]{%
    \rput[B](0,0){\psframe*[linecolor=gray]
                  (-12pt,8.5pt)(0,-3.5pt)}

    \hspace*{0.5em}%
  }%
  \makebox[0.5em][r]{#1}%
  \psline[linewidth=0.4pt,linecolor=gray]
        (0.5em,-3.5pt)(0.5em,8.5pt)
}
\lstset{basicstyle=\ttfamily, numberstyle=\bord}

\begin{lstlisting}[language=C, numbers=left]
int factorielle(int n){
  if (n > 1) return n*factorielle(n-1);
  else return 1;
}
\end{lstlisting}

```

Chapitre 12

Exercices supplémentaires

Exercice E-12-4 Construction d'une bibliographie

- a** Construire la base bibliographique `BiblioExercice.bib` du listing 12.2 page suivante en essayant de trouver, sur Internet, les trois premières entrées déjà saisies.
- b** Expliquer le résultat obtenu dans l'exemple suivant.

E-41

```
\usepackage[style=authoryear-comp]{biblatex}
\bibliography{BiblioExercice}
```

On n'oubliera pas de lire `\cite{ftb,texbook}`.

On n'oubliera pas de lire D. KNUTH 2003 ; D. E. KNUTH 1986.

- c** Quels reproches pourrait-on faire au fichier `BiblioExercice.bib` ?

Exercice E-12-5 Modification des mots clés

La locution latine *et al.*, employée pour remplacer les blocs occultés lorsqu'une liste est tronquée, est très répandue dans la littérature anglophone. Mais certains auteurs préfèrent, au moins pour les publications en français, utiliser l'expression « et collaborateurs » ou son abréviation « et coll. ».

Dans cet exercice de niveau assez difficile, on demande de répondre à cette préférence en proposant une commande *surchargeant* la configuration par défaut de `biblatex`.

On donne les indications suivantes :

- Les traductions des mots clés de `biblatex` dans la langue $\langle \text{langue} \rangle$ sont rassemblées dans le fichier $\langle \text{langue} \rangle.lbx$ livré avec ce package.
- Le mot clé gouvernant la substitution par la locution *et al.* est `andothers`.
- La commande permettant les $\langle \text{redéfinitions} \rangle$ des traductions des mots clés dans la langue $\langle \text{langue} \rangle$, à utiliser dans le fichier source `.tex`, est :

```
\DefineBibliographyStrings{\langle \text{langue} \rangle}{\langle \text{redéfinitions} \rangle}
```

```

1 |@article{journals/ijns/Chauvin91,
2 |  author   = {Yves Chauvin},
3 |  title    = {Constrained Hebbian Learning: Gradient Descent to Global
4 |             Minima in an N-Dimensional Landscape.},
5 |  journal  = {Int. J. Neural Syst.},
6 |  volume   = {2},
7 |  number   = {1-2},
8 |  year     = {1991},
9 |  pages    = {35-46},
10 | ee       = {http://dx.doi.org/10.1142/S0129065791000042},
11 | bibsource = {DBLP, http://dblp.uni-trier.de}
12 | }
13 |@BOOK{ftb,
14 |  AUTHOR = "Donald~E. Knuth",
15 |  TITLE={\LaTeX}book, {Composition informatique},
16 |  PUBLISHER={Vuibert},
17 |  ADDRESS={Paris},
18 |  ISBN={2-7117-4819-7},
19 |  NOTE={{F}rench translation of{\cite{texbook}}, by {{Jean-C{\^o}me} {
20 |      Charpentier}},
21 |  YEAR=2003}
22 |@Book{texbook,
23 |  author   = "Donald Ervin Knuth",
24 |  title    = "Computers \& Typesetting: The {\TeX}book",
25 |  volume   = "A",
26 |  publisher = "Addison Wesley",
27 |  year     = "1986",
28 |  address  = "Reading, Massachusetts",
29 |  ISBN     = "0-201-13447-0"
30 | }
31 |@Book{RAB1,
32 |  author   = {Gotlib, Marcel},
33 |  title    = {Rubrique-à-brac},
34 |  volume   = {1},
35 |  publisher = {Dargaud},
36 |  address  = {Paris},
37 |  isbn     = {2-205-01104-9},
38 |  series   = {16/22},
39 |  year     = {1979}

```

Listing 12.2: Base bibliographique BiblioExercice.bib

Solutions des exercices

Solution de l'exercice 12-1 page 313

Cette entrée ne présente pas beaucoup de mystères. Remarquons seulement que *J'ai lu* est l'éditeur, à inscrire dans l'attribut `publisher`, et que *Librio* est la collection qui peut être saisie dans l'attribut `series`. Rappelons par ailleurs que l'édition se note sous la forme d'un entier et non d'un ordinal, `biblatex` étant alors en mesure à la fois de le transformer en ordinal et de le traduire. Le fichier `.bib` répondant à la question est donc le suivant.

