

1 Commandes Unix de base

1.1 Introduction

Les PC de l'Institut Fourier utilisent le système d'exploitation Linux¹. Vous pouvez simuler l'environnement Linux sous Windows, en installant le logiciel libre `cygwin` disponible gratuitement sur www.cygwin.com.

Si vous n'avez pas de terminal ouvert, il faut utiliser l'interface graphique pour en lancer un, en cliquant sur une icône représentant une fenêtre terminal ou en utilisant le menu System de l'interface ou en cliquant avec le bouton droit de la souris dans l'arrière-plan, puis en sélectionnant `shell` ou `Terminal` si l'option existe, ou `Executer une commande` puis tapez `konsole` (ou `xterm` si `konsole` n'est pas reconnu) puis `Entrée`.

Attention, les conventions des interfaces graphiques Unix peuvent différer des interfaces que vous connaissez : par exemple, on utilise le click avec le bouton du milieu (ou click simultané sur les boutons droits et gauche) pour insérer le contenu du "presse-papier". Certains logiciels compatibles Windows acceptent toutefois les raccourcis clavier `Ctrl-C` (copier), `Ctrl-V` (coller) et `Ctrl-X` (effacer et copier).

1.2 Les répertoires

Les commandes de cette section et de la section suivante sont des commandes Unix, également utilisables sur Mac OS X (depuis l'application Terminal), ou avec Cygwin sous Windows.

Les données accessibles depuis un ordinateur sont placés dans une arborescence contenant des fichiers ordinaires (qui contiennent des données, dans l'arborescence ce sont les feuilles de l'arbre) et des répertoires (qui contiennent des fichiers ou des répertoires, ce sont les noeuds de l'arbre, là où plusieurs branches se rencontrent). La racine de l'arbre est un répertoire, désignée par `/`. La position d'un fichier sur l'arbre se repère par la suite des noms de répertoires, sous-répertoires, etc. permettant d'y accéder soit de manière absolue (en partant de la racine de l'arbre) soit de manière relative à un répertoire dit répertoire courant. Au login, le répertoire courant est appelé répertoire d'origine, c'est là que vous pouvez stocker vos fichiers personnels, en les organisant dans des répertoires.

- `ls` : afficher la liste des fichiers et des sous répertoires du répertoire courant. Pour avoir plus de détails, `ls -l`. Pour avoir une liste triée par date, `ls -l --sort time`. Si la liste est longue, appuyez simultanément sur la touche `shift` et la touche `page précédente`, ou faites suivre la commande par `| less`, par exemple
`ls -l --sort time | less`
- `pwd` : afficher le nom du répertoire courant.
- `cd` : permet de changer de répertoire.
 - `cd rep` : pour aller dans le sous répertoire appelé `rep`;
 - `cd` ou `cd ~` : pour aller dans votre répertoire d'origine.
 - `cd /usr/local` : pour aller dans le répertoire `/usr/local`
 - `cd ~dupont` : pour aller dans le répertoire d'origine de l'utilisateur `dupont`.
 - `cd ..` : pour aller dans le répertoire parent.
- `~` : désigne le chemin d'accès à votre répertoire d'origine.
`~dupont` : désigne le répertoire d'origine de l'utilisateur dont le nom de login est `dupont`
- `mkdir` : Pour créer un sous répertoire, par exemple
`mkdir rep` crée un sous-répertoire appelé `rep`
- `rmdir` : effacer un répertoire vide, par exemple `rmdir rep`.
- `mount/umount` : permet d'associer un périphérique à un répertoire, par exemple pour accéder à une clef USB, la commande
`mount /usb`
permet d'accéder aux fichiers de la clef USB comme s'ils étaient dans le répertoire `/usb`. Avant de

¹Pour les futurs candidats à l'agrégation, il s'agit du seul système d'exploitation proposé pour l'épreuve de modélisation

retirer la clef USB de l'ordinateur, il faut la désassocier du répertoire par la commande
`umount /usb`.

