



Edición de textos científicos

# LATEX 2012

$a \xrightarrow{f} b$

Composición

Diseño editorial

Gráficos, Inkscape, TikZ

Presentaciones Beamer

Ubuntu - TeXLive

Windows - MikTeX

$$\lim_{x \rightarrow 0} f(x)$$

$$\begin{pmatrix} a \\ b \end{pmatrix}$$

$$\sum_{\substack{0 < i < m \\ 0 < j < n}} a_i b_j$$

$$\prod_{\substack{i=0 \\ i \neq k}}^n \frac{w_i}{(w_{\textcolor{violet}{i}} - w_{\textcolor{blue}{k}})}$$

Alexánder Borbón A., Walter Mora F.

Instituto Tecnológico de Costa Rica



Revista digital Matemática, Educación e Internet [www.cidse.itcr.ac.cr/revistamate/](http://www.cidse.itcr.ac.cr/revistamate/)

---

EDICIÓN DE TEXTOS CIENTÍFICOS

# LAT<sub>E</sub>X

2da edición, 2012.

## Composición, Diseño Editorial, Gráficos, Inkscape, Tikz y Presentaciones Beamer

---

**Walter Mora F.,  
Alexánder Borbón A.**

Escuela de Matemática  
Instituto Tecnológico de Costa Rica.  
([www.tec-digital.itcr.ac.cr/revistamatematica/](http://www.tec-digital.itcr.ac.cr/revistamatematica/))



Este libro se distribuye bajo la licencia Creative Commons Reconocimiento - No Comercial - Sin obra derivada 3.0 Unported License. Esta licencia permite copiado y distribución gratuita, pero no permite venta ni modificaciones de este material. Ver <http://creativecommons.org/>.

Límite de responsabilidad y exención de garantía: El autor o los autores han hecho su mejor esfuerzo en la preparación de este material. Esta edición se proporciona "tal cual". Se distribuye gratuitamente con la esperanza de que sea útil, pero sin ninguna garantía expresa o implícita respecto a la exactitud o completitud del contenido.

La Revista digital Matemáticas, Educación e Internet es una publicación electrónica. El material publicado en ella expresa la opinión de sus autores y no necesariamente la opinión de la revista ni la del Instituto Tecnológico de Costa Rica.

Copyright© Revista digital Matemática Educación e Internet ([www.cidse.itcr.ac.cr/revistamate](http://www.cidse.itcr.ac.cr/revistamate)). Segunda Edición.

Correo Electrónico: [wmora2@gmail.com](mailto:wmora2@gmail.com)

Escuela de Matemática

Instituto Tecnológico de Costa Rica

Apdo. 159-7050, Cartago

Teléfono (506)25502225

Fax (506)25502493

Mora Flores, Walter.

Edición de Textos Científicos con L<sup>A</sup>T<sub>E</sub>X. Composición, Gráficos, Inkscape y Presentaciones Beamer/Walter Mora F. Alexander Borbón A. – 2da ed.

– Escuela de Matemática, Instituto Tecnológico de Costa Rica. 2012.

173 p.

ISBN 978-9977-66-227-5

1. TeX. 2. Composición tipográfica-automatizada 3. Tipos - símbolos matemáticos.

# Contenido

---

Prefacio	vii
<b>1    <math>\text{\LaTeX}</math>: Primeros pasos</b>	<b>1</b>
1.1    ¿Qué es $\text{\LaTeX}$ ?	1
1.2    Distribuciones $\text{T\kern-1.61em\kern.14emEX}/\text{\LaTeX}$ y editores.	2
1.2.1    Distribuciones $\text{T\kern-1.61em\kern.14emEX}/\text{\LaTeX}$	3
1.2.2    Un Editor	3
<b>2    Acciones en una sesión con <math>\text{\LaTeX}</math></b>	<b>4</b>
2.1    Editar, compilar y ver el resultado.	4
2.2    Convertir documentos a $\text{\LaTeX}$	7
2.3    Detalles del preámbulo y el cuerpo	7
2.3.1    Idioma	10
2.4    Tipos y tamaños de fuentes.	12
2.4.1    Caracteres especiales.	12
2.4.2    Algunos tipos de fuentes (fonts).	12
2.4.3    Tamaños de letras.	13
2.5    (*) Las fuentes y sus atributos	15
2.6    Párrafos y efectos especiales.	18
2.6.1    Centrar	19
2.6.2    Cajas	19
2.6.3    Texto en columnas: <code>multicol</code> , <code>minipage</code> y <code>parbox</code> .	20
2.7    Color, cajas y líneas.	22
2.7.1    Otros efectos de texto	24
2.7.2    Texto como en la pantalla	25
2.8    Notas al pie de página.	26
2.8.1    Espacio horizontal y vertical	27
2.8.2    Notas en el margen	28
2.9    Enumerado automático.	28
2.10    Enumeración usando el paquete <code>TikZ</code> .	31

2.11	Título, contenido, secciones y bibliografía	32
2.12	Modulación	34
<b>3</b>	<b>Texto en modo matemático</b>	<b>35</b>
3.1	Potencias, subíndices y superíndices	36
3.2	Tamaño natural	36
3.3	Raíces	37
3.4	Fracciones y expresiones de dos niveles	37
3.5	Tres puntos consecutivos	39
3.6	Delimitadores	39
3.7	LLaves y barras horizontales	41
3.8	Acentos y “sombreados” en modo matemático	42
3.9	Negritas en modo matemático	42
3.10	Espacio en modo matemático	42
3.11	Centrado	43
3.12	Entorno <b>equation</b> . Contadores automáticos	43
3.13	Arreglos	44
3.14	Matrices	46
3.15	Alineamiento	47
3.16	Tablas de símbolos matemáticos frecuentes	51
3.16.1	Letras griegas	51
3.16.2	Operadores binarios	51
3.16.3	Relaciones	51
3.16.4	Negación de relaciones	51
3.16.5	Otros símbolos	52
3.16.6	Especiales	52
3.16.7	Símbolos del paquete <i>amssymb</i>	52
3.17	Cómo hacer nuevos Comandos.	52
3.18	Comandos con opciones	55
<b>4</b>	<b>Tablas</b>	<b>56</b>
4.1	Los ambientes <i>figure</i> y <i>table</i>	56
4.2	Fuentes en tabular.	57
4.3	Color en tablas.	58
4.4	Rotación de texto en celdas.	61
4.5	Unir celdas.	62
4.6	Escalar una tabla	62
4.7	Espaciado en celdas.	64
4.8	Ancho de las columnas	65
4.9	Modo matemático en tablas con <b>tabularx</b>	68
4.10	Problemas con los objetos flotantes: Paquete <b>float</b>	68
4.11	Tablas sofisticadas con <b>TikZ</b>	69
<b>5</b>	<b>Insertar gráficos y figuras en documentos L<sup>A</sup>T<sub>E</sub>X</b>	<b>71</b>
5.1	Introducción	71
5.2	¿Cómo insertar las figuras?	71
5.3	Edición adicional de figuras con Inkscape	77
5.4	De nuevo: Paquete <b>float</b>	80
5.5	Paquete <b>subfigure</b>	81
5.6	Los ambientes <b>wrapfigure</b> y <b>floatflt</b>	81

5.7	Crear figuras nativas con <code>TikZ</code>	83
<b>6</b>	<b>Diseño Editorial</b>	<b>88</b>
6.1	Bajar la carga cognitiva	88
6.2	Amenidad: Los Cuatro Principios Básicos	90
6.2.1	Proximidad.	90
6.2.2	Alineamiento.	90
6.2.3	Repetición.	91
6.2.4	Contraste.	91
6.3	Legibilidad: Cómo escoger las fuentes.	91
6.4	Color	93
6.5	Editar un PDF con <code>Inkscape</code>	94
<b>7</b>	<b>Personalizar el Documento</b>	<b>97</b>
7.1	Numeración automática de definiciones, teoremas y ejemplos.	97
7.2	El paquete <code>ntheorem</code>	98
7.3	Personalización de teoremas, definiciones, etc.	99
7.4	Personalización de secciones, subsecciones, etc.	99
7.5	El paquete <code>todonotes</code>	100
7.6	El paquete <code>boibootes</code>	100
7.7	Paquete <code>algorithm2e</code>	102
7.8	Código de lenguajes de programación en color	104
7.8.1	Paquete <code>minted</code>	105
7.8.2	Paquete <code>verbments</code> para TeXLive	105
7.9	Cómo hacer listas de ejercicios con solución	107
7.10	Cabeceras	108
<b>8</b>	<b>Citas bibliográficas consistentes con BibTeX</b>	<b>110</b>
8.1	Entorno <code>thebibliography</code>	110
8.2	BibTeX	111
8.3	JabRef	114
<b>9</b>	<b>Cómo hacer Transparencias con la clase Beamer</b>	<b>116</b>
9.1	Introducción	116
9.2	Un documento Beamer	117
9.3	Marcos (frames)	118
9.4	Velos (overlays)	119
9.4.1	Opciones <code>&lt;i-&gt;</code> y <code>\uncover&lt;i-&gt;</code>	120
9.4.2	Opción <code>&lt;i- alert@ i&gt;</code>	121
9.5	Comando <code>pause</code> .	122
9.6	Entornos para teoremas, definición, etc.	122
9.7	Blocks.	123
9.8	Opción <code>fragile</code>	125
9.9	Entornos para código de programas	125
9.9.1	Entorno <code>semiverbatim</code>	125
9.9.2	Entorno <code>minted</code>	126
9.10	Beamer y el paquete <code>algorithm2e</code>	127
9.11	Gráficos	128
9.12	Ligas y botones.	129

9.13 Efectos de Transición. Color	130
9.14 Ligas a Documentos Externos	132
9.15 Animaciones	133
<b>10 Documentos LaTeX en Internet</b>	<b>134</b>
10.1 LaTeX2HTML Translator	134
10.2 Otra Opción: PDFScreen	135
Bibliografía	137
<b>Apéndice A: Instalar una distribución y un editor</b>	<b>138</b>
A.0.1 Distribuciones T <small>E</small> X	138
A.0.2 Un Editor	139
A.1 Software adicional	142
<b>Apéndice B: Ubuntu</b>	<b>145</b>
B.1 Instalar Ubuntu desde Windows	145

# Prefacio

---

Esta es la versión 2012 del libro. Se han corregido varios errores en el código de los ejemplos, se han creado nuevas secciones y se han ampliado otras, además de reubicar algunas. El texto cubre aspectos básicos e intermedios sobre composición tipográfica  $\text{\LaTeX}$ , diseño editorial, presentaciones Beamer, edición adicional de gráficos y figuras con Inkscape y Tikz.

También se desarrollan tópicos que tienen que ver con paquetes especiales. Algunas veces la descripción se hace “por ejemplos”, dada la bastedad del tema. Los temas que se han incluido son los tópicos más frecuentes en la edición de libros y artículos sobre matemáticas, educación, software y programación, según nuestra experiencia. Incluye nuevos paquetes y nuevos comandos que resuelven problemas cotidianos de edición de textos matemáticos de una manera más sencilla. Ahora se considera [TeXLive - Ubuntu](#) y [MiK\TeX-Windows](#), nuevas cosas en diseño editorial e infografía y una nueva presentación de los ejemplos (basada en TikZ).

La edición 2010 fue implementada en Windows XP con [MiK\TeX 2.8](#) y [Texmaker](#). Esta nueva edición se implementó con la distribución [TeX Live 2011](#) y los editores [TeXmaker 3.2.1](#) y [TeXstudio 2.2](#) (en algunas ocasiones) sobre [Ubuntu 11.10](#). En general, no debería haber problema en usar el motor [MiK\TeX](#) o [TeX Live](#) para compilar el código de los ejemplos (excepto tal vez un par de casos en Windows 7, pero se indica cómo solucionar el problema). Este texto se ha usado en algunos cursos en el Instituto Tecnológico de Costa Rica y se usa frecuentemente en la revista digital de Matemática, Educación e Internet en la edición de artículos y libros.

Agradecemos a todas las personas que nos han ayudado con la lectura detenida del libro, señalando errores en el texto y el código, inconsistencias, sugiriendo nuevas secciones y por todos sus comentarios.

Se puede pedir un machote del código [LaTeX](#) de este libro, solo debe solicitarlo a los autores.

W. MORA, A. BORBÓN.

*Cartago, Costa Rica. Enero 2012.*



# LATEX: PRIMEROS PASOS

## 1.1 ¿Qué es LATEX?

*"TeX is intended for the creation of beautiful books - and especially for books that contain a lot of mathematics".*

Donald Knuth



Donald Knuth, 1938-

El sistema TeX (se pronuncia [tej]) fue diseñado y desarrollado por Donald Knuth en la década del 70. Es un sofisticado programa para la composición tipográfica de textos *científicos* tales como artículos, reportes, libros, etc. TeX es en la práctica un estándar para publicaciones científicas en áreas como matemática, física, computación, etc. LATEX es un conjunto de macros TeX preparado por Leslie Lamport. LATEX no es un procesador de textos, es un lenguaje que nos permite preparar automáticamente un documento de apariencia estándar y de alta calidad. En general, solo necesitamos editar texto y algunos comandos y LATEX se encarga de componer automáticamente el documento.

A diferencia de un procesador de textos, con LATEX tenemos un control más fino sobre cualquier aspecto tipográfico del documento<sup>1</sup>.

LATEX formatea las páginas de acuerdo a la *clase* de documento especificado por el comando `\documentclass{}`, por ejemplo, `\documentclass{book}`.

Un documento LATEX puede tener texto ordinario junto con texto en *modo matemático*. Los comandos vienen precedidos por el símbolo “\” (barra invertida).