```

1 @Book{          voltaire200403,
2   title         = {Candide, ou l'optimisme},
3   author        = {Voltaire},
4   publisher     = {J'ai lu},
5   date          = {2004-03},
6   origdate     = {1759-01},
7   edition       = {2},
8   series        = {Librio},
9   language      = {french},
10  isbn          = {978-2290335369},
11  pagetotal     = {95}
12 }
```

Solution de l'exercice 12-2 page 313

Compte tenu de l'apparence des étiquettes, le style utilisé est très vraisemblablement le style `alphabetic`. Il suffit donc de saisir :

12-22

```

\usepackage[style=alphabetic]{biblatex}
\bibliography{MaBiblio}

Lorsque \textsc{Knuth} rédigeait en 1963 sa thèse \cite{knu63},
imaginait-il \dots{}
\printbibliography
```

Solution de l'exercice 12-3 page 313

Notons le fait que la citation prend ici la forme du numéro de la référence inscrit en exposant, comme dans la revue *Nature*. Nous employons donc la commande `\superscript`. Rappelons par ailleurs les commandes `\citeauthor` et `\citeyear`, vues page 306, qui permettent de citer les données des attributs `author` et `title` :

```
\usepackage{biblatex}
\bibliography{MaBiblio}
```

Lorsque `\citeauthor{knu63}` rédigeait en `\citeyear{knu63}` sa thèse `\supercite{knu63}`, imaginait-il `\dots` ?

```
\printbibliography
```

Solution de l'exercice E-12-4 page 476

a La première entrée a été trouvée en saisissant « Yves Chauvin Constrained Hebbian Learning » dans un moteur de recherche ; la deuxième a été obtenue grâce à un moteur de recherche ; la troisième a été recueillie en procédant à une recherche sur le titre « texbook » sur l'un des sites indiqués page 294 ; la dernière, quant à elle, a été saisie à la main (en utilisant toutefois les outils d'aide à la saisie de données bibliographiques de notre éditeur).

b Puisque le style utilisé compresse les citations, on s'attend à ce que le nom de l'auteur ne soit pas répété. Il l'est ici car biblatex considère qu'il s'agit de deux auteurs distincts puisque leurs noms sont saisis de façons différentes (même si elles sont implicitement identiques). Cet exemple montre la nécessité d'être rigoureux dans la saisie des données bibliographiques. Dans le cas qui nous occupe, nous verrons à la question suivante la meilleure méthode pour ce faire.

c Ce fichier souffre de quelques défauts :

1. Nous en constatons tout d'abord certains peu gênants : la casse n'est pas homogène sur l'ensemble du document ; il peut être aussi commode d'aligner verticalement les signes = ; bien que les guillemets, comme les accolades, puissent encadrer les données des attributs, il est préférable d'utiliser toujours les mêmes délimiteurs au sein d'une base bibliographique. La page <http://www.ctan.org/tex-archive/biblio/bibtex/utills> recense une série d'utilitaires permettant de gérer des fichiers .bib et notamment le programme bibtool¹ qui permet de les homogénéiser. MITTELBACH et GOOSSENS (2006, section 13.4) en détaille l'usage.
2. Les noms d'auteurs ne sont pas tous saisis selon le format

```
author = {<Nom>, <Prénom>},
```

3. Compte tenu de la citation insérée ligne 19, les deux ouvrages entrés lignes 13 et 21 ont le même auteur. Ses nom et prénoms, saisis différemment, posent le problème traité à la question question b. Pour qu'ils soient identiques dans toutes les entrées, la meilleure façon de procéder est de recourir à l'attribut @string et donc d'insérer en début de fichier :

```
@String{knuth = {Knuth, Donald Ervin}}
```

et de remplacer les lignes 14 et 22 par (noter l'absence d'accolades) :

