

Jérôme BASTIEN

ACTIVITÉ D'INTERSEMESTRE

Notice d'introduction à \LaTeX 2 ϵ

Du 29 au 31 janvier 2007

(Version révisée par Benjamin Blunier et compilée le 23 janvier 2007)

Table des matières

Avant-propos	4
1 Prise en main	6
1.1 Comment créer un document	6
1.2 Les premiers fichiers \LaTeX	7
2 Structuration et mise en forme d'un document	11
2.1 Quelques principes généraux	11
2.1.1 Espaces	11
2.1.2 Caractères spéciaux	11
2.1.3 Commandes \LaTeX	11
2.1.4 Commentaires	12
2.2 Les principes de mises en page du document	12
2.2.1 Classes de documents	12
2.2.2 Gros documents	13
2.3 La mise en page proprement dite	14
2.3.1 Chapitres et sections	14
2.3.2 Tables des matières	15
2.3.3 Références croisées	15
2.3.4 Environnements	17
2.3.5 Modification de l'aspect des caractères	18
3 Écrire des mathématiques	19
3.1 Les mathématiques standards sous \LaTeX	19
3.1.1 Généralités	19
3.1.2 Groupements en mode mathématique	21
3.1.3 Éléments d'une formule mathématique	21
3.1.4 Alignements verticaux	24
3.1.5 Taille des polices mathématiques	25
3.1.6 Insertion de texte en mode mathématique	25
3.2 Les mathématiques avec les packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$	26
3.3 Liste des symboles mathématiques	26
3.4 Écrire des textes mathématiques	26

4	Tableaux, flottants, figures	28
4.1	Tableaux	28
4.2	Objets flottants	29
4.2.1	Généralités	29
4.2.2	Créations de tableaux flottants	30
4.2.3	Créations de figures flottantes	31
5	Bibliographie et index	36
5.1	Bibliographie	36
5.2	Index	38
6	Compléments	41
6.1	Présentations à la façon PowerPoint	41
6.2	Réalisation d'un sujet d'examen	43
6.3	Algorithmes	43
6.4	Listings	44
6.5	L ^A T _E X et d'autres logiciels de calculs	45
6.6	Créer ses propres packages	46
6.7	Un dernier exercice pour finir	47
6.8	Le reste de L ^A T _E X?	47
A	Ce qu'on peut trouver sur L^AT_EX	48
A.1	Ouvrages	48
A.2	Sur sa propre machine	48
A.3	Sur le web	49
B	Les mathématiques avec les packages de l'$\mathcal{A}\mathcal{M}\mathcal{S}$-L^AT_EX [8]	50
	Bibliographie	75

Avant-propos

Introduction

L^AT_EX est un logiciel de traitement de texte scientifique, libre et utilisé par l'ensemble de la communauté mathématique dans le monde (éditeur, revues scientifiques, enseignants, chercheurs, ...). Sa portabilité et sa disponibilité pour la plupart des plateformes en font un format d'échange très utilisé. Logiciel libre, il s'enrichit chaque jour de nouvelles macros, mises à la disposition de la communauté.

Très complet, il permet de gérer la mise en page de façon très précise, d'écrire des équations mathématiques, de faire une bibliographie, d'insérer des graphiques ... On peut aussi dessiner des symboles chimiques, des partitions de musique, des hiéroglyphes, écrire en arabe, en hébreux, en cyrillique.

Ses nombreuses possibilités font de lui un outil utilisé par une communauté de plus en plus vaste (égyptologues, musiciens, linguistes, et même les rédacteurs du Mat'matronche, le trombinoscope des étudiants de l'UTBM...).

En annexe A, on trouvera une liste non exhaustive¹ de quelques références (ouvrages ou URL) qu'on pourra trouver sur L^AT_EX.

Les objectifs de l'atelier L^AT_EX sont les suivants :

- Donner en quelques heures les bases minimales du traitement de texte L^AT_EX.
- Les équations mathématiques
- Compléments : Différents styles L^AT_EX, bibliographie, index, graphiques, écrire en cyrillique, grec ancien, transférer des résultats matlab/maple vers L^AT_EX, faire des figures, des tableaux, ...

En fait, L^AT_EX est surtout une (formidable!!) façon de voir. Une fois qu'on a compris les bases et les grands principes, on doit pouvoir se débrouiller seul.

Avertissement

Il existe des tas d'ouvrage, de références sur le web permettant de s'initier à L^AT_EX (voir annexe A). Je me suis inspiré de certaines d'entre elles, que je citerai, notamment du très bon « joli manuel pour L^AT_EX 2_ε »² de Benjamin Bayart [5], ou encore de la non moins bonne « courte (?) introduction à L^AT_EX 2_ε » [15], qui est livré avec ses sources, et dont je me suis énormément inspiré³.

J'ai essayé dans cette documentation de rassembler les éléments de base, vous permettant de vous initier à L^AT_EX. Puis, vous irez consulter différentes documentations.

¹impossible d'être exhaustif!!

²Voir la bibliographie page 75.

³De plus en plus en regardant cette introduction, je me dis que j'aurais pu la donner telle qu'elle; elle est vraiment bien faite ...

Pour chaque chapitre, se trouvent des fichiers d'exemple en \LaTeX , organisés en répertoires portant des noms identiques à ceux des chapitres. Étudiez-les, compilez-les, essayez de les améliorer. Dans cette documentation, ceux-ci sont repris et commentés, les commentaires disparaissent au fur et à mesure de vos progrès.

Il est impossible d'être exhaustifs ; aussi, ai-je tenté de rassembler les notions minimales pour être un bon \LaTeX iste. Je me suis appuyé sur quelques années de pratiques de \LaTeX , mais il se peut que les exemples fournis comportent des erreurs ou des lourdeurs. J'ai acquis un certain nombre d'habitudes, parfois mauvaises ; en vous présentant cette notice et en travaillant avec vous, je compte vous voir progresser mais aussi apprendre moi-même.

Les versions de \LaTeX 2.09 et \LaTeX 2 ϵ

Je ne rentrerai ici pas dans les détails, mais la version de \LaTeX présentée ici s'appelle en fait la version $\text{\LaTeX}2\epsilon$, par opposition à la version \LaTeX 2.09, qui a disparu. Cette version précédente n'est plus développée, mais les compilateurs $\text{\LaTeX}2\epsilon$ acceptent encore des documents écrits pour \LaTeX 2.09, pour ne pas pénaliser les gens qui auraient encore des vieux fichiers⁴. La version \LaTeX 2.09 est caractérisée, entre autres, par le fait que le document commence par `\documentstyle` et non `\documentclass`. Dans cette ancienne version, il n'est pas fait de différence entre les options et les packages. Pour plus de détails, on pourra consulter [12]. Quoi qu'il en soit, ne préparez plus vos fichiers \LaTeX avec \LaTeX 2.09.

Les versions de \LaTeX évoluent de toute façon. Un projet est mené pour créer une version qui remplacerait $\text{\LaTeX}2\epsilon$, il s'agit du projet \LaTeX 3 (voir par exemple <http://tex.loria.fr/texlive-htmldoc/latex/ltx3info/ltx3info.html>).

Site web

Ce document est mis en ligne sur le site <http://utbmjb.chez-alice.fr/>. On y trouvera aussi les petits fichiers \LaTeX qui apparaissent dans cette notice.

Merci !!

Merci à l'UTBM qui a permis l'ouverture de l'activité \LaTeX pendant l'intersemestre.

Merci à l'équipe technique du département Génie Informatique qui a installé et testé \LaTeX dans ses salles, L'ouverture de l'activité n'aurait pu avoir lieu sans leur aide précieuse.

J'ai pu faire cette documentation en m'aidant des sources \LaTeX de la célèbre « courte (?) introduction à $\text{\LaTeX}2\epsilon$ », écrite par Tobias Oetiker, traduite en français par Matthieu Herrb. Voir [15]. Cette courte introduction et ses sources sont disponibles à l'adresse suivante : <http://www.laas.fr/~matthieu/cours/latex2e>.

Merci enfin à tout ceux qui, de près ou de loin, travaillent pour le développement des logiciels libres, dont \LaTeX fait partie.

Bonne découverte de \LaTeX .

⁴D'autres logiciels, que je ne citerai pas ici, n'ont pas ce soucis !

Chapitre 1

Prise en main

1.1 Comment créer un document

Il faut d'abord éditer un document avec un éditeur de texte; des exemples élémentaires sont fournis et commentés dans la section 1.2.

Ensuite il faut compiler son document soit en utilisant les boutons de raccourci des logiciels (Kile ou TeXniccenter), soit avec la console en tapant :

```
latex monfichier
```

où `monfichier.tex` est le nom sous lequel vous avez sauvegardé votre fichier latex. Apparaissent un certain nombre de messages, puis des fichiers se créent. On peut alors, si tout s'est bien passé, prévisualiser¹ le résultat en ouvrant le fichier dvi :

```
xdvi monfichier
```

On modifie alors le fichier \LaTeX , on le complète, on recompile,

En fin de travail, on en fait ensuite un fichier postscript :

```
dvips -o monfichier.ps monfichier
```

ou le visualiser en faisant

```
gv monfichier.ps
```

Si on veut créer à partir du postscript un document pdf, il suffit de faire

```
ps2pdf monfichier.ps monfichier.pdf
```

Il est aussi possible de compiler directement le document en pdf avec la commande suivante (il est nécessaire dans ce cas que les figures soient au format pdf ou jpeg) :

```
pdflatex monfichier.pdf
```

¹On voit un document proche du document final, hormis certains détails parfois.

1.2 Les premiers fichiers L^AT_EX

Vous verrez dans l'exemple `exemple01.tex` un exemple très simple de fichier L^AT_EX :

```
Fichier : exemple01.tex

\documentclass[11pt,a4paper]{article}

\begin{document}

Cela est un premier exemple de fichier \LaTeX.
Mais il est très simple et les accents
sont lourds à écrire et la façon de le faire n'est pas naturelle !

\end{document}
```

Ce document, comme tous les fichiers L^AT_EX commencera par `\documentclass`. Cette instruction est suivie d'un certain nombre d'argument entre crochets : ce sont les options choisies². Puis vient entre accolades la classe du document, qui sera différentes selon les besoins de l'utilisateur (voir chapitre 2).

Entre les deux instructions `\begin{document}` et `\end{document}`, se trouve le corps du texte.

La partie située entre les instructions `\documentclass` et `\begin{document}` s'appelle l'entête ou le préambule du texte. Cette partie est indépendante du corps du texte.

Comme le montre l'exemple `exemple01.tex`, il est pénible de saisir les accents³. On se sert alors du préambule (ou de l'entête) : cette partie contiendra toutes les instructions permettant la mise en page et l'utilisation de packages supplémentaires⁴ ou de macros personnelles⁵. On rajoute dans le préambule les packages `inputenc` et `fontenc` qui permettent de gérer les mots accentués. Voir le fichier `exemple05.tex` :

```
Fichier : exemple05.tex

\documentclass[11pt,a4paper]{article}

% utilisation des caractères accentués du clavier français
\usepackage[latin1]{inputenc}

% Codage de la fonte pour césure des mots accentués
\usepackage[T1]{fontenc}

\begin{document}

Cela est un deuxième exemple de fichier \LaTeX.
Il est très simple et les accents
sont moins pénibles à écrire. On peut aussi
taper ses cédilles de façon plus agréable.

\end{document}
```

À la base, L^AT_EX était fait par et pour les anglo-Saxons. Il a donc fallu le franciser, ce que fait le package `babel`, comme le montre le fichier `exemple10.tex`.

²Ici, on veut un texte en 11 points. On aurait pu choisir 10 ou 12 points. On a choisi un format A4.

³Néanmoins, à la base, cela permet d'éviter les problèmes sur les claviers sans lettres accentuées : L^AT_EX n'est pas franchouillard !

⁴Un package permet de faire de chose que ne permet pas de faire le L^AT_EX de base. Ce sont les différentes contributions de toutes les personnes qui améliore L^AT_EX et elles sont distribuées gratuitement. Certains auteurs parlent « d'extensions » ou de « modules », à la place du terme anglais « package ».

⁵C'est l'équivalent de fonctions, que chacun programme à sa guise.

Fichier : exemple10.tex

```

\documentclass[11pt,a4paper]{book}

% utilisation des caractères accentués du clavier français
\usepackage[latin1]{inputenc}

% Codage de la fonte pour césure des mots accentués
\usepackage[T1]{fontenc}

% déclaration de babel.
% Si multilingue : \usepackage[english,french,serbian, tec ...]{babel}
\usepackage[french,english]{babel}

\title{Exemple 10}
\author{Moi Même}

%%% corps du document
\begin{document}
\selectlanguage{french}

\maketitle

\chapter{Intérêt de babel}
Dans ce document, on voit un des innombrables intérêts
d'utiliser babel.

\selectlanguage{english}

\chapter{Interest of babel}
In this chapter, we can see how it is interesting
to use babel.
\end{document}

```

On voit apparaître dans l'entête du fichier `exemple10.tex`, deux commandes qui permettent de définir un titre (`\title`) et un auteur (`\author`). Ces données permettent à la commande `\maketitle`⁶ d'imprimer le titre⁷ et le nom de l'auteur. L'emplacement de ces éléments est totalement géré par \LaTeX . Elles sont optionnelles. Si on fait un titre (en appelant `\maketitle`) les deux commandes `\title` et `\author` sont obligatoires⁸. Si on omet la date, \LaTeX mettra par défaut la date qui correspond au moment de la compilation⁹. Si on veut forcer une date particulière, il suffit d'écrire par exemple :

```
\date{30 décembre 2003}
```

Nous verrons au chapitre 2 ce que fait l'instruction `\chapter`.

Les Anglo-Saxons ont aussi leurs habitudes de marges et il faut les personnaliser, comme le montre l'entête du fichier `exemple15.tex`.

⁶souvent placée en début de document, logique ...

⁷Le titre des fichiers exemples reprendra le nom du fichier, mais c'est une convenance personnelle, pour m'y retrouver. Vous y mettez ce que vous voulez.

⁸l'oubli de l'une d'entre elle provoquera un avertissement à la compilation.

⁹c'est bien pratique quand on fait plusieurs versions.

L'utilisation du package `vmargin` permet de redéfinir les marges adaptées à un document A4 avec la commande `\setpapersize{A4}`.

Fichier : `exemple15.tex`

```
\documentclass[11pt,a4paper]{article}

%%% différents package

% utilisation des caractères accentués du clavier français
\usepackage[latin1]{inputenc}

% Codage de la fonte pour césure des mots accentués
\usepackage[T1]{fontenc}

% déclaration de babel.
% Si multilingue : \usepackage[english,french,serbian, tec ...]{babel}
\usepackage[french]{babel}

%%% gestion des marges.avec vmargin
\usepackage{vmargin}
\setpapersize[portrait]{A4}

\title{Exemple 15}
\author{Moi Même}

%%% corps du document

\begin{document}
\selectlanguage{french}

\maketitle

Dans ce document, on voit qu'il faut
gérer la mise en page, gérer la mise en page, gérer la mise en page,
gérer la mise en page, gérer la mise en page, gérer la mise en page,
gérer la mise en page, gérer la mise en page, gérer la mise en page,
gérer la mise en page, gérer la mise en page, gérer la mise en page,
gérer la mise en page, gérer la mise en page, gérer la mise en page,
gérer la mise en page, gérer la mise en page, gérer la mise en page,
gérer la mise en page, gérer la mise en page, gérer la mise en page.
\end{document}
```

On voit que l'entête du document prend de plus en plus d'importance; qu'à cela ne tienne, comme le montre le fichier `exemple20.tex`.

Fichier : `exemple20.tex`

```
\documentclass[11pt,a4paper]{article}

\input{stylebase}

\title{Exemple 20}
\author{Moi Même}
```

```
\begin{document}
\maketitle

Dans ce document, on voit qu'on peut dissocier
le texte de la mise en page, et même en terme de fichier.
\end{document}
```

Dans ce fichier, l'instruction `input` permet d'aller chercher un autre fichier, appelé fichier de style, où on gère tous ses paramètres, indépendamment du texte. Voir le fichier `stylebase.tex` :

Fichier : `stylebase.tex`

```
%%% différents package

% utilisation des caractères accentués du clavier français
\usepackage[latin1]{inputenc}

% Codage de la fonte pour césure des mots accentués
\usepackage[T1]{fontenc}

% déclaration de babel.
% Si multilingue : \usepackage[english,french,serbian, tec ...]{babel}
\usepackage[french]{babel}

%%% gestion des marges.avec vmargin
\usepackage{vmargin}
\setpapersize[portrait]{A4}
```

Avec un peu d'habitude, on se constituera différents fichiers de style, personnalisés.

Chapitre 2

Structuration et mise en forme d'un document

Ce chapitre reprend une partie des chapitre 1 et 2 de [15].

2.1 Quelques principes généraux

2.1.1 Espaces

Les caractères d'espacement, tels que les blancs ou les tabulations sont traités de manière unique comme un seul espace \LaTeX . Plusieurs blancs consécutifs sont considérés comme une seule espace¹. Les espaces en début d'une ligne sont en général ignorées et un seul retour à la ligne est traité comme une espace.

Une ligne vide entre deux lignes de texte marque la fin d'un paragraphe. Plusieurs lignes vides sont considérées comme une seule ligne vide.

2.1.2 Caractères spéciaux

Les symboles suivants sont des caractères réservés qui, soit ont une signification spéciale dans \LaTeX , soit ne sont pas disponibles dans toutes les polices. Si vous les saisissez directement dans votre texte, ils ne seront pas imprimés et forceront \LaTeX à faire des choses que vous n'avez pas voulues.

`$ & % # _ { } ~ ^ \`

Comme vous le voyez, certains de ces caractères peuvent être utilisés dans vos documents en les préfixant par un antislash :

Exemple 2.1. `$ & % # _ { }` seront codés par `\$ \& \% \# _ \{ \}`

2.1.3 Commandes \LaTeX

Les commandes \LaTeX sont sensibles à la casse des caractères (majuscules ou minuscules) et utilisent l'un des deux formats suivants :

- soit elles commencent par un antislash `\` et ont un nom composé uniquement de lettres. Un nom de commande est terminé par une espace, un chiffre ou tout autre caractère qui n'est pas une lettre ;
- soit elles sont composées d'un antislash et d'un caractère spécial (non-lettre) exactement.

¹En langage typographique, *espace* est un mot féminin

\LaTeX ignore les espaces après les commandes. Si vous souhaitez obtenir un blanc après une commande, il faut ou bien insérer `{ }` suivi d'un blanc ou bien utiliser une commande d'espacement spécifique de \LaTeX , ou taper un antislash juste après la commande.

Exemple 2.2. L'écriture de

```
\LaTeX{} et \TeX\ sont chouettes. \LaTeX est beau.
Il aime \LaTeX.
```

donnera

\LaTeX et \TeX sont chouettes. \LaTeX est beau. Il aime \LaTeX .

2.1.4 Commentaires

Quand \LaTeX rencontre un caractère `%` dans le fichier source, il ignore le reste de la ligne en cours. C'est utile pour ajouter des notes qui n'apparaîtront pas dans la version imprimée.

2.2 Les principes de mises en page du document

2.2.1 Classes de documents

La première information dont \LaTeX a besoin en traitant un fichier source est le type de document que son auteur est en train de créer. Ce type est spécifié par la commande `documentclass` :

```
\documentclass[option]{classe}
```

Ici `classe` indique le type de document à créer. Le tableau 2.1 donne la liste de quelques classes de documents. Le paramètre `option` permet de modifier le comportement de la classe de document. Les options sont séparées par des virgules. Les principales options disponibles sont présentées dans le tableau 2.2 on the next page.

article	pour des articles dans des revues scientifiques, des présentations, des rapports courts, des documentations, des invitations, etc.
report	pour des rapports plus longs contenant plusieurs chapitres, des petits livres, des thèses, etc.
book	pour des vrais livres.
slides	pour des transparents.
beamer	pour des présentations à la façon PowerPoint,
amsart	analogue de la classe article, faite par l'AMS.
amsbook	analogue de la classe book, faite par l'AMS.

TAB. 2.1 – Classes de documents.

Exemple : un fichier source pour un document \LaTeX pourrait commencer par la ligne

```
\documentclass[11pt,twoside,a4paper]{article}
```

10pt, 11pt, 12pt	définit la taille de la police principale du document. Si aucune option n'est présente, la taille par défaut est de 10pt .
a4paper, letterpaper, ...	définit la taille du papier. Le papier par défaut est letterpaper , le format standard américain. Les autres valeurs possibles sont : a5paper , b5paper , executivepaper , et legalpaper .
fleqn	aligne les formules mathématiques à gauche au lieu de les centrer.
leqno	place la numérotation des formules à gauche plutôt qu'à droite.
twocolumn	demande à L ^A T _E X de formater le texte sur deux colonnes.
twoside, oneside	indique si la sortie se fera en recto-verso ou en recto simple. Par défaut, les classes article et report sont en simple face alors que la classe book est en double-face
draft	Cette option, que l'on utilise lors de l'élaboration du document, permet de voir les débordements de lignes, indiqués par une marque noire dans la marge. Elle cache aussi les figures ce qui accélère la compilation du document. Option à désactiver pour faire une sortie finale du document.

TAB. 2.2 – Quelques options.

elle informe L^AT_EX qu'il doit composer ce document comme un article avec une taille de caractère de base de onze points et qu'il devra produire une mise en page pour une impression *double face* sur du papier au format A4².

2.2.2 Gros documents

Lorsque l'on travaille sur de gros documents, il peut être pratique de couper le fichier source en plusieurs morceaux. L^AT_EX a deux commandes qui vous permettent de gérer plusieurs fichiers sources.

```
\include{fichier}
```

Vous pouvez utiliser cette commande dans le corps de votre document pour insérer le contenu d'un autre fichier source. L^AT_EX ajoute automatiquement le suffixe .tex au nom spécifié. Remarquez que L^AT_EX va sauter une page pour traiter le contenu de fichier.tex.

La seconde commande peut être utilisée dans le préambule. Elle permet de dire à L^AT_EX de n'inclure que certains des fichiers désignés par les commandes `\include`.

```
\includeonly{fichier}
```

²Sans l'option **a4paper**, le format de papier sera américain : 8,5 × 11 pouces, soit 216 × 280 mm.

Après avoir rencontré cette commande dans le préambule d'un document, seules les commandes `include` dont les fichiers sont cités en paramètre de la commande `includeonly` seront exécutées.

La commande `include` saute une page avant de commencer le formatage du texte inclus. Parfois ce comportement n'est pas souhaitable. Dans ce cas, vous pouvez utiliser la commande :

```
\input{fichier}
```

qui insère simplement le fichier indiqué sans aucun traitement sophistiqué, mais interdit l'utilisation de `\includeonly`.

J'insiste sur l'utilité de cette commande. Quand il y a beaucoup de pages, \LaTeX peut être lent à compiler et c'est extrêmement pratique de ne compiler qu'une partie du texte, tout en gardant les liens vers les autres!

2.3 La mise en page proprement dite

2.3.1 Chapitres et sections

Pour aider le lecteur à suivre votre pensée, vous souhaitez séparer vos documents en chapitres, sections ou sous-sections. \LaTeX utilise pour cela des commandes qui prennent en argument le titre de chaque élément. C'est à vous de les utiliser dans l'ordre.

Dans la classe de document **article**, les commandes de sectionnement suivantes sont disponibles :

- `part`
- `section`
- `subsection`
- `subsubsection`
- `paragraph`
- `subparagraph`
- `appendix`

Dans les classes **report** et **book**, la commande `chapter` est également reconnue.

L'espacement entre les sections, la numérotation et le choix de la police et de la taille des titres sont gérés automatiquement par \LaTeX .

Essayer de compiler les fichiers `exemple01.tex`, `exemple05.tex` et `exemple10.tex`. Attention, à partir de maintenant, l'ensemble des fichiers de style est regroupé dans le répertoire **style**. Étudiez ensuite, le fichier `exemple15.tex` :

Fichier : `exemple15.tex`

```
\documentclass[11pt,a4paper]{book}

\input{../style/stylebase01}

\title{Exemple 15}
\author{Moi Même}

%%% corps du document

\begin{document}
\maketitle

\include{exemple15_chapitre1}
\include{exemple15_chapitre2}

\end{document}
```

Il produira un document identique à l'exemple `exemple10.tex`. Seule, l'organisation du fichier en sous fichiers est modifiée.

2.3.2 Tables des matières

L^AT_EX peut ensuite créer la table des matières : la commande : `\tableofcontents` imprime la table des matières. Un document doit être traité (on dit aussi compilé) deux fois par L^AT_EX pour avoir une table des matières correcte. Dans certains cas, un troisième passage est même nécessaire. L^AT_EX vous indique quand cela est nécessaire.