Hay comandos que funcionan en *modo texto* y hay comandos que solo funcionan en *modo matemático*. Todo lo que es lenguaje matemático se edita en modo matemático. Hay varios en-

<sup>1</sup>Además de LATEX, existe otra opción, llamada ConTeXt. Este conjunto de macros TeX es menos famosa pero tal vez es más sencilla de usar y ofrece más posibilidades de edición TeX

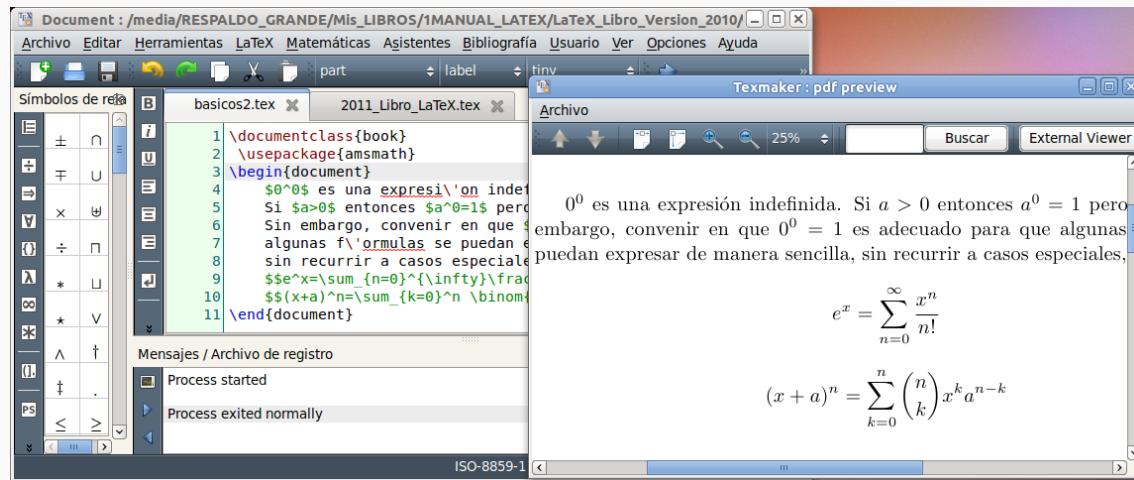
tornos para este modo, el más común es el entorno delimitado por dos signos de dólar ( $\$ \dots \$$ ).

Un ejemplo de código LATEX es el siguiente:

### Ejemplo 1.1

```
\documentclass{book}
\usepackage{amsmath}
\usepackage[T1]{fontenc}
\begin{document}
$0^0$ es una expresión indefinida.
Si $a>0$ entonces $a^0=1$ pero $0^a=0$.
Sin embargo, convenir en que $0^0=1$ es adecuado para que
algunas fórmulas se puedan expresar de manera sencilla,
sin recurrir a casos especiales, por ejemplo
$$e^x=\sum_{n=0}^{\infty}\frac{x^n}{n!}$$
$$(x+a)^n=\sum_{k=0}^n \binom{n}{k} x^k a^{n-k}$$
\end{document}
```

Este código se digita en un editor (en la figura de abajo se usó **Texmaker**) y luego se *compila*. La ventana a la derecha en la figura que sigue, muestra la salida:



## 1.2 Distribuciones TEX/LATEX y editores.

Una distribución **TeX** contiene el núcleo principal del programa, paquetes y extensiones adicionales: Integra todo lo que hace falta para poner a funcionar **TeX** y **LATEX** sobre un sistema operativo. Una vez que instalamos **TeX** usando alguna distribución, es conveniente tener un editor no solo para editar de manera cómodo el texto, también para acceder de manera sencilla a las tareas usuales de una sesión **LATEX**: editar, compilar y visualizar (en DVI o PDF).

MikTeX,  
TeXLive,  
MacTeX

### 1.2.1 Distribuciones TeX/LaTeX

Hay varias distribuciones TeX por ejemplo: TeXLive (Windows, Linux, Mac), Mactex (Mac OS X) y Miktex (Windows). Las pruebas de este libro se hicieron con MiKTeX2.9 (Windows XP, 32 bits) y con TeXLive 2011 (Ubuntu 10.10, 64 bits). Asumimos que el lector tiene la distribución respectiva completa a mano.

Texmaker,  
Texstudio,  
Winshell

### 1.2.2 Un Editor

Después de la instalación de la distribución TeX, instalamos un editor. Hay varios editores: Texmaker, Texstudio, Winshell, Kile, etc. Los editores buscan la instalación TeX/LaTeX de manera automática. Luego se pueden configurar algunas cosas adicionales.

Los detalles relacionados con la instalación de una distribución TeX, la instalación de un editor y la instalación de software de apoyo (Windows y Ubuntu) se puede consultar en el Apéndice A.





# 2

# ACCIONES EN UNA SESIÓN CON LATEX

En una sesión LATEX ejecutamos varias acciones: Ponemos un preámbulo con la clase de documento, paquetes que se van a usar, cosas de maquetación, etc. y editamos el cuerpo del documento, luego compilamos ([LaTeX](#) o [PDFLaTeX](#)) y vemos el resultado en un visor (DVI o PDF).

## 2.1 Editar, compilar y ver el resultado.

- 1 Ponemos un **preámbulo**: La clase de documento, indicaciones sobre márgenes, largo y ancho de página, numeración, etc., y cargamos los paquetes adicionales (fuentes, símbolos, gráficos, etc.).

```
\documentclass{article}
  \textheight    = 20cm
  \textwidth     = 18cm
  \topmargin    = -2cm
  \oddsidemargin= -1cm
  \parindent    = 0mm
  \usepackage{amsmath,amssymb,amsfonts, latexsym}
  \usepackage{graphicx}

\begin{document}
\section{Preliminares}
{ \bf Distancia entre dos puntos.} Recordemos que la distancia euclíadiana
de un punto $A=(a,b)$ a otro punto $B=(p,q)$ es
$ d(A,B)=||A-B||=\sqrt{(a-p)^2+(b-q)^2} $
Sean $A=(1,1)$ y $B=(5,3)$. El punto medio es
$M=\frac{(1+5, 1+3)}{2}=(3,2).$ \\
\end{document}
```

Preámbulo

- 2 **Editamos**: Escribimos texto corriente y texto en *modo matemático* (posiblemente combinando ambos). Mucho del texto en modo matemático es edita en los entornos `$...$` o `$$...$$`. Esto le indica al programa que interprete el texto y lo convierta en símbolos matemáticos.
- 3 **Compilamos**: En el menú del editor está la opción [LaTeX](#) o la opción [PDFLaTeX](#) para compilar. Esto nos permite detectar, por ejemplo, errores en los comandos o en la sintaxis de

una fórmula.

#### ④ Ver el archivo DVI o el PDF

- Una vez que hemos compilado con la opción [LaTeX](#), usamos la opción [DVI](#) para ver el documento (esto hace que un visualizador ejecute el programa 'dvips' para ver el documento en pantalla). Si queremos una versión PDF, usamos la opción [DVI->PDF](#).
  - Una vez que hemos compilado con la opción [PDFLaTeX](#), usamos la opción [Ver PDF](#) para ver el documento.
- ⑤ **Imprimir el archivo DVI.** Formalmente, imprimir la interpretación en formato PostScript (muy fino) del archivo [DVI](#).

Después de compilar se producen varios archivos: \*.tex, \*.dvi (o .pdf), \*.aux, \*.log, \*.toc. El archivo de edición tiene extensión \*.tex mientras que el archivo .log contiene un informe del proceso de compilación. Para imprimir un documento [LaTeX](#) (generado con la opción [LaTeX](#)) solo necesitamos el archivo \*.dvi y los archivos de los gráficos incluidos en el documento (si hubiera).

#### ¿Compilar con PDFLaTeX o LaTeX?

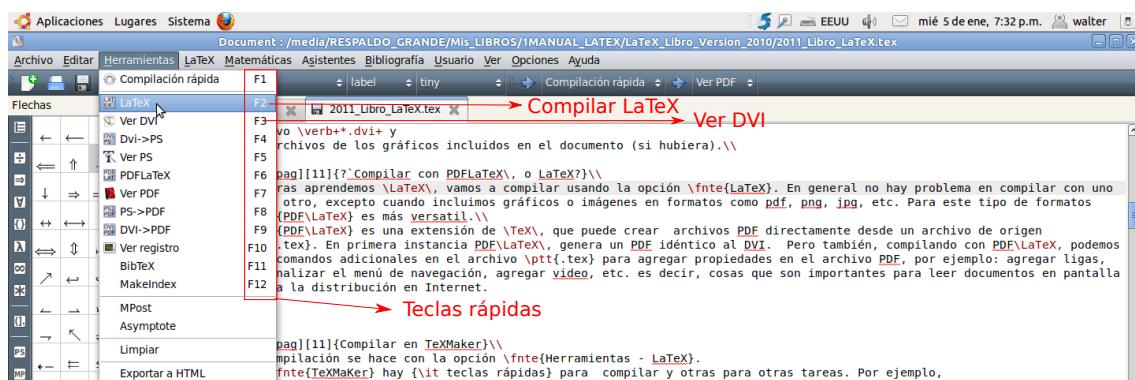
El Formato PDF se ha convertido en uno de los formatos de documentos electrónicos más utilizados para la publicación de documentos en la Web. Hay muchas ventajas que la hicieron muy popular: Es adecuado para la visualización y para imprimir, permiten búsquedas, etc. Pero también, compilando con [PDFLaTeX](#), podemos usar comandos adicionales en el archivo .tex para agregar propiedades en el archivo PDF: Agregar ligas, personalizar el menú de navegación, agregar video, etc., es decir, cosas que son importantes para leer documentos en pantalla y para la distribución en Internet.

[PDFLaTeX](#) es una extensión de [TeX](#) que puede crear archivos PDF directamente desde un archivo de origen .tex. Cuando compilamos con [PDFLaTeX](#), generamos un PDF de igual apariencia que el DVI.

Este libro fue compilado con [PDFLaTeX](#) porque usa ligas a otros documentos y tiene muchas figuras en distintos formatos.

#### Compilar desde el editor TeXMaker

La compilación se hace con la opción [Herramientas - LaTeX](#). En [TeXMaker](#) hay *teclas rápidas* para compilar y otras para otras tareas. Por ejemplo, la compilación [LaTeX](#) se hace con la tecla [F2](#) y el documento DVI se ve con la tecla [F3](#).



## Resultado de la compilación.

Si no hay errores de sintaxis, el mensaje en la ventana inferior sería,

Process exited normally

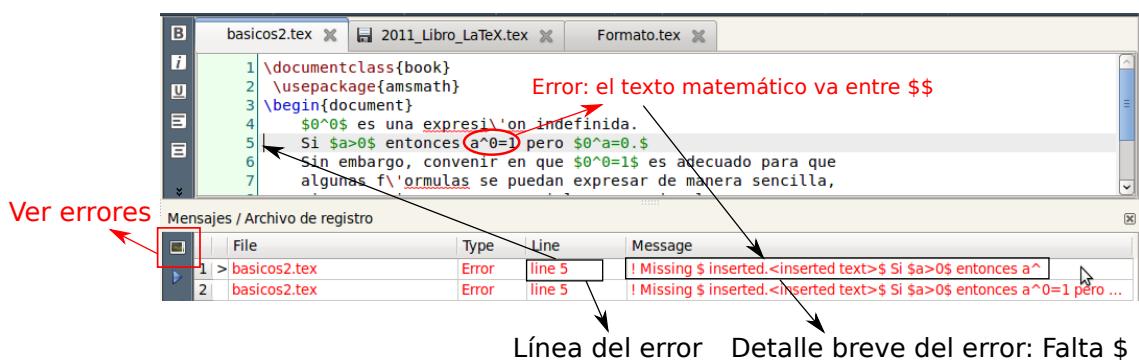
Si hay errores de sintaxis, el mensaje en la ventana inferior sería,

Process exited with error(s)

En este caso, la compilación genera una información de salida en la 'ventana de mensajes' (se habilita o deshabilita en el menú Ver). Si la compilación encuentra algún error se indicará con un mensaje corto (en rojo), por ejemplo

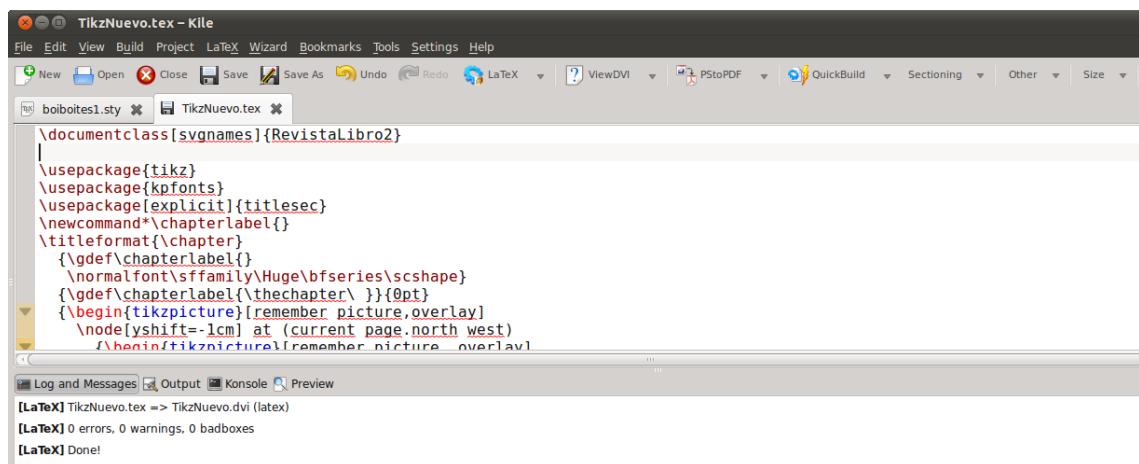
```
Error line 323 ! Missing $ <inserted text>...
```

En este ejemplo, esto nos indica que falta '\$' en la línea 323. Como se ve,  $\frac{x}{x+1}$  es texto matemático (inicia con un comando de fracción) y por tanto no se puede interpretar como texto corriente, necesita estar entre \$ \$. Al hacer clic en 'line 323' nos llevará a la línea del error.