```
author = knuth,
```

1. Le package biblatex est livré avec le fichier bibtoolrsc pour que bibtool, au départ prévu seulement pour bibtex, puisse tenir compte des nouveaux types d'entrées et attributs de biblatex. Voir le README de ce dernier pour plus de précisions à ce sujet.

Le listing 12.3 page suivante est une version du fichier `BiblioExercice.bib`, obtenue avec le programme `bibtool` lancé dans un terminal (cf. page 50) par la commande

```
bibtool -r braces BiblioExercice.bib
```

Toutes les modifications de forme (*casse*, alignement, accolades plutôt que guillemets) ont été effectuées automatiquement. On a aussi procédé manuellement :

- Aux modifications suggérées aux points 2 et 3.
- À l'adaptation de `bibtex` à `biblatex`, notamment le remplacement de l'attribut `note` (ligne 19 page 477) par `translator` et de l'attribut `ee` (ligne 10 page 477) par `doi`. Ce dernier, que nous n'avons pas étudié, est prévu pour les DOI (*Digital Object Identifier*) et, vu l'adresse <http://dx.doi.org>, c'est ce que semble être la donnée du champ `ee` original.
- À quelques autres ajustements, mineurs.

Solution de l'exercice E-12-5 page 476

Nous souhaitons ici ne modifier la traduction du mot clé `andothers` que pour le français. On recherche donc le fichier `french.lbx` dans notre distribution \LaTeX et le plus simple pour ce faire est de saisir dans un terminal :

```
kpsewhich_french.lbx
```

Nous ouvrons alors ce fichier ainsi localisé dans notre éditeur et, grâce à la fonction de recherche de ce dernier, nous y trouvons le mot clé `andothers`, ainsi déclaré :

```
andothers = {{et\addabbrvspace al\adddot}}{et\addabbrvspace al\adddot}},
```

Cette déclaration amène plusieurs remarques :

1. Les commandes `\addabbrvspace` et `\adddot` nous sont inconnues, mais leur signification peut se deviner : la première ajoute une espace après l'abréviation et la deuxième un point (abréviatif). Il semble donc être nécessaire, pour faire figurer cet espace et ce point, de ne pas les ajouter manuellement mais de recourir aux commandes employées. Même si cette nécessité est obscure, nous pressentons qu'il va être préférable de s'y conformer pour la redéfinition que nous devons mettre en œuvre.
2. Les deux parties qui composent la définition du mot clé `andothers` semblent identiques, ce qui ne manque pas d'être intrigant. Pour en comprendre la raison, examinons les autres déclarations ; un peu plus bas, celle du mot clé `newseries` :

```
newseries = {{nouvelle s\ 'erie}}{nouv\adddot\ s\ 'er\adddot}},
```

met en évidence le fait que la première partie définit la version complète du mot clé tandis que la seconde définit la version abrégée.

Il semble donc nécessaire de *surcharger* comme suit la traduction de `andothers` :

```
andothers = {{et collaborateurs}}{et coll\adddot}},
```

```

1 | @String{knuth = {Knuth, Donald Ervin}}
2 | @Article{
3 |   author      = {Chauvin, Yves},
4 |   title       = {Constrained Hebbian Learning: Gradient Descent to
5 |     Global Minima in an  $N$ -Dimensional Landscape},
6 |   journaltitle = {International Journal of Neural Systems},
7 |   volume      = {2},
8 |   number      = {1-2},
9 |   date        = {1991},
10 |  pages       = {35-46},
11 |  doi         = {10.1142/S0129065791000042},
12 |  bibsource    = {DBLP, http://dblp.uni-trier.de}
13 | }
14 | @Book{
15 |   author      = knuth,
16 |   title       = {Le \TeX{}book, Composition informatique},
17 |   publisher   = {Vuibert},
18 |   location    = {Paris},
19 |   isbn        = {2-7117-4819-7},
20 |   translator  = {Charpentier, Jean-Côme},
21 |   date        = {2003}
22 | }
23 | @Book{
24 |   author      = knuth,
25 |   title       = {Computers \& Typesetting: The \TeX{}book},
26 |   volume      = {A},
27 |   publisher   = {Addison Wesley},
28 |   date        = {1986},
29 |   location    = {Reading, Massachusetts},
30 |   isbn        = {0-201-13447-0}
31 | }
32 | @Book{
33 |   author      = {Gotlib, Marcel},
34 |   title       = {Rubrique-à-brac},
35 |   volume      = {1},
36 |   publisher   = {Dargaud},
37 |   location    = {Paris},
38 |   isbn        = {2-205-01104-9},
39 |   series      = {16/22},
40 |   date        = {1979}

```

Listing 12.3: Base bibliographique épurée

Reste maintenant à savoir où procéder à cette redéfinition. De façon analogue à ce que nous avons vu pour le fichier `biblatex.cfg`, elle peut être faite :

- directement dans ce fichier `french.lbx` (déconseillé) ;
- dans une copie de ce fichier ou, mieux, dans le fichier `biblatex.cfg` placés :
 - dans l’arborescence `TEXMFLOCAL`, pour une utilisation partagée (cf. page 90) ;
 - dans l’arborescence `TEXMFHOME`, pour une utilisation personnelle (cf. page 90) ;
 - dans le répertoire où se situe le fichier `.tex` ;
- directement dans le fichier `.tex`.

Enfin, puisque notre redéfinition ne doit concerner que le français, nous l’effectuons dans l’un des fichier et répertoire ci-dessus au moyen de la commande :