La table des matières est mise en forme grâce aux réglages par défaut de L^AT_EX. On peut choisir le degré de précision de la table des matières : par défaut, L^AT_EX inscrira les noms de parties, chapitres, sections et sous-sections, si on se trouve dans la classe `book` ou `amsbook` (voir tableau 2.1 on page 12) et ira jusqu'aux sous-sous-sections dans la classe `article`. C'est ce qu'on appelle la profondeur de la table des matières et elle est contrôlée par un compteur nommé `tocdepth`. Par convention, les sections sont considérées comme étant le niveau 1. On en déduit le niveau des autres subdivisions : -1 pour les parties, 0 pour les chapitres pour la classe `book`, 1 pour les sections, 2 pour les sous-sections... On pourra modifier cette valeur³ parties en tapant

```
\setcounter{tocdepth}{2}
```

Toutes les commandes citées ci-dessus existent dans une forme « étoilée » obtenue en ajoutant une étoile `*` au nom de la commande. Ces commandes produisent des titres de sections qui n'apparaissent pas dans la table des matières et qui ne sont pas numérotés. On peut ainsi remplacer la commande `\section{Introduction}` par `\section*{Introduction}`.

Voir le document `exemple20.tex`.

2.3.3 Références croisées

Dans les livres, rapports ou articles, on trouve souvent des références croisées vers des figures, des tableaux ou des passages particuliers du texte. L^AT_EX dispose des commandes suivantes pour faire des références croisées :

- `\label{marque}`;
- `\ref{marque}`;
- `\pageref{marque}`.

Ici, `marque` est un identificateur choisi par l'utilisateur. L^AT_EX remplace `\ref` par le numéro de la section, de la sous-section, de la figure, du tableau, ou du théorème où la commande `\label` correspondante a été placée. L'utilisation de références croisées rend nécessaire de compiler deux fois le document : à la première compilation les numéros correspondant aux étiquettes `\label{}` sont inscrits dans le fichier `.aux` et, à la compilation suivante, `\ref{}` et `\pageref{}` peuvent imprimer ces numéros⁴.

Voir le document `exemple25.tex` :

Fichier : `exemple25.tex`

```
\documentclass[11pt,a4paper]{article}

\input{../style/stylebase01}

\title{Exemple 25}
\author{Moi Même}
```

³parfois, en classe `article`, il est parfois peu commode de voir apparaître les sous-sous-sections, ce qui rend la table des matières trop longue.

⁴Ces commandes ne connaissent pas le type du numéro auquel elles se réfèrent, elles utilisent le dernier numéro généré automatiquement.

```

%%% corps du document

\begin{document}
\selectlanguage{french}

\maketitle

\section{Introduction}
\label{introduction}
En introduction, on étudiera le vol des mouches.

\section{Le vol des mouches}
\subsection{Temp humide}
\label{voltempshumide}
Elles volents par temps humide.
\subsection{Temp sec}
Elles volents par temps sec,
comme dans le cas étudié en section \ref{voltempshumide}.

\section{Conclusion}
Ainsi, dans cet article, on a montré que les mouches volaient,
ainsi que le prédisait l'introduction (voir section \ref{introduction}
page \pageref{introduction}).

\end{document}

```

Citons aussi le package très pratique, **showkeys**, qui permet de voir apparaître les marques utilisées. Ce package est naturellement désactivée pour la sortie finale.

Voir le document `exemple27.tex` :

Fichier : `exemple27.tex`

```

\documentclass[11pt,a4paper]{article}

\input{../style/stylebase01}

\title{Exemple 27}
\author{Moi Même}

% pour voir les etiquettes
\usepackage{showkeys}

%%% corps du document
\begin{document}
\selectlanguage{french}

\maketitle

D'après l'équation
\begin{equation}
\label{eq01}

```



```
p=-\nabla v,
\end{equation}
Les mouches volents par temps humide.

Elles volents aussi par temps sec.
Dans ce cas, l'équation (\ref{eq01}) n'est plus valable.

\end{document}
```

2.3.4 Environnements

Pour composer du texte dans des contextes spécifiques, L^AT_EX définit des environnements différents pour divers types de mise en page :

```
\begin{nom}
texte (ou autre).
\end{nom}
```

Ici, nom est le nom de l'environnement. Les environnements peuvent être imbriqués :

```
\begin{aaaa}
\begin{bbbb}
\begin{cccc}

\end{cccc}
\end{bbbb}

\begin{bbbb}

\end{bbbb}
\end{aaaa}
```

Dans les sections suivantes quelques environnements importants sont présentés.

On trouvera des exemples dans le fichier `exemple30.tex`.

Listes, énumérations et descriptions

L'environnement `itemize` est utilisé pour des listes simples et `enumerate` est utilisé pour des énumérations (listes numérotées), ou `description` pour des descriptions.

Alignements à gauche, à droite et centrage

Les environnements `flushleft` et `flushright` produisent des textes alignés à gauche ou à droite. L'environnement `center` produit un texte centré.

Verbatim

Tout texte inclus entre `\begin{verbatim}` et `\end{verbatim}` est imprimé tel quel, comme s'il avait été tapé à la machine, avec tous les retours à la ligne et les espaces, sans qu'aucune commande L^AT_EX ne soit exécutée.

2.3.5 Modification de l'aspect des caractères

L'option 10pt, 11pt ou 12 pt du tableau 2.2 on page 13 permet de gérer la taille des caractères de l'ensemble du document. Localement⁵, il faut parfois modifier la taille des caractères :

Victor Hugo était un grand homme. {\Large Victor Hugo} \’etait un
{\tiny grand} homme.

Les différentes tailles proposées, du plus petit au plus grand sont regroupées dans le tableau 2.3

<code>\tiny</code>	<code>\large</code>
<code>\scriptsize</code>	<code>\Large</code>
<code>\footnotesize</code>	<code>\LARGE</code>
<code>\small</code>	<code>\huge</code>
<code>\normalsize</code>	<code>\Huge</code>

TAB. 2.3 – Les différentes tailles de caractères

L^AT_EX utilise deux sortes de caractères penchés : les italiques et les obliques. Il existe aussi une forme dite «petite capitale». On peut utiliser ces trois types de caractères, comme le montre l'exemple suivant⁶ :

On pourra utiliser une estimation *a priori* pour démontrer ce **résultat** de Monsieur EIFFEL. On pourra utiliser
une estimation `\textit{a priori}`
pour d'émontrer ce `\textbf{r}\’esultat`
de Monsieur `\textsc{Eiffel}`.

On utilise l'une des quatre commandes du tableau 2.4 ; l'argument texte est placé entre accolade.

<code>\textup{texte}</code>	caractère normal
<code>\textit{texte}</code>	caractère italique
<code>\textsl{texte}</code>	caractère oblique
<code>\textsc{texte}</code>	caractère en petites capitales

TAB. 2.4 – Les différents styles de caractères

On pourra aussi mélanger tout ces styles⁷ comme le montre l'exemple suivant :

On pourra *utiliser* une **estimation** *a priori* pour démontrer ce **résultat** de MONSIEUR EIFFEL. On pourra `\textsl{utiliser}`
une `{\huge estimation}`
`\textbf{\textit{a priori}}`
pour d'émontrer ce
`\textbf{r}\’esultat` de
`{\tiny \textsc{Monsieur}}`
`\textsc{Eiffel}`.

⁵ Attention, la gestion des titres, des titres de sections, ... est gérée par L^AT_EX et il ne faut pas les modifier soi-même.

⁶ l'italique est utilisé, par exemple, pour citer une expression d'origine étrangère, entrée dans le français. Par exemple *a posteriori*.
On ne met pas d'accent sur le a !

⁷ avec modération !

Chapitre 3

Écrire des mathématiques

On verra dans cet chapitre, le plus important de ce document, comment écrire des mathématiques sous \LaTeX .

Dans la section¹ 3.1, on verra comment écrire en \LaTeX standard. On verra ensuite en section 3.2, l'utilisation des packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$. On pourra lire aussi la section facultative 3.4, dans laquelle on voit qu'écrire des mathématiques, même en connaissant très bien \LaTeX , n'est pas si simple!

3.1 Les mathématiques standards sous \LaTeX

Vous êtes prêts! Dans ce chapitre nous allons aborder l'atout majeur de \TeX : la composition de formules mathématiques. Mais attention, ce chapitre ne fait que décrire les commandes de base. Bien que ce qui est expliqué ici soit suffisant pour la majorité des utilisateurs, ne désespérez pas si vous n'y trouvez pas la solution à votre problème de mise en forme d'une équation mathématique. Il y a de fortes chances pour que la solution se trouve dans les packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$ ².

Personnellement, j'ai une préférence pour les packages de $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$. Ainsi, je vous invite à passer directement à la section 3.2. En fait, les packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$ reconnaissent les commandes mathématiques standard de \LaTeX , donc tout ce qui est présenté dans cette section; néanmoins, ces packages utilisent des commandes permettant de refaire tout cela de façon plus précise. De plus, la documentation annexe relative à $\mathcal{A}\mathcal{M}\mathcal{S}\text{\LaTeX}$ (voir [8]) est très bien faite.

3.1.1 Généralités

\LaTeX dispose d'un mode spécial pour la mise en page de formules mathématiques : pour passer en mode mathématiques, il y a deux méthodes :

- l'environnement `displaymath`, ou ce qui équivalent `\[... \]` ou encore `$$... $$`;
- l'environnement `math`, ou ce qui équivalent `(\ ... \)` ou encore `$... $`.

La première permet de réaliser des équations hors texte et la seconde dans le texte.

Pour tout ce chapitre, on présentera le résultat produit \LaTeX à gauche et le code source à droite³.

¹qui reprend une partie du chapitre 3 de [15].

²*American Mathematical Society*, voir section 3.2.

³C'est le package `example` qui permet cela.

Voir le fichier `exemple01.tex` qui fournit le résultat suivant :

On peut calculer le carré de la diagonale en utilisant les valeurs de a et b en écrivant $c^2 = a^2 + b^2$, soit encore

$$c^2 = a^2 + b^2.$$

On peut réécrire cela sous la forme

$$a^2 = c^2 - b^2,$$

ou encore

$$b^2 = c^2 - a^2.$$

```
On peut calculer le carr\'e de la diagonale
en utilisant les valeurs de
\begin{math}
a
\end{math}
et
\(\backslash\)
en \'ecrivant
$c^2=a^2+b^2$,
soit encore
\begin{displaymath}
c^2=a^2+b^2.
\end{displaymath}
On peut r\'e\'ecrire cela sous la forme
$$
a^2=c^2-b^2,
$$
ou encore
\[
b^2=c^2-a^2.
\]
```

On obtient ainsi des formules qui ne sont pas numérotées. Si vous voulez qu'elles soient numérotées par L^AT_EX, utilisez l'environnement `\equation`. Avec `\label` et `\ref`, vous pouvez faire référence à une équation, comme le montre le fichier `exemple05.tex` qui produit :

On peut calculer le carré de la diagonale en utilisant :

$$c^2 = a^2 + b^2. \quad (3.1)$$

L'équation (3.1) se met sous la forme équivalente :

$$c = \sqrt{a^2 + b^2}.$$

```
On peut calculer le carr\'e de la diagonale
en utilisant :
\begin{equation}
\label{eq1}
c^2=a^2+b^2.
\end{equation}
L'\ref{eq1} se met sous
la forme \'equivalente :
$$
c=\sqrt{a^2+b^2}.
$$
```

Remarquez que les expressions mathématiques sont formatées différemment selon qu'elles sont composées en ligne ou hors texte . Comparez les deux exemples suivants :

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6}$$

```
$\lim_{n \to \infty}
\sum_{k=1}^n \frac{1}{k^2}
= \frac{\pi^2}{6}$
```

$$\lim_{n \rightarrow \infty} \sum_{k=1}^n \frac{1}{k^2} = \frac{\pi^2}{6}$$

```
\begin{displaymath}
\lim_{n \to \infty}
\sum_{k=1}^n \frac{1}{k^2}
= \frac{\pi^2}{6}
\end{displaymath}
```

Il y a des différences entre le mode *mathématique* et le mode *texte*. Par exemple, en mode *mathématique* :

1. la plupart des espaces et des retours à la ligne n'ont aucune signification. Les espaces sont déduites de la logique de la formule ou indiquées à l'aide de commandes spécifiques telles que : `quad` ou `qqquad` ;
2. les lignes vides ne sont pas autorisées. Un seul paragraphe par formule ;
3. chaque lettre est considérée comme étant le nom d'une variable et sera imprimée comme tel. Pour insérer du texte normal (police et espacement standard) dans une formule, il faut utiliser la commande `\textrm{...}`.

$$\forall x \in \mathbf{R} : \quad x^2 \geq 0 \quad (3.2)$$

```
\begin{equation}
\forall x \in \mathbf{R} :
\quad x^2 \geq 0
\end{equation}
```

$$x^2 \geq 0 \quad \text{pour tout } x \in \mathbf{R} \quad (3.3)$$

```
\begin{equation}
x^2 \geq 0 \quad \text{pour tout }
\textrm{pour tout }
x \in \mathbf{R}
\end{equation}
```

3.1.2 Groupements en mode mathématique

La plupart des commandes du mode mathématique ne s'applique qu'au caractère suivant. Pour qu'une commande s'applique à un ensemble de caractères, il faut les grouper en utilisant des accolades : `{...}`.

$$a^x + y \neq a^{x+y} \quad (3.4)$$

```
\begin{equation}
a^x + y \neq a^{x+y}
\end{equation}
```

3.1.3 Éléments d'une formule mathématique

Dans cette section nous allons voir les commandes les plus importantes du mode mathématique. Pour voir l'ensemble des symboles mathématiques, voir la section 3.3.

Les lettres grecques minuscules sont saisies de la manière suivante : `\alpha`, `\beta`, `\gamma`, etc. Les lettres grecques majuscules⁴ sont saisies ainsi : `\Gamma`, `\Delta`, etc.

$$\lambda, \xi, \pi, \mu, \Phi, \Omega$$

```
\lambda, \xi, \pi, \mu, \Phi, \Omega
```

⁴Il n'y a pas de Alpha majuscule dans L^AT_EX 2_ε parce que c'est le même caractère que le A romain. Lorsque le nouveau codage mathématique sera terminé, cela changera.

Les indices et exposants sont positionnés en utilisant les caractères `_` et `^`.

a_1	x^2	$e^{-\alpha t}$	a_{ij}^3	<code>\$a_{1}\$ \quad \$x^{2}\$ \quad</code>
$e^{x^2} \neq (e^x)^2$				<code>\$e^{-\alpha t}\$ \quad</code>
				<code>\$a^3_{ij}\$ \quad</code>
				<code>\$e^{x^2} \neq (e^x)^2\$</code>

La racine carrée est saisie ainsi : `\sqrt`, la racine $n^{\text{ième}}$ est produite par la commande `\sqrt[n]`. La taille du symbole racine est calculée par L^AT_EX. Pour obtenir le symbole seul, utilisez `\surd`.

\sqrt{x}	$\sqrt{x^2 + \sqrt{y}}$	$\sqrt[3]{2}$	<code> \$\sqrt{x}\$ \quad</code>
$\sqrt{x^2 + y^2}$			<code> \$\sqrt{x^2 + \sqrt{y}}\$ \quad</code>
			<code> \$\sqrt[3]{2}\$ \quad</code>
			<code> \$\surd[x^2 + y^2]\$</code>

Les commandes `\overline` et `\underline` créent un trait horizontal au-dessus ou au-dessous d'une expression.

$\overline{m+n}$	<code> \$\overline{m+n}\$</code>
------------------	----------------------------------

Les commandes `\overbrace` et `\underbrace` créent une grande accolade horizontale au-dessus ou au-dessous d'une expression.

$\underbrace{a+b+\cdots+z}_{26}$	<code> \$\underbrace{a+b+\cdots+z}_{26}\$</code>
----------------------------------	--------------------------------------------------

Les vecteurs sont en général marqués en ajoutant une flèche au-dessus du nom de la variable. Ceci est obtenu par la commande `\vec`. Pour coder le vecteur de A à B , les commandes `\overrightarrow` et `\overleftarrow` sont bien utiles.

\vec{a}	\overrightarrow{AB}	<code> \begin{displaymath}</code>
		<code> \vec{a} \quad \overrightarrow{AB}</code>
		<code> \end{displaymath}</code>

En général, les points indiquant une opération de multiplication ne sont pas imprimés. Cependant, il arrive qu'il soit nécessaire de les faire apparaître pour aider la lecture. Utilisez alors `\cdot`.

$v = \sigma_1 \cdot \sigma_2 \tau_1 \cdot \tau_2$	<code> \[v = \sigma_1 \cdot \sigma_2 \tau_1 \cdot \tau_2</code>
	<code> \tau_1 \cdot \tau_2 \]</code>

Les noms des fonctions telles que sinus doivent être imprimés à l'aide d'une police droite et non en italique comme les variables. L^AT_EX fournit donc les commandes suivantes pour les fonctions les plus utilisées :

<code>\arccos</code>	<code>\cos</code>	<code>\csc</code>	<code>\exp</code>	<code>\ker</code>	<code>\limsup</code>	<code>\min</code>	<code>\sinh</code>
<code>\arcsin</code>	<code>\cosh</code>	<code>\deg</code>	<code>\gcd</code>	<code>\lg</code>	<code>\ln</code>	<code>\Pr</code>	<code>\sup</code>
<code>\arctan</code>	<code>\cot</code>	<code>\det</code>	<code>\hom</code>	<code>\lim</code>	<code>\log</code>	<code>\sec</code>	<code>\tan</code>
<code>\arg</code>	<code>\coth</code>	<code>\dim</code>	<code>\inf</code>	<code>\liminf</code>	<code>\max</code>	<code>\sin</code>	<code>\tanh</code>

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

```
\[ \lim_{x \rightarrow 0}
\frac{\sin x}{x} = 1 \]
```

Pour la fonction modulo, il y a deux commandes possibles : `\bmod` pour l'opérateur binaire et `\pmod` pour l'opérateur unaire :

$$a \bmod b$$

$$x \equiv a \pmod{b}$$

```
$a\bmod b$\\
$x\equiv a \pmod{b}$
```

Un trait de fraction est produit par la commande :

```
\frac{numérateur}{dénominateur}
```

La forme utilisant un *slash* (1/2) est souvent préférable pour des petits éléments.

$$1\frac{1}{2} \text{ heures}$$

$$\frac{x^2}{k+1} \quad x^{\frac{2}{k+1}} \quad x^{1/2}$$

```
$1\frac{1}{2}$~heures
\begin{displaymath}
\frac{x^2}{k+1} \quad x^{\frac{2}{k+1}} \quad x^{1/2}
\end{displaymath}
```

Il est parfois utile de pouvoir superposer des symboles, la commande `stackrel` place son premier argument en taille réduite au dessus du second :

$$X \stackrel{!}{=} 1$$

```
\begin{displaymath}
X \stackrel{!}{=} 1
\end{displaymath}
```

Les intégrales sont produites par la commande `\int`, les sommes par la commande `\sum`, les produits par la commande `\prod`. Les limites inférieures et supérieures sont indiquées avec `_` et `^` comme pour les indices et les exposants.

$$\sum_{i=1}^n \int_0^{\frac{\pi}{2}} \prod_{\epsilon}$$

```
\begin{displaymath}
\sum_{i=1}^n \quad \int_0^{\frac{\pi}{2}} \quad \prod_{\epsilon}
\end{displaymath}
```

Pour les crochets et les autres délimiteurs, il existe toutes sortes de symboles en T_EX (par exemple `[` `<` `||` `↕`). Les parenthèses et les crochets sont obtenus avec les caractères correspondants, les accolades avec `\{`, et tous les autres délimiteurs sont obtenus par des commandes spéciales (par exemple `\updownarrow`).

$$a, b, c \neq \{a, b, c\}$$

```
\begin{displaymath}
\{a,b,c\} \neq \{a,b,c\}
\end{displaymath}
```

Si vous ajoutez la commande `\left` avant un délimiteur ouvrant ou `\right` avant un délimiteur fermant, \TeX détermine automatiquement la taille appropriée pour ce caractère. Remarquez qu'il est nécessaire de fermer chaque délimiteur ouvrant (`\left`) avec un délimiteur fermant (`\right`). Si vous ne voulez pas de délimiteur fermant, utilisez le délimiteur invisible `\right.` !

$$1 + \left(\frac{1+x}{1-x^2} \right)^3$$

```
\begin{displaymath}
1 + \left( \frac{1+x}{1-x^2} \right)^3
\end{displaymath}
```

Pour saisir des points de suspension dans une formule, vous pouvez utiliser plusieurs commandes. `\ldots` imprime les points sur la base de la ligne, `\cdots` les imprime au milieu. En plus il y a les commandes `\vdots` pour les imprimer verticalement et `\ddots` pour les imprimer en diagonale. Vous trouverez un autre exemple dans la section 3.1.4.

$$x_1, \dots, x_n \quad x_1 + \cdots + x_n$$

```
\begin{displaymath}
x_{1}, \ldots, x_{n} \quad x_1 + \cdots + x_n
\end{displaymath}
```

3.1.4 Alignements verticaux

Pour imprimer des matrices, utilisez l'environnement `array`. La commande `\\` est utilisée pour séparer les lignes.

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots \\ x_{21} & x_{22} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

```
\begin{displaymath}
\mathbf{X} = \begin{array}{ccc}
x_{11} & x_{12} & \ldots \\
x_{21} & x_{22} & \ldots \\
\vdots & \vdots & \ddots
\end{array}
\end{displaymath}
```

L'environnement `array` peut également être utilisé pour imprimer des expressions qui ont un délimiteur invisible obtenu par la commande `\right.` :

$$y = \begin{cases} a & \text{si } d > c \\ b + x & \text{le matin} \\ l & \text{la journée} \end{cases}$$

```
\begin{displaymath}
y = \left\{ \begin{array}{ll}
a & \text{\textrm{si } $d > c$}} \\
b+x & \text{\textrm{le matin}} \\
l & \text{\textrm{la journ'ee}}
\end{array} \right.
\end{displaymath}
```

L'environnement `array` permet, comme l'environnement `tabular`, d'insérer des lignes horizontales ou

verticales :

$$\left(\begin{array}{c|c} 1 & 2 \\ \hline 3 & 4 \end{array} \right)$$

```
\begin{displaymath}
\left(\begin{array}{c|c}
1 & 2 \\
\hline
3 & 4
\end{array}\right)
\end{displaymath}
```

Pour les formules qui prennent plusieurs lignes ou pour des systèmes d'équations, utilisez les environnements `eqnarray` et `eqnarray*` plutôt que `equation`. Avec `eqnarray` chaque ligne est numérotée, alors que la variante `eqnarray*` ne produit aucun numéro.

$$\begin{array}{rcl} f(x) & = & \cos x \\ f'(x) & = & -\sin x \\ \int_0^x f(y) dy & = & \sin x \end{array}$$

```
\begin{eqnarray}
f(x) & = & \cos x \\
f'(x) & = & -\sin x \\
\int_0^x f(y) dy & = & \sin x
\end{eqnarray}
```

3.1.5 Taille des polices mathématiques

En mode mathématique T_EX choisit la taille de la police en fonction du contexte. Les exposants, par exemple, sont imprimés avec une police plus petite.

Malgré tout, il peut être nécessaire d'indiquer à L^AT_EX la taille exacte. En mode mathématique, la taille de la police est déterminée par les quatre commandes :

`\displaystyle (123)`, `\textstyle (123)`, `\scriptstyle (123)` et `\scriptscriptstyle (123)`.

Changer de style modifie également la façon dont les limites sont affichées.

$$\mathrm{corr}(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\left[\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2 \right]^{1/2}}$$

```
\begin{displaymath}
\mathrm{corr}(X,Y)=
\frac{\displaystyle
\sum_{i=1}^n(x_i-\overline{x})
(y_i-\overline{y})}
{\displaystyle\biggl[
\sum_{i=1}^n(x_i-\overline{x})^2
\sum_{i=1}^n(y_i-\overline{y})^2
\biggr]^{1/2}}
\end{displaymath}
```

Cet exemple est un cas où on a besoin de crochets plus petits que ceux qui seraient produits par les commandes standards `\left[` `\right]`.

3.1.6 Insertion de texte en mode mathématique

La commande `\textrm` qui fait passer en mode *texte* permet l'utilisation des lettres accentuées et respecte les espaces.

3.2 Les mathématiques avec les packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{L}\text{A}\text{T}\text{E}\text{X}$

On consultera le chapitre 8 de [8] donné en annexes B page 50, très complète.

3.3 Liste des symboles mathématiques

On consultera les pages⁵ 53 à 59 de [15] très complètes ou le guide de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{L}\text{A}\text{T}\text{E}\text{X}$ [2]. Attention, les symboles présentés dans [15] nécessitent parfois les packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{L}\text{A}\text{T}\text{E}\text{X}$.

3.4 Écrire des textes mathématiques

Combien de professeur de science ont dû vous dire que « ce qui se conçoit clairement, s'exprime clairement ». Vous pourriez être amené à croire que grâce à $\text{L}\text{A}\text{T}\text{E}\text{X}$, tout est facile et qu'il suffit d'écrire des formules aux kilomètres pour présenter une pensée mathématique.

Il n'en est rien ! Je m'appuierai sur le bref mais excellent papier de Michèle Audin [3].

Le défaut que l'on peut avoir en découvrant $\text{L}\text{A}\text{T}\text{E}\text{X}$ est de vouloir écrire de grosses formules mathématiques, puisque toutes ces équations sont si bien gérées.