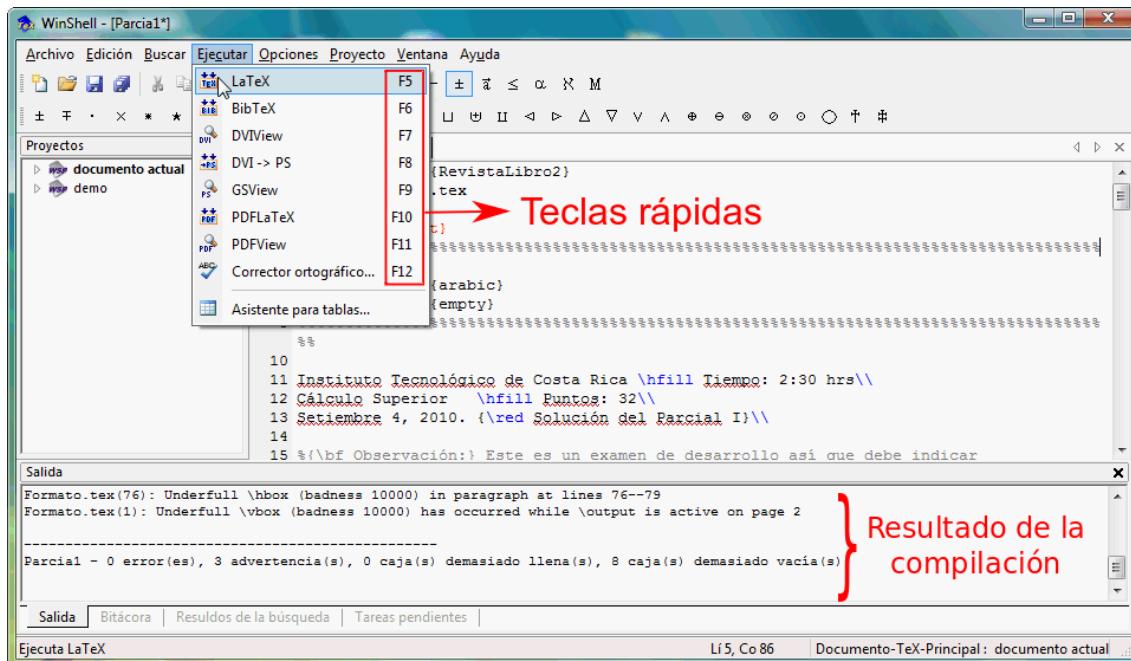
## Compilar desde el editor Kile

Se usa el menú Build-Compile-LaTeX o Build-Compile-PDFLaTeX. Las teclas rápidas son Alt-2 para compilar y Alt-3 para ver el DVI. Igual que en TeXMaker, tenemos una ventana para el resultado de la compilación.



## Compilar desde el editor WinShell

La compilación se hace con la opción Ejecutar - LaTeX. En WinShell hay *teclas rápidas* para compilar y otras para otras tareas. Por ejemplo, la compilación LaTeX se hace con la tecla F5 y el documento DVI se ve con la tecla F7.

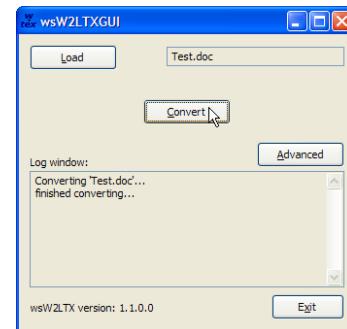


## 2.2 Convertir documentos a L<sup>A</sup>T<sub>E</sub>X

Possiblemente haya interés en convertir documentos (con o sin ecuaciones) de Microsoft Word o de OOoWriter, a documentos L<sup>A</sup>T<sub>E</sub>X. Los programas gratuitos pueden hacer esto con un resultado limitado.

### wsW2LTX (Windows)

Los archivos en formato Word (97-2003) se pueden convertir a L<sup>A</sup>T<sub>E</sub>X (en Windows) usando el programa gratuito wsW2LTX ([http://www.winshell.org/modules/w2ltx\\_download/](http://www.winshell.org/modules/w2ltx_download/)). Se debe descargar el paquete **wsW2LTXSDK** que incluye el programa **wsW2LTXGUI** (figura de la derecha). Para hacer la conversión hay que ejecutar el programa **wsW2LTXGUI** que está en la carpeta bin de este paquete. La conversión es limitada. Hay software no gratuito que hace muy bien el trabajo, por ejemplo **Word2TeX** ( $\approx \$54$ ) y **Word-to-LaTeX** ( $\approx \$44$ ).



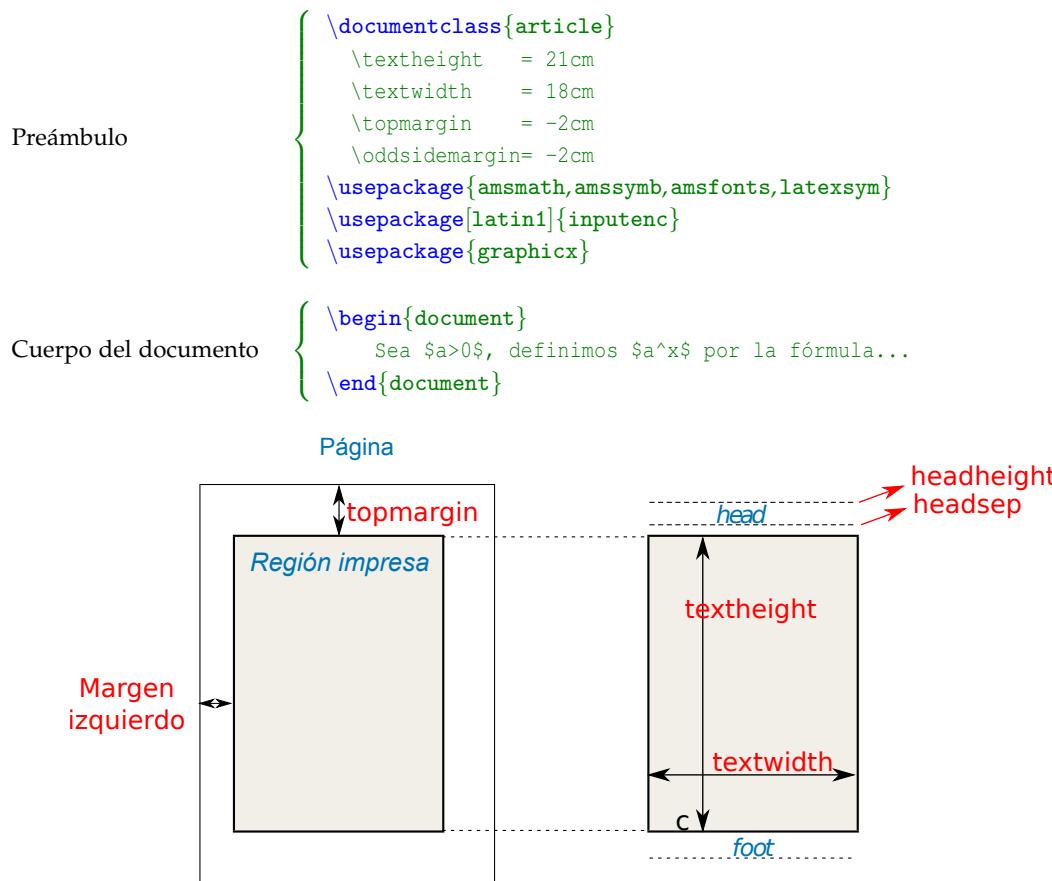
### Writer2LaTeX (Ubuntu)

Writer2LaTeX es una extensión para Writer de OpenOffice.org. Se instala con 'Gestor de Paquetes Synaptic' (se busca **Writer2LaTeX**). Para hacer la conversión se carga el documento .odt (OOoWriter) y se usa la opción **Archivo-Exportar**. La conversión es limitada.

## 2.3 Detalles del preámbulo y el cuerpo

Un documento básico en L<sup>A</sup>T<sub>E</sub>X se compone de dos partes: el *preámbulo* del documento y el *cuerpo*. Al inicio del documento se debe especificar la clase de documento y lo relativo al ajuste de las páginas, nada de lo que pongamos en el preámbulo aparecerá en el documento que se imprime al final. En el cuerpo se escribe el texto (normal y matemático). Es la parte que aparecerá impresa

como producto final.



**Figura 2.1** Diseño del documento.

- `\documentclass{article}`: Es la clase de documento, article se utiliza para editar documentos con formato de artículo. article se refiere al archivo article.cls. Estos archivos .cls implementan la estructura específica de un documento. También se usa “report” o “book” para un reporte o un libro.
- `\textheight=21cm`: Establece el largo del texto en cada página (en este caso, de 21 cm). El default es 19 cm.
- `\textwidth=18cm`: Establece el ancho del texto en cada página (en este caso, de 18 cm). El default es 14 cm.
- `\topmargin=-2cm`: Establece el margen superior. El default es de 3 cm, en este caso la instrucción `-2cm` sube el margen 2 cm hacia arriba.
- `\oddsidemargin=-2cm`: Establece el margen izquierdo de las páginas impares. El default es de 4.5 cm; sin embargo, con sólo poner esta instrucción el margen queda en 2.5 cm. Si el parámetro es positivo se aumenta este margen y si es negativo disminuye. Note que esto combinado con el ancho del texto, determina el ancho del otro margen!
- `\usepackage{amsmath,amssymb,amsfonts, latexsym}`: Esta instrucción indica que en este documento se usarán paquetes de símbolos adicionales (símbolos de la AMS).
- `\usepackage[latin1]{inputenc}`: Esta instrucción se usa para incluir un paquete que nos permite usar los acentos y otros símbolos, directamente del teclado.

- `\usepackage{graphicx}`: Esta instrucción se usa para incluir un paquete para el manejo de gráficos y figuras en el documento.

## Otros comandos para el preámbulo

- `\renewcommand{\baselinestretch}{1.5}`: Genera un texto a espacio y medio. Si se pone 2 en vez de 1.5, lo hace a doble espacio.
- `\pagestyle{empty}`: Elimina la numeración de las páginas.
- `\parskip=Xmm`: Genera un espacio de X mm entre los párrafos.
- `\headheight`: Altura de la cabecera (page header) de la página.
- `\headsep`: Distancia desde la parte inferior de la cabecera al cuerpo de texto en una página.
- `\parindent=0mm`: Elimina la sangría.
- `\pagestyle{myheadings}`: Coloca la numeración de página en la parte superior.
- `\markright{'texto'}`: Coloca 'texto' en la parte superior de la página. Se pueden poner varios `\markright` en el texto (en cada sección, por ejemplo).

Ejemplo: `\markright{\LaTeX \quad \hrulefill W. Mora, A. Borb'on \$\; \$}`

- `\newpage`: Le indica a L<sup>A</sup>T<sub>E</sub>X que siga imprimiendo en la página siguiente.

## Tópico adicional: La manera fácil.

El diseño de documento se puede simplificar con el paquete `geometry`. Por ejemplo, si queremos un documento 18cm×21cm con margen superior de 2cm y margen izquierdo de 2cm, cambiamos nuestras instrucciones agregando en el *preámbulo*

```
\usepackage[total={18cm,21cm},top=2cm, left=2cm]{geometry}
```

es decir,

---

```
\documentclass{article}
\usepackage[total={18cm,21cm},top=2cm, left=2cm]{geometry}
\usepackage{latexsym,amsmath,amssymb,amsfonts}
\usepackage[latin1]{inputenc}
\usepackage{graphicx}
\begin{document}
...
\end{document}
```

---

Un manual de referencia para este paquete se puede encontrar en <ftp://ftp.tex.ac.uk/tex-archive/macros/latex/contrib/geometry/geometry.pdf>.

### 2.3.1 Idioma

El idioma oficial que utiliza LATEX es el *inglés*, sin embargo, utilizando algunas instrucciones se puede lograr que soporte otros idiomas, en particular, veremos cómo hacer para que soporte el español.

#### Acentos y otros caracteres

LATEX normalmente no acepta tildes, ni la “ñ”, tampoco el signo de pregunta ‘¿’, ni la apertura o el cierre de comillas. Para que acepte estos caracteres se deben utilizar las instrucciones que aparecen en la tabla 2.1.

Comando	Símbolo	Comando	Símbolo
\'a	á	?`	¿
\'e	é	!`	¡
\'{\i}	í	`` ``	“ ”
\'o	ó	` ' ` '	‘ ’ ‘ ’
\'u	ú	\~n	ñ

Tabla 2.1 Acentos en modo texto y otros símbolos

Para que LATEX reconozca los acentos que usamos en español directamente del teclado (como ‘ú’ en vez de \'u) y para que genere una salida adecuada para un pdf, colocamos en el *preámbulo*

inputenc

```
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
```

Con estos paquetes se tendrá un soporte para los acentos en español, ahora sólo las comillas se tendrán que seguir poniendo mediante la instrucción dada en la tabla.

Si la codificación es *utf-8*, como en muchas distribuciones recientes de Linux, colocamos en el *preámbulo* la instrucción,

```
\usepackage[utf8]{inputenc}
```

**Nota:** En este libro vamos a usar la codificación latin1 en las plantillas.

#### Cambiar las plantillas a español

Otro problema que tiene LATEX con el idioma es que los textos para "capítulo", "sección", etc., están en inglés, por lo tanto, en un libro no saldría **Capítulo 1** sino **Chapter 1**.

Existen dos formas sencillas de solucionar este problema. La forma más simple y recomendada es usar el paquete *babel*, ponemos en el *preámbulo* del documento la instrucción

babel

```
\usepackage[spanish]{babel}
```

que carga la opción en español de la librería *babel*. Además quedará habilitada la división correcta de las palabras. Dos opciones más aparecen como tópico adicional, ¿porqué?. El problema es que este paquete tiene conflictos con otros paquetes que podríamos querer usar.

Si **babel** tuviera conflictos con otros paquetes...

#### Tópico adicional: Cambios en el idioma.

A veces el paquete *babel* tiene conflictos con algún otro paquete que queremos usar. Por eso

tenemos que tener en cuenta un par de opciones más.