```
\DefineBibliographyStrings{french}{%  
  andothers = {{et collaborateurs}{et coll\adddot}},  
}
```

Chapitre 13

Solutions des exercices

Solution de l'exercice 13-1 page 328

a Le direct `\index{\LaTeX}` fonctionnerait très bien mais il est assez classique de vouloir classer le mot `\LaTeX` avec les mots commençant par la lettre L. Pour cela, on peut employer la syntaxe :

```
\index{latex@\LaTeX}
```

Il faut être cohérent dans le choix des majuscules et minuscules. Une même lettre ne sera pas confondue si elle est utilisée une fois sous forme capitale et une autre fois sous forme minuscule : il y aura deux entrées d'index.

b Si on saisit directement `\index{#include}`, on peut penser qu'on va avoir le même problème que celui de la question précédente mais en fait, c'est bien pire ici.

Au moment de la première compilation `PDF \LaTeX` , il n'y aura aucun souci. Tout continuera à se passer sans anicroche lorsqu'on utilisera le programme `texindy`. C'est lors de la compilation `PDF \LaTeX` suivante que les ennuis surviendront :

```
! You can't use `macro parameter character #' in vertical mode.
1.<ligne>  \item #
           include, <num. page>
```

qui permet peut-être de mieux se souvenir que le caractère `#` est spécial et qu'il ne doit pas être placé n'importe où. La syntaxe correcte pour indexer ce mot est :

```
\index{#include@#include}
```

ou, si on veut classer ce mot avec les mots commençant par la lettre I :

```
\index{include@#include}
```

c Pour composer l'expression en italique, on va employer `\textit` ou, mieux, `\emph`. Le problème de classement est alors le même que dans les questions précédentes et la solution est similaire :

```
\index{that's all folks@\emph{That's all Folks}}
```

On notera l'emploi des minuscules pour la clé de classement (à gauche du signe `@`).

Solution de l'exercice 13-2 page 328

Il faut faire apparaître `\textit{@}` en classant avec la clé @. Voici le code qui fonctionne. Le lecteur est invité à bien comprendre son fonctionnement !

```
\index{"@@"\textit{"@}}
```

Solution de l'exercice 13-3 page 328

Plusieurs coquilles se sont glissées dans le livre. Tout d'abord une partie de la solution se trouve à la suite de l'énoncé. Ensuite cette solution explique qu'il a fallu redéfinir l'environnement `\printindex` alors qu'il s'agit de l'environnement `theindex`. Enfin, on y indique comment les lettres encadrées ont été produites mais on ne précise pas où placer ce code. Reprenons tout cela.

Pour obtenir ce résultat, on a redéfini l'environnement `theindex` afin d'utiliser l'environnement `multicols` du package `multicol` pour avoir des colonnes équilibrées et permettre, au-dessus, un texte sur tout l'empagement. Comme il a été indiqué, cet environnement doit également redéfinir la commande `\item`. On a pris sa définition habituelle pour les index. Cela donne :

```
\renewenvironment{theindex}
  {\clearpage \section*{Index des concepts}
   Ce premier index ne présente que les concepts généraux rencontrés
   dans cet ouvrage. On verra par la suite \dots
   \renewcommand{\item}{\par\hangindent 40pt}
   \begin{multicols}{2}
   }
  {\end{multicols}}
```

Ensuite, le code qui sert à mettre les lettres dans un cadre avec des *filets* est placé dans la macro `\lettergroup` qui est celle appelée à chaque changement de lettre dans l'index. La macro `\nopagebreak` finale est une précaution pour éviter qu'une telle lettre se trouve en bas de colonne avec aucune entrée au-dessous.