Commençons par un exemple tout simple : Andrew John Wiles a démontré⁶ en 1993 le dernier théorème de Fermat⁷ :

Théorème 3.1 (Théorème de Wiles-Fermat). Pour tout entier n supérieur ou égal à trois, l'équation $x^n + y^n = z^n$ n'admet pas de solutions entières non nulles.

Logiquement, on peut aussi l'écrire de résultat sous la forme

Théorème 3.2 (Théorème de Wiles-Fermat (deuxième version)). On a :

$$\forall n \in \mathbb{N}, \quad n \geq 3 \implies$$

$$\left(\forall (x, y, z) \in \mathbb{N}^3, \quad (x^n + y^n - z^n = 0 \implies ((x = 0) \wedge (y = 0) \wedge (z = 0))) \right). \quad (3.8)$$

Je vous laisse le soin d'apprécier laquelle des deux versions est la plus agréable à lire ! L'équation (3.8) utilise certes beaucoup de commandes⁸ de $\text{L}\text{A}\text{T}\text{E}\text{X}$. Mais l'énoncé du théorème 3.1 est plus naturel, plus agréable à lire ; le lecteur se sent moins idiot, puisqu'il comprend sans trop avoir mal à la tête. De plus, même en mathématique, on écrit dans une langue. On ne mélange pas les mots et les symboles mathématiques, on ponctue ses phrases et on écrit de façon correcte, grammaticalement comme scientifiquement.

Par exemple, on évitera de réécrire le théorème 3.1 sous la forme

Théorème 3.3 (Théorème de Wiles-Fermat (troisième version)). $\forall n \in \mathbb{N}$, si $n \geq 3$, alors $\exists (x, y, z) \in (\mathbb{N}^*)^3$ tels que $x^n + y^n = z^n$.

⁵présent sur le DVD.

⁶Je n'insisterai pas sur l'extrême importance de ce résultat. Fermat aurait démontré au milieu du dix-septième siècle un résultat pourtant sur les nombres entiers (voir théorème 3.1). La légende dit, que n'ayant pas la place d'en écrire la preuve dans son manuscrit, il aurait laissé ce résultat sans preuve. Ce résultat est resté non démontré pendant plus de trois siècles et a fait coulé beaucoup d'encre.

⁷plus précisément il a démontré la conjecture de Fermat, connue sous le nom du dernier théorème de Fermat. On l'appelle maintenant le théorème de Wiles-Fermat. On pourra consulter le livre très intéressant sur le théorème de Fermat [1] ; très vulgarisateur, c'est un véritable récit sur la science.

⁸et même de l' $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{L}\text{A}\text{T}\text{E}\text{X}$, voir section 3.2.

De façon générale, si on veut faire passer un message scientifique, on aura intérêt à le présenter par des phrases et des mots simples. Naturellement, rien n'interdit d'utiliser le moment venu une sémantique mathématique adéquate pour démontrer le résultat en question⁹.

Pour plus de détails, on pourra aussi consulter [14].

Je finis cette section par un petit exercice mathématique : dans tout les exemples qui suivent, essayez de comprendre ce qu'il vaudrait mieux écrire (ils sont tous extraits de [3]).

1. Si $a \in X$, alors on a $b \in Y$. Indication : où sont les verbes ?
2. Montrer qu'un nombre complexe s'écrit comme le produit d'un nombre réel positif et d'un nombre complexe de module 1. Indication : lequel ?
3. La fonction $x^2 + 3$ est paire. Indication : $x^2 + 3$ est un nombre, une suite de symbole, ce que vous voulez, mais pas une fonction.
4. Sur tout compact, toute fonction continue f est bornée. Indication : qu'en est-il d'une fonction continue g ?
5. Si $0 \leq u_n^{1/n} = \rho < 1$, alors $\lim_n u_n = 0$.

Le dernier exercice serait de trouver dans vos différents cours de l'UTBM les endroits où apparaissent de telles choses¹⁰.

⁹La preuve du théorème 3.1 constitue un ensemble de plus de mille pages, paraît-il, et les concepts utilisés ainsi que les notations employées sont beaucoup plus sophistiqués que son énoncé !

¹⁰Dans les UV que j'enseigne, j'en suis conscient, de telles détails apparaissent sûrement !

Chapitre 4

Tableaux, flottants, figures

Ce chapitre reprend une partie des chapitres 2 et 4 [15].

Il n'est donné dans ce chapitre qu'un minimum d'indications et d'exemples. On consultera par exemple [15], [8], [5], [16] ou [6].

4.1 Tableaux

L'environnement `tabular` permet de réaliser des tableaux avec ou sans lignes de séparation horizontales ou verticales. \LaTeX ajuste automatiquement la largeur des colonnes.

L'argument *description du tableau* de la commande : `\begin{tabular}{description du tableau}` définit le format des colonnes du tableau. Utilisez un `l` pour une colonne alignée à gauche, `r` pour une colonne alignée à droite et `c` pour une colonne centrée. `p{largeur}` permet de réaliser une colonne justifiée à droite sur plusieurs lignes et enfin `|` permet d'obtenir une ligne verticale.

À l'intérieur de l'environnement `tabular`, le caractère `&` est le séparateur de colonnes, `\\` commence une nouvelle ligne et `\hline` insère une ligne horizontale.

7C0	hexadécimal
3700	octal
11111000000	binaire
1984	décimal

```
\begin{tabular}{|r|l|}
\hline
7C0 & hexad\'ecimal \\
3700 & octal \\
11111000000 & binaire \\
\hline \hline
1984 & d\'ecimal \\
\hline
\end{tabular}
```

Bienvenue dans ce cadre.
Merci de votre visite.

```
\begin{tabular}{|p{4.7cm}|}
\hline
Bienvenue dans ce
cadre.\\
Merci de votre visite.\\
\hline
\end{tabular}
```

Voir aussi l'exemple suivant, stocké dans le fichier `exemple01.tex` :

i	0	1	2
η_i^c	—	0.999921	1.999945
η_i	—	1	2
k_i^c	$3.22 \cdot 10^{-14}$	0.999892	0.999892
k_i	0	1	0.999892

```
\begin{tabular} {|c|c|c|c|}
\hline
i & $0$ & $1$ & $2$
\\ \hline \hline
 $\eta_i^c$  & $-$ & $0.999921$ & $1.999945$
\\ \hline \hline
 $k_i^c$  & $3.22 \cdot 10^{-14}$ & $0.999892$ & $0.999892$
\\ \hline \hline
 $k_i$  & $0$ & $1$ & $0.999892$
\\ \hline \hline
\end{tabular}
```

4.2 Objets flottants

4.2.1 Généralités

De nos jours, la plupart des publications contiennent un nombre important de figures et de tableaux. Ces éléments nécessitent un traitement particulier car ils ne peuvent être coupés par un changement de page. On pourrait imaginer de commencer une nouvelle page chaque fois qu'une figure ou un tableau ne rentrerait pas dans la page en cours. Cette façon de faire laisserait de nombreuses pages à moitié blanches, ce qui ne serait réellement pas beau.

La solution est de laisser « flotter » les figures et les tableaux qui ne rentrent pas sur la page en cours, vers une page suivante et de compléter la page avec le texte qui suit l'objet « flottant ». \LaTeX fournit deux environnements pour les objets flottants adaptés respectivement aux figures (`figure`) et aux tableaux (`table`). Pour faire le meilleur usage de ces deux environnements, il est important de comprendre comment \LaTeX traite ces objets flottants de manière interne. Dans le cas contraire ces objets deviendront une cause de frustration intense car \LaTeX ne les placera jamais à l'endroit où vous souhaitez les voir.

Commençons par regarder les commandes que \LaTeX propose pour les objets flottants :

Tout objet inclus dans un environnement `figure` ou `table` est traité comme un objet flottant. Les deux environnements flottants ont un paramètre optionnel :

```
\begin{figure}[placement]
```

ou

```
\begin{table}[placement]
```

appelé *placement*. Ce paramètre permet de dire à \LaTeX où vous autorisez l'objet à flotter. Un *placement* est composé d'une chaîne de caractères représentant des *placements possibles*. Reportez-vous au tableau 4.1 on the next page.

Si aucun placement n'est indiqué, les classes standard utilisent `[tbp]` par défaut.

\LaTeX place tous les objets flottants qu'il rencontre dans l'ordre en suivant les indications fournies par l'auteur. Si un objet ne peut être placé sur la page en cours, il est placé soit dans la file des figures soit dans la

Caractère	Emplacement pour l'objet flottant. . .
h	<i>here</i> , ici, à l'emplacement dans le texte où la commande se trouve. Utile pour les petits objets.
t	<i>top</i> , en haut d'une page
b	<i>bottom</i> , en bas d'une page
p	<i>page</i> , sur une page à part ne contenant que des objets flottants.
!	ici, sans prendre en compte les paramètres internes (que le nombre maximum d'objets flottants sur une page) qui pourraient empêcher ce placement.

TAB. 4.1 – Placements possibles

file des tableaux¹. Quand une nouvelle page est entamée, L^AT_EX essaye d'abord de voir si les objets en tête des deux files pourraient être placés sur une page spéciale, à part. Si cela n'est pas possible, les objets en tête des deux files sont traités comme s'ils venaient d'être trouvés dans le texte : L^AT_EX essaye de les placer selon les placements possibles restants. Tous les nouveaux objets flottants rencontrés dans la suite du texte sont ajoutés à la queue des files. L^AT_EX respecte scrupuleusement l'ordre d'apparition des objets flottants. C'est pourquoi un objet flottant qui ne peut être placé dans le texte repousse tous les autres à la fin du document.

D'où la règle :

Si L^AT_EX ne place pas les objets flottants comme vous le souhaitez, c'est souvent à cause d'un seul objet trop grand qui bouche l'une des deux files d'objets flottants.

Essayer d'imposer à L^AT_EX un emplacement particulier en utilisant l'option [h] pour un flottant est une idée à *proscrire*, les versions modernes de L^AT_EX changent d'ailleurs automatiquement l'option [h] en [ht].

Voici quelques éléments supplémentaires qu'il est bon de connaître sur les environnements table et figure. Avec la commande :

```
\caption{texte de la légende}
```

vous définissez une légende pour l'objet. Un numéro (incrémenté automatiquement) et le mot « Figure » ou « Table »² sont ajoutés par L^AT_EX.

Avec \label et \ref vous pouvez faire référence à votre objet à l'intérieur de votre texte.

Dans certains cas il peut s'avérer nécessaire d'utiliser l'une des deux commandes : \clearpage ou \cleardoublepage. La première d'entre elles ordonne à L^AT_EX de placer tous les objets en attente immédiatement puis de commencer une nouvelle page. La seconde commence une nouvelle page de droite.

Dans les sections 4.2.2 et 4.2.3, on présente quelques exemples de tableaux et de figures flottantes.

4.2.2 Créations de tableaux flottants

Compte tenu des sections 4.1 et 4.2.1, un tableau flottant se déclare de la façon suivante :

```
\begin{table} [h]
\begin{center}
\begin{tabular} {|c|c|c|c|c|}
\hline
$i$ & $0$ & $1$ & $2$

```

¹ Il s'agit de files FIFO (*First In, First Out*) : premier arrivé, premier servi.

² Avec l'extension **babel**, la présentation des légendes est modifiée pour obéir aux règles françaises.

```

\\ \hline \hline
$\eta_i^c$ & $-$ & $0.999921$ & $1.999945$
\\ \hline
$\eta_i$ & $-$ & $1$ & $2$
\\ \hline \hline
$k_i^c$ & $3.22 \cdot 10^{-14}$ & $0.999892$ & $0.999892$
\\ \hline
$k_i$ & $0$ & $1$ & $0.999892$
\\ \hline
\end{tabular}
\caption{\label{taban1} Constants  $\eta_i$ ,  $k_i$  and constants
 $\eta_i^c$  and  $k_i^c$  computed from the cycle of the Fig. (2.13).}
\end{center}
\end{table}

```

On pourra consulter le fichier `exemple03.tex`, qui crée le tableau 4.2.

i	0	1	2
η_i^c	—	0.999921	1.999945
η_i	—	1	2
k_i^c	$3.22 \cdot 10^{-14}$	0.999892	0.999892
k_i	0	1	0.999892

TAB. 4.2 – Constants η_i , k_i and constants η_i^c and k_i^c computed from the cycle of the Fig. (2.13).

4.2.3 Créations de figures flottantes

Nous avons étudié le cadre de flottant, pour faire des figure. Maintenant, il faut étudier comment créer les figures. Un format souvent utilisé est le format postscript.

Insertions de figures

Un moyen d'inclure des graphiques dans un document est de les produire à l'aide d'un logiciel spécialisé³ puis d'inclure le résultat dans le document. Ici encore \LaTeX offre de nombreuses solutions.

On supposera que l'on dispose de fichier postscript ou pdf (l'inclusion d'images jpeg est possible si le document est compilé avec `pdflatex`), crée par un autre logiciel que \LaTeX . Voir par l'exemple `exemple05.tex`. On consultera la section 4.2.3 pour la création de tels fichiers.

Dans l'entête de son fichier, on déclare `\usepackage{graphicx}`. Puis, on écrit dans le corps du fichier :

```

\begin{figure}[h]
\begin{center}
\includegraphics[width=5cm]{exemplefigure01}
\end{center}
\caption{Une jolie figure.}
\label{figure01}
\end{figure}

```

³Tel que Matlab, XFig, Gnuplot...

Ici, `[h]` permet d'essayer de placer la figure là où elle est déclarée, `width` est un paramètre qui permet de régler la largeur de la figure (d'autres paramètres existent comme `scale`, `height`, `angle`). Voir le résultat en figure 4.1.



FIG. 4.1 – Une jolie figure.

On consultera aussi le fichier `exemple05.tex`.

On évitera d'ajouter une extension au nom de la figure car suivant la compilation, \LaTeX choisira (si les images aux format `eps` et `pdf` existent) de prendre l'image au format `eps` (compilation avec `latex`) ou `pdf` (compilation avec `pdflatex`).

Afin d'assurer la lisibilité du dossier où sont stockées les images il est conseillé d'adopter un arborescence claire comme par exemple un dossier `images` contenant des dossier ayant les noms des extension des fichiers `eps`, `pdf`, `jpg`, `Xfig`, etc.

Le compilateur doit savoir où trouver les images. La commande `\graphicspath` permet de préciser quelles sont les chemins des images. Ainsi dans l'arborescence proposée on déclarera dans le préambule :

```
\graphicspath{{./images/eps/}{./images/pdf/}{./images/jpg/}}
```

Créations de figures postscript

On pourra utiliser le logiciel libre XFIG. Grâce à lui, on crée une image, que l'on peut exporter au format `eps` et `pdf`.

Voir par exemple la figure (l'ouvrir avec XFIG) `exemplefigure05.fig`. Elle a été exportée au format `eps` et `pdf` et insérée dans le fichier, comme le montre la figure 4.2.

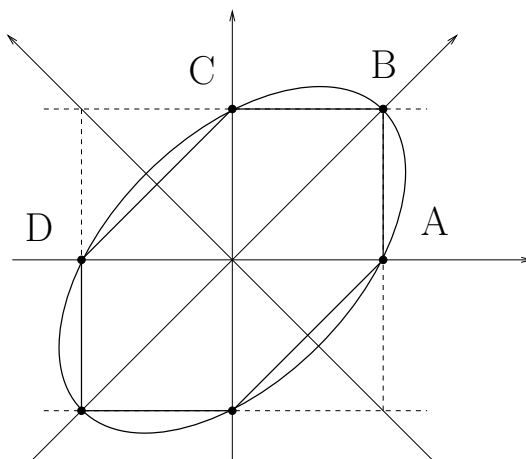


FIG. 4.2 – Une ellipse et un hexagone s'aimait d'amour tendre.

On peut aussi faire une figure grâce à `matlab`⁴, l'exporter en `eps` (voir le script `exemplematlab01.m` à faire

⁴Il existe aussi des clones libres de `Matlab`, tels que `Scilab` ou `Octave`, mais ils ont des capacités graphiques (entre autres) moins

tourner sous matlab).

```
% petit script pour comparer les deux criteres de Von Mises et de Tresca.
n=5000;
h1=pi/4/(fix(n/4));
h2=pi/2/(fix(n/2));
x1=0:h1:pi/4;
x2=pi/4:h1:pi/2;
x3=pi/2:h2:pi;
X=[x1 x2 x3];
y1=1./cos(x1);
y2=1./sin(x2);
y3=1./abs(cos(x3)-sin(x3));
t1=[y1 y2 y3];
t2=sqrt(1./abs(1-(sin(2*X))/2));
eccart=abs((t2-t1)./t1);
plot(X,t1,X,t2);
legend('Tresca','Von_Mises',0);
print -depsc comparaisontrescamis.eps;
pause;
plot(X,eccart);
legend('écart relatif entre Tresca et Von Mises');
```

Voir les figures produites 4.3, insérées dans le source \LaTeX .

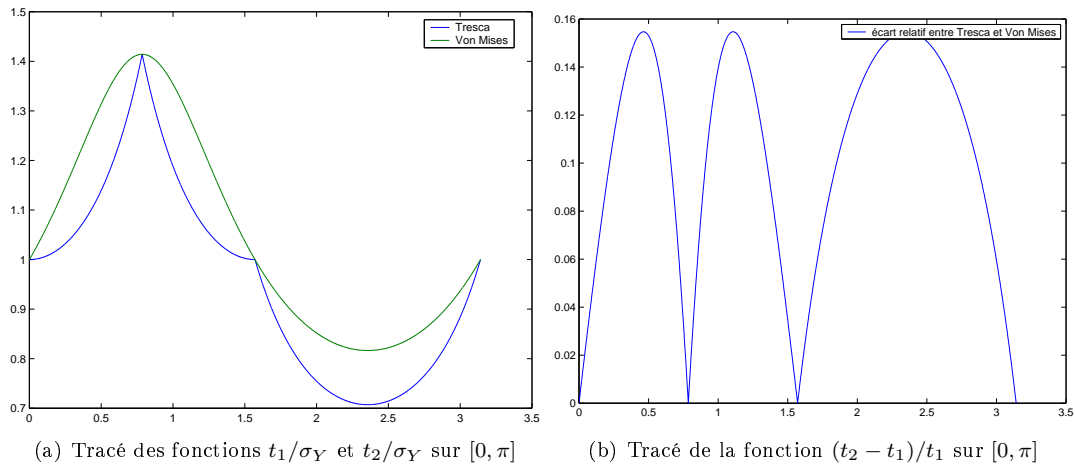


FIG. 4.3 – Comparaison des critères de Tresca et de Von Mises

Compléments sur les figures

On peut aussi faire des sous-figures grâce au package `subfigure`.

développées que matlab.

Il est possible d'exporter des images Xfig contenant du texte L^AT_EX, soit en exportant l'image en deux parties (`.pstex`, `pstex_t`) dont l'une contient l'image au format eps ou pdf et l'autre, le texte L^AT_EX qui sera superposé à l'image lors de la compilation.

Une autre méthode, qui à l'aide d'un script (en Perl) `fig2eps`, permet d'exporter la figure en un seul fichier au format eps et pdf. La syntaxe est suivante :

```

Usage:
fig2ps [-h|--help]          Prints this help
fig2ps [options] file.fig  Converts file.fig into ps using LaTeX for texts.
    --[no]gv runs or not gv at the end;
    --packages=pack1,pack2,... packages to be used
    --add=pack1,pack2,... supplementary packages to be used
    -k|--keep wether to keep or not the temporary files
    --bbox=dvips|gs|a,b,c,d method for the bounding box
    --input=file use file as a TeX template (\input file)
    --dvips=s options to go to dvips
    --preamble=s add string to the preamble
    --[no]pdf wether fig2ps should produce ps or pdf output
    --eps wether the extension of the target file is eps or ps for postscript
    --keepps when producing pdf, tells to keep the intermediary ps file
    --[no]forcespecial forces every text object to be exported as special, that
    is processed with LaTeX.

```

Il est possible de créer une script qui automatise la conversion des figures Xfig en eps et pdf. On suppose l'arborescence du dossier suivante :

- Fichiers `.tex`
- Dossier `images`
 - Dossier `eps` qui contient les images `.eps`
 - Dossier `pdf` qui contient les images `.pdf`
 - Dossier `Xfig` qui contient les images `.fig`
 - Dossier `jpg` qui contient les images `.jpg`
 - etc.
- Dossier `scripts` qui contient les scripts d'automatisation

On peut alors créer un script `convertFig2eps.sh` dans le dossier `scripts` qui convertira toutes les images `.fig` du dossier `images/Xfig` en `.eps` et `.pdf` en les enregistrant respectivement dans les répertoires `images/eps` et `images/pdf`. Le script est le suivant :

```

#----- Converti les fichiers ".fig" en ".eps et .pdf"
for i in `ls ../images/Xfig/*.fig`
do fig2eps --add=graphicx,amsmath,amsfonts,color --pdf --forcespecial
--keepps --nogv --input=../notations $i
done

# -- move the eps files to ../images/eps
for i in `ls ../images/Xfig/*.eps`
do mv $i ../images/eps/
done

# -- move the pdf files to ../images/pdf
for i in `ls ../images/Xfig/*.pdf`

```

```
    do mv $i ../images/pdf/  
done  
##
```

On notera que l'option `--add` permet d'ajouter des packages utiles à la compilation du texte contenu dans l'image. L'option `--input` permet quant à elle d'ajouter un fichier comme par exemple un fichier où sont définies des notations (commandes définies par l'utilisateur par exemple) utilisées dans la figure.

Chapitre 5

Bibliographie et index

Il n'est donné dans ce chapitre qu'un minimum d'indications et d'exemples. On consultera par exemple [15], [8], [5], [16] ou [6].

5.1 Bibliographie

Grâce à un utilitaire très pratique de \LaTeX , qui s'appelle bibtex, on peut faire très facilement une bibliographie.

On écrit d'abord un fichier *.bib dans lequel on écrit les différentes références citées. On respecte un format particulier : selon le type d'ouvrages cités (articles, livre, thèse, ...), il faut indiquer des champs obligatoires et des champs facultatifs. Voir le fichier `exemplebiblio01.bib`.

Chacune des ces références est repérée par une clé, comme le montre l'exemple suivant :

```
@Book{schatzmanbook,
  title =      {Analyse numérique,
                une approche mathématique,
                Cours et exercices},
  author =     {Schatzmann, Michelle},
  publisher =   {Dunod},
  year =       {2001},
}
```

Dans le corps du texte, on cite l'ouvrage écrit par Michelle Schatzman en en citant la clé, précédée de la commande `\cite`, ici `\cite{schatzmanbook}`.

Enfin, dans l'article, on indique à quel endroit la bibliographie doit apparaître, en écrivant :

```
\bibliographystyle{alpha}
\bibliography{exemplebiblio01}
```

Ici, alpha définit le style de la bibliographie et exemplebiblio01 est le nom du fichier (sans extension) où sont stockés les différentes références.

Enfin, il faut compiler le fichier \LaTeX , puis exécuter bibtex en tapant dans la fenêtre de commande (ou le raccourci de l'éditeur)

```
bibtex nomfichier
```

On recompile alors au moins deux fois¹ le fichier L^AT_EX.

Attention, on compilera le même fichier (sans taper l'extension), une fois avec latex, une fois avec bibtex, puis (au moins) une fois de nouveau avec latex.

Voir et compiler le fichier `exemple01.tex`, associé au fichier de bibliographie `exemplebiblio01.bib`.

Fichier : `exemple01.tex`

```
\documentclass[11pt,a4paper]{article}

\input{../style/stylebase01}

\title{exemple01.tex}
\author{Moi Même}

%%% corps du document

\begin{document}
\selectlanguage{french}

\maketitle

On consultera les trois références suivantes :
\cite{schatzmanbook},
\cite{butcher} et
\cite{bornemann}.

\bibliographystyle{alpha}
\bibliography{exemplebiblio01}

\end{document}
```

Cela devrait fournir le résultat suivant :

On consultera les trois références suivantes : [Sch01], [But63] et [Bor02].

Bibliographie

- [Bor02] Folkmar Bornemann. Runge-kutta methods, trees, and *mathematica*. Disponible sur le web : <http://xxx.lanl.gov/>, novembre 2002.
- [But63] J. C. Butcher. Coefficients for the study of runge kutta integration processes. *J. Austral. Math. Soc.*, 3 :185–201, 1963.
- [Sch01] Michelle Schatzmann. *Analyse numérique, une approche mathématique, Cours et exercices*. Dunod, 2001.

Pour ceux qui utilisent Emacs, lorsqu'un fichier *.bib est ouvert, il suffit de sélectionner le menu Entry

¹Principe général à L^AT_EX, puisque des fichiers auxiliaires sont successivement créés.