Exercice :

A l'aide des commandes expliquées ci-dessus :

1. Placez vous dans votre répertoire d'origine : **cd**. Vérifiez que vous y êtes bien : **pwd**. Regardez la liste des fichiers par : **ls** , puis avec : **ls -al**
2. En utilisant la commande **mkdir**, créer un répertoire de nom : **essai** Vérifier son existence (avec **ls**).
3. Aller dans ce répertoire : **cd essai** Vérifier que vous y êtes (avec **pwd**).
4. Revenir dans le répertoire d'origine par **cd ..** Vérifiez que vous y êtes. Détruire (**rmdir**) le répertoire **essai**.

1.3 Les fichiers

- **cp** : Copier un fichier :
`cp essai.cc test.cc` : crée une copie du fichier `essai.cc` nommée `test.cc` (dans le répertoire courant)
`cp fichier1.cc ..` : copie le fichier `fichier1.cc` dans le répertoire parent (situé juste avant le répertoire courant).
 - **mv** : Déplace ou/et renomme un fichier.
`mv fichier1.cc fichier2.cc` : renomme le fichier `fichier1.cc`
Si `rep` est un répertoire, `mv fichier1.cc rep` déplace le fichier `fichier1.cc` dans le sous répertoire `rep`.
 - **rm** : effacer un fichier
`rm fichier.cc` : efface le fichier `fichier.cc`.
`rm -rf rep` : efface le répertoire `rep` et tous ses sous-répertoires. Attention, regardez ce que vous avez tapé avant de valider, il n'y a pas de commande de récupération.
 - **emacs** ou **nedit** : éditer un fichier texte, par exemple
`emacs fichier.cc &`
ouvre le fichier `fichier.cc` (le crée s'il n'existe pas).
 - ***** désigne n'importe quelle chaîne de caractère dans un nom de fichier, par exemple
`cp *.tex /usb`
permet de recopier tous les fichiers dont le nom termine par `.tex` sur la clef USB (après l'avoir associé au répertoire `/usb` par la commande `mount /usb`).
 - **?** permet de remplacer n'importe quel caractère dans un nom de fichier
 - **mcop**y (en local) : permet de copier des fichiers sur une disquette, par exemple :
`mcop Programmes/essai.cc a:`
recopie le fichier `essai.cc` du répertoire `Programmes` sur la disquette. Inversement :
`mcop a:essai.cc Programmes`
recopie le fichier `essai.cc` de la disquette vers le répertoire `Programmes`.
 - **mdir** `a:` (en local) : liste les fichiers contenus sur la disquette
- Remarques :

1. Notez que l'appui sur la touche de tabulation (à gauche de la touche **A** sur un clavier français) est un raccourci clavier qui permet de compléter automatiquement le début d'un nom de fichier ou d'afficher les différentes possibilités de complétion.
2. Le caractère `&` placé à la fin d'une commande permet de l'exécuter en tâche de fond (on peut lancer d'autres commandes pendant que cette commande s'exécute).
3. Pour avoir plus de détails sur une commande Unix, vous pouvez utiliser la commande `man`, par exemple, taper dans une fenêtre de commande :
`man ls`
pour en savoir plus sur la commande `ls`.

Exercice :

Créez un fichier appelé `essai` contenant quelques lignes. Déconnectez-vous et laissez votre binome se connecter. Recopiez le fichier `essai` de votre binome dans votre répertoire. Renommez-le `essai.de.mon.binome`. Recopiez le fichier `essai` sur une disquette.

1.4 Noms et fonction de quelques logiciels.

Ces noms de commande vous permettront de lancer les logiciels correspondants depuis n'importe quel interpréteur de commande (ce qui permet de les lancer indépendamment de l'environnement, il suffit de connaître leur nom et non pas dans quel menu ils se trouvent).

