

Escritura de textos en \LaTeX para humanidades

Clase 3

Santiago Gualchi
santiagogualchi@gmail.com

miércoles 15 de mayo de 2019

Contenidos

1. Modo matemático

Qué es, cómo usarlo, letras griegas, símbolos, subíndices y superíndices

2. Ítems y enumeraciones

Listas ordenadas y no ordenadas

3. Entorno `description`

Listas de definiciones

4. Manejo de ejemplos

Inserción de ejemplos numerados, paquetes `gb4e` y `expex`

Modo matemático

Qué es el modo matemático

El modo matemático es uno de los mayores fuertes de \LaTeX y es, en parte, lo que lo llevó a ser ampliamente adoptado por la comunidad científica. El modo matemático nos ofrece la posibilidad de insertar fórmulas y otros elementos altamente formateados y de forma flexible.

Algunos ejemplos

- ❖ Fórmulas matemáticas:

$$H(X) = \mathbb{E}_X[I(x)] = - \sum_{x \in \mathbb{X}} p(x) \log p(x)$$

- ❖ Matrices:

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix}$$

Algunos ejemplos

- ❖ Pruebas lógicas:

$$\frac{\frac{\Gamma \vdash A \quad A, B \vdash C}{\Gamma, B \vdash C}}{\Gamma \vdash B \rightarrow C}$$

- ❖ Derivaciones semánticas:

$$\llbracket \text{TP} \rrbracket = \llbracket \text{Pas} \rrbracket^{a,t} (\llbracket \text{VP} \rrbracket_{\check{c}}^a)$$

Algunos ejemplos

- ❖ Caracteres griegos:

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

- ❖ Otros caracteres especiales:

$\leq, \not\leq, \sqsubset, \therefore, \approx, \cap, \exists, \nexists, \rightarrow, \perp, \emptyset$

Diferencias con el modo texto

- ❖ Algunos comandos funcionan para un modo sí y para el otro no.
- ❖ El espaciado en modo matemático es automático.
- ❖ En modo matemático, las letras se consideran variables por defecto y se formatean de tal modo.

Cómo ingresar al modo matemático

- ❖ *Text style*: introducimos el contenido dentro del párrafo entre signos \$.

La primera letra del alfabeto griego es α y la segunda es β .

La primera letra del alfabeto griego es α y la segunda es β .

Cómo ingresar al modo matemático

- ❖ *Display style*: encerramos las fórmulas entre dos pares de signos \$.

El tercer axioma establece que la relación de dominancia es transitiva:

```
$$(\forall x y z \in N)[((x \triangleleft^* y) \wedge (y \triangleleft^* z)) \rightarrow (x \triangleleft^* z)].$$
```

El tercer axioma establece que la relación de dominancia es transitiva:

$$(\forall xyz \in N)[((x \triangleleft^* y) \wedge (y \triangleleft^* z)) \rightarrow (x \triangleleft^* z)].$$

Letras griegas

Comando		Comando		Comando		Comando	
α	<code>\alpha</code>	A	<code>A</code>	ν	<code>\nu</code>	N	<code>N</code>
β	<code>\beta</code>	B	<code>B</code>	ξ	<code>\xi</code>	Ξ	<code>\Xi</code>
γ	<code>\gamma</code>	Γ	<code>\Gamma</code>	o	<code>o</code>	O	<code>O</code>
δ	<code>\delta</code>	Δ	<code>\Delta</code>	π	<code>\pi</code>	Π	<code>\Pi</code>
ϵ	<code>\epsilon</code>	E	<code>E</code>	ρ	<code>\rho</code>	P	<code>P</code>
ζ	<code>\zeta</code>	Z	<code>Z</code>	σ	<code>\sigma</code>	Σ	<code>\Sigma</code>
η	<code>\eta</code>	H	<code>H</code>	τ	<code>\tau</code>	T	<code>T</code>
θ	<code>\theta</code>	Θ	<code>\Theta</code>	v	<code>\upsilon</code>	Υ	<code>\Upsilon</code>
ι	<code>\iota</code>	I	<code>I</code>	ϕ	<code>\phi</code>	Φ	<code>\Phi</code>
κ	<code>\kappa</code>	K	<code>K</code>	χ	<code>\chi</code>	X	<code>X</code>
λ	<code>\lambda</code>	Λ	<code>\Lambda</code>	ψ	<code>\psi</code>	Ψ	<code>\Psi</code>
μ	<code>\mu</code>	M	<code>M</code>	ω	<code>\omega</code>	Ω	<code>\Omega</code>