```
\newcommand\lettergroup[1]{%
  \par\hrulefill\fbbox{#1}\hrulefill\par
  \nopagebreak
}
```

Chapitre 14

Solutions des exercices

Solution de l'exercice 14-1 page 349

Nous avons vu que, en cas d'homographie dans la base terminologique, l'attribut `parent` peut être utilisé à la place de `name` pour spécifier la clé du terme parent. Une recherche sur le mot « parent » dans la documentation du package indique comment procéder dans une telle situation.

On commence par créer une entrée « parent » dont la description doit être vide, son attribut `description` ne contenant que la commande `\nopostdesc`, destinée à supprimer le point final automatiquement ajouté lors d'une utilisation normale :

```
\newglossaryentry{bois}{%
  name=bois,
  description={\nopostdesc},
  plural={bois}
}
```

Ensuite sont créées les entrées « enfants » dont l'entrée parent doit être stipulée dans l'attribut `parent` :

```
\newglossaryentry{bois-arbre}{%
  parent={bois},
  description={1. tissu végétal des arbres},
  sort={1}
}
\newglossaryentry{bois-cervide}{%
  parent={bois},
  description={2. organe osseux ramifié sur la tête des cervidés},
  sort={2}
}
```

Ces entrées enfants ayant le même nom (celui de leur entrée parent), il convient d'indiquer au moyen de l'attribut `sort` l'ordre dans lequel on souhaite les voir figurer.

Notons par ailleurs que les termes tant enfants que parents peuvent être employés dans le texte courant, les numéros de pages des occurrences respectives apparaissant normalement dans chacun des cas.

Enfin, dès lors qu'on recourt à l'attribut `parent`, le style de glossaire employé doit nécessairement être l'un de ceux de nature arborescente. Celui chargé dans l'exemple suivant, `treenoname`, a la particularité de ne pas répéter le nom des entrées enfants.

E-42

```

\glossarystyle{treenoname}
\newglossaryentry{bois}{%
  name=bois,
  description={\nopostdesc},
  plural={bois}
}
\newglossaryentry{bois-arbre}{%
  parent={bois},
  description={1. tissu végétal des arbres},
  sort={1}
}
\newglossaryentry{bois-cervide}{%
  parent={bois},
  description={2. organe osseux ramifié sur la tête des cervidés},
  sort={2}
}

```

% page 1

Le mot `\og \gls{bois}\fg{}` est un exemple typique d'homographie. Le `\gls{bois-arbre}` tel que le fournissent les arbres...

% page 3

Les `\glspl{bois-cervide}` qui ornent la tête des cervidés...

Glossaire

bois 1

1. tissu végétal des arbres. 1
2. organe osseux ramifié sur la tête des cervidés. 3

Le mot « bois » est un exemple typique d'homographie. Le bois tel que le fournissent les arbres...

1

Les bois qui ornent la tête des cervidés...

3

Solution de l'exercice 14-2 page 349

Contrairement à `\printglossary` qui n'affiche que le glossaire principal, la commande `\printglossaries` affiche *tous* les glossaires définis mais, de ce fait, ne supporte pas l'option `title`. Cependant, la commande `\printglossary` accepte qu'en option lui soit spécifié le type de glossaire à afficher et nous avons vu que celui propre aux acronymes est de type `\acronymtype` qui, si l'option `acronym` est passée au package `glossaries`, vaut `acronym`. Il suffit donc de remplacer la commande `\printglossaries` par deux occurrences de `\printglossary`, la première sans option

pour composer normalement le glossaire principal (celui des termes ordinaires) et la deuxième avec les options adéquates :