Si el documento es de tipo `article`, podemos poner en el *preámbulo*

---

```
\renewcommand{\contentsname}{Contenido}
\renewcommand{\partname}{Parte}
\renewcommand{\appendixname}{Apéndice}
\renewcommand{\figurename}{Figura}
\renewcommand{\tablename}{Tabla}
\renewcommand{\abstractname}{Resumen}
\renewcommand{\refname}{Bibliografía}
```

---

Si el documento es `book` se puede agregar en el *preámbulo*

---

```
\renewcommand{\contentsname}{Contenido}
\renewcommand{\partname}{Parte}
\renewcommand{\appendixname}{Apéndice}
\renewcommand{\figurename}{Figura}
\renewcommand{\tablename}{Tabla}
\renewcommand{\chaptername}{Capítulo}
\renewcommand{\bibname}{Bibliografía}
```

---

Una tercera opción es hacer este cambio permanente: Se puede editar los archivos `article.cls`, `report.cls` y/o `book.cls`. En [Ubuntu](#) estos archivos están en `/usr/share/texmf-texlive/tex/latex/base` (debe tener permisos de escritura para modificarlos). En la distribución MiK<sub>T</sub>E<sub>X</sub> están en `C:/Archivosdeprograma/MiKTeX2.7/tex/latex/base`. En ambos casos, se busca y se abre el archivo el archivo de texto `article.cls` (o `report.cls` o `book.cls`) y se buscan las líneas

---

```
\newcommand{\contentsname}{Contents}
\newcommand{\listfigurename}{List of Figures}
...
```

---

y se cambian por

---

```
\newcommand{\contentsname}{Contenido}
\newcommand{\partname}{Parte}
\newcommand{\indexname}{Lista Alfabética}
...
```

---

Luego, simplemente se guarda el archivo.

## División de palabras: Paquete `babel`.

**babel** Agregando '`\usepackage[spanish]{babel}`' en el preámbulo (junto con `inputenc` y `fontenc`), se logra que L<sup>A</sup>T<sub>E</sub>X divide correctamente la mayoría de palabras en español, sin embargo, hay algunos casos en donde no será así; si al componer el texto observamos que hay una palabra que se ha dividido mal, vamos a esa palabra en el archivo `*.tex`, y le indicamos exactamente donde la puede dividir, usando guiones. Por ejemplo, `e-xa-men`, `ac-ción`, `am-nis-tí-a`

Este sistema tiene el inconveniente de que **LaTeX** sólo divide bien la palabra en ese punto del documento y si dicha palabra aparece otra vez habrá que volver a decirle cómo se divide, y tiene la ventaja de que funciona con palabras que tienen acento. También se puede usar `\hyphenation{deci-sión,e-xa-men, otras palabras}` al inicio del documento para que **LaTeX** divida las palabras *tal y como se especifica* en la lista.

## 2.4 Tipos y tamaños de fuentes.

### 2.4.1 Caracteres especiales.

Algunos caracteres están **reservados** para que cumplan alguna función, por eso no se pueden obtener digitándolos (tecleándolos) directamente como cualquier letra. El hacerlo puede producir algún error de compilación, o puede pasar que el carácter sea ignorado. En las siguientes dos tablas se especifica el uso de algunos caracteres y el comando que se debe digitar (teclar) para imprimirlos.

Caracteres reservados

Carácter	Descripción
\	carácter inicial de comando
{ }	abre y cierra bloque de código
\$	abre y cierra el modo matemático
&	tabulador (en tablas y matrices)
#	señala parámetro en las macros
_ , ^	para subíndices y exponentes
~	para evitar cortes de renglón
%	para comentarios

Carácter	Comando
\	<code>\backslash</code>
{, }	<code>\{, \}</code>
\$	<code>\\$</code>
&	<code>\&amp;</code>
_ , ^	<code>\_ , \^{}{}</code>
#	<code>\#</code>
~	<code>\sim{}{}</code>
%	<code>\%</code>

### Otros símbolos y caracteres especiales

La barra invertida (\) se obtiene con el comando `\tt\char'\\`. Podemos escribir por ejemplo,

`\tt\char'\\begin` para obtener: `\begin`. Note que no es igual al comando '`\backslash`'.

Use "textcomp" para otros caracteres especiales.

### Paquete `textcomp`

Hay algunos caracteres especiales que requieren el paquete `textcomp`. Para usar este paquete agregamos en el *preámbulo*,

```
\usepackage{textcomp}
```

Por ejemplo, el acento: ` se obtiene con el comando `\textasciigrave`, el apóstrofo: ' se obtiene con el comando `\textquotesingle` y el símbolo de copyright: © se obtiene con `\textcopyright`. La lista de símbolos se puede obtener en

<http://home.online.no/~pjackson/latex/textcomp.pdf>

### El comando `verb`

El comando `\verb` permite imprimir los caracteres tal y como aparecen en pantalla. Por ejemplo, si se digita `\verb@{ $ x^2+1 $ }@` se imprimirá `{ $ x^2+1 $ }`. El símbolo '@' se usa como delimitador de lo que se quiere imprimir. Se pueden usar otros delimitadores no presentes en el texto a imprimir.

\verb

### 2.4.2 Algunos tipos de fuentes (fonts).

Para cambiar el tipo de letra se pone `\tipo texto`. Por ejemplo, para escribir en negrita se pone `\bf texto`

Comando	produce
<code>\rm Roman }</code>	Roman
<code>\em Enfático}</code>	<i>Enfático</i>
<code>\bf Negrita }</code>	<b>Negrita</b>
<code>\it Itálica }</code>	<i>Itálica</i>

Comando	produce
<code>\sl Slanted }</code>	<i>Slanted</i>
<code>\sf Sans Serif }</code>	Sans Serif
<code>\sc Small Caps }</code>	<b>SMALL CAPS</b>
<code>\tt Typewriter }</code>	Typewriter
<code>\underline{Subrayado}</code>	<u>Subrayado</u>

Observe el uso de las llaves para delimitar el alcance del comando, es decir, el comando no tiene efecto más allá de lo que está entre llaves.

**Nota:** Los comandos de la tabla anterior en realidad quedaron en desuso con la nueva versión LATEX2e (1994), pero estaban tan arraigados que se decidió mantenerlos aún con sus limitaciones. Por ejemplo, '`\it\bf Negrita en itálica`' no produce negrita en itálica, produce: '**Negrita en itálica**'.

En el nuevo LATEX2e se usa `\textit{...}`, `\textbf{...}`, `\texttt{...}` etc. Ahora podemos escribir '`\textit{\textbf{Negrita en itálica}}`' que produce ***Negrita en itálica***.

#### 2.4.3 Tamaños de letras.

Una manera de especificar el tamaño de la fuente es la siguiente,

Comando	produce
<code>\tiny</code>	Tiny
<code>\scriptsize</code>	Script
<code>\footnotesize</code>	Foot
<code>\small</code>	Small
<code>\normalsize</code>	Normal
<code>\large</code>	large
<code>\Large</code>	Large
<code>\huge</code>	<b>huge</b>
<code>\Huge</code>	<b>Huge</b>

Se pueden hacer combinaciones de tipos de letras con diferentes tamaños.

#### Ejemplo 2.1

`\large \bf Nota:` produce: **Nota**:  
`\large\textit{\textbf{Nota}}` produce: ***Nota***.

#### Tamaño global de fuentes.

La fuente default que usa LATEX es de tamaño 10pt (72pt = 1 pulgada). Este tamaño lo podemos

cambiar a 11pt o 12pt agregando esta opción, por ejemplo `\documentclass[12pt]{article}`.

Si por alguna razón queremos variar este tamaño, por ejemplo para adaptar un documento para leer en un lector *Kindle*, para un libro de cuentos, poesía, tesis, etc., podemos usar la clase `memoir`, así tendremos soporte para fuentes de tamaño 9pt, 10pt, 11pt, 12pt, 14pt, 17pt, 20pt, 25pt, 30pt, 36pt, 48pt y 60pt, además de muchas cosas adicionales de estilo. Esta clase se usan de la manera usual,

```
\documentclass[letterpaper,12pt,extrafontsizes]{memoir}
%paquetes, etc
\begin{document}
....
```

8pt	10pt	14pt
Después de haber definido $e^x$ para $x$ real cualquiera, es preferible definir $a^x$ por la fórmula $a^x = e^{x \log a}$	Después de haber definido $e^x$ para $x$ real cualquiera, es preferible definir $a^x$ por la fórmula $a^x = e^{x \log a}$	Después de haber definido $e^x$ para $x$ real cualquiera, es preferible definir $a^x$ por la fórmula $a^x = e^{x \log a}$

## Otros tamaños de fuente.

El tamaño de la fuente se puede controlar usando el paquete `anyfontsize`. Debemos poner en el *preámbulo*

```
\usepackage{anyfontsize}
```

Luego podemos usar el comando `\fontsize{x}{0}` para establecer el tamaño de la fuente en `xpt`.

### Ejemplo 2.2

El código:

```
\documentclass{article}
\usepackage{fix-cm}           % En algunos casos es necesario.
\usepackage{anyfontsize}
\begin{document}
Este es  {\fontsize{80}{0}\selectfont LaTeX}
\end{document}
```

Produce: Este es

**LaTeX**

## 2.5 (\*) Las fuentes y sus atributos

---

$\text{\LaTeX}$  no usa las fuentes del sistema operativo, más bien usa las fuentes instaladas por default en la distribución  $\text{\TeX}$ . Otras fuentes especiales se pueden agregar de manera automática usando paquetes. Si una fuente está disponible, puede aplicar esta fuente a parte de un texto o de manera global.

Una lista de fuentes disponibles en  $\text{\LaTeX}$  se puede encontrar en

“The LaTeX Font Catalogue” en <http://www.tug.dk/FontCatalogue/>

Cualquier fuente en  $\text{\LaTeX}$  tiene cinco atributos: `encoding`, `family`, `series`, `shape`, `size`. Ya hemos usado estos atributos para la fuente estándar. En un texto normal uno usa varios tipos de fuentes por eso es conveniente conocer como se hace en  $\text{\LaTeX}$  estos cambios.

`\fontencoding{}`: Es la manera de identificar los caracteres usando números. Por ejemplo, el carácter ~ lo podemos obtener directamente del teclado (pues estamos usando el paquete `inputenc` en la codificación `latin1`), pero lo podemos obtener también usando su codificación `latin1`: Este carácter se puede obtener con el comando `\char126`. El mismo código nos da otro carácter en el caso de que cambiemos a la codificación `OML`, por ejemplo.

`\fontfamily{}`: Nombre de la colección de fuentes. Familias comunes son

---

```
cmr Computer Modern Roman (default)
cmss Computer Modern Sans
cmtt Computer Modern Typewriter
cmm Computer Modern Math Italic
cmsy Computer Modern Math Symbols
cmex Computer Modern Math Extensions
ptm Adobe Times
phv Adobe Helvetica
pcr Adobe Courier
```

---

`\fontseries{}`: “Peso” de la fuente.

---

```
m Medium
b Bold
bx Bold extended
sb Semi-bold
c Condensed
```

---

`\fontshape{}`: Forma de la fuente.

---

```
n Normal
it Italic
sl Slanted (''oblicua'')
sc Caps and small caps
```

---

`\fontsize{tamaño}{baselineskip}`: Tamaño de la fuente y separación vertical entre líneas en un mismo párrafo (`baselineskip`).

Usualmente usamos instrucciones tales como

```
{\fontfamily{...}\selectfont{ texto}}  

{\fontencoding{...}\fontfamily{...}\selectfont{ texto }}  

{\fontencoding{...}\fontfamily{...} \fontseries{b}\selectfont{ texto }}
```

### Ejemplo 2.3

Para usar la fuente `Calligra`, debemos poner en el *preámbulo*

```
\usepackage[T1]{fontenc}  
\usepackage{calligra}
```

Luego podemos cambiar la fuente de una parte del texto. Por ejemplo, en la palabra ‘Había’ podemos cambiar la fuente y el tamaño para la ‘H’ solamente:

```
{\fontfamily{calligra}\fontsize{30}{1}\selectfont{H}}\normalfont abía una vez ...
```

Esta instrucción produce:



Otros ejemplos son

CM Mathematics  
`\fontencoding{OMS}\fontfamily{cmsy}\selectfont{x}` 2.3 § 2.3

Zapf Chancery Medium Italic:  
`{\fontfamily{pzc}\fontsize{12}{1}\selectfont{Ejemplo}}` Ejemplo

Sans Serif ( $\approx$  Helvetica):  
`{\fontfamily{hv}\selectfont{Ejemplo}}` Ejemplo

CM ‘Dunhill’  
`{\fontfamily{cmdh}\selectfont{Ejemplo}}` Nuevo Ejemplo

Podemos aplicar el tipo de fuente a partes determinadas del documento y regresar después a la fuente normal. Esto se hace usando el comando `\normalfont`. Por ejemplo,

```
Gauss publicó {\fontfamily{hv}\selectfont{ \blue Disquisitiones Arithmeticae}}  

\normalfont tres años más tarde...
```

Gauss publicó [Disquisitiones Arithmeticæ](#) tres años más tarde...

## Cambio global de fuente.

Para hacer un cambio global solo debe declarar el paquete correspondiente en el *preámbulo* del documento. Por ejemplo

---

\usepackage{pslatex}	\usepackage{concrete}
\usepackage{bookman}	\usepackage{cmbright}
\usepackage{helvet}	\usepackage{fourier}
\usepackage{palatino}	\usepackage{mathptmx}
\usepackage{newcent}	\usepackage{mathpazo}
\usepackage{pxfonts}	\usepackage{concrete,eulervm}
\usepackage{txfonts}	\usepackage{pslatex,concrete}

---

Este documento usa el paquete `mathpazo`.