Type et de choisir l'une des possibilité (article, book, report, ...) pour voir apparaître la structure à remplir, champ par champ, en éliminant, le cas échéant, les champs optionnels. On aurait par exemple

```
@Book{,
  ALTAauthor =      {},
  ALTeditor =       {},
  title =          {},
  publisher =       {},
  year =           {},
  OPTkey =          {},
  OPTvolume =       {},
  OPTnumber =       {},
  OPTseries =       {},
  OPTaddress =      {},
  OPTedition =      {},
  OPTmonth =        {},
  OPTnote =         {},
  OPTannote =       {}
}
```

Sur le web, il existe une base de données² qui permet de récupérer directement les données bibliographiques au format bibtex : voir <http://www.ams.org/mrlookup>. Cela fournit en deux clics le genre de chose suivante :

```
@article {MR1977105,
  AUTHOR = {Br{\`e}zis, Ha{\`i}m and Arendt, Wolfgang and
    Barth{\`e}lemy, Louise and Crandall, Michael and Carillo, Jose
    and Diaz, J. Ildefonso and Goldstein, Jerome A. and Pierre,
    Michel and Vazquez, Juan Luis},
  TITLE = {Introduction},
  JOURNAL = {J. Evol. Equ.},
  FJOURNAL = {Journal of Evolution Equations},
  VOLUME = {3},
  YEAR = {2003},
  NUMBER = {1},
  PAGES = {1--9},
  ISSN = {1424-3199},
  MRCLASS = {01A70},
  MRNUMBER = {1 977 105},
}
```

Le logiciel JabRef (fourni avec le DVD) permet de créer une base de donnée bibliographique (fichier .bib) de manière simple et rapide. Le logiciel dispose de fonctionnalités de recherche et permet d'autre part d'accéder à certaines bases de données bibliographique en ligne (IEEEExplore par exemple), de faire des recherches par mots-clés (titre, auteur, etc.) et de télécharger directement les références trouvées au format de BibTeX et de les inclure dans la base de donnée.

5.2 Index

Cette section s'inspire beaucoup de [15].

²Cette base de données est la partie gratuite, de mathscinet, une base de données très pratique en mathématiques et physique. Elle est payante !

L'index est un élément fort utile de nombreux ouvrages. L^AT_EX et le programme associé `makeindex`³ permettent de créer des index assez facilement. Dans cette introduction, seules les commandes élémentaires de gestion d'un index sont présentées. Pour une description plus détaillée, reportez-vous à [8] ou à [6].

Pour utiliser cette fonctionnalité, le package `makeidx` doit être chargée dans le préambule avec :

```
\usepackage{makeidx}
```

La création de l'index doit être activée par la commande :

```
\makeindex
```

placée dans le préambule.

Le contenu de l'index est défini par une série de commandes :

```
\index{clefs}
```

où `clef` est un mot-clef de l'index. Vous insérez des commandes `\index` aux endroits du texte que vous voulez voir référencés par l'index. Le tableau 5.1 explique la syntaxe de l'argument *clef* avec plusieurs exemples.

Exemple	Résultat	Commentaires
<code>\index{hello}</code>	hello, 1	Entrée normale
<code>\index{hello!Peter}</code>	Peter, 3	Sous-entrée de 'hello'
<code>\index{Sam@\textsl{Sam}}</code>	<i>Sam</i> , 2	Entrée formatée
<code>\index{Lin@\textbf{Lin}}</code>	Lin , 7	idem
<code>\index{Jenny textbf}</code>	Jenny, 3	Numéro de page formaté
<code>\index{Joe textit}</code>	Joe, 5	idem
<code>\index{eolienne@éolienne}</code>	éolienne, 4	clef accentuée

TAB. 5.1 – Exemples de clefs d'index

On écrit dans le source l'instruction

```
\printindex
```

à l'endroit où on veut voir l'index apparaître.

On compile le fichier L^AT_EX, puis on lance `makeindex` en tapant :

```
makeindex nomfichier
```

et on recompile alors plusieurs fois le fichier L^AT_EX.

Le package `showidx` permet de visualiser les entrées de l'index dans la marge gauche du texte. Cela permet la relecture et la mise au point de l'index.

On consultera l'index produit en fin de ce document, où figurent différents exemples du tableau 5.1. . Cet index n'a qu'une valeur démonstrative et n'est pas complet!

On pourra aussi consulter le fichier `exemple05.tex` :

³sur les systèmes qui ne supportent pas les noms de fichiers de plus de huit caractères, ce programme s'appelle `makeidx`.

Fichier : exemple05.tex

```
\documentclass[11pt,a4paper]{article}

\input{../style/stylebase01}
\usepackage{makeidx}

\makeindex

\title{exemple05.tex}
\author{Moi Même}

%%% corps du document

\begin{document}
\selectlanguage{french}

\maketitle

On parlera de matrices \index{matrice}, et
plus particulièrement de matrices
symétriques\index{matrice!symétrique}.
Ces types de matrices est souvent utilisés
en mécanique, notamment pour la méthode
des éléments finis\index{éléments finis}.

\printindex

\end{document}
```

On consultera aussi le fichier `exemple10.tex` où est chargé le package `showidx`.

Chapitre 6

Compléments

Dans cette section, on verra quelques unes des nombreuses possibilités offertes par \LaTeX .

6.1 Présentations à la façon PowerPoint

Cette partie est provient du site <http://www.tuteurs.ens.fr>¹.

Historiquement, il y avait les classes slides, et seminar qui sont beaucoup trop basiques. Ensuite sont apparus pdfscreen et prosper, qui sont beaucoup plus évolués. Enfin est venu beamer, qui est nettement le meilleur choix à l'heure actuelle.

Beamer est donc un package \LaTeX prévu pour faire des présentations. Il est conçu pour produire du pdf. Pour ce faire, on compile le fichier .tex en utilisant pdflatex au lieu de latex. Le pdf obtenu peut alors être visualisé avec n'importe quel afficheur de pdf, comme Acrobat Reader.

Beamer vient avec une documentation très étoffée, accessible à l'adresse suivante : <http://www.ctan.org/tex-archive/macros/latex/contrib/beamer/doc/>.

Exemple de base :

Fichier : Exemple de base du package beamer

```
\documentclass{beamer}
\usepackage[latin1]{inputenc}
\usetheme{Warsaw}

\title{Hello World!}
\author{B. Meuhr}\institute{École Normale Supérieure, département de pipologie}

\begin{document}

\begin{frame}
\titlepage
\end{frame}

\begin{frame}
\textsc{Beamer}, c'est vachement tout beau!
\end{frame}

\begin{frame}
```

¹<http://www.tuteurs.ens.fr/logiciels/latex/beamer.html>

```
Et c'est facile.
\end{frame}

\end{document}
```

Beamer permet de définir des séquences de slides différant seulement par des apparitions ou disparitions (ou mise en grisé) de morceaux de texte. Par exemple, ce code :

```
\begin{frame}
\begin{itemize}
\item<13> Ploum!
\item<2-4> Plim?
\item Plum...
\end{itemize}
\end{frame}
```

va produire une séquence de 4 slides. Toutes les 4 auront un « Plum... » sur la troisième ligne. La première ligne contiendra un « Ploum! » sur la première et la troisième slide, tandis qu'elle sera vide sur la deuxième et la quatrième. La deuxième ligne, elle, sera vide sur la première slide, et contiendra « Plim? » sur les suivantes.

Comme le montre cet exemple, un certain nombre de commandes ont été modifiées pour prendre un argument optionnel entre <>. Dans cet argument on peut mettre un nombre, un intervalle (deux nombres séparés par un -), ou plusieurs nombres ou intervalles séparés par des virgules. Ces nombres spécifient sur quelles slides de la séquence qui va être créée la commande doit « avoir un effet ». Pour `\item`, cela veut dire afficher la bille et le texte qui en dépend. Mais pour les commandes de changement de police ou de couleur, cela veut dire que le changement ne doit être effectué que sur les slides en question. Par exemple, un :

```
\textbf<2>{ploum}
```

aura pour effet de mettre « ploum » en gras sur la deuxième slide, et en normal sur la première.

L'argument optionnel entre <> n'est pas un effet magique : les auteurs de Beamer ont du modifier toutes les commandes... qu'ils ont modifiées. Pour les autres, il faut se débrouiller à la main. On dispose des commandes `\only` et `\uncover` qui prennent un argument entre <>, puis un argument normal entre {}, qu'elles n'affichent que sur les slides spécifiées. La différence entre les deux est qu'avec `only`, le texte est jeté, tandis qu'avec `uncover`, il occupe sa place mais est transparent. Par exemple, avec :

```
pif \uncover<2>{paf} Abcd-Efgh \only<1>{Ijklmn} ouais!
```

on obtiendra :

```
pif      Abcd-Efgh Ijklmn ouais!
```

sur la première slide, et :

```
pif paf Abcd-Efgh ouais!
```

sur la deuxième.

Par défaut, le texte caché est complètement invisible. Il est possible de l'avoir en grisé en mettant un :

```
\setbeamercovered{transparent}
```

Voir aussi l'exemple `exemple02.tex` plus complet (disponible dans le répertoire `beamer`).

6.2 Réalisation d'un sujet d'examen

Pour la réalisation de sujet d'examen, on peut utiliser la classe de document `exam`. Un exemple est présent sur le DVD, il vous suffit d'adapter le document à votre besoin pour réaliser un examen. Lorsque le code \LaTeX est écrit, il suffit d'ajouter la commande (`\printanswers`) afin de compiler le sujet d'examen avec ou sans le corrigé.

Afin de spécifier les questions du sujet, il est nécessaire de déclarer l'environnement `questions` (délimité par `\begin{questions}` et `\end{questions}`). Ensuite, on met en évidence chaque question de l'énoncé avec (`\begin{question}[nb points]` et `\end{question}`), avec entre crochet le nombre de points attribué à la question. La numérotation des questions est automatique. Il est possible de calculer automatiquement la somme des points de l'examen avec le commande `\addpoints`. L'affichage du total s'effectue avec `\numpoints`. Afin d'indiquer la réponse à la question, on utilise (`\begin{solution}` et `\end{solution}`). Ainsi, lorsque l'affichage de la solution est demandée, le compilateur inclus dans le document la réponse désirée juste en dessous de la question.

Exemple de code \LaTeX avec la classe de document `exam` :

Entourez la (ou les) lettre(s) correspondant à votre choix.

```
\begin{questions}

\begin{question}[4]
\LaTeX{} a été écrit par L. Lamport en: \\
\begin{oneparchoices}
  \choice 1960
  \choice 1982
  \choice 1885
  \choice 2000
\end{oneparchoices}
\end{question}

\begin{solution}
\LaTeX{} a été écrit par L. Lamport en 1982.
\end{solution}

\end{questions}
```

6.3 Algorithmes

Grâce aux packages `algorithm` et `algorithmic` on peut aussi insérer des algorithmes, comme le montre l'algorithme 1 on the following page.

On pourra consulter [18].

On pourra consulter le fichier `exemplealgo01.tex`.

On verra dans le fichier style qui lui est associé (fichier `/style/stylealgo01`) des macros de redéfinition de commandes telles que

```
\renewcommand{\algorithmicend}{\textbf{fin}}
```

Algorithme 1 Détermination des différences divisées : $diff_div_dist(n, x, y \longrightarrow d)$

entrée :

n : entier naturel tel que $n + 1$ désigne le nombre de points d'interpolation ;

x : vecteur des $n + 1$ abscisses réelles des points d'interpolation : pour tout i , $x(i) = x_i$;

y : vecteur des $n + 1$ valeurs réelles prises par f en $(x_i)_{0 \leq i \leq n}$: pour tout i , $y_i = f(x_i)$.

sortie :

d : vecteur des différences divisées. Pour tout i de $\{0, \dots, n\}$, $d(i) = f[x_0, \dots, x_i]$.

Début de corps

pour $i = 0$ à n **faire**

$d(i) \longleftarrow y(i)$

fin pour

pour $i = 1$ à n **faire**

pour $j = n$ à i **faire**

$d(j) \longleftarrow \frac{d(j) - d(j-1)}{x(j) - x(j-i)}$

fin pour

fin pour

Fin de corps

Cette macro permet de remplacer le mot clé de fin de boucle **end** par **fin**.

Ces macros permettent de franciser les termes de l'algorithme, qui par défaut² sont en anglais.

Vous pouvez essayer de supprimer cette francisation pour voir apparaître les termes anglais.

6.4 Listings

Le package `listings` permet d'insérer des codes de programmes directement dans un document \LaTeX .

On pourra consulter [6] ou [11].

Vous pouvez voir page 33, ou un peu plus bas, le résultat produit. Ce package respecte les décalages créés dans le texte pour mettre en évidence les différentes boucles par exemple.

```
function s=somme(n)
% calcul de somme
s=0;
for i=1:n
    s=s+1/(i ^ 2);
end
```

On entrera dans l'entête du document \LaTeX les instructions :

```
\usepackage{listings}
\lstset{language=Matlab}
```

²Il y a des cas, comme celui-ci, où `babel` ne francise pas les termes. En fait, cela est logique, `babel` est un package général, utile à tous, et qui ne peut prendre en compte les différents packages qui apparaissent régulièrement. Heureusement, l'auteur des packages `algorithm` et `algorithmic` (comme les bons auteurs de packages) laisse aux utilisateurs la possibilité de redéfinir les termes dans leur langue.

Ici, le langage spécifié est matlab. Cela permet de faire apparaître en gras les mots réservés de matlab³ Ce package reconnaît un certain nombre de langage de programmation comme mathematica, fortran, C ...

Un exercice supplémentaire consisterait à voir que sous matlab, le soft de l'exemple précédent est très mal écrit⁴ et qu'il peut être avantageusement remplacé par

```
function s=somme(n)
% calcul de somme
s=sum((1./(1:n)).^2);
```

6.5 L^AT_EX et d'autres logiciels de calculs

Page 33, on donné l'exemple d'un soft matlab qui permet de faire des courbes et des les exporter au format postscript pour être insérer sous L^AT_EX (voir figure 4.3 on page 33).

Il existe bien entendu d'autres logiciels (libres ou non) qui permettent de faire ce genre de graphiques.

Il existe aussi une possibilité offerte par matlab⁵ (ou maple). On exécute un calcul en symbolique dont on récupère le résultat sous forme de chaîne dédiée à L^AT_EX qu'on peut insérer ensuite dans un document.

Par exemple, sous matlab, on tape la séquence suivante :

```
syms C R X Y theta lambda;
M=C+R*X*(sin(theta)-sin(lambda))-R*Y*(cos(theta)-cos(lambda));
W=int(M^2,theta,0,lambda);
xB=simplify(diff(W,X));
yB=simplify(diff(W,Y));
omegaB=simplify(diff(W,X));
xB=latex(subs(xB,lambda,pi/2));
yB=latex(subs(yB,lambda,pi/2));
omegaB=latex(subs(omegaB,lambda,pi/2));

disp(xB);
disp(yB);
disp(omegaB);
```

On obtient alors «à peu près»⁶ le résultat suivant :

$$\begin{aligned}x_B &= \frac{R^2}{4EI} \left((3R - 8R\pi)X + 2RY + (4 - 2\pi)C \right), \\y_B &= \frac{R^2}{4EI} \left(2RX + RY\pi - 4C \right), \\\omega_B &= \frac{R}{2EI} \left((2R - R\pi)X - 2RY + C\pi \right),\end{aligned}$$

On pourra vérifier que les calculs faits en symboliques correspondent au calcul d'énergie de déformation dans

³Parfois, matlab progresse plus vite que les packages L^AT_EX et quand des nouvelles fonctions apparaissent en matlab il faut les redéfinir pour que listings les reconnaissent.

⁴À la compilation, L^AT_EX ne reconnaît toujours pas les codes matlab mal écrits !

⁵Je ne parle que de matlab dans ce cas ; peut-être existe-t-il des logiciels libre qui offre aussi cette possibilité.

⁶car il faut parfois modifier les expressions obtenues.

des poutres circulaire :

$$\begin{aligned} M(\theta) &= C + RX(\sin \theta - \sin \lambda) - RY(\cos \theta - \cos \lambda), \\ W &= \frac{R}{2EI} \int_0^\lambda M^2 d\theta, \\ x_B &= \frac{\partial W}{\partial X}, \\ y_B &= \frac{\partial W}{\partial Y}, \\ \omega_B &= \frac{\partial W}{\partial C}, \end{aligned}$$

et λ est égal à $\pi/2$.

6.6 Créer ses propres packages

Les différents packages que vous avez utilisés sont en fait des fichiers *.sty où sont redéfinis toutes les nouvelles fonctions. Vous n'avez pas besoin de comprendre comment ces fichiers fonctionnent⁷. Cependant, contrairement à ce que j'ai écrit dans le chapitre 1, page 10, on ne parlera pas de fichier style, mais de package personnel : vous mettez dans un fichier qui s'appelle **monpackage.sty** l'ensemble de vos paramètres personnels et des packages que vous appelez. La première ligne de ce fichier doit indiquer à L^AT_EX qu'on déclare des package en écrivant

```
\ProvidesPackage{monpackage}
```

Au début de votre fichier tex principal, vous n'avez plus qu'à écrire⁸ :

```
\usepackage{monpackage}
```

Par exemple, le début du fichier principal de cette documentation est

```
\documentclass[11pt,reqno,final,a4paper]{amsbook}

\usepackage{noticeinitiationlatex}

\begin{document}
\selectlanguage{french}
```

La début de mon fichier **noticeinitiationlatex.sty** est

```
\ProvidesPackage{noticeinitiationlatex}
```

Attention, cela suppose que L^AT_EX sache trouver où se trouve votre fichier **monpackage.sty** ; si ce n'est pas le cas, il faut indiquer en utilisant **\usepackage** et **\ProvidesPackage** le chemin du fichier **monpackage.sty**.

⁷Du moins, dans un premier temps.

⁸à la place de input, comme j'avais écrit dans le chapitre 1.

6.7 Un dernier exercice pour finir

Si vous vous sentez un peu plus à l'aise avec \LaTeX , essayez de créer un document en utilisant la classe `book` (voir section 2.1) et en faisant différents chapitres⁹. Tâchez de regrouper dans ce fichier l'ensemble des éléments qui vous paraissent utiles pour la suite. Il faudra réfléchir aux différents éléments et packages à déclarer dans votre entête, éventuellement à stocker dans un fichier style à part.

6.8 Le reste de \LaTeX ?

Il reste encore beaucoup à apprendre : comprendre les erreurs de compilation de \LaTeX , comprendre les différents fichiers créés, utiliser des nouveaux packages, personnaliser \LaTeX , être capable d'installer \LaTeX , créer ses propres macros

Tout cela se fera progressivement, au fur et à mesure de vos besoins. On pourra consulter, entre autres, quelques références indiquées dans l'annexe A.

⁹Vous verrez l'avantage d'utiliser les commandes `include` et `includeonly` décrites dans la section 2.2.2 on page 13.

Annexe A

Ce qu'on peut trouver sur L^AT_EX

Vaste sujet !

A.1 Ouvrages

La beauté de L^AT_EX est qu'il n'appartient à personne et on pourra trouver sur le web des tas de références. Néanmoins, il existe quelques ouvrages de références, déjà cités dans cet ouvrage. Je ne peux m'empêcher de les citer : elles sont très complètes et certaines d'entre elles sont devenues des références célèbres¹, comme par exemple [8, 6, 16, 12].

À la bibliothèque de l'UTBM, vous en trouverez quelques unes d'entre elles. Il existe aussi d'autres ouvrages comme [13].

Citons aussi, pour la curiosité le livre de J. Hicgham Nicholas [14] ou le papier de Michèle Audin, destinés à ceux qui veulent écrire des mathématiques [3] (cette fois-ci, ce n'est pas la forme qui est évoquée, mais le fond).

Il existe un très bon ouvrage écrit par deux des trois des auteurs de [8], utile à tous ceux qui veulent dessiner sous L^AT_EX, faire des partitions, dessiner des positions d'échecs, de jeu de go, faire de la chimie, des schémas électroniques : [9].

A.2 Sur sa propre machine

Souvent, les distributions standard de L^AT_EX contiennent des documentations² sur L^AT_EX. Ces documentations sont souvent relatives à des packages et sont écrites par leur propres auteurs. Il en est ainsi de [10, 7, 17, 18, 11, 2].

Je tâcherai d'apporter un exemplaire de chacun de ces rapports pour l'atelier L^AT_EX.

¹C'est le chapitre 8 de [8] que je distribuerai photocopié pour l'atelier L^AT_EX.

²On pourra consulter, pour ceux qui ont linux, une aide en html, très bien faite, avec un certain nombre de doc sur les packages notamment. Sur les distributions standard (à moduler bien entendu), on consultera le fichier `/usr/share/texmf/doc/index.html`. Cette aide n'est pas destinée au débutant, mais permet de trouver des informations, entre autres, sur la plupart des packages existant ou des références pratiques.

A.3 Sur le web

Tapez sous un bon moteur de recherche (google, pour ne pas le citer), «introduction à Latex» et vous aurez énormément de références.

On trouvera une définition très succincte sur L^AT_EX et de nombreux liens internet sur la Wikipedia sur <http://fr.wikipedia.org/wiki/LaTeX>.

Des FAQ (Frequently asked questions ou «foire aux questions») L^AT_EX permettent de répondre à un certain nombre de questions que se posent les L^AT_EXistes. On pourra consulter par exemple la FAQ de l'université Lille 3 (<http://www.grappa.univ-lille3.fr/FAQ-LaTeX/>).

On ira aussi consulter le célèbre (La)TeX navigator (<http://tex.loria.fr/index.html>), où vous trouverez des documentations, des distributions de L^AT_EX, des packages. On pourra aussi télécharger par FTP des packages sur les archives du CTAN (<http://www.ctan.org>).

On peut trouver sur le web deux incontournables documentations : un joli manuel pour L^AT_EX_{2 ϵ} de Benjamin Bayart [5] (voir <http://www.loria.fr/services/tex/general/manuel2ep.ps.gz>; une nouvelle édition est en préparation : voir <http://jmpl.fr.eu.org/>) et une Une courte (?) introduction à L^AT_EX_{2 ϵ} ³ de Tobias Oetiker [15] (voir <http://www.laas.fr/~matthieu/cours/latex2e/>). Cette dernière présente le gros avantage d'être distribuée avec ses sources⁴ L^AT_EX.

Une documentation sur L^AT_EX est disponible au format html sur <ftp://ftp.giss.nasa.gov/pub/sgreen/latex/latex.tar.gz> ou <http://www.giss.nasa.gov/latex/>.

On pourra aussi consulter [4], disponible sur <http://www.loria.fr/services/tex/general/apprends-latex.ps.gz>.

Enfin, on pourra lire l'aide mémoire L^AT_EX de Vincent Suguin disponible sur <http://tex.loria.fr/general/aide-memoire-latex-seguin1998.pdf>, qui constitue un très bon résumé des principales commandes L^AT_EX.

³Traduction de «The not so short introduction to L^AT_EX_{2 ϵ} », assurée par M. Herrb ; la traduction quelque compléments français spécifiques.

⁴J'en m'en suis beaucoup inspiré pour réaliser les chapitres 2, 3 et une partie du chapitre 4.

Annexe **B**

Les mathématiques avec les packages de l' $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX [8]

La pages suivantes sont un extrait du livre (chapitre 8) « The \LaTeX Companion » [8].

Higher Mathematics

Basic L^AT_EX offers a high level of mathematical typesetting capabilities. However, when complex equations or other mathematical constructs have to be input repeatedly, it is up to you to define new commands or environments to ease the burden of typing. The American Mathematical Society (AMS), recognizing that fact, has sponsored the development of extensions to T_EX, known as $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX. They make the preparation of mathematical compuscripts less time-consuming and the copy more consistent.

Recently these extensions were ported to L^AT_EX in the form of a set of packages known as “ $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX” [?]. As some parts of $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX had to do with mathematics fonts the corresponding L^AT_EX packages went into a separate distribution called “AMSF_{onts}”, rather than into $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX.

8.1 The $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX Project

$\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX was originally released for general use in 1982. Its main strength is that it facilitates mathematical typesetting, while producing output that satisfies the high standards of mathematical publishing. It provides a predefined set of natural commands such as `\matrix` and `\text` that make complicated mathematics reasonably convenient to type. These commands incorporate the typesetting experience and standards of the American Mathematical Society, to handle problematic possibilities, such as matrices within matrices or a word of text within a subscript, without burdening the user.

$\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX lacks certain useful L^AT_EX features such as automatic numbering that adjusts to addition or deletion of material being the primary one. Nor does

it have the laborsaving abilities of L^AT_EX for preparing indexes, bibliographies, tables, or simple diagrams. These features are such a convenience for authors that the use of L^AT_EX spread rapidly in the mid-1980s (a reasonably mature version of L^AT_EX was available by the end of 1983), and the American Mathematical Society began to be asked by its authors to accept electronic submissions in L^AT_EX.

Thus, the $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX project came into being in 1987 and three years later $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX version 1.0 was released. The conversion of $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX's mathematical capabilities to L^AT_EX, and the integration with the NFSS, were done by Frank Mittelbach and Rainer Schöpf, working as consultants to the AMS, with assistance from Michael Downes of the AMS technical support staff.