```
\usepackage[xindy,acronym]{glossaries}  
\printglossary  
\printglossary[type=acronym,title=Acronymes et sigles]
```

Chapitre C

Solutions des exercices

Solution de l'exercice C-1 page 393

Les espacements verticaux sont tous gérés par des `\vspace` prenant un paramètre basé sur `\stretch`. Pour composer l'année complètement à gauche et la thèse à droite en réservant deux centimètres, on utilise un `\hfill` entre les deux éléments et on termine avec une commande `\hspace*`. Il est important d'utiliser la forme étoilée de cette commande, sinon, l'espace disparaîtrait puisqu'on se trouve en fin de ligne.

Le *retrait* est supprimé en affectant une valeur nulle à la longueur `\parindent`. On notera que comme cette affectation a lieu dans un environnement, elle n'aura d'action qu'à l'intérieur de cet environnement. Cela signifie qu'en dehors de la page de titre, on retrouvera le *retrait* habituel.

Le code permettant tout ce travail est le suivant :

C-23	<pre> \begin{titlepage} \setlength{\parindent}{0pt} \begin{center} Université de Munich \end{center} \vspace{\stretch{1}} Année 1884 \hfill Thèse\hspace*{1cm}\par \vspace{\stretch{2}} \begin{center} \bfseries\large Théodore Escherich\par Le phage T4 \end{center} \vspace{\stretch{3}} Imprimé le 24 septembre 2010 \end{titlepage} </pre>
------	---

Solution de l'exercice C-2 page 394

La commande `\newcommand` permet de définir une commande n'ayant qu'un seul paramètre optionnel et ce dernier en première position. Or on veut que le premier argument soit un paramètre obligatoire. La définition va donc commencer par :

```
1| \newcommand*\lasse[1]{%
```

L'astuce consiste alors à composer le texte en se servant de ce premier paramètre avant d'appeler une autre commande ayant également un paramètre optionnel.

```

2 | Les #1
3 | \lassetype
4 | }
5 | \newcommand*\lassetype[1][assertions]{%
6 |   #1 suivant%
7 |   \lassegenre
8 | }

```

On notera le caractère pourcent qui permet de ne pas avoir d'espace après le début du mot « suivant ».

Cette commande en appelle alors une dernière qui gère le deuxième argument optionnel et qui termine la phrase.

```

9 | \newcommand*\lassegenre[1][e]{%
10 |   #1s sont équivalent#1s
11 | }

```

Le résultat est le suivant :

E-43	<pre> \lasse{trois}\par \lasse{quatre}[thèses]\par \lasse{deux}[arbres][] </pre>	<pre> } Les trois assertions suivantes sont équivalentes } Les quatre thèses suivantes sont équivalentes } Les deux arbres suivants sont équivalents </pre>
------	--	---

Solution de l'exercice C-3 page 394

Le nombre 2 147 483 647 est le plus grand nombre positif acceptable par T_EX, donc il ne devrait normalement pas y avoir d'erreur.

Cela dit, il suffit de penser que ce nombre en notation romaine devrait s'écrire avec 2 147 483 lettres « M » suivies de « DCXLVII » ! Du point de vue de T_EX, il s'agit d'un mot de plus de deux millions de lettres et on peut alors suspecter qu'il aura du mal à gérer un tel monstre.

Incidentement, on peut se poser des questions sur la santé mentale d'un utilisateur qui voudrait afficher un nombre ne tenant même pas dans un tome complet de *l'Encyclopædia Universalis* !

Il ne faut pas en déduire que T_EX est trop limité. Sur l'ordinateur d'un des auteurs, avec une distribution classique, le nombre 732 424 000 donnait également une erreur, mais avec 732 423 000, la compilation se passait bien. Pour cela, T_EX a quand même manipulé en bloc un matériel de plus de sept cent mille caractères.

Solution de l'exercice C-4 page 395

Il est clair qu'il faut utiliser un compteur pour gérer le numéro d'exercice. Comme il est demandé que ce compteur soit réinitialisé à chaque section, on le déclarera avec la syntaxe suivante :

```
\newcounter{exercice}[section]
```

D'autre part, ce compteur devra être affiché sous la forme du numéro de section et du numéro d'exercice séparés par un point. Le plus simple sera de redéfinir la commande `\theexercice`.

Le reste de la présentation est relativement simple. Il convient cependant d'être attentif pour penser à tous les détails qui font la différence.