### Ejemplo 2.4

#### Fuentes concrete y mathptmx

##### concrete

La expresión en paréntesis cuadrados es un *promedio* de los valores de  $f''$  en  $[a, b]$ , por lo tanto este promedio está entre el máximo y el mínimo absoluto de  $f''$  en  $[a, b]$  (asumimos  $f''$  continua). Finalmente, por el teorema del valor intermedio, existe  $\xi \in ]a, b[$  tal que  $f''(\xi)$  es igual a este valor promedio, es decir

$$-\frac{h^3}{12} \sum_{k=0}^{n-1} f''(\eta_k) = -\frac{(b-a)h^2}{12} \cdot f''(\xi), \quad \xi \in ]a, b[$$

##### mathptmx

La expresión en paréntesis cuadrados es un *promedio* de los valores de  $f''$  en  $[a, b]$ , por lo tanto este promedio está entre el máximo y el mínimo absoluto de  $f''$  en  $[a, b]$  (asumimos  $f''$  continua). Finalmente, por el teorema del valor intermedio, existe  $\xi \in ]a, b[$  tal que  $f''(\xi)$  es igual a este valor promedio, es decir

$$-\frac{h^3}{12} \sum_{k=0}^{n-1} f''(\eta_k) = -\frac{(b-a)h^2}{12} \cdot f''(\xi), \quad \xi \in ]a, b[$$

Información adicional se puede obtener en [2] y [16].



Usando las fuentes del sistema: Xe<sup>La</sup>T<sub>E</sub>X. L<sup>A</sup>T<sub>E</sub>X no usa las fuentes del sistema. Xe<sup>La</sup>T<sub>E</sub>X es una variante de T<sub>E</sub>X que puede usar las fuentes instaladas en el sistema operativo (si requiere un trabajo más profesional). Viene incluido en Mik<sup>T</sup>E<sub>X</sub> (2.8 en adelante) y en T<sub>E</sub>XLive 2010.

Un documento Xe<sup>La</sup>T<sub>E</sub>X lo editamos de la manera usual (la codificación debe ser UTF8) por ejemplo

---

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage{xltextra}
\setmainfont[Mapping=tex-text]{Lucida Bright}%fuente en el sistema

\begin{document}
Después de haber definido \$e^x\$ para \$x\$ real cualquiera, no hay ninguna
dificultad para dar una definición de \$a^x\$ para cada \$a>0\$.

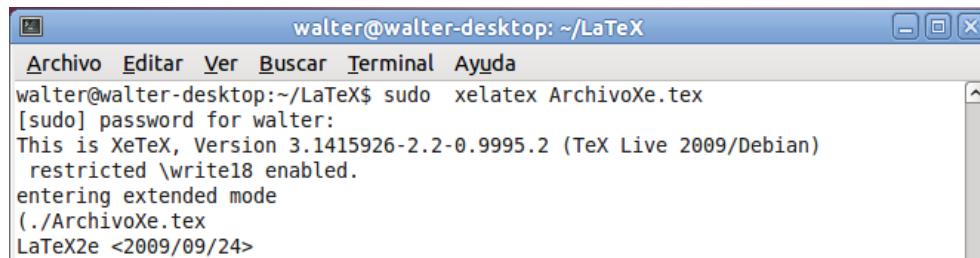
```

Un método es definir  $a^x$  como el número  $y$  tal que  $\log y = x$ ; claro que este método no sirve para  $a=1$  puesto que el logaritmo de base 1 no está definido. Otro modo es definir  $a^x$  por la fórmula:

```
 $$a^x=e^{x\log a}$$
\end{document}
```

---

Para compilar se usa el comando `xelatex` (presente en `Kile`; en `TeXMaker` hay que configurarlo). Si no se tiene el comando en el editor, abrimos una *terminal* y ejecutamos (en la carpeta correcta): `xelatex ArchivoTal.tex`. Por ejemplo en `Ubuntu` sería,



```
walter@walter-desktop: ~/LaTeX
Archivo Editar Ver Buscar Terminal Ayuda
walter@walter-desktop:~/LaTeX$ sudo xelatex ArchivoXe.tex
[sudo] password for walter:
This is XeTeX, Version 3.1415926-2.2-0.9995.2 (TeX Live 2009/Debian)
restricted \write18 enabled.
entering extended mode
(./ArchivoXe.tex
LaTeX2e <2009/09/24>
```

El resultado es un texto con fuente `Lucida Bright` (presente en mi sistema).

Después de haber definido  $e^x$  para  $x$  real cualquiera, no hay ninguna dificultad para dar una definición de  $a^x$  para cada  $a > 0$ . Un método es definir  $a^x$  como el número  $y$  tal que  $\log y = x$ ; claro que este método no sirve para  $a = 1$  puesto que el logaritmo de base 1 no está definido. Otro modo es definir  $a^x$  por la fórmula:

$$a^x = e^{x \log a}$$

Un manual de referencia se encuentra en <http://tug.ctan.org/tex-archive/info/xetexref/XeTeX-reference.pdf>.

## 2.6 Párrafos y efectos especiales.

---

En LATEX se puede escribir de manera ordenada o desordenada, el programa acomoda el texto e interpreta los comandos que se digitaron. Pero, por tratarse de un código, mejor es identar correctamente el texto. Para indicarle a LATEX que un párrafo ha terminado hay que *dejar un renglón en blanco*.

Si entre dos palabras se deja más de dos espacios en blanco solo se imprimirá uno. También se tiene que dejar doble paso de línea (doble 'enter') para separar párrafos o usar '\\" para cambiar de renglón. Si usamos '\\\\' avanzamos dos renglones.

**Ejemplo 2.5**

Cambio de renglón:

El texto:

---

```
{\bf Introducción.} \\  
Se parte de un conjunto $\Omega$  
de $n$ patrones, objetos  
o 'individuos',  
descritos por un vector  
de $p$ atributos.\\\\  
{\bf Nota:}....
```

---

produce:

**Introducción.**

Se parte de un conjunto  $\Omega$  de  $n$  patrones, objetos o 'individuos', descritos por un vector de  $p$  atributos.

**Nota:....****2.6.1 Centrar**

Para centrar un texto se pone éste en el entorno `\begin{center} ... \end{center}`

**Ejemplo 2.6 (Centrado).**

El texto:

---

```
\begin{center}  
Manual de\\  
\LaTeX  
\end{center}
```

---

produce:

Manual de  
**LATEX**

**2.6.2 Cajas**

Para encerrar palabras o un texto en una caja se usan los comandos `\fbox{ texto }` o `\framebox{ }`

**Ejemplo 2.7 (Cajas)**

El texto:

---

```
\begin{center}  
Manual de\\  
\fbox{\LaTeX}  
\end{center}
```

---

produce:

Manual de  
**LATEX**

### 2.6.3 Texto en columnas: `multicol`, `minipage` y `parbox`.

En LATEX existen varias maneras de escribir doble columna, ahora veremos algunas de ellas.

**El comando `multicol`.** La forma más sencilla es utilizar el paquete `multicol`, para ello se pone en el preámbulo,

```
\usepackage{multicol}
```

Antes del texto que se quiere escribir a doble columna se escribe `\begin{multicols}{#}` en donde # representa el número de columnas que se quieren utilizar, cuando se finaliza se escribe `\end{multicols}{#}`.

#### Ejemplo 2.8

El texto:

---

```
\begin{multicols}{2}
```

Hace que el texto que está entre los delimitadores salga a doble columna, LATEX{} se encarga de manera automática de distribuir el texto entre las columnas de la mejor manera, además ...  
`\end{multicols}`

---

produce:

Hace que el texto que está entre los delimitadores salga a doble columna, LATEX se encarga de manera automática de distribuir el texto entre las columnas de la mejor manera, además de acomodar el texto cuando hay un cambio de línea.

El campo de separación que se deja entre las columnas se define en el *preámbulo* con la instrucción `\setlength{\columnsep}{7mm}` la cual dejaría un espacio de 7 milímetros entre ellas.

#### El ambiente `minipage`

Otra manera de insertar texto a doble columna (o más columnas), es utilizar el entorno `minipage` (también se puede utilizar `\twocolumn`), la sintaxis es como sigue,

---

```
\begin{minipage}[t]{Xcm}
% primera columna
\end{minipage} \hfill \begin{minipage}[t]{Ycm}
% segunda columna
\end{minipage}
```

---

Donde ‘Xcm’ y ‘Ycm’ especifica el ancho de cada columna. En vez de ‘Xcm’ y ‘Ycm’ se puede usar un porcentaje del ancho de página: `\textwidth`. Por ejemplo, para usar la mitad del ancho de página ponemos:

```
\begin{minipage}[t]{0.5\textwidth}
```

La opción ‘b’ (bottom) se usa para alinear las dos columnas en el ‘fondo’ del ambiente `minipage`. También se puede usar la opción ‘t’ (top) o ‘c’ (center). La elección depende del contenido de cada ambiente ‘`minipage`’. En el ejemplo que sigue se incluye texto y una figura.

En este ejemplo se incluyen figuras: ver capítulo 5

### Ejemplo 2.9 (Figuras en un ambiente `minipage`).

El texto:

---

```
\begin{figure} [h]
\begin{minipage} [b] {0.3\textwidth}
    La imagen de la derecha muestra un icosaedro junto con un
    dodecaedro (figura central), los satélites son un icosaedro,
    un dodecaedro y un tetraedro. Las figuras fueron generadas con
    {\sc Mathematica} y maquilladas con {\it Inkscape}.
\end{minipage} \hfill \begin{minipage} [b] {0.6\textwidth}
    \begin{center}
        % Insertar figuras: ver capítulo 5
        \includegraphics{images/ML_fig3.eps}
        \caption{ Poliedros}
    \end{center}
\end{minipage}
\end{figure}
```

---

produce:



La imagen de la derecha muestra un icosaedro junto con un dodecaedro (figura central), los satélites son un icosaedro, un dodecaedro y un tetraedro. Las figuras fueron generadas con **MATHEMATICA** y maquilladas con *Inkscape*.

**Figura 2.2** Poliedros

### El comando `parbox`

Un comando similar a `minipage` es `parbox`. Se usa así:

```
\parbox{xcm}{texto}\parbox{ycm}{texto}.
```

`parbox` también se puede usar en combinación con `\framebox` para hacer un recuadro:

```
\framebox{\parbox{xcm}{texto}}.
```

Este comando no está pensado para grandes bloques de texto. Veamos un ejemplo:

**Ejemplo 2.10 (Incluir una figura usando `parbox`).**

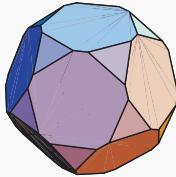
El texto:

---

```
\parbox{3cm}{\includegraphics{images/ML_fig4}} \parbox{12cm}{En
{\sc Mathematica}, podemos eliminar una o varias caras de un dodecaedro,
seleccionar el color y el grosor de las aristas y poner color a las caras.
Para esto debemos utilizar los comandos ... } %Sale del 2do parbox!
```

---

produce:



En MATHEMATICA, podemos eliminar una o varias caras de un dodecaedro, seleccionar el color y el grosor de las aristas y poner color a las caras. Para esto debemos utilizar los comandos ...

**Nota 1:** `verbatim` es un comando *frágil*. En ciertos ambientes no funciona.

**Nota 2:** Cuando se pone un gráfico en una columna debe tenerse el cuidado de ajustar el ancho del gráfico al ancho de la columna.

**Nota 3:** Cuando se utilice `\parbox` se debe tener el cuidado de terminar con `\}` para cambiar de renglón

## 2.7 Color, cajas y líneas.

---

- Para usar colores se podría llamar al paquete `xcolor`: `\usepackage{xcolor}`. Aunque se pueden usar los colores predefinidos (black, red, blue,...), también podemos personalizar los colores.

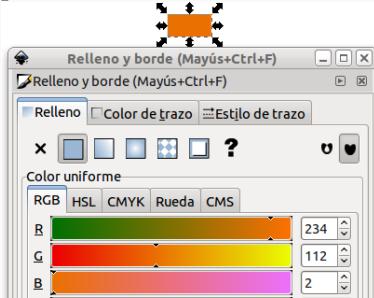
Para definir un color personalizado podemos usar alguna aplicación que tenga un selector de colores. Si, por ejemplo, el selector de colores codifica en el formato RGB, el color se puede definir así:

`\definecolor{micolor1}{rgb}{x, y, z}`, donde  $x, y, z \in [0,1]$ .

La definición de color se pone en el *preámbulo*. El formato RGB es adecuado para generar archivos PDF para ver en pantalla. Para imprimir se usa el formato CMYK.

### Ejemplo 2.11

Usando un selector de colores (en este caso, usamos [Inkscape](#)), podemos definir un anaranjado personalizado.



El anaranjado personalizado tiene parámetros 234 112 2, pero [LATEX](#) recibe valores entre 0 y 255, así que los dividimos por 255. Ponemos en el *preámbulo*

```
\definecolor{miorange}{rgb}{0.91, 0.43, 0.0}
```

- Para poner texto en una caja usando un color de fondo determinado, se usa

```
\fcolorbox{color fondo}{color borde}{ texto}
```

### Ejemplo 2.12

El código

```
\fcolorbox{orange}{orange}{ \color{white} LaTeX}
```

produce:

LaTeX

- Para entrar párrafos en una caja se puede usar “`minipage`” para que el texto se acomode de manera adecuada dentro de la caja. También podemos usar “`\fboxsep`” para controlar el espacio entre el contenido de la caja y el rectángulo. El valor por defecto es 0pt.

**Ejemplo 2.13**

El texto:

```
\begin{center}
\fboxsep 12pt
\fcolorbox{orange}{white}{
\begin{minipage}[t]{10cm}
$0^0$ es una expresión indefinida. Si $a>0$, $a^0=1$ pero $0^a=0$. Sin embargo, convenir en que $0^0=1$ es adecuado para que algunas fórmulas se puedan expresar de manera sencilla, sin recurrir a casos especiales, por ejemplo

$$e^x=\sum_{n=0}^{\infty}\frac{x^n}{n!}$$
$$ (x+a)^n=\sum_{k=0}^n \binom{n}{k} x^k a^{n-k} $$
\end{minipage}
}
\end{center}
```

produce:

$0^0$  es una expresión indefinida. Si  $a > 0$ ,  $a^0 = 1$  pero  $0^a = 0$ . Sin embargo, convenir en que  $0^0 = 1$  es adecuado para que algunas fórmulas se puedan expresar de manera sencilla, sin recurrir a casos especiales, por ejemplo

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

$$(x+a)^n = \sum_{k=0}^n \binom{n}{k} x^k a^{n-k}$$

**2.7.1 Otros efectos de texto**

- `\hfill` texto: Se usa para alinear el texto a la derecha.

**Ejemplo 2.14**

El texto: Instituto Tecnológico de Costa Rica `\hfill` Tiempo: 2:45 hrs

produce:

Instituto Tecnológico de Costa Rica

Tiempo: 2:45 hrs

- `\rulefill` y `\dotfill`. Veamos ejemplos de su uso

**Ejemplo 2.15**

El texto:

Instituto Tecnológico de Costa Rica \hrulefill Tiempo: 2:45 hrs

produce:

Instituto Tecnológico de Costa Rica ..... Tiempo: 2:45 hrs

**Ejemplo 2.16**

El texto:

Instituto Tecnológico de Costa Rica \dotfill Tiempo: 2:45 hrs

produce:

Instituto Tecnológico de Costa Rica ..... Tiempo: 2:45 hrs

- \rule[xcm]{ycm}{zcm}. Este comando se usa para dibujar una línea horizontal o vertical de ycm y grosor zcm. La distancia de la línea a la base del texto se controla con el primer parámetro [xcm].

**Ejemplo 2.17**

El texto:

Instituto Tecnológico de Costa Rica \hfill Tiempo: 2:45 hrs\\

\rule[0.5cm]{15cm}{0.01cm}

produce:

Instituto Tecnológico de Costa Rica ..... Tiempo: 2:45 hrs

### 2.7.2 Texto como en la pantalla

Para reproducir exactamente lo que está en la pantalla (incluyendo espacios) se pone el texto en el entorno \begin{verbatim}... \end{verbatim}.

**Ejemplo 2.18 (Usando `verbatim`).**

El texto:

```
\begin{verbatim}
Sub Trapecio(a,b,n,delta)
Dim N As Integer
Dim F As New clsMathParser
    suma = 0
    h = (b - a) / N
For i = 1 To N - 1
    xi = a + i * h
    suma = suma + F.Eval1(xi)
Next i
End Sub
\end{verbatim}
```

produce:

```
Sub Trapecio(a,b,n,delta)
Dim N As Integer
Dim F As New clsMathParser
    suma = 0
    h = (b - a) / N
For i = 1 To N - 1
    xi = a + i * h
    suma = suma + F.Eval1(xi)
Next i
End Sub
```

En la sección (7.8) se muestran varias opciones para obtener código de programación en color de acuerdo al lenguaje de programación.

## 2.8 Notas al pie de página.

Las notas al pie de página se producen con `\footnote{ ...texto... }`

El comando se escribe exactamente donde se quiere que quede la etiqueta que hará referencia al pie de página. La nota en el pie de esta página se generó con el código<sup>2</sup>.

.... esta página se generó con el código`\footnote{Ejemplo de \tt footnote}.`

### El Paquete `fancyvrb`

Hay cosas que no podemos hacer con el entorno `verbatim` pero que en algunos contextos son necesarios. Por ejemplo, usar `verbatim` para una nota al pie de página o usar símbolos matemáticos, color, etc.; dentro del ambiente `verbatim`. Estas cosas se pueden hacer con el paquete `fancyvrb`: Ponemos en el preámbulo

```
\usepackage{fancyvrb}
```

Con este paquete ahora ya podríamos usar `verbatim` en notas al pie de página. Se usa `footnote` de la manera usual pero, en este caso, precedido por `\VerbatimFootnotes`. Por ejemplo,

---

El peor ejemplo de programación recursiva es la de la función factorial

```
\VerbatimFootnotes \footnote{ Se refiere al código
\begin{verbatim}
int factorial(int n) {
    if (n == 0) return 1;
```

<sup>2</sup>Ejemplo de `footnote`

```

        return n * factorial(n-1);
\end{verbatim}
} aunque es un ejemplo muy claro.
```

---

A veces es adecuado introducir texto en modo matemático en un ambiente `verbatim` así como otros efectos. Para hacer esto, usamos el entorno `Verbatim` del paquete `fancyvrb`. Notar la mayúscula: `Verbatim`. El entorno sería

```
\begin{Verbatim}[opciones]
...
\end{Verbatim}
```

En el ejemplo que sigue, se usan varias opciones: '`frame=lines`' para poner segmentos de línea al inicio y al final, '`xleftmargin`' y '`xrightmargin`' para ajustar estos segmentos. '`commandchars`' se usa para indicar que `\` se va usar para aplicar comando `LATEX` en el entorno y `catcode` para especificar los caracteres de código matemático que se van a permitir en este ambiente, en este ejemplo caso: `$`, `^`, `\` y `_`.

### Ejemplo 2.19

El código:

```
\begin{Verbatim}[xleftmargin=3.1cm,xrightmargin=4.5cm,resetmargins=true,
               frame=lines,formatcom=\color{blue},fontfamily=ptm,
               commandchars=\{\}, codes={\catcode`$=3\catcode`^=7\catcode`_=8}]
$x$           $y=x^2$
0.000005      2.5$\times 10^{-11}$
\end{Verbatim}
```

produce:

$$\begin{array}{ll} x & y = x^2 \\ 0.000005 & 2.5 \times 10^{-11} \end{array}$$

Apóstrofe  
en VBA y  
OOoBasic

**Nota:** Si usamos el ambiente `Verbatim` para escribir código Basic (como VBA, OOoBasic, etc.), los comentarios usan el apóstrofe: `'`. Este apóstrofe se introduce en el ambiente `Verbatim` con el comando `\textquotesingle` del paquete `textcomp`.

### 2.8.1 Espacio horizontal y vertical

Para dejar espacio horizontal se usa el comando `\hspace{Xcm}`. El efecto es abrir espacio o correr horizontalmente texto, tablas o gráficos. `Xcm` es el corrimiento a la derecha o a la izquierda en centímetros, según sea `X` positivo o negativo. También se puede usar, por supuesto, `Xmm`, `Xin`, etc.

**Ejemplo 2.20**

```
$\subset \hspace{-3.5mm} / \hspace{-1.5mm} \$ 30 produce: \# 30
```

Para abrir espacio verticalmente se usa el comando `\vspace{Xcm}` que funciona de manera análoga a `\hspace`. En el ejemplo que sigue se debe dejar un renglón en blanco para lograr el efecto.

**Ejemplo 2.21**

El texto:

```
COLORES\\
\vspace{-1.5cm}
\hspace{2cm} VERDE
```

produce:

VERDE  
COLORES

### 2.8.2 Notas en el margen

Podemos poner figuras o texto en el margen con '`\hspace*`'. LATEX quita el espacio horizontal (en blanco) que hay al final de una línea. Si queremos que LATEX no elimine este espacio, se incluye el argumento opcional \*.

**Texto** → `\hspace*{-2.8cm} {\color{cyan} \small Texto}`

Hay varios paquetes para poner notas en el margen, por ejemplo el paquete "marginnote". Una vez que hemos puesto `\usepackage{marginnote}` en el preámbulo, podemos poner una nota en el margen como se muestra en el margen; el código que produce esta nota es

**Regla de la cadena:**

$$z_t = \nabla z \cdot \vec{v}$$

... podemos poner una nota en el margen como se muestra en el margen derecho;  
el código que produce esta nota es

```
\marginnote{\color{cyan} Regla de la cadena: $z_t=\nabla z \cdot \vec{v}$}
```

También podemos usar el paquete "todonotes" (es el que usamos en este libro), ver la sección 7.3.

## 2.9 Enumerado automático.

Uno de los comandos más usados para hacer listas es `enumerate`. Cada nuevo ítem se indica con `\item`, con esto se obtiene una enumeración automática. También uno puede controlar la enumeración con la etiqueta deseada. `enumerate` admite anidamiento hasta el cuarto nivel.

**Ejemplo 2.22**

El texto:

---

```
\bf Instrucciones.
Este es un examen de desarrollo, por lo tanto deben aparecer
todos los pasos que lo llevan a su respuesta.
Trabaje de manera clara y ordenada.\

\begin{enumerate}
\item[\bf [3 Puntos]] Sea $A=\{1,b,c,d,7\}$ y $B=\{1,2,c,d\}$.
    Calcule ${\cal P}(A \Delta B)$.
\item[\bf [5 Puntos]] Muestre que $A - (B \cap C) = (A - B) \cup (A - C)$

\item[\bf [5 Puntos]] Mostrar que $[A \cup C \subseteq B \cup C; A \cap C = \emptyset] \Rightarrow A \subseteq B$.
\Longleftrightarrow A \subseteq B
\item[\bf [2 Puntos]] Sea $\Re = (\mathbb{R}^*, \mathbb{R}^*, R)$
    definida por $x \Re y \iff xy > 0$.

\begin{enumerate}
\item[\bf [3 Puntos]] Muestre que $\Re$ es una relación de equivalencia.
\item[\bf [2 Puntos]] Determine las clases de equivalencia $\overline{1}$ y $\overline{-1}$.
\item[\bf [1 Punto]] Determine $\mathbb{R}^*/\Re$ (el conjunto cociente).
\end{enumerate}
\end{enumerate}

```

---

**Ejemplo 2.22 (continuación).**

produce:

**Instrucciones.** Este es un examen de desarrollo, por lo tanto deben aparecer todos los pasos que lo llevan a su respuesta. Trabaje de manera clara y ordenada.

1. **[3 Puntos]** Sea  $A = \{1, b, c, d, 7\}$  y  $B = \{1, 2, c, d\}$ . Calcule  $\mathcal{P}(A \Delta B)$ .
2. **[5 Puntos]** Muestre que  $A - (B \cap C) = (A - B) \cup (A - C)$
3. **[5 Puntos]** Mostrar que  $[A \cup C \subseteq B \cup C \wedge A \cap C = \emptyset] \implies A \subseteq B$
4. **[2 Puntos]** Sea  $\Re = (\mathbb{R}^*, \mathbb{R}^*, R)$  definida por  $x \Re y \iff xy > 0$ .
  - (a) **[3 Puntos]** Muestre que  $\Re$  es una relación de equivalencia.
  - (b) **[2 Puntos]** Determine las clases de equivalencia  $\bar{1}$  y  $\bar{-1}$ .
  - (c) **[1 Punto]** Determine  $\mathbb{R}^*/\Re$  (el conjunto cociente).

**Ejemplo 2.23**

Ejemplo con 4 niveles, usamos `itemize` en vez de 'enumerate'. El texto:

---

```
\begin{enumerate}
\item[\fbox{1.}] {\bf Procedimiento}
    {\em Aprendizaje}
\item[\fbox{2.}] {\bf comienzo}
\begin{enumerate}
\item Paso a.
\item Paso b.
    \begin{enumerate}
        \item Paso c.1
        \item Paso c.2
        \begin{itemize}
            \item Paso c.2.1
            \item Paso c.2.2
        \end{itemize}
        \item Paso c.3
    \end{enumerate}
\end{enumerate}
\item Paso d.
\end{enumerate}
\item[\fbox{3.}] {\bf fin}
\end{enumerate} % fin del primero
```

---

**Ejemplo 2.23 (continuación).**

produce:

- 1.** **Procedimiento** *Aprendizaje*
- 2.** **comienzo**
  - (a) Paso a.
  - (b) Paso b.
    - i. Paso c.1
    - ii. Paso c.2
      - Paso c.2.1
      - Paso c.2.2
    - iii. Paso c.3
  - (c) Paso d.
- 3.** **fin**

Los símbolos que `enumerate` pone por defecto para enumerar una lista se pueden cambiar redefiniendo los comandos `labelenumi`, `labelenumii`, `labelenumiii` y `labelenumiv`.

**Ejemplo 2.24**

Si escribimos:

---

```
\renewcommand{\labelenumi}{\Roman{enumi}.}
\renewcommand{\labelenumii}{\arabic{enumii}$} }
\renewcommand{\labelenumiii}{\alph{enumiii}$} }
\renewcommand{\labelenumiv}{$\bullet$ } }

\begin{enumerate}
    \item Primer nivel
        \begin{enumerate}
            \item Segundo nivel
                \begin{enumerate}
                    \item Tercer nivel
                        \begin{enumerate}
                            \item Cuarto nivel
                        \end{enumerate}
                \end{enumerate}
            \end{enumerate}
        \end{enumerate}
    \end{enumerate}
```

---

produce:

- I. Primer nivel (en Romanos)
  - 1) Segundo nivel (en numeración arábiga)
    - a) Tercer nivel (numeración alfabética)
      - Cuarto nivel (usamos bullet)

**Nota:** Si se va a utilizar el mismo sistema de numeración durante todo el documento, estas instrucciones se pueden poner en el preámbulo.

## 2.10 Enumeración usando el paquete **TikZ**.

Podemos crear ‘bolas’ 3D con números para usar en un entorno `enumerate`. Las bolas las creamos con el paquete `tikz` y debemos definir un comando para llamar a estas bolas. Ponemos en el preámbulo,

---

```
\usepackage{tikz}
\usepackage{enumitem}
\newcommand*\itembolasazules[1]{%
    \footnotesize\protect\tikz [baseline=-3pt]{%
        \protect\node [scale=.5, circle, shade, ball color=blue]{\color{white}\Large\bf#1};}}
```

Y ya podemos usar las listas enumeradas,

**Ejemplo 2.25 (Listas con el paquete tikz).**

El texto:

produce:

```
\begin{enumerate} [label=\itembolasazules{\arabic*}]
\item paso
\item paso
\item paso
\end{enumerate}
```

➊ paso

➋ paso

➌ paso

## 2.11 Título, contenido, secciones y bibliografía

La primera versión de este documento (2005) era de clase `report`.