- `emacs`, `nedit` : éditeurs de texte.
- `firefox` (parfois `mozilla`) : navigateur Internet. Permet également de lire des fichiers `.html`
- `soffice` ou `abiword` : traitement de texte non scientifique. Permet de lire la plupart des fichiers `.doc` de Microsoft Word.
- `texmacs` : traitement de texte scientifique particulièrement adapté à la saisie d'équations. Il permet aussi d'effectuer des calculs en s'interfaçant avec d'autres logiciels.
- `soffice` ou `gnnumeric` : tableur, permet de lire la plupart des fichiers `.xls` de Microsoft Excel.
- `kpdf` (parfois `xpdf` ou `acroread`) : pour lire des fichiers PDF (d'extension `.pdf`)
- `latex`, `pdflatex` : compilation d'un fichier \LaTeX
- `xdvi` : visualiser des fichiers DVI (d'extension `.dvi`), cf. la micro-introduction à \LaTeX ci-dessous,
- `gv` ou `ghostview` : visualiser des fichiers Postscript (d'extension `.ps`)
- `gimp` : traitement d'images par exemple de fichiers d'extension `.gif`, `.png`
- `lpr` : imprime un fichier.
- `zip/unzip` : utilitaires de compression/décompression d'archives au format zip, très répandu sur les machines Windows.
- `scilab` : logiciel de calcul numérique
- `xcas` : logiciel libre de calcul formel et géométrie interactive, interfacé avec le logiciel d'arithmétique PARI-GP.
- `xmaple` : logiciel commercial de calcul formel
- etc.

2 Une micro introduction à L^AT_EX

2.1 Introduction

L^AT_EX est le format standard utilisé dans le monde de l'édition mathématique. Un document au format L^AT_EX est un texte (au format ASCII) contenant des commandes de formatage. Ces commandes servent à structurer le texte (chapitres, sections, etc.) en laissant au compilateur le soin de rendre cette structure au mieux en fonction du format de sortie (texte imprimé, fichier PDF, sortie HTML pour mettre sur un site Web). Elles gèrent aussi l'affichage des symboles mathématiques, la numérotation des chapitres (chapter) sections, sous-sections (section, subsection), les références (on place un repère nom avec label puis on se réfère à nom avec ref ou pageref)

```
\label{toto} puis \ref{toto} ou \pageref{toto}
```

ou permettent de créer automatiquement la table des matières (`\tableofcontents`). Une fois saisi, le texte source L^AT_EX doit être compilé (parfois deux fois de suite pour mettre à jour les références et la table des matières). Le compilateur ne tient pas compte de la mise en page du texte source, le nombre d'espace entre deux mots est ignoré de même que le passage à la ligne. Seuls les sauts de lignes sont interprétés comme signalant un début de paragraphe.

La syntaxe d'une commande de formatage L^AT_EX est :

```
\command[option]{argument}
```

Il existe dix caractères réservés qui ne sont donc pas imprimés tels quels :

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

Pour les imprimer, il faut taper :

```
\$ \& \% \# \_ \{ \} \^{} \~{} \symbol{92}
```

Le passage à la ligne (changement de paragraphe) se fait en insérant une ligne vide, la ligne suivante est alors indentée²

Un espace est créé avec `_`.

2.2 L'édition

Utilisez Emacs pour éditer votre texte en L^AT_EX :

```
emacs essai.tex &
```

Tapez, les lignes suivantes (sans les commentaires qui commencent par %).

```
\documentclass[a4paper,11pt]{article} % 11 ou 12pt, article ou report ou book
\usepackage[utf8]{inputenc}           % pour taper les caractères accentués
\usepackage[T1]{fontenc}              % idem
\usepackage[francais]{babel}          % français (chapter -> chapitre...)
\usepackage{amsmath,amsfonts,amssymb} % symboles AMS
\newcommand{\N}{\mathbb{N}}           % définit la commande \N
\title{Un essai\Semaine info licence} % définit le titre (ici sur 2 lignes)
\author{Mon Nom}                      % indiquez votre nom

\begin{document}                      % début du document

\maketitle                            % écrit le titre (cf. \title et \author)

\section{Calcul de  $A^P \bmod N$ } % un paragraphe
```

²On peut forcer un passage à la ligne sans indentation en tapant `\\` mais ceci n'est pas recommandé pour la lisibilité du texte.

```

Soit $A \in \mathbb{N}$ un entier, ...           % on utilise la commande \N

\subsection{Traduction Algorithmique} % un sous paragraphe
\label{sec:tradu} % définit un label
L'algorithme de la puissance rapide se compose de plusieurs parties
\begin{enumerate}
\item On commence par ...
\item Ensuite ...
\end{enumerate}

\subsection{Le programme en $C^{++}$} % un autre sous paragraphe
On a vu (section \ref{sec:tradu}) ... % une référence au label

\newpage % nouvelle page
\tableofcontents % table des matières

\end{document} % fin du document