The most often used packages are `amsmath` (from $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX) and `amssymb` (from the AMSFonts distribution). To invoke them in a document you write, e.g., `\usepackage{amsmath}` in the usual way. Installation and usage documentation is included with the packages. For `amssymb` the principal piece of documentation is the *AMSFonts User's Guide* (`amsfndoc.tex`); for `amsmath` it is the *$\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX User's Guide* (`amsldoc.tex`).¹

8.2 Fonts and Symbols in Formulae

8.2.1 Mathematical Symbols

(\mathcal{L} 42–47)

Tables 8.2 on the next page to 8.11 on page 227 review the mathematical symbols available in standard L^AT_EX. You can put a slash through a L^AT_EX symbol by preceding it with the `\not` command, for instance,

(\mathcal{L} 44)

$u \not\prec v$ or $a \not\in \mathbf{A}$ `$u \not< v$` or `$a \not\in \mathbf{A}$`

Tables 8.12 on page 227 to 8.19 on page 229 show the extra math symbols of the $\mathcal{A}\mathcal{M}\mathcal{S}$ -Fonts, which are automatically available when you specify the `amssymb` package.² However, if you want to define only some of them (perhaps because your T_EX installation has insufficient memory to define all the symbol names), you can use the `amsfonts` package and the `\DeclareMathSymbol` command, which is explained in section 7.7.6.

¹ The AMS distribution also contains a file `diff12.tex` which describes differences between version 1.1 and 1.2 of $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX. Note in particular that in versions 1.0 and 1.1 of $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX, which predated L^AT_EX 2_ε, the `amsmath` package was named “`amstex`” and included some of the font-related features that are now separated in the `amssymb` and `amsfonts` packages.

² Note that the Companion uses Lucida math fonts which contain the standard L^AT_EX and $\mathcal{A}\mathcal{M}\mathcal{S}$ symbols but with different shapes compared to the Computer Modern math fonts.

\hat{a}	<code>\hat{a}</code>	\acute{a}	<code>\acute{a}</code>	\bar{a}	<code>\bar{a}</code>	\dot{a}	<code>\dot{a}</code>	\breve{a}	<code>\breve{a}</code>
\check{a}	<code>\check{a}</code>	\grave{a}	<code>\grave{a}</code>	\vec{a}	<code>\vec{a}</code>	\ddot{a}	<code>\ddot{a}</code>	\tilde{a}	<code>\tilde{a}</code>

Table 8.1: Math mode accents (available in L^AT_EX)

α	<code>\alpha</code>	β	<code>\beta</code>	γ	<code>\gamma</code>	δ	<code>\delta</code>	ϵ	<code>\epsilon</code>
ε	<code>\varepsilon</code>	ζ	<code>\zeta</code>	η	<code>\eta</code>	θ	<code>\theta</code>	ϑ	<code>\vartheta</code>
ι	<code>\iota</code>	κ	<code>\kappa</code>	λ	<code>\lambda</code>	μ	<code>\mu</code>	ν	<code>\nu</code>
ξ	<code>\xi</code>	\omicron	<code>\omicron</code>	π	<code>\pi</code>	ϖ	<code>\varpi</code>	ρ	<code>\rho</code>
ϱ	<code>\varrho</code>	σ	<code>\sigma</code>	ς	<code>\varsigma</code>	τ	<code>\tau</code>	υ	<code>\upsilon</code>
ϕ	<code>\phi</code>	φ	<code>\varphi</code>	χ	<code>\chi</code>	ψ	<code>\psi</code>	ω	<code>\omega</code>
Γ	<code>\Gamma</code>	Δ	<code>\Delta</code>	Θ	<code>\Theta</code>	Λ	<code>\Lambda</code>	Ξ	<code>\Xi</code>
Π	<code>\Pi</code>	Σ	<code>\Sigma</code>	Υ	<code>\Upsilon</code>	Φ	<code>\Phi</code>	Ψ	<code>\Psi</code>
Ω	<code>\Omega</code>								

Table 8.2: Greek letters (available in L^AT_EX)

\pm	<code>\pm</code>	\cap	<code>\cap</code>	\diamond	<code>\diamond</code>	\oplus	<code>\oplus</code>
\mp	<code>\mp</code>	\cup	<code>\cup</code>	\triangleup	<code>\triangleup</code>	\ominus	<code>\ominus</code>
\times	<code>\times</code>	\uplus	<code>\uplus</code>	\triangledown	<code>\triangledown</code>	\otimes	<code>\otimes</code>
\div	<code>\div</code>	\sqcap	<code>\sqcap</code>	\triangleleft	<code>\triangleleft</code>	\oslash	<code>\oslash</code>
$*$	<code>\ast</code>	\sqcup	<code>\sqcup</code>	\triangleright	<code>\triangleright</code>	\odot	<code>\odot</code>
\star	<code>\star</code>	\vee	<code>\vee</code>	\lhd^a	<code>\lhd^a</code>	\bigcirc	<code>\bigcirc</code>
\circ	<code>\circ</code>	\wedge	<code>\wedge</code>	\rhd^a	<code>\rhd^a</code>	\dagger	<code>\dagger</code>
\bullet	<code>\bullet</code>	\setminus	<code>\setminus</code>	\unlhd^a	<code>\unlhd^a</code>	\ddagger	<code>\ddagger</code>
\cdot	<code>\cdot</code>	\wr	<code>\wr</code>	\unrhd^a	<code>\unrhd^a</code>	\amalg	<code>\amalg</code>

^a Not predefined in NFSS. Use the `latexsym` or `amssymb` package.

Table 8.3: Binary operation symbols (available in L^AT_EX)

\leq	<code>\leq</code>	\geq	<code>\geq</code>	\equiv	<code>\equiv</code>	\models	<code>\models</code>	\prec	<code>\prec</code>
\succ	<code>\succ</code>	\sim	<code>\sim</code>	\perp	<code>\perp</code>	\preceq	<code>\preceq</code>	\succcurlyeq	<code>\succcurlyeq</code>
\simeq	<code>\simeq</code>	\mid	<code>\mid</code>	\ll	<code>\ll</code>	\gg	<code>\gg</code>	\asymp	<code>\asymp</code>
\parallel	<code>\parallel</code>	\subset	<code>\subset</code>	\supset	<code>\supset</code>	\approx	<code>\approx</code>	\bowtie	<code>\bowtie</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\cong	<code>\cong</code>	\Join	<code>\Join</code>	\sqsubset	<code>\sqsubset</code>
\sqsupseteq	<code>\sqsupseteq</code>	\neq	<code>\neq</code>	\smile	<code>\smile</code>	\sqsubseteq	<code>\sqsubseteq</code>	\sqsupseteq	<code>\sqsupseteq</code>
\doteq	<code>\doteq</code>	\frown	<code>\frown</code>	\in	<code>\in</code>	\ni	<code>\ni</code>	\propto	<code>\propto</code>
$=$	<code>=</code>	\vdash	<code>\vdash</code>	\dashv	<code>\dashv</code>	$<$	<code><</code>	$>$	<code>></code>

Table 8.4: Relation symbols (available in L^AT_EX)

\leftarrow	<code>\leftarrow</code>	\longleftarrow	<code>\longleftarrow</code>	\uparrow	<code>\uparrow</code>
\Leftarrow	<code>\Leftarrow</code>	\Longleftarrow	<code>\Longleftarrow</code>	\Uparrow	<code>\Uparrow</code>
\rightarrow	<code>\rightarrow</code>	\longrightarrow	<code>\longrightarrow</code>	\downarrow	<code>\downarrow</code>
\Rightarrow	<code>\Rightarrow</code>	\Longrightarrow	<code>\Longrightarrow</code>	\Downarrow	<code>\Downarrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\longleftrightarrow	<code>\longleftrightarrow</code>	\updownarrow	<code>\updownarrow</code>
\Leftrightarrow	<code>\Leftrightarrow</code>	\Longleftrightarrow	<code>\Longleftrightarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\mapsto	<code>\mapsto</code>	\longmapsto	<code>\longmapsto</code>	\nearrow	<code>\nearrow</code>
\hookrightarrow	<code>\hookrightarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>	\searrow	<code>\searrow</code>
\leftharpoonup	<code>\leftharpoonup</code>	\rightharpoonup	<code>\rightharpoonup</code>	\swarrow	<code>\swarrow</code>
\leftharpoondown	<code>\leftharpoondown</code>	\rightharpoondown	<code>\rightharpoondown</code>	\nwarrow	<code>\nwarrow</code>

Table 8.5: Arrow symbols (available in L^AT_EX)

\dots	<code>\ldots</code>	\cdots	<code>\cdots</code>	\vdots	<code>\vdots</code>	\ddots	<code>\ddots</code>	\aleph	<code>\aleph</code>
\prime	<code>\prime</code>	\forall	<code>\forall</code>	∞	<code>\infty</code>	\hbar	<code>\hbar</code>	\emptyset	<code>\emptyset</code>
\exists	<code>\exists</code>	∇	<code>\nabla</code>	\surd	<code>\surd</code>	\Box^a	<code>\Box^a</code>	\triangle	<code>\triangle</code>
\diamond	<code>\Diamond^a</code>	\imath	<code>\imath</code>	\jmath	<code>\jmath</code>	ℓ	<code>\ell</code>	\neg	<code>\neg</code>
\top	<code>\top</code>	\flat	<code>\flat</code>	\natural	<code>\natural</code>	\sharp	<code>\sharp</code>	\wp	<code>\wp</code>
\perp	<code>\bot</code>	\clubsuit	<code>\clubsuit</code>	\diamondsuit	<code>\diamondsuit</code>	\heartsuit	<code>\heartsuit</code>	\spadesuit	<code>\spadesuit</code>
\mho^a	<code>\mho^a</code>	\Re	<code>\Re</code>	\Im	<code>\Im</code>	\angle	<code>\angle</code>	∂	<code>\partial</code>

^a Not predefined in NFSS. Use the latexsym or amssymb package.Table 8.6: Miscellaneous symbols (available in L^AT_EX)

\sum	<code>\sum</code>	\prod	<code>\prod</code>	\coprod	<code>\coprod</code>	\int	<code>\int</code>	\oint	<code>\oint</code>
\bigcap	<code>\bigcap</code>	\bigcup	<code>\bigcup</code>	\bigsqcup	<code>\bigsqcup</code>	\bigvee	<code>\bigvee</code>	\bigwedge	<code>\bigwedge</code>
\bigodot	<code>\bigodot</code>	\bigotimes	<code>\bigotimes</code>	\bigoplus	<code>\bigoplus</code>	\biguplus	<code>\biguplus</code>		

Table 8.7: Variable-sized symbols (available in L^AT_EX)

\arccos	<code>\arccos</code>	\cos	<code>\cos</code>	\csc	<code>\csc</code>	\exp	<code>\exp</code>	\ker	<code>\ker</code>	\limsup	<code>\limsup</code>	\min	<code>\min</code>	\sinh	<code>\sinh</code>
\arcsin	<code>\arcsin</code>	\cosh	<code>\cosh</code>	\deg	<code>\deg</code>	\gcd	<code>\gcd</code>	\lg	<code>\lg</code>	\ln	<code>\ln</code>	\Pr	<code>\Pr</code>	\sup	<code>\sup</code>
\arctan	<code>\arctan</code>	\cot	<code>\cot</code>	\det	<code>\det</code>	\hom	<code>\hom</code>	\lim	<code>\lim</code>	\log	<code>\log</code>	\sec	<code>\sec</code>	\tan	<code>\tan</code>
\arg	<code>\arg</code>	\coth	<code>\coth</code>	\dim	<code>\dim</code>	\inf	<code>\inf</code>	\liminf	<code>\liminf</code>	\max	<code>\max</code>	\sin	<code>\sin</code>	\tanh	<code>\tanh</code>

Table 8.8: Log-like symbols (available in L^AT_EX)

\uparrow	<code>\uparrow</code>	\Uparrow	<code>\Uparrow</code>	\downarrow	<code>\downarrow</code>	\Downarrow	<code>\Downarrow</code>
$\{$	<code>\{</code>	$\}$	<code>\}</code>	\updownarrow	<code>\updownarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\lfloor	<code>\lfloor</code>	\rfloor	<code>\rfloor</code>	\lceil	<code>\lceil</code>	\rceil	<code>\rceil</code>
\langle	<code>\langle</code>	\rangle	<code>\rangle</code>	\angle	<code>\angle</code>	\backslash	<code>\backslash</code>
$ $	<code> </code>	$\ $	<code>\ </code>				

Table 8.9: Delimiters (available in L^AT_EX)

$\}$	<code>\rmoustache</code>	\int	<code>\lmoustache</code>	$\}$	<code>\rgroup</code>	$\{$	<code>\lgroup</code>
$ $	<code>\arrowvert</code>	$\ $	<code>\Arrowvert</code>	$ $	<code>\bracevert</code>		

Table 8.10: Large delimiters (available in L^AT_EX)

\widetilde{abc}	<code>\widetilde{abc}</code>	\widehat{abc}	<code>\widehat{abc}</code>
\overleftarrow{abc}	<code>\overleftarrow{abc}</code>	\overrightarrow{abc}	<code>\overrightarrow{abc}</code>
\overline{abc}	<code>\overline{abc}</code>	\underline{abc}	<code>\underline{abc}</code>
\overbrace{abc}	<code>\overbrace{abc}</code>	\underbrace{abc}	<code>\underbrace{abc}</code>
\sqrt{abc}	<code>\sqrt{abc}</code>	$\sqrt[n]{abc}$	<code>\sqrt[n]{abc}</code>
f'	<code>f'</code>	$\frac{abc}{xyz}$	<code>\frac{abc}{xyz}</code>

Table 8.11: L^AT_EX math constructs

Γ	<code>\digamma</code>	\varkappa	<code>\varkappa</code>	\beth	<code>\beth</code>	\daleth	<code>\daleth</code>	\gimel	<code>\gimel</code>
----------	-----------------------	-------------	------------------------	---------	--------------------	-----------	----------------------	----------	---------------------

Table 8.12: AMS Greek and Hebrew (available with amssymb package)

\ulcorner	<code>\ulcorner</code>	\urcorner	<code>\urcorner</code>	\llcorner	<code>\llcorner</code>	\lrcorner	<code>\lrcorner</code>
-------------	------------------------	-------------	------------------------	-------------	------------------------	-------------	------------------------

Table 8.13: AMS delimiters (available with amssymb package)

\Rightarrow	<code>\Rightarrow</code>	\rightsquigarrow	<code>\rightsquigarrow</code>	\Leftleftarrow	<code>\Leftleftarrow</code>
\Leftrightarrow	<code>\Leftrightarrow</code>	\Lleftarrow	<code>\Lleftarrow</code>	\twoheadleftarrow	<code>\twoheadleftarrow</code>
\leftarrowtail	<code>\leftarrowtail</code>	\looparrowleft	<code>\looparrowleft</code>	\leftrightharpoons	<code>\leftrightharpoons</code>
\curvearrowleft	<code>\curvearrowleft</code>	\circlearrowleft	<code>\circlearrowleft</code>	\Lsh	<code>\Lsh</code>
\Uparrow	<code>\Uparrow</code>	\upharpoonleft	<code>\upharpoonleft</code>	\downharpoonleft	<code>\downharpoonleft</code>
\multimap	<code>\multimap</code>	\leftrightsquigarrow	<code>\leftrightsquigarrow</code>	\rightleftarrows	<code>\rightleftarrows</code>
\Rightarrow	<code>\Rightarrow</code>	\twoheadrightarrow	<code>\twoheadrightarrow</code>	\rightarrowtail	<code>\rightarrowtail</code>
\looparrowright	<code>\looparrowright</code>	\rightleftharpoons	<code>\rightleftharpoons</code>	\curvearrowright	<code>\curvearrowright</code>
\circlearrowright	<code>\circlearrowright</code>	\Rsh	<code>\Rsh</code>	\downdownarrows	<code>\downdownarrows</code>
\downharpoonright	<code>\downharpoonright</code>	\upharpoonright	<code>\upharpoonright</code>	\restriction	<code>\restriction</code>

Table 8.14: AMS arrows (available with amssymb package)

\nleftarrow	<code>\nleftarrow</code>	\nrightarrow	<code>\nrightarrow</code>	\nLeftarrow	<code>\nLeftarrow</code>
\nRightarrow	<code>\nRightarrow</code>	\nleftrightarrow	<code>\nleftrightarrow</code>	\nLeftrightarrow	<code>\nLeftrightarrow</code>

Table 8.15: AMS negated arrows (available with amssymb package)

\leq	<code>\leqq</code>	\leq	<code>\leqslant</code>	\leq	<code>\eqslantless</code>
\lessapprox	<code>\lessapprox</code>	\approx	<code>\approx</code>	\approx	<code>\approx</code>
\lessdot	<code>\lessdot</code>	\lll	<code>\lll</code>	\lessgtr	<code>\lessgtr</code>
\lesseqgtr	<code>\lesseqgtr</code>	\doteqdot	<code>\doteqdot</code>	\doteq	<code>\doteq</code>
\risingdotseq	<code>\risingdotseq</code>	\fallingdotseq	<code>\fallingdotseq</code>	\backsimeq	<code>\backsimeq</code>
\backsim	<code>\backsim</code>	\subseteq	<code>\subseteq</code>	\subset	<code>\subset</code>
\sqsubset	<code>\sqsubset</code>	\preccurlyeq	<code>\preccurlyeq</code>	\curlyeqprec	<code>\curlyeqprec</code>
\prec	<code>\prec</code>	\preccurlyeq	<code>\preccurlyeq</code>	\vartriangleleft	<code>\vartriangleleft</code>
\trianglelefteq	<code>\trianglelefteq</code>	\dashv	<code>\dashv</code>	\vdash	<code>\vdash</code>
\smallsmile	<code>\smallsmile</code>	\smallfrown	<code>\smallfrown</code>	\bumpeq	<code>\bumpeq</code>
\Bumpeq	<code>\Bumpeq</code>	\geq	<code>\geq</code>	\geqslant	<code>\geqslant</code>
\eqslantgtr	<code>\eqslantgtr</code>	\gtrsim	<code>\gtrsim</code>	\gtrapprox	<code>\gtrapprox</code>
\gtrdot	<code>\gtrdot</code>	\ggg	<code>\ggg</code>	\gtrless	<code>\gtrless</code>
\gtreqless	<code>\gtreqless</code>	\gtreqless	<code>\gtreqless</code>	\eqcirc	<code>\eqcirc</code>
\circeq	<code>\circeq</code>	\triangle	<code>\triangle</code>	\thicksim	<code>\thicksim</code>
\thickapprox	<code>\thickapprox</code>	\supseteq	<code>\supseteq</code>	\supset	<code>\supset</code>
\sqsupseteq	<code>\sqsupseteq</code>	\succcurlyeq	<code>\succcurlyeq</code>	\curlyeqsucc	<code>\curlyeqsucc</code>
\succsim	<code>\succsim</code>	\succapprox	<code>\succapprox</code>	\vartriangleright	<code>\vartriangleright</code>
\trianglerighteq	<code>\trianglerighteq</code>	\vdash	<code>\vdash</code>	\shortmid	<code>\shortmid</code>
\shortparallel	<code>\shortparallel</code>	\between	<code>\between</code>	\pitchfork	<code>\pitchfork</code>
\varpropto	<code>\varpropto</code>	\blacktriangleleft	<code>\blacktriangleleft</code>	\therefore	<code>\therefore</code>
\backepsilon	<code>\backepsilon</code>	\blacktriangleright	<code>\blacktriangleright</code>	\because	<code>\because</code>

Table 8.16: AMS binary relations (available with amssymb package)

\nless	<code>\nless</code>	\nleq	<code>\nleq</code>	\nleqslant	<code>\nleqslant</code>
\nleqq	<code>\nleqq</code>	\nleq	<code>\nleq</code>	\nleqq	<code>\nleqq</code>
\nvertneqq	<code>\nvertneqq</code>	\nlsim	<code>\nlsim</code>	\nlnapprox	<code>\nlnapprox</code>
\nprec	<code>\nprec</code>	\npreceq	<code>\npreceq</code>	\nprecnsim	<code>\nprecnsim</code>
\nprecnapprox	<code>\nprecnapprox</code>	\nnsim	<code>\nnsim</code>	\nshortmid	<code>\nshortmid</code>
\nmid	<code>\nmid</code>	\nvDash	<code>\nvDash</code>	\nvDash	<code>\nvDash</code>
\ntriangleleft	<code>\ntriangleleft</code>	\ntrianglelefteq	<code>\ntrianglelefteq</code>	\nsubseteq	<code>\nsubseteq</code>
\nsubsetneq	<code>\nsubsetneq</code>	\nvarsubsetneq	<code>\nvarsubsetneq</code>	\nsubseteq	<code>\nsubseteq</code>
\nvarsubsetneqq	<code>\nvarsubsetneqq</code>	\ngeq	<code>\ngeq</code>	\ngeq	<code>\ngeq</code>
\ngeqslant	<code>\ngeqslant</code>	\ngeq	<code>\ngeq</code>	\gneq	<code>\gneq</code>
\gneq	<code>\gneq</code>	\nvertneqq	<code>\nvertneqq</code>	\ngsim	<code>\ngsim</code>
\gnapprox	<code>\gnapprox</code>	\nsucc	<code>\nsucc</code>	\nsucceq	<code>\nsucceq</code>
\succsim	<code>\succsim</code>	\succnapprox	<code>\succnapprox</code>	\ncong	<code>\ncong</code>
\nshortparallel	<code>\nshortparallel</code>	\nparallel	<code>\nparallel</code>	\nvDash	<code>\nvDash</code>
\nVDash	<code>\nVDash</code>	\ntriangleright	<code>\ntriangleright</code>	\ntrianglerighteq	<code>\ntrianglerighteq</code>
\nsubseteq	<code>\nsubseteq</code>	\nsubseteq	<code>\nsubseteq</code>	\nsubseteq	<code>\nsubseteq</code>
\nvarsubseteq	<code>\nvarsubseteq</code>	\nsubseteq	<code>\nsubseteq</code>	\nsubseteq	<code>\nsubseteq</code>

Table 8.17: AMS negated binary relations (available with amssymb package)

\dotplus	<code>\dotplus</code>	\smallsetminus	<code>\smallsetminus</code>	\Cap	<code>\Cap</code>
\Cup	<code>\Cup</code>	\barwedge	<code>\barwedge</code>	\veebar	<code>\veebar</code>
\doublebarwedge	<code>\doublebarwedge</code>	\boxminus	<code>\boxminus</code>	\boxtimes	<code>\boxtimes</code>
\boxdot	<code>\boxdot</code>	\boxplus	<code>\boxplus</code>	\divideontimes	<code>\divideontimes</code>
\times	<code>\times</code>	\rtimes	<code>\rtimes</code>	\leftthreetimes	<code>\leftthreetimes</code>
\rightthreetimes	<code>\rightthreetimes</code>	\curlywedge	<code>\curlywedge</code>	\curlyvee	<code>\curlyvee</code>
\circleddash	<code>\circleddash</code>	\circledast	<code>\circledast</code>	\circledcirc	<code>\circledcirc</code>
\centerdot	<code>\centerdot</code>	\intercal	<code>\intercal</code>		

Table 8.18: AMS binary operators (available with amssymb package)

\hbar	<code>\hbar</code>	\hslash	<code>\hslash</code>	\vartriangle	<code>\vartriangle</code>
\triangledown	<code>\triangledown</code>	\square	<code>\square</code>	\lozenge	<code>\lozenge</code>
\circledS	<code>\circledS</code>	\angle	<code>\angle</code>	\measuredangle	<code>\measuredangle</code>
\nexists	<code>\nexists</code>	\mho	<code>\mho</code>	\Finv	<code>\Finv</code>
\Game	<code>\Game</code>	\Bbbk	<code>\Bbbk</code>	\backprime	<code>\backprime</code>
\varnothing	<code>\varnothing</code>	\blacktriangle	<code>\blacktriangle</code>	\blacktriangledown	<code>\blacktriangledown</code>
\blacksquare	<code>\blacksquare</code>	\blacklozenge	<code>\blacklozenge</code>	\bigstar	<code>\bigstar</code>
\sphericalangle	<code>\sphericalangle</code>	\complement	<code>\complement</code>	\eth	<code>\eth</code>
\diagup	<code>\diagup</code>	\diagdown	<code>\diagdown</code>		

Table 8.19: AMS miscellaneous (available with amssymb package)

8.2.2 Names of Math Font Commands

The list of math font commands provided by the $\mathcal{A}\mathcal{M}\mathcal{S}$ packages is shown in table 8.20 on the next page, where for each case an example is shown. In addition, the math font commands of table 7.4 on page 183 can be used.

In the `amsmath` package, `\boldsymbol` is to be used for individual bold math symbols and bold Greek letters—everything in math except for letters (where one would use `\mathbf`). For example, to obtain a bold ∞ , or `\boldsymbol{\infty}`, `\boldsymbol{+}`, `\boldsymbol{\pi}`, or `\boldsymbol{0}`.