Otros símbolos

	Comando		Comando
\leftarrow	<code>\leftarrow</code>	\Leftarrow	<code>\Leftarrow</code>
\rightarrow	<code>\rightarrow</code>	\Rightarrow	<code>\Rightarrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\Leftrightarrow	<code>\Leftrightarrow</code>
\uparrow	<code>\uparrow</code>	\Uparrow	<code>\Uparrow</code>
\downarrow	<code>\downarrow</code>	\Downarrow	<code>\Downarrow</code>
\forall	<code>\forall</code>	\neg	<code>\neg</code>
\exists	<code>\exists</code>	\nexists	<code>\nexists</code>
$=$	<code>=</code>	\neq	<code>\neq</code>
\in	<code>\in</code>	\notin	<code>\notin</code>
\subset	<code>\subset</code>	$\not\subset$	<code>\not\subset</code>
\subseteq	<code>\subseteq</code>	$\not\subseteq$	<code>\not\subseteq</code>
\sqsubset	<code>\sqsubset</code>	$\not\sqsubset$	<code>\not\sqsubset</code>

Superíndices y subíndices

- Podemos poner superíndices en modo matemático con \wedge .

```
$$$N(t) = e^{-\lambda t + C} = e^{-\lambda t} \cdot e^C$$$
```

$$N(t) = e^{-\lambda t + C} = e^{-\lambda t} \cdot e^C$$

- Podemos poner subíndices en modo matemático con $_$.

```
Juan$_{i}$ salió a pasear con su$_{i}$ perro.
```

```
Juani salió a pasear con sui perro.
```

Superíndices y subíndices

Recién vimos cómo poner superíndices y subíndices en modo matemático. Pero también podemos hacerlo en modo texto con los comandos `\textsuperscript` y `\textsubscript`. Compárense los siguientes fragmentos de código:

```
Él$_{Joaquín}$ salió a pasear con su$_{Joaquín}$  
perro.
```

Él_{Joaquín} salió a pasear con su_{Joaquín} perro.

```
Él\textsubscript{Joaquín} salió a pasear con  
su\textsubscript{Joaquín} perro.
```

Él_{Joaquín} salió a pasear con su_{Joaquín} perro.

Ítems y enumeraciones

Listas no ordenadas

Podemos definir listas no ordenadas usando el entorno `itemize` y el comando `\item`:

```
\begin{itemize}
  \item Este es un ítem.
  \item Este es otro ítem.
\end{itemize}
```

- ❖ Este es un ítem.
- ❖ Este es otro ítem.

Listas ordenadas

Podemos definir listas ordenadas usando el entorno `enumerate` y el comando `\item`:

```
\begin{enumerate}
  \item Este es el primer ítem.
  \item Este es el segundo ítem.
\end{enumerate}
```

1. Este es el primer ítem.
2. Este es el segundo ítem.

Listas anidadas

Podemos anidar listas dentro de otras listas definiendo un entorno dentro de otro:

```
\begin{enumerate}
  \item Este es el primer ítem.
  \begin{itemize}
    \item Este es un ítem anidado.
    \item Este es otro ítem anidado.
  \end{itemize}
\end{enumerate}
```

1. Este es el primer ítem.
 - ❖ Este es un ítem anidado.
 - ❖ Este es otro ítem anidado.

Entorno description

Qué es el entorno `description`

El entorno `description` sirve para armar listas de definiciones. Funciona igual que `itemize` y `enumerate`, pero vamos a definir un parámetro optativo con el comando `\item`.

Un ejemplo

Tipos de lenguajes según su complejidad:

```
\begin{description}
  \item[Tipo 3] Lenguajes regulares
  \item[Tipo 2] Lenguajes independientes al
    contexto
  \item[Tipo 1] Lenguajes sensibles al contexto
  \item[Tipo 0] Lenguajes irrestrictos
\end{description}
```

Tipos de lenguajes según su complejidad:

Tipo 3 Lenguajes regulares

Tipo 2 Lenguajes independientes al contexto

Tipo 1 Lenguajes sensibles al contexto

Tipo 0 Lenguajes irrestrictos

Manejo de ejemplos

Paquetes

Podemos elegir entre dos paquetes:

- ▣ gb4e

- ▣ expex

Recordemos que vamos a tener que importarlos:

```
\usepackage{gb4e}
```

o

```
\usepackage{expex}
```

El entorno `exe` con `gb4e`

El primer paso consiste en definir un entorno `exe`. Luego, podemos insertar ejemplos con el comando `\ex` (en forma análoga a como definíamos listas).

```
\begin{exe}
  \ex El chico sonrió.
  \ex El auto es rojo.
\end{exe}
```

- (1) El chico sonrió.
- (2) El auto es rojo.