```

2.3 La compilation

Pour traduire les différentes commandes de votre texte, il faut le compiler. Vous devez d'abord sauvegarder votre texte, à la souris, menu `Files -> Save current buffer` ou au clavier en tapant (`Ctrl-X Ctrl-S`). Dans la fenêtre de commandes (Konsole ou xterm), tapez :

```
latex essai
```

La compilation se fait avec traduction en un fichier `essai.dvi` ou, avec un arrêt à la première erreur rencontrée. Lorsque une erreur est détectée, un message apparaît indiquant, la nature de l'erreur et la ligne où elle se situe. Tapez sur la touche `Entrée` pour continuer ou tapez `x` puis `Entrée` pour interrompre la compilation. Corrigez votre erreur dans la fenêtre `emacs` et recompilez.

2.4 La visualisation

Pour visualiser votre texte avant l'impression, tapez dans la fenêtre de commandes :

```
xdvi essai &
```

Si la page de visualisation n'est pas mise à jour lorsque vous compilez à nouveau, vous devez quitter `xdvi` en tapant sur la touche `q` et le relancer avec la commande ci-dessus.

2.5 L'impression

Pour imprimer, tapez dans la fenêtre de commandes :

```
dvips essai
```

2.6 Créer des fichiers PDF et HTML à partir d'un source \LaTeX

Si vous utilisez la commande `pdflatex` à la place de la commande `latex`, le compilateur génère un fichier `.pdf` au format PDF (que l'on peut lire avec Acrobat Reader ou sous Unix avec `gv`).

Pour obtenir une sortie HTML, utilisez la commande `latex2html` (disponible sur certains systèmes seulement)

2.7 Les environnements \LaTeX

C'est une partie du document délimitée par :

```
\begin{type d'environnement}...\end{type d'environnement}
```

Voici quelques environnements souvent utilisés :

- `\begin{verbatim} ... \end{verbatim}` : pas d'interprétation des commandes, le texte est mis en style `\texttt` (contrairement à `\tt ...` qui met en style `\texttt` mais interprète...)
- `\begin{itemize} ... \end{itemize}` ou `\begin{enumerate} ... \end{enumerate}` : permet d'énumérer une liste ; chaque élément de la liste doit commencer par `\item`
La différence est que `enumerate` numérote les items
- `\begin{center} ... \end{center}` permet de centrer un texte
- `\begin{tabular}{|l|c|r|r|} ... \end{tabular}` : crée un tableau. Le nombre d'arguments (ici 4) indique le nombre de colonnes. Ces arguments définissent l'alignement `l` (left), `c` (center), `r` (right). On tape les lignes du tableau en séparant les colonnes par `&`. Chaque ligne est terminée par la commande `\\` . Si on écrit la commande `\hline` après une fin de ligne, cela affichera un trait de séparation horizontal, Pour les traits de séparation verticaux, utiliser `|` dans l'argument.
- “Exception” : pour mettre une partie de texte en italique, on écrit `\em ...` , en gras `\bf ...` .

2.8 L'environnement mathématique

2.8.1 Le mode mathématique

Dans le corps d'un texte, les formules mathématiques sont délimitées par un dollar, alors que les formules devant apparaître sur une ligne séparée sont délimitées par deux dollars. On tape par exemple :

Considérons les équations $x+y=0$ et $x-y=2$.

et on obtient :

Considérons les équations $x + y = 0$ et $x - y = 2$ alors que si on tape :

Considérons les équations $\begin{cases} x+y=0 \\ x-y=2 \end{cases}$

on obtient (la commande `\mbox` permet d'écrire du texte dans une formule) :

Considérons les équations

$$x + y = 0 \text{ et } x - y = 2$$

On peut aussi obtenir une équation numérotée avec l'environnement `equation` :

```
\begin{equation} \label{eq:def_x}
x = \sqrt{y+z}
\end{equation}
```

ce qui donne :

$$x = \sqrt{y + z} \quad (1)$$

2.8.2 Les fractions

Une fraction s'obtient avec la commande `\frac{\overline{...}}` (surligne)

$$\frac{x}{2y}=0.4\overline{230769}$$

donne :

$$\frac{x}{2y} = 0.4\overline{230769}$$

2.8.3 Les indices, les exposants et les flèches de vecteurs

Les indices s'obtiennent avec le caractère `_`, les exposants avec le caractère `^` et les flèches de vecteurs avec la commande `\overrightarrow`