Since `\boldsymbol` takes a lot of typing, you can introduce new commands for bold symbols to be used frequently:

```

\newcommand{\bpi}{\boldsymbol{\pi}}
\newcommand{\binfty}{\boldsymbol{\infty}}
\[ B_{\infty} + \pi B_1 \sim B_{\infty} + \pi B_1
\mathbf{B}_{\infty} + \pi \mathbf{B}_1 \sim \mathbf{B}_{\infty} + \pi \mathbf{B}_1
\]

```

For those math symbols where the command `\boldsymbol` has no effect because the bold version of the symbol does not exist in the currently available fonts, there exists a command “Poor man’s bold” (`\pmb`), which simulates bold

<code>\mathbb</code>	Blackboard bold alphabet, e.g., <code>\mathbb{NQRZ}</code> gives: \mathbb{NQRZ} (not available in <code>amsmath</code> , need to load <code>amssymb</code>).
<code>\mathfrak</code>	Euler Fraktur alphabet, e.g., <code>\mathfrak{E}=\mathfrak{mc}^2</code> gives: $\mathfrak{E} = \mathfrak{mc}^2$ (not available in <code>amsmath</code> , need to load <code>amssymb</code>).
<code>\boldsymbol</code>	Used to obtain bold numbers and other nonalphabetic symbols, as well as bold Greek letters (defined in <code>amsbsy</code>).
<code>\pmb</code>	“Poor man’s bold,” used for math symbols when bold versions don’t exist in the available fonts, e.g., <code>\pmb{\oint}</code> gives: \oint and <code>\pmb{\triangle}</code> gives: \triangle (defined in <code>amsbsy</code>).
<code>\text</code>	Produce normal text with correct text-spacing in the current font used outside math, e.g., <code>\text{E=mc^2\quad\text{(Einstein)}}</code> gives: $E = mc^2$ (Einstein) (defined in <code>amstext</code>).

Table 8.20: Font commands available in mathematics with the \mathcal{AMS} packages

by typesetting several copies of the symbol with slight offsets. This procedure must be used for the extension and large operator symbols from the `cmex` font, as well as the \mathcal{AMS} extra math symbols from the `msam` and `msbm` fonts.

$$\frac{\partial w}{\partial u} \bigg| \frac{\partial u}{\partial v}$$

```
\[ \frac{\partial w}{\partial u} \bigg| \frac{\partial u}{\partial v}
```

With large operators and extension symbols (for example, \sum and \prod) `\pmb` does not currently work very well because the proper spacing and treatment of limits is not preserved. Therefore, the `\mathop` operator `\mathop` needs to be used (see table 7.13 on page 213).

$$\sum_{j < P} \prod_{\lambda} \lambda R(r_i) \quad \prod_{x_j} \lambda R(x_j)$$

```
\[ \sum_{j < P} \prod_{\lambda} \lambda R(r_i) \quad \prod_{x_j} \lambda R(x_j)
```

To make an entire math formula bold (or as much of it as possible, depending on the available fonts), use `\boldmath` preceding the formula.

The sequence `\mathbf{\hat{A}}` produces a bold accent character over the **A**. However, combinations like `\mathcal{\hat{A}}` will not work in ordinary \LaTeX because the `\mathcal` font does not have its own accents. In the `amsmath` package the font change commands are defined in such a way that accent characters will be taken from the `\mathrm` font if they are not available in the current font (in addition to the `\mathcal` font, the `\mathbb` and `\mathfrak` fonts don’t contain accents).

8.3 Compound Symbols, Delimiters, Operators

This section³ presents the math commands that are available through the `amsmath` package, which supplements \LaTeX in the area of compound symbols, large delimiters, etc. In the examples, `amsmath`’s alignment environments are used. In principle a detailed understanding of how they work is not necessary at this stage, but an interested reader can turn to section 8.5 for more information.

8.3.1 Multiple Integral Signs

`\iint`, `\iiint`, and `\iiiint` give multiple integral signs, with the spacing between them nicely adjusted, in both text and display style. `\idotsint` gives two integral signs with dots between them.

$$\iint_V \mu(u, v) du dv \quad (8.1)$$

```
\begin{gather} \iint\limits_V \mu(u, v) \, du \, dv \\ \iiint\limits_V \mu(u, v, w) \, du \, dv \, dw \\ \idotsint\limits_V \mu(t, u, v, w) \, dt \, du \, dv \, dw \\ \end{gather}
```

$$\iiint_V \mu(u, v, w) du dv dw \quad (8.2)$$

$$\iiint_V \mu(t, u, v, w) dt du dv dw \quad (8.3)$$

$$\int \cdots \int_V \mu(u_1, \dots, u_k) \quad (8.4)$$

8.3.2 Over and Under Arrows

Some extra over and under arrow operations are available. (In standard \LaTeX one has `\overrightarrow` and `\overleftarrow`.)

$$\overrightarrow{\psi_\delta(t) E_t h} = \psi_\delta(t) E_t h$$

```
\begin{align*} \overrightarrow{\psi_\delta(t) E_t h} &= \psi_\delta(t) E_t h \\ \overleftarrow{\psi_\delta(t) E_t h} &= \psi_\delta(t) E_t h \\ \overleftrightarrow{\psi_\delta(t) E_t h} &= \psi_\delta(t) E_t h \\ \overleftrightarrow{\psi_\delta(t) E_t h} &= \psi_\delta(t) E_t h \\ \end{align*}
```

These arrows all scale properly in subscript sizes, as seen in the following integral $\int_{uv} vt \, dt$, which was coded as `\int_{\overrightarrow{uv}} vt \, dt`.

³ Some material in this and the following sections is reprinted from the electronic document `testmath.tex` (distributed with $\mathcal{AMS}\text{-}\text{\LaTeX}$) with permission of the American Mathematical Society.

8.3.3 Dots

Ellipsis dots should almost always be typed as `\dots`. Positioning (on the baseline or centered) is automatically selected according to whatever follows the `\dots`. If the next character is a plus sign, the dots will be centered; if it's a comma, they will be on the baseline. These default dot placements provided by the `amsmath` package can be changed if different conventions are wanted.

If the dots fall at the end of a math formula, the next character will be something like `\end` or `\)` or `$`, which does not give any information about how to place the dots. If that is the case, you must help by using `\dotsc` for “dots with commas,” or `\dotscb` for “dots with binary operators/relations,” or `\dotscms` for “multiplication dots,” or `\dotsci` for “dots with integrals.” In the examples below, low dots are produced in the first instance and centered dots in the others, with the spacing on either side of the dots nicely adjusted.

A series H_1, H_2, \dots , a regional sum $H_1 + H_2 + \dots$, an orthogonal product $H_1 H_2 \dots$, and an infinite integral

A series $\sum_{H_1, H_2} \dots$,
a regional sum $\sum_{H_1 + H_2} \dots$, an
orthogonal product $\sum_{H_1 H_2} \dots$, and
an infinite integral
 $\int \int \dots$.

$$\int_{H_1} \int_{H_2} \dots$$

$$\backslash[\backslash\mathrm{int}\{H\ 1\}\backslash\mathrm{int}\{H\ 2\}\backslash\mathrm{dotsi}\backslash].$$

8.3.4 Accents in Math

The following accent commands automatically position double accents correctly:

\tilde{A}	\tilde{B}	\tilde{C}	\tilde{D}	<code>\begin{gather*}</code>
$\breve{\tilde{E}}$	$\breve{\tilde{F}}$	$\breve{\tilde{G}}$	$\breve{\tilde{H}}$	<code>\Acute{\Acute{A}} \quad \quad \Bar{\Bar{B}} \quad \quad</code>
				<code>\Breve{\Breve{C}} \quad \quad \Check{\Check{D}} \quad \quad</code>
				<code>\Ddot{\Ddot{E}} \quad \quad \quad \Dot{\Dot{F}} \quad \quad</code>
				<code>\Grave{\Grave{G}} \quad \quad \Hat{\Hat{H}} \quad \quad</code>
				<code>\Tilde{\Tilde{I}} \quad \quad \quad \Vec{\Vec{J}}</code>
				<code>\end{gather*}</code>

This double accent operation is complicated and tends to slow down the processing of a L^AT_EX file. If the document contains many double accents, you can load the `amstrax` package. It defines the `\accentsymbol` command, which you can use in the preamble of your document to help speed things up. It stores the result of the double accent command in a box register for quick retrieval. `\accentsymbol` is used like `\newcommand`:

This is a double hat $\hat{\hat{A}}$ and this $\ddot{\delta}$ a delta with a bar and a dot.

$\hat{\hat{A}}$
 $\dot{\bar{\Delta}}$
 This is a double hat $\hat{\hat{A}}$ and this
 $\dot{\bar{\Delta}}$ a delta with a bar and a dot.

8.3.5 Superscripted Accents

Some accents have a wide form: typing `\widehat{xy}`, `\widetilde{xy}` produces \widehat{xy} , \widetilde{xy} . Because these wide accents have a certain maximum size, the `amsxtra` package introduces a different notation to handle extremely long expressions: \widehat{AmBD} instead of $\widehat{Am}BD$. `amsxtra` has the following control sequences to achieve this easily:

$(AmBD)^{\wedge}$	$(AmBD)^{\vee}$	(8.5)	<code>\begin{gather}</code>	<code>(AmED)\spat</code>	<code>\qqquad (AmED)\spcheck</code>	<code>\</code>
$(AmBD)^{\sim}$	$(AmBD)^{\cdot}$	(8.6)	<code>(AmED)\spilde</code>	<code>\qqquad (AmED)\spdot</code>	<code>\</code>	
$(AmBD)^{\circ}$	$(AmBD)^{\circ\circ}$	(8.7)	<code>(AmED)\spddot</code>	<code>\qqquad (AmED)\spdddot</code>	<code>\</code>	
$(AmBD)^{\circ}$		(8.8)	<code>(AmED)\spbreve</code>			
			<code>\end{gather}</code>			

8.3.6 Dot Accents

`\dddot` and `\ddddot` are available to produce tripled and quadrupled dot accents in addition to the `\dot` and `\ddot` accents already available in L^AT_EX:

$$\overset{\cdots}{Q} \qquad \overset{\cdots}{R} \qquad \$ \quad \backslash \text{dddot}\{Q\} \quad \backslash \text{qqquad} \quad \backslash \text{ddddot}\{R\} \quad \$$$

8.3.7 Roots

In ordinary L^AT_EX the placement of root indices is sometimes not good. With `amsmath` the commands `\leftroot` and `\uproot` allow the adjustment of the position of the root. Positive arguments to these commands will move the root index to the left and up respectively, while a negative argument will move them right and down. The units of increment are quite small, which is useful for such adjustments. In the example below, the root index β is moved 2 units to the left and 4 units up.

$$\sqrt[k]{\beta} \quad \sqrt[k]{\beta}$$

8.3.8 Boxed Formulae

The command `\boxed` puts a box around its argument, similar to `\fbox`, except that the contents are in math mode:

$$\boxed{W_t - F \subseteq V(P_i) \subseteq W_t} \quad \backslash [\boxed{W_t - F \subseteq V(P_i) \subseteq W_t}]$$

8.3.9 Extensible Arrows

`\xleftarrow` and `\xrightarrow` produce arrows that extend automatically to accommodate unusually wide subscripts or superscripts. The text of the subscript or superscript are given as an optional and mandatory argument, respectively:

$$0 \stackrel{\alpha}{\leftarrow} F \times \Delta[n-1] \stackrel{\partial_0 \alpha(b)}{\rightarrow} E^{\partial_0 b}$$

```
\[0 \xleftarrow[\alpha]{\partial_0 \alpha(b)} E^{\partial_0 b}
\]
```

8.3.10 `\overset`, `\underset`, and `\sideset`

L^AT_EX provides `\stackrel` for placing a superscript above a binary relation. `amsmath` introduces somewhat more general commands, `\overset` and `\underset`. These can be used to place one symbol above or below another symbol, independently of whether it is a relation or something else. The input `\overset{*}{X}` will place a superscript-size `*` above the `X`; `\underset` performs the parallel operation that one would expect.

$$\overset{*}{X} \quad \underset{*}{X} \quad \overset{a}{\underset{b}{X}}$$

```
\[ \overset{*}{X} \quad \underset{*}{X} \quad \overset{a}{\underset{b}{X}}
\]
```

There is also a command called `\sideset` that serves a rather special purpose: it puts symbols in the subscript and superscript positions of large operator symbols such as \sum and \prod . A prime example is the case when you want to put a prime on a sum symbol. If there are no limits above or below the sum, you could just use `\nolimits`:

$$\sum' E_n. \quad (8.9)$$

```
\begin{equation}
\sum\nolimits' E_n.
\end{equation}
```

But if you want not only the prime but also limits on the sum symbol, things are not so easy. Suppose you want to add a prime on the sum symbol in an expression, like

$$\sum_{n < k, n \text{ odd}} n E_n \quad (8.10)$$

```
\begin{equation}
\sum_{n < k, \backslash n \mathrm{odd}} n E_n
\end{equation}
```

then you can use `\sideset` like this: `\sideset{}{'}\sum_{\dots} n E_n`. The extra pair of empty braces is explained by the fact that `\sideset` has the capability of putting an extra symbol or symbols at each corner of a large operator.

$$\prod_{i=1}^4 \sum_{0 \leq i \leq m}' E_i \beta x$$

```
\[
\sideset{}{'}\prod_k
\sideset{}{'}\sum_{0 \leq i \leq m} E_i \beta x
\]
```

8.3.11 The `\smash` Command

The plain T_EX command `\smash` retains the contents of a box but annihilates its height and depth. The `amsmath` package introduces the optional arguments `t` and `b` with the `\smash` command. `\smash[t]{...}` annihilates only the top of the box contents, retaining the bottom part, while `\smash[b]{...}` annihilates the bottom part and keeps the top.

$$X_j = (1/\sqrt{\lambda_j})X'_j \quad X_j = (1/\sqrt{\lambda_j})X'_j$$

```
\[ X_j=(1/\sqrt{\smash[b]{\lambda_j}})X_j'
\quad
X_j=(1/\sqrt{\lambda_j})X_j'
\]
```

The previous example shows how the `\smash` command was used to limit the depth of the radical, which otherwise extends to encompass the depth of the subscript (right-hand formula in the above example).

8.3.12 The `\text` Command

The main use of the `\text` command, which is also available separately with the `amstext` package, is for words or phrases in a display. It is similar to the L^AT_EX command `\mbox` in its effects, but has a couple of advantages. If you would like a word or phrase of text in a subscript, you can type

`..._{\text{word or phrase}}`

which, apart from having a more descriptive name, is also slightly easier to enter than the equivalent `\mbox`, since the correct size is automatically chosen:

`..._{\mbox{\scriptsize word or phrase}}`

$$y = y' \quad \text{if and only if} \quad y'_k = \delta_k y_{\tau(k)}$$

```
\[ \mathbf{y}=\mathbf{y}' \quad \text{if and only if} \quad
y'_k=\delta_k y_{\tau(k)}
\]
```

8.3.13 Operator Names

Math functions such as \log , \sin , and \lim are traditionally set in roman type to help avoid confusion with single math variables, set in math italic. The more common ones have predefined names: `\log`, `\sin`, `\lim`, and so forth (see table 8.8 on page 226). New ones, however, come up all the time in mathematical papers. The `amsmath` package provides `\DeclareMathOperator` and `\DeclareMathOperator*` for producing new function names that will have the same typographical treatment. For instance, `\DeclareMathOperator{\xxx}{xxx}` produces \textit{xxx} in the proper font and automatically adds proper spacing on either side when necessary, so that you get $A \textit{xxx} B$ instead of $Axxx B$. Examples of definitions of operator names are shown below (the `\`, in the definition of `\esssup` adds some space; see table 8.21 on page 252):

Input text	
<code>\DeclareMathOperator*{\esssup}{ess\,sup}</code>	
<code>\DeclareMathOperator{\meas}{meas}</code>	
<code>\newcommand{\abs}[1]{\lvert#1\rvert}</code>	
<code>\newcommand{\norm}[1]{\lVert#1\rVert}</code>	
<code>\begin{align*}</code>	
<code>\norm{f}_{\infty}</code>	$\ f\ _\infty$
<code>\esssup_{x \in R^n} \abs{f(x)}</code>	$\operatorname{ess\,sup}_{x \in R^n} f(x) $
<code>\meas_1\{u \in R_+^1 : f^*(u) > \alpha\}</code>	$\operatorname{meas}_1\{u \in R_+^1 : f^*(u) > \alpha\}$
<code>\meas_n\{x \in R^n : \abs{f(x)} \geq \alpha\}</code>	$\operatorname{meas}_n\{x \in R^n : f(x) \geq \alpha\}$
<code>\end{align*}</code>	

$$\|f\|_\infty = \operatorname{ess\,sup}_{x \in R^n} |f(x)|$$

$$\operatorname{meas}_1\{u \in R_+^1 : f^*(u) > \alpha\} = \operatorname{meas}_n\{x \in R^n : |f(x)| \geq \alpha\} \quad \forall \alpha > 0.$$

Output text

The starred form `\DeclareMathOperator*` is like `\DeclareMathOperator`; the only difference is the placement of subscripts and superscripts, as seen in the example above. In order to make the use of the vertical bar notation more flexible, `amsmath` defines the new commands `\lvert`, `\rvert`, `\lVert`, and `\rVert`, which are comparable to L^AT_EX's `\langle` and `\rangle`.

With `amsmath` the following operators are predefined: `\varlimsup`, `\varliminf`, `\varinjlim`, and `\varprojlim`. Here's what they look like in use:

$$\begin{aligned} \overline{\lim}_{n \rightarrow \infty} Q(u_n, u_n - u^\#) &\leq 0 & (8.11) \\ \underline{\lim}_{n \rightarrow \infty} |a_{n+1}| / |a_n| &= 0 & (8.12) \\ \varliminf (m_i^\lambda)^* &\leq 0 & (8.13) \\ \varinjlim_{p \in S(A)} A_p &\leq 0 & (8.14) \end{aligned}$$

8.3.14 \mod and Its Relatives

Commands `\mod`, `\bmod`, `\pmod`, and `\pod` are provided by the `amsopn` package to deal with the rather special spacing conventions of “mod” notation. `\bmod` and `\pmod` are available in L^AT_EX, but with `amsopn` the spacing of `\pmod` will adjust to a smaller value if it is used in a nondisplay formula. `\mod` and `\pod` are variants of `\pmod` preferred by some authors; `\mod` omits the parentheses, whereas `\pod` omits the “mod” and retains the parentheses.

$$\begin{aligned} \gcd(k, l \bmod k) & \quad (8.15) \\ u \equiv v + 1 \pmod{n^2} & \quad (8.16) \\ u \equiv v + 1 \bmod n^2 & \quad (8.17) \\ u \equiv v + 1 \pod{n^2} & \quad (8.18) \end{aligned}$$

8.3.15 Fractions and Related Constructions

In addition to `\frac` (available in L^AT_EX), `amsmath` provides `\dfrac` and `\tfrac` as convenient abbreviations for `\displaystyle\frac ...` and `\textstyle\frac ...`.

$$\begin{aligned} \frac{1}{k} \log_2 c(f) & \quad \frac{1}{k} \log_2 c(f) \\ \text{and } \sqrt{\frac{1}{k} \log_2 c(f)} & \quad \sqrt{\frac{1}{k} \log_2 c(f)} \end{aligned}$$

For binomial expressions such as $\binom{n}{k}$ the `amsmath` packages defines the commands `\binom`, `\dbinom`, and `\tbinom`.

$$\binom{k}{1}2^{k-1} + \binom{k}{2}2^{k-2} \quad (8.19)$$

and $\binom{k}{1}2^{k-1} + \binom{k}{2}2^{k-2}$.

`\binom`, and its variants `\dbinom` and `\tbinom`, as well as `\frac` and its variants `\dfrac` and `\tfrac` are implemented using the generalized fraction command `\genfrac`, which has six parameters.

`\genfrac{ldelim}{rdelim}{thick}{style}{num}{denom}`

The first two parameters `ldelim` and `rdelim` are the left and right delimiters, respectively. The third parameter `thick` allows you to override the line thickness (for instance `\binom` uses this to set the line thickness to zero, i.e., invisible). If this argument is left empty, the line thickness defaults to “normal”. The fourth parameter is the mathematics style override. It can take integer values in the range 0–3 to select, respectively, `\displaystyle`, `\textstyle`, `\scriptstyle`, and `\scriptscriptstyle`. Finally, the fifth argument `num` is the numerator, while the sixth `denom` is the denominator of the fraction.

To illustrate, here is how `\frac`, `\tfrac`, and `\binom` might be defined.

```
\newcommand{\frac}[2]{\genfrac{}{}{}{}{#1}{#2}}
\newcommand{\tfrac}[2]{\genfrac{}{}{}{}{#1}{#2}}
\newcommand{\binom}[2]{\genfrac{}{}{}{}{#1}{#2}}
```

Other examples are the following re-implementation of TeX’s fraction primitives.

$$\frac{n+1}{n} \quad \left\langle \frac{n+1}{n} \right\rangle$$

$$n+2 \quad \left(\frac{n+2}{n} \right)$$

$$\frac{n-3}{n} \quad \left[\frac{n-3}{n} \right]$$

```
\renewcommand{\over}[2]{%
  \genfrac{}{}{}{}{#1}{#2}}
\renewcommand{\overwithdelims}[2]{%
  \genfrac{\langle\rangle}{\langle\rangle}{}{}{#1}{#2}}
\[\over{n+1}{n}\qquad\overwithdelims{n+1}{n}\]
\renewcommand{\atop}[2]{%
  \genfrac{}{}{}{}{#1}{#2}}
\renewcommand{\atopwithdelims}[2]{%
  \genfrac{}{}{}{}{#1}{#2}}
\[\atop{n+2}{n}\qquad\atopwithdelims{n+2}{n}\]
\renewcommand{\above}[2]{%
  \genfrac{}{}{}{}{#1}{#2}}
\renewcommand{\abovewithdelims}[2]{%
  \genfrac{}{}{}{}{#1}{#2}}
\[\above{n-3}{n}\qquad\abovewithdelims{n-3}{n}\]
```

Of course, if you want to use a particular notation implemented with `\genfrac` repeatedly throughout your document you will do yourself (and your publisher) a favor if you define a meaningful command name with `\newcommand` as an abbreviation for that notation, as in the examples above.

8.3.16 Continued Fractions

A continued fraction can be obtained as follows:

$$\frac{1}{\sqrt{2} + \frac{1}{\sqrt{3} + \frac{1}{\sqrt{4} + \frac{1}{\sqrt{5} + \frac{1}{\sqrt{6} + \dots}}}}} \quad (8.20)$$

```
\begin{equation}
\cfrac{1}{\sqrt{2}+}
\cfrac{1}{\sqrt{3}+}
\cfrac{1}{\sqrt{4}+}
\cfrac{1}{\sqrt{5}+}
\cfrac{1}{\sqrt{6}+\dotsb}
\end{equation}
```

Left or right positioning of any of the numerators is achieved by using the optional argument `[l]` or `[r]` with the `\cfrac` command.

8.3.17 Big-g-g Delimiters

In order to better control the sizes of math delimiters, basic TeX introduces four commands `\big`, `\Big`, `\bigg` and `\Bigg`, which produce ever larger versions of the delimiter specified as parameter. These commands can be used with any of the delimiters that can follow the `\left` or `\right` command (see tables 8.9, 8.10, and 8.13 on page 227). Moreover, for each of the four commands above, three variants exist for use as an opening symbol (e.g., `\bigl`), as a binary relation (e.g., `\Bigm`), or as a closing symbol (e.g., `\Biggr`).⁴ Whereas, with basic TeX, the sizes of these delimiters are fixed, with the `amsmath` package the sizes adapt to the size of the surrounding material.

$$\left(\mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) ds \right)$$

$$\left(\mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y^x(s)} \varphi(x) ds \right)$$

```
\[
\biggl(\mathbf{E}_y\int_0^{t_\varepsilon}L_{x,y^x(s)}\varphi(x)\,ds\biggr)
\]
\[\Large
\biggl(\mathbf{E}_y\int_0^{t_\varepsilon}L_{x,y^x(s)}\varphi(x)\,ds\biggr)
\]
```

⁴ See table 7.13 on page 213 for a discussion of the various math symbol types.

$$\begin{array}{ccccc}
 \text{cov}(\mathcal{L}) & \longrightarrow & \text{non}(\mathcal{K}) & \longrightarrow & \text{cf}(\mathcal{K}) \\
 \downarrow & & \uparrow & & \uparrow \\
 \text{add}(\mathcal{L}) & \longrightarrow & \text{add}(\mathcal{K}) & \longrightarrow & \text{cov}(\mathcal{K})
 \end{array}$$

```

\begin{equation*}
\DeclareMathOperator{\add}{add}
\DeclareMathOperator{\cf}{cf}
\DeclareMathOperator{\cov}{cov}
\DeclareMathOperator{\non}{non}
\begin{CD}
\cov{\mathcal{L}} @>>> \non{\mathcal{K}} \\
@V \add VV @. \\
\add{\mathcal{L}} @>>> \add{\mathcal{K}} @>>> \cov{\mathcal{K}}
\end{CD}
\end{equation*}

```

8.5 Alignment Structures for Equations

The `amsmath` package defines several environments for creating multiline display equations. They perform similarly to L^AT_EX's `equation` and `eqnarray` environments. The following structures are discussed in the next sections.