Información de juicios con gb4e

Podemos añadir información de juicios entre corchetes. Para ello vamos a necesitar encerrar los ejemplos entre llaves. Haciéndolo de esta forma, gb4e se encarga de que los ejemplos queden alineados.

```
\begin{exe}  
  \ex[] {Lo obligó a trabajar hasta las cuatro.}  
  \ex[*] {Lo obligó a ser agredido a las  
          cuatro.}  
\end{exe}
```

- (3) Lo obligó a trabajar hasta las cuatro.
- (4) * Lo obligó a ser agredido a las cuatro.

Ejemplos anidados con gb4e

Podemos anidar ejemplos definiendo entornos `xlist`.

```
\begin{exe}  
  \ex Guardé el informe sin revisarlo.  
  \ex \begin{xlist}  
    \ex El informe que guardé sin haber  
      revisado.  
    \ex El informe que guardé sin haberlo  
      revisado.  
  \end{xlist}  
\end{exe}
```

- (5) Guardé el informe sin revisarlo.
- (6)
 - a. El informe que guardé sin haber revisado.
 - b. El informe que guardé sin haberlo revisado.

Referencias en gb4e

Para referir a ejemplos en el cuerpo del documento vamos a usar el comando `\label` para etiquetar el ejemplo y el comando `\ref` para nombrarlo.

Por ejemplo, es obvio que la oración (`\ref{cart}`) es gramatical.

```
\begin{exe}
  \ex \label{cart} El cartero entregó un
      paquete a mi vecino.
\end{exe}
```

Por ejemplo, es obvio que la oración (7) es gramatical.

(7) El cartero entregó un paquete a mi vecino.

La sintaxis de `expex`

El paquete `expex` nos permite definir ejemplos en forma más abreviada. Para ello, vamos a encerrar cada ejemplo entre los comandos `\pex` y `\xe`.

```
\pex El chico sonrió. \xe  
\pex El auto es rojo. \xe
```

- (8) El chico sonrió.
- (9) El auto es rojo.

Ejemplos anidados en `expex`

Para anidar ejemplos con `expex` vamos a introducir cada subejemplo con el comando `\a`.

```
\pex
  \a El informe que guardé sin haber revisado.
  \a El informe que guardé sin haberlo
      revisado.
\xe
```

- (10) a. El informe que guardé sin haber revisado.
b. El informe que guardé sin haberlo revisado.

Referencias en expex

Dos comandos muy útiles que provee el paquete `expex` son `\nextx` y `\lastx` que permiten insertar referencias al ejemplo siguiente y al anterior, respectivamente.

Ver el ejemplo (`\nextx`).

```
\pex El presentador dio [un sobre] [al ganador  
del concurso]. \xe
```

Como se vio en el ejemplo (`\lastx`), `\dots`

Ver el ejemplo (11).

(11) El presentador dio [un sobre] [al ganador del concurso].

Como se vio en el ejemplo (11), ...

Referencias en `expex`

También podemos definir etiquetas explícitamente encerrándolas entre corchetes angulares seguido de los comandos `\pex` y `\a`. Luego, para referirlos usamos los comandos `\getref` o `\getfullref`. Pero esto solo funciona si no está activado el paquete `babel`. Si necesitamos importar ese paquete, siempre podemos usar los comandos `\label` y `\ref` como hacíamos con `gb4e`.

Referencias en expex

```
\pex<parmin>
  \a<buena> Este es un ejemplo.
  \a<mala> \ljudge{*} Este es un otro ejemplo.
\xe
```

Con `\texttt{expex}` podemos referir de varias formas a los ejemplos. Por ejemplo:

`(\getref{parmin.mala})`, `(\getref{parmin})` o `(\getfullref{parmin.mala})`.

- (12) a. Este es un ejemplo.
- b. * Este es un otro ejemplo.

Con `expex` podemos referir de varias formas a los ejemplos. Por ejemplo: (b), (12) o (12b).

La yapa

La yapa: referencias a secciones

Así como los comandos `\label` y `\ref` nos permitían incluir referencias a ejemplos, también podemos incluir referencias a partes, secciones y otras divisiones.

La yapa: referencias a secciones

En la sección `\ref{ant}` introduciremos los antecedentes y en la sección `\ref{proc}` el procedimiento.

```
\section{Antecedentes} \label{ant}  
...  
\section{Procedimiento} \label{proc}  
...
```

En la sección 2 introduciremos los antecedentes y en la sección 3 el procedimiento.

2. Antecedentes

...

3. Procedimiento

...

Bibliografía consultada

Frampton, J. (2014). ExPex for linguists.

Kolber, H.-P. (s.f.). Preliminary documentation for gb4e.sty and c gloss4e.sty.

Oetiker, T. (2018). *The Not So Short Introduction To $\text{\LaTeX} 2_{\epsilon}$* . Olten.

Wikibooks (2019). LaTeX — Wikibooks, The Free Textbook Project. [Online; acceso 10 de mayo de 2019].