```
\renewcommand\theexercice{\thesection.\arabic{exercice}}
\newenvironment{exercice}[1]
{%
  \par\medskip
  \refstepcounter{exercice}
  \noindent\textbf{Exercice \theexercice} : #1\par\nobreak
}
{%
  \par\nobreak\hspace*\stretch{1}***
  \par\medskip
}
```

Dans le code, on a forcé des fins de paragraphe au début et à la fin de l'exercice. Un espace vertical a également été ajouté avec la commande `\medskip`.

L'incréméntation du compteur utilise `\refstepcounter` afin de permettre à l'utilisateur de référencer l'exercice avec une commande `\label`.

Enfin, on a interdit les coupures de page juste après le numéro d'exercice et entre la fin de l'exercice et la ligne composant les trois étoiles. Pour cela on a utilisé la commande `\nobreak` juste après la commande `\par`.

Il n'est pas évident de penser à tous ces détails lorsqu'on programme l'environnement pour la première fois. Ces réflexes s'acquièrent avec l'expérience... et lorsque des circonstances spéciales font que le code ne donne pas un résultat très heureux.

Solution de l'exercice C-5 page 395

Le code nécessite un grand nombre d'emboîtements de commandes `\ifthenelse` et une certaine rigueur dans la saisie sera très utile pour ne pas se perdre!

Lorsqu'on programme ce type d'interface afin de traduire des données brutes en données « humaines », le code, présenté page suivante, n'est pas très complexe mais il est long. Cela n'a rien de spécifique à \LaTeX d'ailleurs.

Les deux premières lignes de la commande `\affH` permettent de récupérer le nombre d'heures et de minutes écoulées depuis minuit. La division sur des entiers est une division euclidienne (division entière), elle permet d'obtenir directement le nombre d'heures. La deuxième ligne est le moyen classique d'obtenir le reste de cette division euclidienne.

On entre ensuite dans une longue suite de tests `\ifthenelse` emboîtés. Ces tests permettent de composer l'heure en traitant les cas 0 (minuit), 1 (heure au singulier), de 2 à 11 (heures au pluriel), 12 (midi) et de 13 à 23 (heures au pluriel en ayant retranché momentanément 12).

Le traitement des minutes est plus simple puisqu'il ne nécessite que deux tests pour traiter les cas 0 (pas de minute affichée), 1 (minute au singulier) et plus d'une minute (minutes au pluriel).

Enfin, le test final permet de savoir s'il faut ajouter « du matin » ou « du soir » ou encore rien du tout pour midi et minuit. Pour ces tests, on fait appel à des conditions complexes. La première regarde si le nombre d'heures est strictement supérieur à 0 et strictement inférieur à 12, et la deuxième si ce nombre est strictement supérieur à 12 et strictement inférieur à 24.

E-44

```

\usepackage{ifthen}
\usepackage{calc}
\newcounter{heure}
\newcounter{minute}
\newcommand\affH[1]{%
  \setcounter{heure}{#1/60}%
  \setcounter{minute}{#1-\value{heure}*60}%
  \ifthenelse{\value{heure}=0}{minuit}{%
    \ifthenelse{\value{heure}=1}{1~heure}{%
      \ifthenelse{\value{heure}<12}{\theheure~heures}{%
        \ifthenelse{\value{heure}=12}{midi}{%
          \ifthenelse{\value{heure}=13}{1~heure}{%
            \ifthenelse{\value{heure}<24}{\addtocounter{heure}{-12}%
              \theheure~heures\addtocounter{heure}{12}}{%
                minuit}}}}}%
  \ifthenelse{\value{minute}=0}{}{%
    \ifthenelse{\value{minute}=1}
      { 1~minute}{ \theminute~minutes}}%
  \ifthenelse{\value{heure}>0\and\value{heure}<12}
    { du matin}{}%
  \ifthenelse{\value{heure}>12\and\value{heure}<24}
    { de l'après-midi}{}%
}

\affH{0} --- \affH{720}\par
\affH{1} --- \affH{722}\par
\affH{360}\par\affH{\the\time}\par

```

Solution de l'exercice C-6 page 396

Il y a en fait deux problèmes : un facile à comprendre et un plus sournois.

Lorsqu'on indique la longueur `\linewidth` à l'environnement `minipage`, c'est la largeur de la minipage qui va être égale à l'empagement. Or la commande `\fbox` ajoute des espacements autour du matériel encadré : il y a un espacement de largeur `\fboxsep` de chaque côté de la minipage. L'épaisseur des deux traits verticaux, donnée par la longueur `\fboxrule`, contribue également au débordement. Il convient donc de retirer deux fois ces deux longueurs à la largeur d'empagement pour obtenir ce qui est souhaité. On utilisera le package `calc` pour faciliter la saisie des calculs.

En réalité, cela ne suffit pas à donner un cadre faisant exactement la largeur d'empagement. Il y a le problème sournois annoncé précédemment ! Dans l'argument de la commande `\fbox`, le `\end{minipage}` se termine par une fin de ligne non protégée.

Pour \LaTeX , une fin de ligne est équivalente à un caractère espace et on a ce que l'on appelle un *espace parasite*. Dans le cas présent, on avait bien pensé à l'espace parasite qui aurait suivi la première accolade ouvrante de la commande `\fbox` en plaçant un pourcent immédiatement après, mais il restait un autre endroit où il fallait se méfier.

L'environnement `minipage` ne compose pas d'espace à la suite de la ligne `\begin`. En revanche, il y en a un en fin d'environnement. Il faut donc soit placer l'accolade de fin de paramètre immédiatement après le `\end` de la `minipage`, soit, si l'on désire garder la même présentation au niveau du source, placer un caractère pourcent pour « cacher » la fin de ligne.

Voici un code qui donne le résultat voulu.

E-45