<code>align</code>	<code>align*</code>	alignment at a single place
<code>flalign</code>	<code>flalign*</code>	spaced-out variants of the above
<code>alignat</code>	<code>alignat*</code>	alignment with space control
<code>equation</code>	<code>equation*</code>	one-line formula
<code>gather</code>	<code>gather*</code>	combining formula without alignment
<code>multline</code>	<code>multline*</code>	multiline equation (one equation number)
<code>split</code>		splitting long formulas

Some of these multiline display environments allow you to align parts of the formula. In contrast to the original L^AT_EX environments `eqnarray` and `eqnarray*`, the structures implemented by the `amsmath` package use a different concept for marking the alignment points: while `eqnarray` is similar to an `array` environment with a `{rcl}` preamble and therefore uses two ampersand characters surrounding the part that should be aligned, in the `amsmath` structures you should mark the alignment point (or points in `alignat`, for example) only with a single ampersand character, placing it to the left of the character that should be aligned with previous or following lines.

The `amsmath` structures give correct spacing around the alignment points, while the `eqnarray` environment produces extra spaces depending on the parameter settings for `array`. The difference can be seen clearly in the next example, where we typeset the same equation using the `equation`, `align`, and `eqnarray` environments; ideally all three should produce the same result, but the `eqnarray` environment comes out too wide.

$$\begin{array}{ll}
 x^2 + y^2 = z^2 & (8.24) \\
 x^2 + y^2 = z^2 & (8.25) \\
 x^3 + y^3 < z^3 & (8.26) \\
 x^2 + y^2 = z^2 & (8.27) \\
 x^3 + y^3 < z^3 & (8.28)
 \end{array}$$

```

\begin{equation}
x^2+y^2 = z^2
\end{equation}
\begin{align}
x^2+y^2 &= z^2 \quad \backslash \quad x^3+y^3 < z^3
\end{align}
\begin{eqnarray}
x^2+y^2 &=& z^2 \quad \backslash \quad x^3+y^3 < z^3
\end{eqnarray}

```

8.5.1 Equation Groups without alignment

The `gather` environment is used for two or more equations, when no alignment is desired among them. Each one is centered separately between the left and right margins.

$$\begin{array}{ll}
 (a+b)^2 = a^2 + 2ab + b^2 & (8.29) \\
 (a+b) \cdot (a-b) = a^2 - b^2 & (8.30)
 \end{array}$$

```

\begin{gather}
(a+b)^2 = a^2 + 2ab + b^2 \\
(a+b) \cdot (a-b) = a^2 - b^2
\end{gather}

```

More examples are shown in section 8.7.3 on page 259.

8.5.2 Equation Groups with alignment

The `align` environment is used for two or more equations when vertical alignment is desired (usually binary relations such as equal signs are aligned). The term “equation” is used rather loosely here to mean any math formula that is intended by an author to be a self-contained subdivision of the larger display, usually, but not always, containing a binary relation.

$$\begin{array}{ll}
 x^2 + y^2 = 1 & x^3 + y^3 = 1 \quad (8.31) \\
 x = \sqrt{1-y^2} & x = \sqrt[3]{1-y^3} \quad (8.32)
 \end{array}$$

```

\begin{align}
x^2 + y^2 &= 1 \quad \& \quad x^3 + y^3 = 1 \\
x &= \sqrt{1-y^2} \quad \& \quad x = \sqrt[3]{1-y^3}
\end{align}

```

More examples are shown in section 8.7.4 on page 259.

With the `align` environment the material is spread out uniformly over the lines. If you want to control the space between equation columns then you can use an `alignat` environment. It has one required argument, for specifying the

number of “align” structures. For an argument of n , the number of ampersand characters per line is $2n - 1$ (one ampersand for alignment within each align structure, and ampersands to separate the align structures from one another).

The special environment `flalign` is a form of the `align` environment with added space between the component align structures.

$L_1 = R_1$	$L_2 = R_2$	(8.33)	<code>\begin{align}</code>
			<code>L_1 &= R_1 & \quad L_2 &= R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$	(8.34)	<code>L_3 &= R_3 & \quad L_4 &= R_4</code>
			<code>\end{align}</code>

$L_1 = R_1$	$L_2 = R_2$	(8.35)	<code>\begin{alignat}{2}</code>
			<code>L_1 &= R_1 & \quad L_2 &= R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$	(8.36)	<code>L_3 &= R_3 & \quad L_4 &= R_4</code>
			<code>\end{alignat}</code>

$L_1 = R_1$	$L_2 = R_2$	(8.37)	<code>\begin{flalign}</code>
			<code>L_1 &= R_1 & \quad L_2 &= R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$	(8.38)	<code>L_3 &= R_3 & \quad L_4 &= R_4</code>
			<code>\end{flalign}</code>

$L_1 = R_1$	$L_2 = R_2$		<code>\begin{flalign*}</code>
			<code>L_1 &= R_1 & \quad L_2 &= R_2 \quad \backslash\backslash</code>
$L_3 = R_3$	$L_4 = R_4$		<code>L_3 &= R_3 & \quad L_4 &= R_4</code>
			<code>\end{flalign*}</code>

More examples are shown in section 8.7.6 on page 261.

8.5.3 Split Equations without Alignment

The `multline` environment is a variation of the `equation` environment used for equations that do not fit on a single line. The first line of a `multline` will be at the left margin and the last line at the right margin except for an indentation on both sides whose amount is equal to `\multlinegap`. The value of `\multlinegap` can be changed using L^AT_EX's `\setlength` and `\addtolength` commands. If `multline` contains more than two lines, any lines other than the first and last will be centered individually within the display width (unless option `fleqn` is in effect). It is, however, possible to force a line to the left or the right with the `\shoveleft` and `\shoveright` commands.

First line of equation	<code>\begin{multline}</code>	
	<code>\text{First line of equation}</code>	<code>\backslash\backslash</code>
Centered Middle line	<code>\text{Centered Middle line}</code>	<code>\backslash\backslash</code>
	<code>\shoveright{\text{Right Middle line}}</code>	<code>\backslash\backslash</code>
Right Middle line	<code>\text{Other centered Middle}</code>	<code>\backslash\backslash</code>
	<code>\shoveleft{\text{Left Middle line}}</code>	<code>\backslash\backslash</code>
Other centered Middle	<code>\text{Last line of equation}</code>	
Left Middle line	<code>\end{multline}</code>	
Last line of equation	(8.39)	

More examples are shown in section 8.7.2 on page 258.

8.5.4 Split Equations with Alignment

Like `multline`, the `split` environment is for single equations that are too long to fit on a single line and hence must be split into multiple lines. Unlike `multline`, however, the `split` environment provides for alignment among the split lines, using an ampersand to mark alignment points, as usual. In addition (unlike the other `amsmath` equation structures) the `split` environment provides no numbering because it is intended to be used only inside some other displayed equation structure, such as `equation`, `align`, or `gather`. These outer environments will provide the numbering.

$(a + b)^4 = (a + b)^2(a + b)^2$	<code>\begin{equation}</code>
	<code>\begin{split}</code>
$= (a^2 + 2ab + b^2)(a^2 + 2ab + b^2)$	<code>(a+b)^4 &= (a+b)^2 (a+b)^2 \quad \backslash\backslash</code>
$= a^4 + 4a^3b + 6a^2b^2 + 4ab^3 + b^4$	<code>&= (a^2+2ab+b^2)(a^2+2ab+b^2) \quad \backslash\backslash</code>
	<code>&= a^4+4a^3b+6a^2b^2+4ab^3+b^4 \quad \backslash\backslash</code>
	<code>\end{split}</code>
	<code>\end{equation}</code>

When the `tbtags` option is specified, the equation number for the `split` environment will be put on the last (resp. first) line if the equation number is on the right (resp. left). By default, the `centertags` option is in effect, putting the equation number centered vertically on the height of the `split`, provided there is enough room for it.

$(a + b)^3 = (a + b)(a + b)^2$	<code>\begin{equation}</code>
	<code>\begin{split}</code>
$= (a + b)(a^2 + 2ab + b^2)$	<code>(a+b)^3 &= (a+b) (a+b)^2 \quad \backslash\backslash</code>
$= a^3 + 3a^2b + 3ab^2 + b^3$	<code>&= (a+b)(a^2+2ab+b^2) \quad \backslash\backslash</code>
	<code>&= a^3+3a^2b+3ab^2+b^3 \quad \backslash\backslash</code>
	<code>\end{split}</code>
	<code>\end{equation}</code>

More examples are shown in section 8.7.1 on page 255.

8.5.5 Alignment Environments as Parts of Displays

In addition to the `split` environment, there are some other equation alignment environments that do not constitute an entire display. They are self-contained units that can be used inside other formulae, or set side by side. The environment names are: `aligned`, `gathered`, and `alignedat`. These environments take an optional argument to specify their vertical positioning with respect to the material on either side. The default alignment is centered (`[c]`), and its effect is seen in the following example.

```


$$\begin{aligned}
 x^2 + y^2 &= 1 & (a+b)^2 &= a^2 + 2ab + b^2 \\
 x &= \sqrt{1-y^2} & (a+b) \cdot (a-b) &= a^2 - b^2
 \end{aligned}$$


```

The same mathematics can now be typeset using different vertical alignments for the environments.

```


$$\begin{aligned}
 x^2 + y^2 &= 1 & (a+b)^2 &= a^2 + 2ab + b^2 \\
 x &= \sqrt{1-y^2} & (a+b) \cdot (a-b) &= a^2 - b^2
 \end{aligned}$$


```

8.5.6 Vertical Spacing and Page Breaks in Equation Structures

You can use the `\[dimension]` command to get extra vertical space between lines in all the `amsmath` displayed equation environments, as is usual in `LATEX`. Unlike `eqnarray`, the `amsmath` environments do not allow page breaks between lines, unless `\displaybreak` or `\allowdisplaybreaks` is used. The reason

for this is that page breaks in such situations should receive individual attention from the author. `\displaybreak` must go before the `\` where it is supposed to take effect. Like `LATEX`'s `\pagebreak`, `\displaybreak` takes an optional argument between zero and four denoting the desirability of the page break. `\displaybreak[0]` means “it is permissible to break here” without encouraging a break; `\displaybreak` with no optional argument is the same as `\displaybreak[4]` and forces a break.

There is also an optional argument for `\allowdisplaybreaks`. This command obeys the usual `LATEX` scoping rules. The normal way of limiting its scope is to put `\allowdisplaybreaks` at the beginning and `\` at the end of the desired range. Within the scope of an `\allowdisplaybreaks` command, the `\`* command can be used to prohibit a page break, as usual.

8.5.7 The `\intertext` Command

The `\intertext` command is used for a short interjection of one or two lines of text in the middle of a display alignment. Its salient feature is the preservation of alignment, which would not be possible if you simply ended the display and then started it up again afterwards. `\intertext` may only appear immediately after a `\` or `\`* command.

```


$$\begin{aligned}
 A_1 &= N_0(\lambda; \Omega') - \phi(\lambda; \Omega'), & (8.42) \\
 A_2 &= \phi(\lambda; \Omega')\phi(\lambda; \Omega), & (8.43)
 \end{aligned}$$


```

and finally

```


$$A_3 = \mathcal{N}(\lambda; \omega). \quad (8.44)$$


```

Here the words “and finally” fall outside the display at the left margin.

8.6 Miscellaneous

This section discusses `amsmath` commands that have not been introduced yet, and it gives a list of the document class files that come with the `AMS-LATEX` distribution.

8.6.1 Equation Numbers

Each environment, except for `split`, has both starred and unstarred forms, where the unstarred forms have automatic numbering, using `LATEX`'s `equation`

counter. The number on any particular line can be suppressed by putting `\notag` before the `\`. You can also override it with a tag of your own design using

```
\tag{label} \tag*{label}
```

where *label* can be any arbitrary text to be used to number the equation.

The starred form, `\tag*`, causes the *label* to be typeset without any annotations like parentheses that might otherwise be added by the document class. `\tag` and `\tag*` can also be used in the starred versions of all the `amsmath` alignment environments.

$x^2 + y^2 = z^2$	(8.45)	<code>\begin{gather}</code>	<code>x^2+y^2 = z^2 \label{eq:r2}</code>	<code>\</code>
$x^3 + y^3 = z^3$			<code>x^3+y^3 = z^3 \notag</code>	<code>\</code>
$x^4 + y^4 = r^4$	(*)		<code>x^4+y^4 = r^4 \tag{*\$}</code>	<code>\</code>
$x^5 + y^5 = r^5$	*		<code>x^5+y^5 = r^5 \tag*{***}</code>	<code>\</code>
$x^6 + y^6 = r^6$	(8.45')	<code>\end{gather}</code>	<code>x^6+y^6 = r^6 \tag{\ref{eq:r2}\$'}</code>	<code>\</code>

Notice the use of the `\label` and `\ref` commands in the previous example to allow subnumbering of equations.

When `leqno` is specified as an option to the `amsmath` package, the equation number will be printed at the left side of the equation (by default, with `amsmath`, it comes out at the right).

(8.46)	$\sin^2 \eta + \cos^2 \eta = 1$	<code>\begin{equation}</code>	<code>\sin^2\eta + \cos^2\eta = 1</code>	<code>\end{equation}</code>
--------	---------------------------------	-------------------------------	------------------------------------------	-----------------------------

8.6.2 Resetting the Equation Counter

In L^AT_EX, if you want to have equations numbered within sections—that is, have equation numbers (1.1), (1.2), . . . , (2.1), (2.2), . . . , in sections 1, 2, and so forth—you would probably redefine `\theequation`:

```
\renewcommand{\theequation}{\thesection.\arabic{equation}}
```

But now you have to reset the equation number by hand at the beginning of each new section or chapter. To make this a little more convenient, `amsmath` provides a command `\numberwithin`. To have equation numbering tied to section numbering, with automatic reset of the equation counter, the command is

```
\numberwithin{equation}{section}
```

As the name implies, `\numberwithin` can be applied to other counters besides the equation counter, but the results may not be satisfactory in all cases because of potential complications. Normal L^AT_EX methods should be used where available, for example, in `\newtheorem`.⁶

To make cross-references to equations easier, an `\eqref` command is provided. This automatically supplies the parentheses around the equation number, and adds an italic correction before the closing parenthesis, if necessary. To refer to an equation that was labeled with the label `e:baset`, the usage would be `\eqref{e:baset}`.

8.6.3 Subordinate numbering sequences

The `amsmath` package provides also a `subequations` environment to make it easy to number equations in a particular group with a subordinate numbering scheme. For example

```
\begin{subequations}
...
\end{subequations}
```

causes all numbered equations within that part of the document to be numbered (4.9a) (4.9b) (4.9c) . . . , if the preceding numbered equation was (4.8). A `\label` command immediately following `\begin{subequations}` produces a `\ref` of the parent number 4.9, not 4.9a. The counters used by the `subequations` environment are `parentequation` and `equation`. They can be set by the L^AT_EX commands `\addtocounter`, `\setcounter`, `\value`, etc.. Moreover, the style of the subordinate numbers, are controlled using standard L^AT_EX methods (see Section A.1.3). For example, redefining `\theequation` as follows will produce roman numerals.

```
\begin{subequations}
\renewcommand{\theequation}{\theparentequation \roman{equation}}
...
```

8.6.4 Fine-Tuning Spacing in Math Mode

Although T_EX generally does a good job of spacing elements of formulae inside mathematics, it is sometimes necessary to fine-tune the position of one or two of those elements. Therefore, the spacing commands shown in table 8.21 on the next page are provided. Both the spelled-out and abbreviated forms of these commands are robust, and they can also be used outside of math.

⁶ See also the discussion of the `\@addtoreset` command on page 23.

Positive space			Negative space		
Abb.	ex.	Spelled out	Abb.	ex.	Spelled out
<code>\,</code>	xx	<code>\thinspace</code>	<code>\!</code>	xx	<code>\negthinspace</code>
<code>\:</code>	xx	<code>\medspace</code>		xx	<code>\negmedspace</code>
<code>\;</code>	xx	<code>\thickspace</code>		xx	<code>\negthickspace</code>
	$x\ x$	<code>\quad</code>			
	$x\ \ x$	<code>\qquad</code>			

Table 8.21: The mathematical spacing commands

For allow you to further fine-tune the spacing in math expressions the command `\mspace` is defined. Its only argument is a \LaTeX length expressed in ‘math units’. One math unit, or `\mu`, is equal to 1/18 em (see also table A.1 on page 476). Thus, to get a negative `\quad` you could write `\mspace{-18.0mu}`.

8.6.5 A Few Points to Note

(\mathcal{L} 151–52)

Many of the commands added by the `amsmath` package are fragile and will need to be `\protected` in commands with “moving arguments.”

With the various alignment environments available in the `amsmath` package, the `eqnarray` environment is no longer needed. Furthermore, since it does not prevent overlapping of the equation numbers with wide formulae, as most of the `amsmath` alignments do, using the `amsmath` alignments seems better. `amsmath` reimplements the \LaTeX `equation` environment as a one-line `gather` environment, and adds an unnumbered version, `equation*`, for symmetry. Note, however, that the command `\verb` might not work in the alignment environments.

`\nonumber` is interchangeable with `\notag`; the latter seems slightly preferable, for consistency with the name `\tag`.

8.6.6 Options and Sub-Packages to the `amsmath` Package

A few options are recognized by the `amsmath` package and the classes provided by $\mathcal{AMS}\text{-}\text{\LaTeX}$.⁷ They affect the positioning of math operator limits or `\tags`.

`centertags` (default) The text of the tag of a `split` environment is vertically centered with respect to its total height.

`tbtags` “Top-or-bottom tags”. The text of the tag of a `split` environment is placed level with the last (resp. first) line, if numbers are on the right (resp. left).

⁷ This is only true for the $\text{\LaTeX}2_{\epsilon}$ release of $\mathcal{AMS}\text{-}\text{\LaTeX}$. Older versions of $\mathcal{AMS}\text{-}\text{\LaTeX}$ realize these options as sub-packages.

`intlimits` Like `sumlimits`, but for integral symbols.

`nointlimits` (default) Opposite of `intlimits`.

`namelimits` (default) Like `sumlimits`, but for certain “operator names” such as `\det`, `\inf`, `\lim`, `\max`, `\min`, that traditionally have subscripts placed underneath when they occur in a displayed equation.

`nonamelimits` Opposite of `namelimits`.

`sumlimits` (default) Place subscripts and superscripts of summation symbols above and below, in displayed equations. This option also affects other symbols of the same type—`\prod`, `\coprod`, `\bigotimes`, `\bigoplus`, and so forth—but excluding integrals (see `intlimits`).

`nosumlimits` Place subscripts and superscripts of summation-type symbols to the side, even in displayed equations.

The following three options are usually global document options and are thus set on the `\documentclass` command. They are, however, also recognized when the `amsmath` package is loaded with the `\usepackage` command.

`leqno` Place equation numbers on the left.

`reqno` Place equation numbers on the right (default).

`fleqn` Position equations at a fixed indent from the left margin rather than centered in the text column.

The $\mathcal{AMS}\text{-}\text{\LaTeX}$ distribution consists of a set of components, which can be loaded independently with the `\usepackage` command. The single most noteworthy package is probably `amsmath`, but the others can be used individually. Note that the `amsbsy`, `amsopn`, and `amstext` packages are included automatically when you use the `amsmath` package.

`amsmath` Defines extra environments for multiline displayed equations, plus a number of other enhancements for math.

`amsbsy` Defines the `\boldsymbol` and `\pmb` (poor man’s bold) commands.

`amsopn` Provides `\DeclareMathOperator` for defining new “operator names” like `\sin` and `\lim`.

`amstext` Provides a `\text` command for typesetting a fragment of text inside a display.

Other packages, providing supplementary functionality, should be loaded explicitly. Only parts of these packages are described in the present chapter.

They are mentioned here for completeness.

- amscd** Defines some commands for easing the generation of commutative diagrams by introducing the `CD` environment (see Section 8.4.4). There is no support for diagonal arrows.
- amsintx** Provides more descriptive command syntax for integrals and sums (not released yet).
- amsthm** Provides a **proof** environment and extensions for the `\newtheorem` command.
- amxtra** Provides certain odds and ends such as `\fracwithdelims` and `\accentedsymbol` (see Section 8.3.4).
- upref** Makes `\ref` print cross-reference numbers always in an upright/roman font regardless of context.

Finally, there are a few packages which come with the **AMSTeX** distribution.

- amsmath** defines the `\mathfrak` and `\mathbb` commands and sets up the fonts **msam** (extra math symbols A), **msbm** (extra math symbols B, and blackboard bold), **eufm** (Euler Fraktur), extra sizes of **cmib** (bold math italic and bold lowercase Greek), and **cmbsy** (bold math symbols and bold script), for use in mathematics.
- amssymb** defines the names of all the math symbols available with the **AMS** fonts collection. This package loads the **amsmath** package.
- eufrak** Set up the Fraktur letters.
- eucal** Makes `\mathcal` use the Euler script instead of the usual Computer Modern script letters.

All these packages recognize the **psamsmath** option, which will use the Y&Y/Blue Sky Research version of the **AMSTeX** collection (which is free available on CTAN).

8.6.7 **AMS-L^AT_EX** Document Classes

The **AMS-L^AT_EX** package comes with a pair of document classes called **amsmath** and **amsbook**, corresponding to L^AT_EX's **article** and **book**. They are primarily designed to prepare manuscripts for submission to the AMS, but there is nothing to prohibit their use for other purposes. With these class files the **amsmath** package is automatically included, so that you can start your document simply with `\documentclass{amsmath}` or `\documentclass{amsbook}`.

8.7 Examples of Multiple-Line Equation Structures

On the following pages we show a lot of real-life examples of the alignment environments discussed earlier. The lines indicating the margins around the typeset examples are not part of the environments but have been added to make the marginal spacing stand out clearly.

8.7.1 The **split** Environment

The **split** environment is not an independent environment but should be used inside something else, such as **equation** or **align**.

If there is not enough room for it, the equation number for a **split** will be shifted to the previous line when equation numbers are on the left; the number shifts down to the next line when numbers are on the right.

When you do not want an equation number, use the **equation*** environment.

$$\begin{aligned}
 f_{h,\varepsilon}(x,y) &= \varepsilon \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y\varepsilon(\varepsilon u)} \varphi(x) du \\
 &= h \int L_{x,z} \varphi(x) \rho_x(dz) \\
 &\quad + h \left[\frac{1}{t_\varepsilon} \left(\mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y\varepsilon(s)} \varphi(x) ds - t_\varepsilon \int L_{x,z} \varphi(x) \rho_x(dz) \right) \right. \\
 &\quad \left. + \frac{1}{t_\varepsilon} \left(\mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y\varepsilon(s)} \varphi(x) ds - \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y\varepsilon(\varepsilon s)} \varphi(x) ds \right) \right] \quad (8.47)
 \end{aligned}$$

This was produced by the following input (the **T_EX** command `\phantom` is used to leave a space equal to the width of its argument):

```

\begin{equation}
\begin{split}
f_{\{h,\varepsilon\}}(x,y)
&= \varepsilon \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y\varepsilon(\varepsilon u)} \varphi(x) du \\
&= h \int L_{x,z} \varphi(x) \rho_x(dz) \\
&\quad + h \left[ \frac{1}{t_\varepsilon} \left( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y\varepsilon(s)} \varphi(x) ds - t_\varepsilon \int L_{x,z} \varphi(x) \rho_x(dz) \right) \right. \\
&\quad \left. + \frac{1}{t_\varepsilon} \left( \mathbf{E}_y \int_0^{t_\varepsilon} L_{x,y\varepsilon(s)} \varphi(x) ds - \mathbf{E}_{x,y} \int_0^{t_\varepsilon} L_{x,y\varepsilon(\varepsilon s)} \varphi(x) ds \right) \right]
\end{split}
\end{equation}

```

```
\end{split}
\end{equation}
```

If the option `centertags` is included in the options list of the `amsmath` package, the equation numbers for `split` environments will be centered vertically on the height of the `split`, as shown in the example below.

$$\left| I_2 \right| = \left| \int_0^T \psi(t) \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^\theta c(\xi) u_t(\xi, t) d\xi \right\} dt \right| \quad (8.48)$$

$$\leq C_6 \left\| f \int_\Omega \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right\| \left\| |u| \xrightarrow{\circ} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.$$

This is produced by the following input:

```
\begin{equation}
\begin{split}
|I_2|
&= \left| \int_0^T \psi(t) \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^\theta c(\xi) u_t(\xi, t) d\xi \right\} dt \right| \\
&\leq C_6 \left\| f \int_\Omega \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right\| \left\| |u| \xrightarrow{\circ} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.
\end{split}
\end{equation}
```

One further example involving `split` and `align`. To obtain unnumbered equations use the `align*` environment instead.

$$\left| I_1 \right| = \left| \int_\Omega g R u d\Omega \right|$$

$$\leq C_3 \left[\int_\Omega \left(\int_a^x g(\xi, t) d\xi \right)^2 d\Omega \right]^{1/2} \quad (8.49)$$

$$\times \left[\int_\Omega \left\{ u_x^2 + \frac{1}{k} \left(\int_a^x c u_t d\xi \right)^2 \right\} c \Omega \right]^{1/2}$$

$$\leq C_4 \left\| f \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \xrightarrow{\circ} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.$$

$$\left| I_2 \right| = \left| \int_0^T \psi(t) \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^\theta c(\xi) u_t(\xi, t) d\xi \right\} dt \right| \quad (8.50)$$

$$\leq C_6 \left\| f \int_\Omega \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \xrightarrow{\circ} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.$$

The input for the above formulae is:

```
\begin{align}
\begin{split}
|I_1| &= \left| \int_\Omega g R u d\Omega \right| \\
&\leq C_3 \left[ \int_\Omega \left( \int_a^x g(\xi, t) d\xi \right)^2 d\Omega \right]^{1/2} \\
&\quad \times \left[ \int_\Omega \left\{ u_x^2 + \frac{1}{k} \left( \int_a^x c u_t d\xi \right)^2 \right\} c \Omega \right]^{1/2} \\
&\leq C_4 \left\| f \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \xrightarrow{\circ} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.
\end{split}
\end{split}
\begin{split}
|I_2| &= \left| \int_0^T \psi(t) \left\{ u(a, t) - \int_{\gamma(t)}^a \frac{d\theta}{k(\theta, t)} \int_a^\theta c(\xi) u_t(\xi, t) d\xi \right\} dt \right| \\
&\leq C_6 \left\| f \int_\Omega \left| \tilde{S}_{a,-}^{-1,0} W_2(\Omega, \Gamma_l) \right| \right\| \left\| |u| \xrightarrow{\circ} W_2^{\tilde{A}}(\Omega; \Gamma_r, T) \right\|.
\end{split}
\end{align}
```

8.7.2 The multiline Environment

Numbered version:

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \end{aligned} \right| \quad (8.51)$$

This was obtained with the lines shown below.

```
\begin{multiline}\label{eq:E}
\int_a^b \biggl\{ \int_a^b [ f(x)^2 g(y)^2 + f(y)^2 g(x)^2 ]
-2f(x) g(x) f(y) g(y) \,dx \biggr\} \,dy
=\int_a^b \biggl\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2
\int_a^b fg \biggr\} \,dy
\end{multiline}
```

An unnumbered version of the above is obtained with the same input, except the `multiline` environment is replaced by `multiline*`.