Exemple :

```
\[x_1=(a^2+b^2)^{\frac{1}{2}}\]  
\[\overrightarrow{OA_{1,i}}=x^{2t}\cdot \overrightarrow{OB_i}\]
```

donne :

$$x_1 = (a^2 + b^2)^{\frac{1}{2}}$$
$$\overrightarrow{OA_{1,i}} = x^{2t} \cdot \overrightarrow{OB_i}$$

2.8.4 Les racines

Une racine s'obtient avec la commande : `\sqrt`

```
\[\sqrt{x^2+1}\]  
\[\sqrt[3]{x^2+1}\]
```

donne :

$$\sqrt{x^2 + 1}$$
$$\sqrt[3]{x^2 + 1}$$

2.8.5 Les limites

Une limite s'obtient avec la commande : `\lim { ... }`

Pour écrire les fonctions mathématiques on les fait précéder de `\`

On tape :

```
\[\lim_{x \rightarrow +\infty} \ln(x) = +\infty\]
```

pour obtenir :

$$\lim_{x \rightarrow +\infty} \ln(x) = +\infty$$

2.8.6 Les matrices

```
\[\left(\begin{array}{ccc}  
2 & 3 & 4 \\ x & x^2 & x^3 \\ 5 & 6 & 7 \end{array}\right)\]
```

donne

$$\begin{pmatrix} 2 & 3 & 4 \\ x & x^2 & x^3 \\ 5 & 6 & 7 \end{pmatrix}$$

2.8.7 Les intégrales et les séries

```
\[\int_a^b f(t) \, dt\]
```

donne

$$\int_a^b f(t) \, dt$$

$\sum_{i=0}^{+\infty} \frac{1}{i^2}$ et $\sum_{i=0}^{+\infty} \frac{1}{i^2}$

donne : $\sum_{i=0}^{+\infty} \frac{1}{i^2}$ et

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

2.8.8 Les dérivées

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

`\[f'(x)=(\exp(2x))^{\prime}=2\exp(2x)\]`

donne

$$f'(x) = (\exp(2x))' = 2 \exp(2x)$$

Pour la dérivée seconde, utiliser `f''`. Pour les dérivées partielles on utilise `\partial` :

`\[\frac{\partial f(x,y)}{\partial x}=2\exp(2x)\]`

donne

$$\frac{\partial f(x,y)}{\partial x} = 2 \exp(2x)$$

2.9 Insérer un graphique

On peut insérer une image au format EPS (encapsulated postscript) dans un source \LaTeX de la manière suivante :

`\includegraphics[width=\textwidth]{image}`

où `image` désigne le nom du fichier `image.eps` (au format Encapsulated Postscript). On peut aussi indiquer une largeur en centimètres après `width=`. Il faut avoir déclaré en tête (avant `\begin{document}`) du fichier source :

`\usepackage{graphicx}`

2.10 Pgiac

`pgiac` est un programme qui permet de faire calculer automatiquement par Giac/Xcas (un logiciel de calcul formel) certaines expressions d'un fichier source au format \LaTeX . Voir le site de J.Michel Sarlat pour des exemples

<http://melusine.eu.org/syracuse/giac/>

2.11 texmacs

`texmacs` est un programme permettant de saisir des documents mathématiques avec une interface similaire à celle des logiciels de traitement de texte usuels tout en conservant une qualité typographique comparable à \LaTeX . Il permet d'importer et d'exporter au format \LaTeX . Il possède également une interface pour lancer certains logiciel de calcul (Menu Insérer, sous-menu session). Pour lancer `texmacs` sous Unix, tapez la commande :

`texmacs &`

Pour ouvrir `texmacs` avec une session de calcul Giac/Xcas, taper :

`xgiac &`

2.12 Pour en savoir plus

- http://fr.wikibooks.org/wiki/Programmation_LaTeX
- Le L^AT_EX navigator : <http://tex.loria.fr/index.html>
- le groupe AmiTeX <http://fr.groups.yahoo.com/group/AmiTeX/>
- Le site de texmacs : www.texmacs.org