```
\usepackage{calc}
\noindent G\hrulefill D\par
\noindent\fbox{%
  \begin{minipage}{\linewidth
    -2\fboxsep-2\fboxrule}
    Matériel encadré
  \end{minipage}%
}
```

GD

Matériel encadré

1

Il est clair qu'en cas d'utilisation fréquente de cette présentation, on a tout intérêt à en faire une commande.

Solution de l'exercice C-7 page 396

Le principe général a été de diviser la largeur du texte par la largeur du caractère pour obtenir le nombre de caractères nécessaires au recouvrement. Une fois ce calcul effectué, on affiche le texte dans une boîte de largeur nulle, et on utilise une commande `\multido` pour afficher le nombre de caractères calculé précédemment. Cela donne le code final :

E-46

```
\usepackage{calc}
\usepackage{multido}
\newcounter{nb}
\newcommand\corrige[2]{%
  \setcounter{nb}{1*\ratio{\widthof{#1}}{\widthof{#2}}}%
  \makebox[0pt][l]{#1}%
  \multido{}{\value{nb}}{#2}{}%
}
\texttt{Un texte \corrige{raturé}{x} barré}
```

On remarquera le 1^* au début du calcul de nb : une commande `\ratio` n'a pas le droit d'apparaître au début d'un calcul. On pourra également remarquer le groupe vide étrange à la suite de la commande `\multido`. Il s'agit d'une précaution pour que l'espace qui suit la commande `\corrige` ne soit pas ignoré.

Solution de l'exercice C-8 page 396

La définition de la suite guide assez facilement l'écriture du code. On doit boucler tant que la valeur 1 n'est pas atteinte. Il s'agira donc d'une boucle `while`. Dans cette boucle, on affiche la valeur du terme courant et on calcule le terme suivant. En sortie de boucle, il ne faut pas oublier d'afficher la valeur 1 puisque la boucle se termine avant que ce dernier terme ne soit affiché.

Pour le calcul, on aura tout intérêt à utiliser la commande `\isodd`. L'utilisation du package `calc` permettra enfin d'écrire les calculs sous une forme simple.

On se servira d'un compteur `term` pour mémoriser la valeur du terme actuel de la suite. La boucle principale sera une commande `\whiledo`. Avant d'entrer dans cette boucle, il faudra initialiser le compteur `term` avec la valeur fournie en paramètre.

Le test de boucle regardera si la valeur du terme actuel est différente de 1. En réalité, on regardera si cette valeur est strictement supérieure à 1, ceci afin d'éviter les boucles infinies sur une donnée initiale négative.

Le code proposé est le suivant :

C-28

```
\usepackage{ifthen}
\usepackage{calc}
\newcounter{term}
\newcommand\syracuse[1]{%
  \setcounter{term}{#1}%
  \whiledo{\value{term}>1}{%
    \theterm,
    \ifthenelse{\isodd{\value{term}}}{
      {\setcounter{term}{\value{term}*3+1}}
      {\setcounter{term}{\value{term}/2}}%
    }
  }
  1
}
\syracuse{7}
```

Nous terminerons par une remarque générale. Quel que soit le langage de programmation, les boucles `while` peuvent être dangereuses puisqu'il faut être certain que la condition de sortie sera effectivement atteinte.

En réalité, nous ne pouvons même pas être totalement certain que cette commande n'entrera jamais dans une boucle sans fin. Nous pouvons toutefois avoir une certaine confiance puisqu'il existe une conjecture qui affirme que la suite de Syracuse atteint toujours la valeur 1, quelle que soit la valeur initiale (strictement positive). Si, par extraordinaire, un lecteur trouvait un contre-exemple, il pourrait en faire part à l'Académie des sciences... ou aux auteurs qui transmettront!