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \end{aligned} \right|$$

And now an unnumbered version numbered with a `\tag*` command.

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \end{aligned} \right| \quad [a]$$

This was generated with:

```
\begin{multiline*}\tag*[a] ... \end{multiline*}
```

This is the same display, but with `\multlinegap` set to zero. Notice that the space on the left of the first line does not change, because of the equation number, while the second line is pushed over to the right margin.

$$\left| \begin{aligned} \int_a^b \left\{ \int_a^b [f(x)^2 g(y)^2 + f(y)^2 g(x)^2] - 2f(x)g(x)f(y)g(y) dx \right\} dy \\ = \int_a^b \left\{ g(y)^2 \int_a^b f^2 + f(y)^2 \int_a^b g^2 - 2f(y)g(y) \int_a^b fg \right\} dy \end{aligned} \right| \quad [a]$$

This was generated with:

```
{\setlength{\multlinegap}{0pt}
\begin{multiline*}\tag*[a] ... \end{multiline*}}
```

8.7.3 The gather Environment

Numbered version with `\notag` on the second line:

$$\left| \begin{aligned} D(a, r) &\equiv \{z \in \mathbf{C} : |z - a| < r\}, & (8.52) \\ \text{seg}(a, r) &\equiv \{z \in \mathbf{C} : \exists z = \Im a, |z - a| < r\}, \\ c(e, \theta, r) &\equiv \{(x, y) \in \mathbf{C} : |x - e| < y \tan \theta, 0 < y < r\}, & (8.53) \\ C(E, \theta, r) &\equiv \bigcup_{e \in E} c(e, \theta, r). & (8.54) \end{aligned} \right|$$

This was generated with:

```
\begin{gather}
D(a,r) \equiv \{ z \in \mathbf{C} : |z-a|<r \}, \\
\operatornamename{seg}(a,r) \equiv \{ z \in \mathbf{C} : \\
\quad \Im z = \Im a, \quad |z-a|<r \}, \quad \notag \\
c(e,\theta,a,r) \equiv \{ (x,y) \in \mathbf{C} : \\
\quad |x-e|<y \tan \theta, \quad 0<y<r \}, \\
C(E,\theta,a,r) \equiv \bigcup_{e \in E} c(e,\theta,a,r). \\
\end{gather}
```

8.7.4 The align Environment

Numbered version:

$$\left| \begin{aligned} \gamma_x(t) &= (\cos tu + \sin tx, v), & (8.55) \\ \gamma_y(t) &= (u, \cos tv + \sin ty), & (8.56) \\ \gamma_z(t) &= \left(\cos tu + \frac{\alpha}{\beta} \sin tv, -\frac{\beta}{\alpha} \sin tu + \cos tv \right). & (8.57) \end{aligned} \right|$$

This was produced using the following input:

```
\begin{align}
\gamma_x(t) &= (\cos tu + \sin tx, v), \\
\gamma_y(t) &= (u, \cos tv + \sin ty), \\
\gamma_z(t) &= \left( \cos tu + \frac{\alpha}{\beta} \sin tv, \right. \\
&\quad \left. - \frac{\beta}{\alpha} \sin tu + \cos tv \right). \\
\end{align}
```

Unnumbered version:

$$\left| \begin{array}{l} \gamma_x(t) = (\cos tu + \sin tx, v), \\ \gamma_y(t) = (u, \cos tv + \sin ty), \\ \gamma_z(t) = \left(\cos tu + \frac{\alpha}{\beta} \sin tv, -\frac{\beta}{\alpha} \sin tu + \cos tv \right). \end{array} \right|$$

This was generated using the following construct:

```
\begin{align*} ... \end{align*}
```

8.7.5 Using the align and split Environments within gather

When using the `align` environment within the `gather` environment, one or the other, or both, should be unnumbered (using the `*` form), since having numbering for both the outer and inner environment would not be meaningful.

Automatically numbered `gather` with `split` and `align*`:

$$\left| \begin{array}{l} \varphi(x, z) = z - \gamma_{10}x - \sum_{m+n \geq 2} \gamma_{mn}x^m z^n \\ \quad = z - Mr^{-1}x - \sum_{m+n \geq 2} Mr^{-(m+n)}x^m z^n \\ \zeta^0 = (\xi^0)^2, \\ \zeta^1 = \xi^0 \xi^1 \end{array} \right| \quad (8.58)$$

Here the `split` environment gets a number from the outer `gather` environment; numbers for individual lines of the `align*` are suppressed because of the star.

```
\begin{gather}
\begin{split}
\varphi(x, z)
&= z - \gamma_{10} x - \sum_{m+n \geq 2} \gamma_{mn} x^m z^n \\
&= z - M r^{-1} x - \sum_{m+n \geq 2} M r^{-(m+n)} x^m z^n
\end{split}
\end{gather}
\begin{align*}
\zeta^0 &= (\xi^0)^2, \\
\zeta^1 &= \xi^0 \xi^1
\end{align*}
```

Shown below, is the `*`-ed form of `gather` with the non-`*`-ed form of `align`.

$$\left| \begin{array}{l} \varphi(x, z) = z - \gamma_{10}x - \sum_{m+n \geq 2} \gamma_{mn}x^m z^n \\ \quad = z - Mr^{-1}x - \sum_{m+n \geq 2} Mr^{-(m+n)}x^m z^n \\ \zeta^0 = (\xi^0)^2, \\ \zeta^1 = \xi^0 \xi^1 \end{array} \right| \quad \begin{array}{l} (8.59) \\ (8.60) \end{array}$$

The latter was produced with the following construct:

```
\begin{gather*}
\begin{split}
\begin{align}
\end{align}
\end{split}
\end{gather*}
```

8.7.6 Using the alignat Environments

Numbered version:

$$\left| \begin{array}{l} V_i = v_i - q_i v_j, \quad X_i = x_i - q_i x_j, \quad U_i = u_i, \quad \text{for } i \neq j; \\ V_j = v_j, \quad X_j = x_j, \quad U_j u_j + \sum_{i \neq j} q_i u_i. \end{array} \right| \quad \begin{array}{l} (8.61) \\ (8.62) \end{array}$$

This example was obtained with the commands below:

```
\begin{alignat}{3}
V_i &= v_i - q_i v_j, & X_i &= x_i - q_i x_j, & U_i &= u_i, & \text{for } i \neq j; \\
V_j &= v_j, & X_j &= x_j, & U_j &= u_j + \sum_{i \neq j} q_i u_i.
\end{alignat}
```

Unnumbered version:

$$\left| \begin{array}{l} V_i = v_i - q_i v_j, \quad X_i = x_i - q_i x_j, \quad U_i = u_i, \quad \text{for } i \neq j; \\ V_j = v_j, \quad X_j = x_j, \quad U_j u_j + \sum_{i \neq j} q_i u_i. \end{array} \right|$$

This was generated using the following construct:

```
\begin{alignat*}{3} ... \end{alignat*}
```

The most common use for `alignat` is for things like

$x = y$	by (8.49)	(8.63)
$x' = y'$	by (8.61)	(8.64)
$x + x' = y + y'$	by Axiom 1.	(8.65)

This example was obtained with the commands below:

```
\begin{alignat}{2}
x      &= y      && \quad \text{by } (\ref{eq:A}) \backslashlabel{eq:C} \backslash
x'     &= y'     && \quad \text{by } (\ref{eq:B}) \backslashlabel{eq:D} \backslash
x + x' &= y + y' && \quad \text{by Axiom 1.}
\end{alignat}
```

The expanded version, `flalign`:

$x = y$	by (8.63)	(8.66)
$x' = y'$	by (8.64)	(8.67)
$x + x' = y + y'$	by Axiom 1.	(8.68)

This was generated using the following construct:

```
\begin{flalign} \dots \end{flalign}
```

8.8 Extensions to the theorem Environment

$\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX comes with the `amsthm` package, which extends \LaTeX 's `\newtheorem` command. Rather than describe `amsthm` (see, for instance, the section “Proclamations” in Grätzer’s book [?] for more details) we will give some details about the `theorem` package, developed by Frank Mittelbach [?]. It also offers an extension of the \LaTeX theorem mechanism by allowing the layout of theorems to be manipulated by specifying a style.

In the present context the word “theorem” is used for any kind of labeled enunciations, often set off from the main text by extra space and a font change. Theorems, corollaries, conjectures, definitions, and remarks are all instances of “theorems.” The header of these structures is composed of a label (such as `THEOREM` or `REMARK`) and a number, which serializes an item in the sequence of items with the same label.

Often it is necessary, in order to satisfy the requirements of different mathematics journals, to customize the layout of the theorem environment. Additionally, different formats may be needed to differentiate the “sort of theorem”: e.g., remarks and definitions are set in roman, while italic is employed for main theorems.

(\mathcal{L} 58, 174)

8.8.1 Defining New Theorem Environments

As in the original \LaTeX version, the command `\newtheorem` defines a new “theorem-like structure.” Two required arguments name the new environment and give the text to be typeset with each instance of the new environment, while an optional argument determines how the environment is enumerated:

```
\newtheorem{env-name}{label-text}
```

The above `\newtheorem` command defines the *env-name* environment and its printed name will be *label-text*. It uses its own counter.

```
\newtheorem{env2-name}[env-name]{label-text2}
```

The above `\newtheorem` command defines the *env2-name* environment, and its printed name will be *label-text2*. It uses the same counter as theorem set *env-name*.

```
\newtheorem{env3-name}{label-text3}[section]
```

The above variant defines the *env3-name* environment and its printed name is *label-text3*. Its counter is enumerated within the counter *section*, that is, with every new `\section` the enumeration starts again with one, and the enumeration is composed from the section number and the theorem counter itself.

```
\theoremstyle{style}
```

The `\theoremstyle` command can define the layout of various, or all, theorem sets. It should be noted that any theorem set defined by `\newtheorem` is typeset in the `\theoremstyle` that is current at the time of the definition.

Thus, the following

```
\theoremstyle{break}      \newtheorem{Cor}{Corollary}
\theoremstyle{plain}      \newtheorem{Exa}{Example}[section]
```

leads to the result that the set `Cor` is formatted in the style `break`, while the set `Exa` and all the following ones are formatted in the style `plain`, unless another `\theoremstyle` follows. Since the definitions installed by `\newtheorem` are global, you can also limit `\theoremstyle` locally by grouping braces.

```
\theorembodyfont{font-declarations}
```

The choice of the font for the theorem body is completely independent of the chosen `\theoremstyle`; this has proven to be very advantageous. For example,

```
{\theorembodyfont{\rmfamily} \newtheorem{Rem}{Remark}}
```

<code>plain</code>	Emulates the original L ^A T _E X definition, except that additionally the parameters <code>\theorempreskipamount</code> and <code>\theorempostskipamount</code> are used.
<code>break</code>	In this style, the theorem header is followed by a line break.
<code>marginbreak</code>	The theorem number is set in the margin, and there is a line break as in <code>break</code> .
<code>changebreak</code>	Like <code>break</code> , but with header number and text interchanged.
<code>change</code>	Header number and text are interchanged, without a line break.
<code>margin</code>	The number is set in the left margin, without a line break.

Table 8.22: List of existing theorem styles

All styles (except `plain`) select `\normalfont\slshape` as the default for `\theorembodyfont`.

defines a theorem set `Rem`, which will be set in `\rmfamily` in the current layout (which in our example is `plain`). As with `\theoremstyle`, the `\theorembodyfont` chosen is that which is current at the time of `\newtheorem`. If `\theorembodyfont` is not specified or you define `\theorembodyfont{}`, then the font used will be defined by `\theoremstyle`.

```
\theoremheaderfont{font-declarations}
```

It is also possible to customize the font used for the theorem headers. This is, however, a global declaration and, therefore, there should be at most one `\theoremheaderfont` command in the preamble. If it is actually necessary to have different header fonts, you will have to define new theorem styles (substituting the desired font).

Two additional parameters affect the vertical space around the theorem environments: `\theorempreskipamount` and `\theorempostskipamount` define, respectively, the spacing before and after such an environment. These parameters apply to all theorem sets and can be manipulated with the ordinary length macros. They are rubber lengths, and therefore can contain `plus` and `minus` parts. These parameters are set using the `\setlength` command.

The commands to define theorem sets, as described in this section, can only be placed in the document preamble or in a package file.

Theorem styles, which exist to date, are shown in table 8.22

8.8.2 Examples of the Definition and Use of Theorems

Suppose that the preamble contains the declarations:

```
\theoremstyle{break}      \newtheorem{Cor}{Corollary}
\theoremstyle{plain}      \newtheorem{Exa}{Example}[section]
```

```
{\theorembodyfont{\rmfamily} \newtheorem{Rem}{Remark}}
\theoremstyle{marginbreak} \newtheorem{Lem}[Cor]{Lemma}
\theoremstyle{change}
\theorembodyfont{\itshape} \newtheorem{Def}[Cor]{Definition}

\theoremheaderfont{\scshape}
```

Then the typical examples below show the typeset output resulting from their use.

```
COROLLARY 1                                \begin{Cor}
This is a sentence typeset in the theorem   This is a sentence typeset in the theorem
environment Cor.                            environment \Lenv{Cor}.
\end{Cor}
```

```
EXAMPLE 8.8.1 This is a sentence typeset \begin{Exa}
in the theorem environment Exa.           This is a sentence typeset in the theorem
\end{Exa}                                environment \Lenv{Exa}.
\end{Exa}
```

```
REMARK 1 This is a sentence typeset in the \begin{Rem}
theorem environment Rem.                  This is a sentence typeset in the theorem
\end{Rem}                                environment \Lenv{Rem}.
\end{Rem}
```

```
2 LEMMA (BEN USER)                        \begin{Lem}[Ben User]
This is a sentence typeset in the theorem   This is a sentence typeset in the theorem
environment Lem.                            environment \Lenv{Lem}.
\end{Lem}
```

```
3 DEFINITION (VERY IMPRESSIVE DEFINITION) \begin{Def}[Very impressive Definition]
This is a sentence typeset in the theorem   This is a sentence typeset in the theorem
environment Def.                            environment \Lenv{Def}.
\end{Def}
```

The last two examples show the effect of the optional argument to a theorem environment (it is typeset in parentheses right after the label).

8.8.3 Special Considerations

The theorem header and body are implemented as a single unit. This means that the `\theoremheaderfont` will inherit characteristics of the `\theorembodyfont` if the NFSS is being used. Thus, if, for example, `\theorembodyfont` is `\itshape`

and `\theoremheaderfont` is `\bfseries` the font selected for the header will have the characteristics “bold extended italic.” If this is not desired you should set it to something like `\theoremheaderfont{\normalfont\bfseries}`. That is, you should supply all the necessary font information explicitly. See chapter 7 for more details about how to do that.

8.9 Mathematical Style Parameters

This section explains how you can globally control the style of your mathematical formulae, and how you can modify the size of certain (sub)formula elements.

8.9.1 Controlling the Size of Characters

Letters and mathematical symbols sometimes get smaller when they appear in fractions, superscripts, or subscripts. In fact, $\mathrm{T\!E\!X}$ has eight different styles in which it can treat formulae, namely:

D, D'	<code>\displaystyle</code>	formulae displayed on lines by themselves
T, T'	<code>\textstyle</code>	formulae embedded in the text
S, S'	<code>\scriptstyle</code>	formulae used as super- or subscripts
SS, SS'	<code>\scriptscriptstyle</code>	second- and higher-order super- or subscripts

The accented symbols represent the so-called *cramped* styles, which are similar to the normal styles except that exponents are not raised so much. $\mathrm{T\!E\!X}$ also uses three different type sizes for mathematics, namely: text size, script size, and scriptscript size.

A formula set inside text (between a $\$$ pair, or between `\(. . \)`) is typeset using text style (style T). A formula on a line by itself, e.g., entered between `\[. . \]`, will be typeset in display style (style D). The size of the different parts of a formula can be determined according to the following scheme:

A symbol in style	will be typeset in	(example)
D, D', T, T'	text size	(text size)
S, S'	script size	(script size)
SS, SS'	scriptscript size	(scriptscript size)

The kind of style used in mathematics formulae is as follows:

style	superscript	subscript	numerator	denominator
D	S	S'	T	T'
D'	S'	S'	T'	T'
T	S	S'	S	S'
T'	S'	S'	S'	S'
S, SS	SS	SS'	SS	SS'
S', SS'	SS'	SS'	SS'	SS'

The last two columns describe the style used in the numerator or denominator of a fraction. An example of the various styles can be seen in the continued fraction below (see also section 8.3.16):

$$b^0 + \frac{a^1}{b_1 + \frac{a^2}{b_2 + \frac{a^3}{b_3}}}$$

```
\normalsize
\[ b^0 + \frac{a^1}{b_1 + \frac{a^2}{b_2 + \frac{a^3}{b_3}}}
\]
```

In the formula above the b of b^0 is in style D , with the 0 in style S ; the a and b of a^1 and b_1 are in style T and T' , respectively, with the exponent 1 in style S and the subscript 1 in style S' ; the a and b of a^2 and b_2 are both in style S' , with the exponent and subscript in style SS' ; finally everything in a^3 and b_3 is in style SS' .

You can give a nicer look to the above example by deciding which style is to be used in each case. Note that to save typing, we define the abbreviation `\D` for the `\displaystyle` command.

$$b^0 + \frac{a^1}{b_1 + \frac{a^2}{b_2 + \frac{a^3}{b_3}}}$$

```
\newcommand{\D}{\displaystyle}
\normalsize
\[ b^0 + \frac{a^1}{\D b_1 + \frac{a^2}{\D b_2 + \frac{a^3}{\D b_3}}}
\]
```

8.9.2 L^AT_EX Math Style Parameters

Because L^AT_EX uses much of the mathematical machinery from $\mathrm{T\!E\!X}$, we briefly describe the mathematical style parameters that L^AT_EX uses to typeset formulae. All these are length parameters which you can redefine with the `\setlength` or `\addtolength` commands (see section A.1.4 on page 474). Moreover, two standard options, `leqno` and `fleqn`, control the numbering and alignment of formulae.

(\mathcal{L} 170)

(\mathcal{L} 82)

The option `fleqn` causes formulae to be aligned on the left, a fixed distance from the left margin (see `\mathindent` below), instead of being centered.

The option `leqno` causes formula numbers to appear on the left instead of at the right (see section 8.6.6 on page 252).

In the list of mathematics style parameters below, all lengths (except `\jot` and `\arraycolsep`) are rubber lengths. With the option `fleqn`, the four `\displayskip` lengths are made equal to the list defining length `\topsep`, to which the value of `\partopsep` is added if the display starts a paragraph (see figure 3.5 on page 64). The four parameters `\abovedisplay...` and `\belowdisplay...` below depend on the current font size. For this reason they cannot be modified in the preamble of the document using `\setlength`, but they must be changed by modifying `\normalsize`, etc.

`\arraycolsep` This gives half the width of the horizontal space between columns in an `array` environment (default value `5pt`, see also section 5.3.2).

`\jot` This is the extra vertical space that is added between rows in an `eqnarray` or `eqnarray*` environment (default value `3pt`).

`\mathindent` This defines the indentation from the left margin of displayed formulae for the `fleqn` option (the default value is equal to the indentation of a first level list, i.e., `2.5em`, and is defined by the option `fleqn`).

`\abovedisplayskip` This specifies the extra space left above a long displayed formula, except with the option `fleqn`, where `\topsep` is used. A long formula is one that lies closer to the left margin than does the end of the preceding line (default value `12pt plus 3pt minus 9pt`).

`\belowdisplayskip` This specifies the extra space left below a long displayed formula, except with the option `fleqn`, where `\topsep` is used (default value `12pt plus 3pt minus 9pt`).

`\abovedisplayshortskip` This specifies the extra space left above a short displayed formula, except with the option `fleqn`, where `\topsep` is used. A short formula is one which starts to the right of where the preceding line ends (default value `0pt plus 3pt`).

`\belowdisplayshortskip` This specifies the extra space left below a short displayed formula, except with the option `fleqn`, where `\topsep` is used (default value `7pt plus 3pt minus 4pt`).

Bibliographie

- [1] Amir D. Aczel. *L'énigme du théorème de Fermat*. Desclée de Brower, 1998.
- [2] American Mathematical Society. *AMS- \LaTeX Version 1.2, User's Guide*, novembre 1996. Version 1.02.
- [3] Michèle Audin. Conseils aux auteurs de textes mathématiques. Disponible sur le web : <http://irmasrv1.u-strasbg.fr/~maudin/publications.html>, octobre 1997.
- [4] M Baudoin. Apprends latex. manuel de l'ENSTA, disponible sur le web par ftp sur <http://www.loria.fr/services/tex/general/apprends-latex.ps.gz>.
- [5] Benjamin Bayart. Joli manuel pour $\text{\LaTeX}2_{\epsilon}$. Guide local de l'ESIEE, compilé le 18/12/1995, disponible sur le web, par exemple à <http://www.loria.fr/services/tex/general/manuel2ep.ps.gz>.
- [6] Bernard Desgraupes. *LaTeX : apprentissage, guide et référence*. Vuibert informatique, Paris, deuxième édition, 2003. Ouvrage disponible à la bibliothèque de l'UTBM sous la cote QA76.754 DES.
- [7] Steven Douglas Cochran. *The subfigure package*, mars 1995.
- [8] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX companion*. Addison, 1994. Includes newly revised LaTeX standard, Ouvrage disponible à la bibliothèque de l'UTBM sous la cote QA76.754 GOO.
- [9] Michel Goossens, Sebastian Rahtz, and Frank Mittelbach. *The LaTeX graphics companion : illustrating documents with Tex and PostScript*. Addison, 1997. Ouvrage disponible à la bibliothèque de l'UTBM sous la cote QA76.754 GOO.
- [10] Michael C. Grant and David Carlisle. *The PSFRAG system, version 3*, avril 1998.
- [11] Carsten Heinz. *Listings.dtx Version 0.19*, 1996.
- [12] K. Kopla and P.W. Daly. *A Guide to $\text{\LaTeX}2_{\epsilon}$, document preparation for beginners and advanced users*. Addison Wesley, 1995.
- [13] Thomas Lachand-Robert. *La maîtrise de TEX et LaTeX*. Masson, 1995. Ouvrage disponible à la bibliothèque de l'UTBM sous la cote QA76.754.
- [14] J. Hicgham Nicholas. *Handbook of writing for the mathematical sciences*. SIAM, 1998. Ouvrage disponible à la bibliothèque de l'UTBM sous la cote Q 226 HIG.
- [15] Tobias Oetiker. Une courte (?) introduction à $\text{\LaTeX}2_{\epsilon}$. based on LKURTZ.TEX Uni Graz & TU Wien, 1987, mail : tobias@ife.ee.ethz.ch or oetiker@dmu.ac.uk ; traduit en français par Matthieu Herrb, 1996 ; disponible sur le web : <http://www.laas.fr/~matthieu/cours/latex2e>.
- [16] Christian Rolland. *Latex, guide pratique*. Addison-Wesley, 1995.
- [17] Timothy Van Zandt. *Seminar.sty, A \LaTeX style for slides and notes, User's guide, version 1.0*, avril 1993.
- [18] Peter Williams. *Algorithms*, avril 1996.

Index

backslash, 11

bibliographie, 36

espace

après une commande, 12

en début de ligne, 11

index, 39

création, 38

utilisation, 39