2.13 Référence.

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

TAB. 1 – Greek Letters

\pm	<code>\pm</code>	\cap	<code>\cap</code>	\diamond	<code>\diamond</code>	\oplus	<code>\oplus</code>
\mp	<code>\mp</code>	\cup	<code>\cup</code>	Δ	<code>\Delta</code>	\ominus	<code>\ominus</code>
\times	<code>\times</code>	\uplus	<code>\uplus</code>	∇	<code>\nabla</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^b	<code>\lhd</code>	\bigcirc	<code>\bigcirc</code>
\circ	<code>\circ</code>	\wedge	<code>\wedge</code>	\rhd^b	<code>\rhd</code>	\dagger	<code>\dagger</code>
\bullet	<code>\bullet</code>	\setminus	<code>\setminus</code>	\unlhd^b	<code>\unlhd</code>	\ddagger	<code>\ddagger</code>
\cdot	<code>\cdot</code>	\wr	<code>\wr</code>	\unrhd^b	<code>\unrhd</code>	\amalg	<code>\amalg</code>
$+$	<code>+</code>	$-$	<code>-</code>				

^b Not predefined in a format based on `basefont.tex`. Use one of the style options `oldfont`, `newfont`, `amssymb` or `amssymb`.

TAB. 2 – Binary Operation Symbols

\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>	\succeq	<code>\succeq</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^b	<code>\Join^b</code>
\sqsubset^b	<code>\sqsubset^b</code>	\sqsupset^b	<code>\sqsupset^b</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>
$:$	<code>:</code>						

^b Not predefined in a format based on `basefont.tex`. Use one of the style options `oldfont`, `newfont`, `amsfonts` or `amssymb`.

TAB. 3 – Relation Symbols

$,$	<code>,</code>	$;$	<code>;</code>	$:$	<code>\colon</code>	\cdot	<code>\ldotp</code>	\cdot	<code>\cdotp</code>
-----	----------------	-----	----------------	-----	---------------------	---------	---------------------	---------	---------------------

TAB. 4 – Punctuation Symbols

\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>
\rightleftharpoons	<code>\rightleftharpoons</code>	\leadsto^b	<code>\leadsto^b</code>		

^b Not predefined in a format based on `basefont.tex`. Use one of the style options `oldfont`, `newfont`, `amsfonts` or `amssymb`.

TAB. 5 – Arrow Symbols

\ldots	<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>	\Box^b	<code>\Box^b</code>
\imath	<code>\imath</code>	∇	<code>\nabla</code>	\neg	<code>\neg</code>	\Diamond^b	<code>\Diamond^b</code>
\jmath	<code>\jmath</code>	\surd	<code>\surd</code>	\flat	<code>\flat</code>	\triangle	<code>\triangle</code>
ℓ	<code>\ell</code>	\top	<code>\top</code>	\natural	<code>\natural</code>	\clubsuit	<code>\clubsuit</code>
\wp	<code>\wp</code>	\bot	<code>\bot</code>	\sharp	<code>\sharp</code>	\diamondsuit	<code>\diamondsuit</code>
\Re	<code>\Re</code>	$\ $	<code>\ </code>	\backslash	<code>\backslash</code>	\heartsuit	<code>\heartsuit</code>
\Im	<code>\Im</code>	\angle	<code>\angle</code>	∂	<code>\partial</code>	\spadesuit	<code>\spadesuit</code>
\mho^b	<code>\mho^b</code>	\cdot	<code>\cdot</code>	$ $	<code> </code>		

^b Not predefined in a format based on `basefont.tex`. Use one of the style options `oldfont`, `newfont`, `amsfonts` or `amssymb`.

TAB. 6 – Miscellaneous Symbols

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

TAB. 7 – Variable-sized Symbols

<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>

TAB. 8 – Log-like Symbols

$($	$($	$)$	$)$	\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>	$/$	<code>/</code>	\backslash	<code>\backslash</code>
$ $	<code> </code>	$\ $	<code>\ </code>				

TAB. 9 – Delimiters

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

TAB. 10 – Large Delimiters

\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>

TAB. 11 – Math mode accents

\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>

TAB. 12 – Some other constructions