

The package `paresse`^{*†}

Le T_EXnicien de surface
le.texnicien.de.surface@wanadoo.fr

August 13, 2008

Résumé

Ce module, reprenant un exemple de T. LACHAND-ROBERT dans [1], fournit un moyen de taper des lettres grecques isolées à l'aide du caractère actif et redéfini. Au lieu de `\(\alpha\)` ou tape `$a` pour obtenir α .

Important : Il doit être chargé **après** `inputenc` si ce dernier est utilisé. De plus, il faut que le signe § soit une lettre pour T_EX.

La **documentation française** de cette extension est `paresse-fr.pdf`. Elle contient le code commenté.

Abstract

This package implements an example from T. LACHAND-ROBERT in [1]. It provides a means of typing isolated greek letters with the character § activated and redefined. Instead of `\(\alpha\)` one types `$a` to obtain α .

Important: You have to load it **after** the `inputenc` package if the latter is used. Moreover the sign § must be a letter for T_EX.

The code is not commented in English any more, see the French documentation for French commented code. Sorry.

Contents

1	Introduction	2
2	Usage	2
2.1	Options	2
2.2	Commands and environment	3
2.3	Table of the §-macros	3
	Bibliography	3

^{*}This document corresponds to the file `paresse.sty` v2, dated 2008/08/13, 50th anniversary edition.

[†]English translation by the author. Any comment about the translation is welcome.

1 Introduction

This is the English version of the documentation of the `paresse` package.

This package provides only a ‘quick and low-cost’ access to greek letters which one can obtain with a macro such as `\alpha` or `\Omega`. It provides also an environment and a macro which make possible the use of `§` to type in those letters. Because of an `\ensuremath` we are not bound to explicitly enter —i.e. by typing `$ $` or `\(\)` or else `\[\]` or anything whatsoever with the same effect— mathematics mode to obtain a greek letter.

The idea of the method is from T. LACHAND-ROBERT and described in [1]. I have just add the `\ensuremath` which is so agreeable to write macros.

There is *no* macros for the lowercase omicron nor for the uppercase alpha, beta... that one can obtain with the latin roman letters with the same look. I have not had the courage nor the strength to build a solution which would provide a means of obtaining an upright uppercase alpha in a math formula embedded in an italic boldfaced text.

Even if the meaning of the French ‘paresse’ is just ‘lazyness’ I would like to emphasize that the name of this package comes from the fact that the sign `§` can be used to point at a paragraph and looks like an S. So there is no connection between the name and the not unfrequent sin of the same (French) name... or maybe...

2 Usage

One loads the package with `\usepackage{paresse}` **after** the package `inputenc`. The sign `§` must be recognised as a letter by `TEX`. One can use for instance `inputenc` with option `latin1` for such a purpose.

By default the package is loaded with option `wild` and so the macros such as `§a` are immediately available. If one prefers one can choose the option `tame` by writing `\usepackage[tame]{paresse}`. One must then use the command `\ActiveLaParsse` or the environment `ParsseActive` to use the ‘§-macros’.

When ‘paresse’ is active, one has just to type `§a` in to obtain α . One has access, by the same means, to all the other greek letters to which a macro is devoted such as `\alpha`, see the table page 3. One obtains α^β with `\(§a~{§b}\)` when `§` is active. One will note that, if the package `amsmath` is loaded, the curly braces are *not* compulsory and that one obtains the same result with just `\(§a~§b\)`.

2.1 Options

- `tame` is the contrary of `wild` which is the option by default. When `tame` reigns, one **must** use an environment `ParsseActive` or a command `\ActiveLaParsse` in order to use the §-macros.
- `ttau` is the contrary of `ttheta` which is selected by default. When `ttheta` is active `§t` gives θ in the contrary `§t` gives τ . In all cases, θ is given by `§v` and τ by `§y`.

Remark: Θ is ‘regularly’ obtained with `§V` and *also* with `§T` whatever is the chosen option.

- `epsilon` is the contrary of `varepsilon` which is selected by default. With `epsilon`, `\Se` gives ϵ otherwise `\Se` gives ε .
- The following ‘couples’ behave as `epsilon`, `varepsilon`: `theta` and `vartheta`; `pi` and `varpi`; `rho` and `varrho`; `sigma` and `varsigma`; `phi` and `varphi`.

The default options are `varepsilon`, `theta`, `pi`, `rho`, `sigma`, `varphi` and `wild`.

2.2 Commands and environment

<code>\makeparesseletter</code>	This command gives the letter-catcode to the ‘character’ <code>\S</code> . After that one can use <code>\S</code> in the name of a macro, for instance. It corresponds to the well-known <code>\makeatletter</code> .
<code>\makeparesseother</code>	This macro gives the catcode <i>other</i> to the character <code>\S</code> . It is the ‘contrary’ of the preceding one. It corresponds to <code>\makeatother</code> .
<code>\ActiveLaParsse</code>	This macro makes <code>\S</code> active and thus enable one to access the macros the name of which begins with <code>\S</code> such as <code>\Sa</code> . A list of these macros and theirs meanings is given in the table 3.
<code>ParsseActive</code>	In this environment <code>\S</code> is active and one can use the <code>\S</code> -macros. One could use this environment if one want to use the <code>\S</code> -macros when the package <code>parsse.sty</code> is loaded whith the option <code>tame</code> .

2.3 Table of the `\S`-macros

<code>\Sa</code>	α	<code>\Sb</code>	β	<code>\Sg</code>	γ	<code>\Sd</code>	δ
<code>\Se</code>	ε	<code>\Sz</code>	ζ	<code>\Sh</code>	η	<code>\Sv</code>	θ
<code>\Si</code>	ι	<code>\Sk</code>	κ	<code>\Sl</code>	λ	<code>\Sm</code>	μ
<code>\Sn</code>	ν	<code>\Sx</code>	ξ	<code>\Sp</code>	π	<code>\Sr</code>	ρ
<code>\Ss</code>	σ	<code>\Sy</code>	τ	<code>\Su</code>	υ	<code>\Sf</code>	φ
<code>\Sc</code>	χ	<code>\Sq</code>	ψ	<code>\Sw</code>	ω		
<code>\SG</code>	Γ	<code>\SD</code>	Δ	<code>\SV</code>	Θ	<code>\SL</code>	Λ
<code>\SX</code>	Ξ	<code>\SP</code>	Π	<code>\SS</code>	Σ	<code>\SU</code>	Υ
<code>\SF</code>	Φ	<code>\SQ</code>	Ψ	<code>\SW</code>	Ω		

Remarks : all the latin letters used in the name of the `\S`-macros, but for θ , τ and ψ , are loaded with reminiscences, I hope :-)) and the greek uppercases are obtained with the (latin) corresponding uppercases.

References

- [1] T. LACHAND-ROBERT. *La maîtrise de T_EX et L^AT_EX*. Masson, Paris, Milan, Barcelone, 1995.
ISBN : 2-225-84832-7.