```
\documentclass{report}
\textheight=20cm
\textwidth=18cm
\topmargin=-2cm
\oddsidemargin=-1cm
\parindent=0mm
\usepackage{graphicx}
\usepackage[T1]{fontenc}
\usepackage{ latexsym, amsmath, amsymb, amsfonts, cancel }
\newcommand{\sen}{\mathop{\rm sen}\nolimits} %seno
\newcommand{\arcsen}{\mathop{\rm arcsen}\nolimits}
\newcommand{\arcsec}{\mathop{\rm arcsec}\nolimits}
\setcounter{chapter}{0}
\newtheorem{teo}{Teorema}[chapter]
\newtheorem{ejemplo}{\it Ejemplo}[chapter]
\newtheorem{defi}{Definición}[chapter]
\begin{document}
\title{\LaTeX\\
\small \sc Instituto Tecnológico de Costa Rica\\
\small \sc Escuela de Matemática\\
\small \sc Enseñanza de la Matemática Asistida por\\
Computadora}
}
\author{Preparado por Prof. Walter Mora F. y Alexander Borbón A.}
\date{2005}
\maketitle %despliega el título
\tableofcontents
\chapter{\LaTeX}
\section{?`Qué es \LaTeX?}
...
\subsection{Preámbulo}
...
\subsubsection{Título}
```

```

...
\begin{thebibliography}{99}
\bibitem{} Hahn, J. ``\LaTeX \$,\$ for everyone''. Prentice Hall,
New Jersey, 1993.
...
\end{thebibliography}
\end{document}

```

---

## Bibliografía

Como se ve en la plantilla anterior,

```

\begin{thebibliography}{99}
\bibitem{Hahn} Hahn, J. {\it \LaTeX\}, for everyone}. Prentice Hall,
New Jersey, 1993.
\end{thebibliography}

```

---

es el ambiente para la bibliografía<sup>3</sup>. Es un ambiente parecido a `enumerate`.

En los documentos tipo `book` o `report`, para que aparezca “Bibliografía”, en la tabla de contenidos se debe poner

```

\addcontentsline{toc}{chapter}{Bibliografía}
\begin{thebibliography}{99}
...

```

---

## Referencias a la bibliografía: `cite`

En el texto uno puede hacer referencia a algún ítem de la bibliografía. Para hacer esto, le ponemos una etiqueta al ítem, digamos:

```

\begin{thebibliography}{99}
...
\bibitem{Hahn} Hahn, J. ``\LaTeX\ \', for everyone''.
Prentice Hall, New Jersey, 1993.
\end{thebibliography}

```

---

`\cite`

Entonces podemos hacer referencia a este libro en el texto con `\cite[nota]{clave}` o solamente con `\cite{clave}`. Por ejemplo: En `\cite[pág. 80]{Gratzer}` se pueden ver los aspectos... produce:

En [6, pág. 80] se pueden ver los aspectos relativos a ...”.

Algo similar podemos hacer en los ejemplos, teoremas, definiciones, etc. Para estos usamos una etiqueta (label) para identificarlos. Por ejemplo, si ponemos

```

\begin{defi}\label{definicion1}
...
\end{defi}

```

---

podemos hacer referencia a esta definición así: En la definición (`\ref{definicion1}`)... Lo mismo podemos hacer en los ambientes `table`, `figure`, `eqnarray`, `equation`, etc.

<sup>3</sup>Una descripción más extensa sobre este tópico se desarrolla en el capítulo 8.

## 2.12 Modulación

---

Para evitar la incomodidad de mantener archivos muy grandes, es conveniente modular el texto separándolo en varios archivos *sin preámbulo ni \begin{document} ... \end{document}*. Por ejemplo, este texto tenía la siguiente plantilla

---

```
\documentclass{report}
\textheight=20cm
\textwidth=18cm
\topmargin=-2cm
...
\begin{document}
\input cap1.tex
\pagebreak
\input cap2.tex
\pagebreak
...
\end{document}
```

---

Cada archivo \*.tex fue editado con el preámbulo hasta que estuviera afinado. Luego se recortó el archivo.





# 3 TEXTO EN MODO MATEMÁTICO

Recordemos que frecuentemente el texto matemático va en el entorno  $\$ \dots \$$ . También hay otros entornos que no requieren estos símbolos de dólar. En este capítulo vamos a usar símbolos especiales (los símbolos son fuentes) que no están presentes en el conjunto de símbolos que por default carga L<sup>A</sup>T<sub>E</sub>X.

Por ejemplo, para escribir “ $\operatorname{sen}(x) \in \mathbb{R}$ ” se requiere definir un comando `\sen` y tener acceso al paquete que me permite usar el comando `\mathbb{R}` para obtener  $\mathbb{R}$ .

Para obtener el texto matemático de este capítulo necesitamos los símbolos que por default carga L<sup>A</sup>T<sub>E</sub>X y adicionalmente varios paquetes: `amsmath`, `amssymb`, `amsfonts`, `latexsym`, `cancel`. También necesitamos cinco comandos especiales para `\sen`, `\arcsen`, etc. La plantilla que sigue viene con todo lo que necesitamos para los ejemplos que siguen en todo este capítulo:

## Plantilla para este capítulo

```
\documentclass{article} %o report o book
  \textheight=20cm
  \textwidth=18cm
  \topmargin=-2cm
    %símbolos matemáticos de la AMS
  \usepackage{amsmath,amssymb,amsfonts,latexsym,cancel}
  \usepackage[spanish]{babel}
  \usepackage[latin1]{inputenc} %acentos desde el teclado
  \usepackage[T1]{fontenc}
    % Comandos especiales
  \newcommand{\sen}{\mathop{\rm sen}\nolimits} %seno
  \newcommand{\arcsen}{\mathop{\rm arcsen}\nolimits}
  \newcommand{\arcsec}{\mathop{\rm arcsec}\nolimits}
  \def\max{\mathop{\rm máx}\nolimits} %máximo
  \def\min{\mathop{\rm mín}\nolimits} %mínimo
\begin{document}
  La fórmula  $\sen^2(x)=1-\cos^2(x)$  ...
\end{document}
```

Aquí suponemos que se tiene una versión completa de MiK<sup>T</sup>E<sub>X</sub> o de T<sub>E</sub>XLive.

## 3.1 Potencias, subíndices y superíndices

Expresión	Código	Expresión	Código
$x^p$	<code>x^p</code>	$x^{n+1}$	<code>x^{n+1}</code>
$(2^2)^n$	<code>(2^2)^n</code>	$2^{2^n}$	<code>2^{2^n}</code>
$\operatorname{sen}^2(x)$	<code>\sen^2(x)</code>	$x^{\operatorname{sen}(x)+\cos(x)}$	<code>x^{(\operatorname{sen}(x)+\cos(x))}</code>
$a_n$	<code>a_n</code>	$a_{n+1}$	<code>a_{n+1}</code>
$u_{N+1}$	<code>u_{N+1}</code>	$u_{N+1}$	<code>u_{N+1}</code>
$a_i^j$	<code>a_i^j</code>	$\int_a^b f(x) dx$	<code>\int_a^b f(x) \, dx</code>
$\sum_{n=1}^N u_n$	<code>\sum_{n=1}^N u_n</code>	$u_{ij}$	<code>u_{ij}</code>

## 3.2 Tamaño natural

Como se ve en la tabla anterior, el texto matemático se ajusta al ancho del renglón. Para desplegarlo en tamaño natural se usa el comando `\displaystyle`. Si sólo se quiere que una parte del texto matemático salga en tamaño natural se escribe `\displaystyle{}` y entre las llaves se pone el texto.

`\displaystyle`

### Ejemplo 3.1

El texto:

La suma parcial  $N$ -ésima  $S_N$  se define con la igualdad  
 $$

produce:

La suma parcial  $N$ -ésima  $S_N$  se define con la igualdad  $S_N = \sum_{k=1}^N a_k$

Se pueden ajustar los subíndice y los superíndices de la siguiente manera

Normal: `$S_{N_j}$` produce:  $S_{N_j}$

Mejor: `$S_{\{N_j\}}$` produce:  $S_{N_j}$

### 3.3 Raíces

---

Raíces cuadradas y raíces  $n$ -ésimas.

Expresión	Código
$\sqrt{x+1}$	<code>\sqrt{x+1}</code>
$\sqrt[n]{x+\sqrt{x}}$	<code>\sqrt[n]{x+\sqrt{x}}</code>
$\sqrt[n]{x+\sqrt{x}}$	<code>\sqrt[n]{x+\sqrt{x}}</code>

### 3.4 Fracciones y expresiones de dos niveles

---

Para hacer fracciones se pueden utilizar los comandos: `\over`, `\frac{ }{ }` o `\atop`. Veamos también otras “fracciones” útiles.

Expresión	Código
$\frac{x+1}{x-1}$	<code>{x+1 \over x-1}</code>
$\frac{x+1}{x-1}$	<code>\frac{x+1}{x-1}</code>
$\frac{\frac{x+1}{3}}{x-1}$	<code>\frac{\frac{x+1}{3}}{x-1}</code>
$\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}}$	<code>\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}}</code>
$\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}}$	<code>\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}}</code>
$\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}}$	<code>\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}}</code>
$\frac{x+1}{x-1}$	<code>\frac{x+1}{x-1}</code>
$\frac{x+1}{x-1}$	<code>\frac{x+1}{x-1}</code> (2pt es el grosor)
$\left\{ \frac{x+1}{x-1} \right\}$	<code>\left\{ \frac{x+1}{x-1} \right\}</code>
$\left[ \frac{x+1}{x-1} \right]$	<code>\left[ \frac{x+1}{x-1} \right]</code>

## Otras expresiones que requieren dos niveles

Expresión	Código
$a \xrightarrow{f} b$	$\displaystyle{a \stackrel{f}{\rightarrow} b}$
$\lim_{x \rightarrow 0} f(x)$	$\displaystyle{\lim_{x \rightarrow 0} f(x)}$
$\binom{a}{b}$	$\displaystyle{a \choose b}$
$\sum_{\substack{0 < i < m \\ 0 < j < n}} a_i b_j$	$\displaystyle{\sum_{\substack{0 < i < m \\ 0 < j < n}} a_i b_j}$
$\prod_{\substack{i=0 \\ i \neq k}}^n \frac{w_i}{(w_i - w_k)}$	$\prod_{\substack{i=0 \\ i \neq k}}^n \frac{w_i}{(w_i - w_k)} = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$

### Ejemplo 3.2

El código:

```
 $$L_{n,k}(x)
 = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{x-x_i}{x_k-x_i}
 = \frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}
 } $$
```

produce:

$$L_{n,k}(x) = \prod_{\substack{i=0 \\ i \neq k}}^n \frac{x-x_i}{x_k-x_i} = \frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

Note el uso de  $\substack{i=0 \\ i \neq k}$  para producir:  $i \neq k$

## Integrales

Expresión	Código
$\int_C \mathbf{F} \cdot d\mathbf{r}$	$\displaystyle{\int_C \mathbf{F} \cdot d\mathbf{r}}$
$\oint_C \mathbf{F} \cdot d\mathbf{r}$	$\displaystyle{\oint_C \mathbf{F} \cdot d\mathbf{r}}$
$\iint_D f(x,y) dA$	$\displaystyle{\iint_D f(x,y) dA}$
$\iiint_Q f(x,y,z) dV$	$\displaystyle{\iiint_Q f(x,y,z) dV}$

En las integrales los subíndices y los superíndices se pueden centrar con el símbolo de integración. Por ejemplo,

`$\displaystyle{\iiint \limits_Q}$` produce:  $\iiint_Q$

## 3.5 Tres puntos consecutivos

---

Se usa un grupo de tres puntos para indicar la continuación de un patrón, se obtienen con los siguientes comandos.

...    \ldots	...    \cdots
:    \vdots	..    \ddots

## 3.6 Delimitadores

---

Para ajustar delimitadores al tamaño de una fórmula se usan los comandos `\left ... \right`. Se puede usar un punto para evitar abrir o cerrar con uno de los delimitadores.

### Ejemplo 3.3

El texto:

`$\displaystyle \left[ \frac{x+1}{(x-1)^2} \right]^n$`

produce:  $\left[ \frac{x+1}{(x-1)^2} \right]^n$

El texto:

`$\int_a^b 2x dx = \left. x^2 \right|_a^b`

produce:  $\int_a^b 2x dx = \left. x^2 \right|_a^b$

El texto:

`$\displaystyle \left\{ \begin{array}{l} n \in \mathbb{N} \\ r \neq 1 \end{array} \right. \neq 1 \right\}`

produce:  $\left\{ \begin{array}{l} n \in \mathbb{N} \\ r \neq 1 \end{array} \right. \neq 1 \right\}$

**Ejemplo 3.4**

El texto:

---

```
\[ f(x)=\left\{ \begin{array}{ll} x^2+1 & \text{si } x \geq 0 \\ & \\ & \ln|x| & \text{si } x < 0 \end{array} \right. \]
\end{array}
\right. \] % Observe el punto que cierra: \left\{ ... \right.
```

---

produce:

$$f(x) = \begin{cases} x^2 + 1 & \text{si } x \geq 0 \\ \ln|x| & \text{si } x < 0 \end{cases}$$

**Delimitadores del paquete `amsmath`.**

También se puede usar los delimitadores del paquete `amsmath`:

`\Biggl,``\Biggr,` `\biggl,``\biggr,` `\Bigl,``\Bigr,` `\bigl,``\bigr.`

En algunos casos estos delimitadores son más eficientes.

**Ejemplo 3.5 (Comparación de delimitadores)**

El texto:

`$$\biggl[ \sum_j \Bigl| \sum_i x_{ij} \Bigr|^2 \biggr]^{1/2}$$`

produce:

$$\left[ \sum_j \left| \sum_i x_{ij} \right|^2 \right]^{1/2}$$

Comparar con `$$\left[ \sum_j \left( \sum_i x_{ij} \right)^2 \right]^{1/2}$$`

$$\left[ \sum_j \left| \sum_i x_{ij} \right|^2 \right]^{1/2}$$

## 3.7 LLaves y barras horizontales

### Barras horizontales.

Las barras horizontales sobre el texto se pueden obtener con el comando `\overline{}`

#### Ejemplo 3.6

El texto:

---

Leyes de DeMorgan:  

$$\begin{aligned} \overline{A \cup B} &= \overline{A} \cap \overline{B} \\ \overline{A \cap B} &= \overline{A} \cup \overline{B} \end{aligned}$$

---

produce:

Leyes de DeMorgan: 
$$\begin{cases} \overline{A \cup B} = \overline{A} \cap \overline{B} \\ \overline{A \cap B} = \overline{A} \cup \overline{B} \end{cases}$$

### LLaves horizontales

Las llaves se ponen con `\{` y `\}`. Esto se usa tanto en texto corriente como en modo matemático.

#### Ejemplo 3.7

El código:

```
$$\max_{x \in A}\{f(x)\} > \min_{x \in A}\{g(x)\}$$
```

produce:

$$\max_{x \in A}\{f(x)\} > \min_{x \in A}\{g(x)\}$$

Los comandos `\max` y `\min` aparecen con acento pues así los definimos en el preámbulo propuesto al inicio del capítulo.

Se puede poner tanto una llave horizontal superior como una llave horizontal inferior y un texto arriba o abajo de la llave, se usa `\overbrace{}^{}_{}`, `\underbrace{}_{}`, etc.

**Ejemplo 3.8**

El texto:

---


$$\overbrace{(x_{i-1})^K f(x)}^{= K_i} + \underbrace{(x_{i-1})_K g(x)}_{= K_i(f(x) + g(x))}$$


---

produce:

$$\overbrace{(x_{i-1})^K f(x)}^{= K_i} + \underbrace{(x_{i-1})_K g(x)}_{= K_i} = K_i(f(x) + g(x))$$

## 3.8 Acentos y “sombreros” en modo matemático

---

 $\hat{t} \quad \$\hat{\imath}\$ \quad \acute{a} \quad \$\acute{a}\$$ 
 $\bar{p} \quad \$\bar{p}\$ \quad \vec{p} \quad \$\vec{p}\$$ 

## 3.9 Negritas en modo matemático

---

En una fórmula matemática, el comando `\boldmath` solo aplica negrita a las fuentes de texto. Para poner en negrita los símbolos se debe usar `\boldsymbol` o `\pmb`. También se puede poner en negrita toda la expresión matemática usando

 $\hbox{\boldmath $ texto $ \unboldmath}$ 
**Ejemplo 3.9 (Negrita en modo matemático)**
 $\cos(x+2\pi) = \mathbf{cos} x \quad \$\pmb{\cos(x+2\pi)}=\cos x\$$ 
 $\cos(x+2\pi) = \mathbf{cos} x \quad \$\cos(x+\pmb{2\pi})=\cos x\$$ 

## 3.10 Espacio en modo matemático

---

*LaTeX* no deja espacios en modo matemático. Para dejar espacio en modo matemático se usan los comandos `\,` `\;` `\!` `\:` tanto como `\hspace{}`

**Ejemplo 3.10 (Espacio en modo matemático)**

Normal:	$n \in \mathbb{N}, x \in \mathbb{R}$	<code>\$n \in \mathbb{N}, x \in \mathbb{R}\$</code>
Mejor:	$n \in \mathbb{N}, x \in \mathbb{R}$	<code>n \in \mathbb{N}, \; ; \; x \in \mathbb{R}</code>
Normal:	$\int f(x)dx$	<code>\$\int f(x) \, dx\$</code>
Mejor:	$\int f(x)dx$	<code>\$\displaystyle{\int f(x) \, dx}\$</code>

**3.11 Centrado**

Para centrar una fórmula se usa `\[ ... \]` o también `$$...$$`, en las fórmulas centradas no es en general necesario utilizar `\displaystyle` para producir texto en tamaño natural.

**Ejemplo 3.11**

El texto:

```
$$ ab \leq \left( \frac{a+b}{2} \right)^2 $$
```

produce:

$$ab \leq \left( \frac{a+b}{2} \right)^2$$

**3.12 Entorno `equation`. Contadores automáticos**

LATEX puede llevar un conteo automático de capítulos, secciones, etc. Podemos llevar también un conteo automático de teoremas, ecuaciones, etc. Por ejemplo con el entorno `equation` (un entorno para el modo matemático que no requiere `$`),

```
\begin{equation} ... \end{equation}
```

podemos poner un número de ecuación a la ecuación actual. Podemos también cambiar ese número usando

```
\setcounter{equation}{k}.
```

El efecto de este comando es sumar (o restar si ponemos `-k`)  $k$  unidades al número de ecuación actual

**Ejemplo 3.12 (Entorno equation)**

```
%estamos en ecuación 3.1 (cap 3)
\begin{equation}
\log_2(xy)=\log_2x + \log_2y
\end{equation} 
$$\log_2(xy) = \log_2 x + \log_2 y \quad (3.1)$$


%sumamos 2 y pasamos a ecuación 3.3
\setcounter{equation}{2}
\begin{equation}
\log_2(a^b)=b\log_2a
\end{equation} 
$$\log_2(a^b) = b \log_2 a \quad (3.3)$$

```

Una vez establecido el contador, se puede usar el ambiente `subequations` para lograr una ‘sub-numeración’:

**Ejemplo 3.13 (subequations)**

```
\begin{subequations}
\begin{equation}
\log_2(xy)=\log_2x + \log_2y
\end{equation} 
$$\log_2(xy) = \log_2 x + \log_2 y \quad (3.4a)$$

\begin{equation}
\log_2(a^b)=b\log_2a
\end{equation} 
$$\log_2(a^b) = b \log_2 a \quad (3.4b)$$

\end{subequations}
```

## 3.13 Arreglos

---

Para editar una matriz se debe indicar:

- Los delimitadores, digamos: `\left[ ... \right]`
- Inicio del “array” y el número y alineación de las columnas (centrado (c), alineado a la izquierda (l) o a la derecha (r)), digamos 3 columnas: `\begin{array}{lcr}`
- Los delimitadores de columnas, para 3 columnas: `& & & \\`
- “`\backslash\backslash`” indica el cambio de fila
- Final del “array”: `\end{array}`

**Ejemplo 3.14 (Entorno array)**

El texto:

```
\[
A = \left( \begin{array}{lcr}
a & a+b & k-a \\
b & b & k-a-b \\
\vdots & \vdots & \vdots \\
z & z+z & k-z
\end{array} \right)
\]
```

produce:

$$A = \begin{pmatrix} a & a+b & k-a \\ b & b & k-a-b \\ \vdots & \vdots & \vdots \\ z & z+z & k-z \end{pmatrix}$$

**Ejemplo 3.15**

Hagamos algunos cambios: Agreguemos dos columnas vacías y cambiemos el alineamiento

El texto:

```
\[
A = \left( \begin{array}{lcccl}
& & & & \\
a & & a+b & & k-a \\
b & & b & & k-a-b \\
\vdots & & \vdots & & \vdots \\
z & & z+z & & k-z
\end{array} \right)
\]
```

produce:

$$A = \begin{pmatrix} & a & a+b & k-a \\ & b & b & k-a-b \\ \vdots & \vdots & \vdots & \vdots \\ z & z+z & k-z \end{pmatrix}$$

**Ejemplo 3.16 (Función a trozos).**

El texto:

```
\[
f(x) = \left( \begin{array}{ll}
x^2 & \text{si } x < 0 \\
x-1 & \text{si } x > 0
\end{array} \right)
\]
```

produce:

$$f(x) = \begin{cases} x^2 & \text{si } x < 0 \\ x-1 & \text{si } x > 0 \end{cases}$$

Nota: `\mbox` se utiliza para escribir texto corriente dentro del modo matemático.

### Ejemplo 3.17

A veces es conveniente anidar un array dentro de otro array,

El código

```

$$
\left.\begin{array}{lclcl}
\cos x &=& 0 &\Longleftrightarrow& x=(2k+1)\cdot\frac{\pi}{2}; & k \in \mathbb{Z} \\
&&&&& \\
\sin x &=& -1 &\Longleftrightarrow& x=(4k+3)\cdot\frac{\pi}{2}; & k \in \mathbb{Z} \\
&&&&& \\
\cos(2x) &=& \frac{1}{2} &\Longleftrightarrow& \left.\begin{array}{lcr}
x &=& \frac{\pi}{6}+k\pi; & z \in \mathbb{Z} \\
&&& \\
x &=& -\frac{\pi}{6}+k\pi; & z \in \mathbb{Z} \\
\end{array}\right. \\
&&&&& \\
\end{array}\right. \\
\end{array}
$$

```

produce:

$$\left\{ \begin{array}{l} \cos x = 0 \implies x = (2k+1)\frac{\pi}{2}; k \in \mathbb{Z} \\ \sin x = -1 \implies x = (4k+3)\frac{\pi}{2}; k \in \mathbb{Z} \\ \cos(2x) = \frac{1}{2} \implies \begin{cases} x = \frac{\pi}{6} + k\pi; z \in \mathbb{Z} \\ x = -\frac{\pi}{6} + k\pi; z \in \mathbb{Z} \end{cases} \end{array} \right.$$

## 3.14 Matrices

El entorno array es útil y versátil. Si solo queremos trabajar con matrices podemos usar los entornos `smallmatrix`, `pmatrix`, `bmatrix`, `Bmatrix`, `vmatrix` y `Vmatrix`. Estos entornos producen, respectivamente,  $( )$ ,  $[ ]$ ,  $\begin{matrix} \end{matrix}$ ,  $| |$  y  $\left| \begin{matrix} \end{matrix} \right|$ .

El entorno `smallmatrix` produce arreglos ajustados (sin delimitadores), para ser usadas en el texto normal, e.g.  $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ .

El código del párrafo anterior es,

---

El entorno `\tt smallmatrix` produce arreglos ajustados (sin delimitadores), para ser usadas en el texto normal, e.g. `$\bigl( \begin{smallmatrix} a & b \\ c & d \end{smallmatrix} \bigr)$.`

---

### Ejemplo 3.18 (Entorno para matrices)

El texto:

---

```
\begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
h_0 & 2(h_0+h_1) & h_1 & \cdots & 0 \\
0 & h_1 & 2(h_1+h_2) & h_2 & \cdots \\
& \ddots & \ddots & \ddots & \\
0 & 0 & h_{n-3} & 2(h_{n-3}+h_{n-2}) & h_{n-2} \\
0 & 0 & & & \cdots 1
\end{pmatrix} \cdot \begin{pmatrix}
c_0 \\
c_1 \\
\vdots \\
c_{n-1} \\
c_n
\end{pmatrix}
```

---

produce:

$$\begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
h_0 & 2(h_0+h_1) & h_1 & \cdots & 0 \\
0 & h_1 & 2(h_1+h_2) & h_2 & \cdots \\
& \ddots & \ddots & \ddots & \\
0 & \cdots & h_{n-3} & 2(h_{n-3}+h_{n-2}) & h_{n-2} \\
0 & 0 & & & \cdots 1
\end{pmatrix} \cdot \begin{pmatrix}
c_0 \\
c_1 \\
\vdots \\
c_{n-1} \\
c_n
\end{pmatrix}$$

## 3.15 Alineamiento

### Ambiente “eqnarray”

Se puede alinear una serie de pasos (o ecuaciones), en un razonamiento usando

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

este comando construye una matriz de 3 *columnas*.

Si se quiere que cada uno de los pasos aparezca numerado se utiliza

```
\begin{eqnarray} . . . \end{eqnarray}
```

Si se usa `\begin{eqnarray} ... \end{eqnarray}`, se puede evitar numerar una ecuación poniendo `\nonumber` al final de la fila (antes de `\backslash\backslash`).

### Ejemplo 3.19 (Arreglos sin numeración)

El texto:

De acuerdo al lema de Euclides tenemos que

```
\begin{eqnarray*}
\mbox{mcd}(a,b) &=& \mbox{mcd}(a-r_0q,r_0) \\
&=& \mbox{mcd}(r_1,r_0) \\
&=& \mbox{mcd}(r_1,r_0-r_1q_2) \\
&=& \mbox{mcd}(r_1,r_2) \\
&=& \mbox{mcd}(r_1-r_2q_2,r_2)
\end{eqnarray*}
```

produce:

De acuerdo al lema de Euclides tenemos que

$$\begin{aligned} \text{mcd}(a,b) &= \text{mcd}(a - r_0q, r_0) \\ &= \text{mcd}(r_1, r_0) \\ &= \text{mcd}(r_1, r_0 - r_1q_2) \\ &= \text{mcd}(r_1, r_2) \\ &= \text{mcd}(r_1 - r_2q_2, r_2) \end{aligned}$$

### Ejemplo 3.20 (Sin numeración)

El texto:

```
% Sin numeración >>
\begin{eqnarray*}
y=\sqrt[n]{x} &\Longleftrightarrow& y^n=x \\
&\Longleftrightarrow& n \log y = \log x, \text{ si } x>0, y>0 \\
&\Longleftrightarrow& \log \sqrt[n]{x}=\frac{1}{n} \log x
\end{eqnarray*}
```

produce:

$$\begin{aligned} y = \sqrt[n]{x} &\implies y^n = x \\ &\implies n \log y = \log x; \text{ si } x > 0, y > 0 \\ &\implies \log \sqrt[n]{x} = \frac{1}{n} \log x \end{aligned}$$

### Ejemplo 3.21 (Con numeración)

El texto:

---

```
% Con numeración >>
\begin{eqnarray}
y=\sqrt[n]{x} & \Longrightarrow & y^n=x \\
& \Longrightarrow & n\log y=\log x, \text{ si } x>0, y>0 \\
& \Longrightarrow & \log \sqrt[n]{x}=\frac{1}{n}\log x
\end{eqnarray}
```

---

produce:

$$y = \sqrt[n]{x} \implies y^n = x \quad (3.5)$$

$$\implies n \log y = \log x, \text{ si } x > 0, y > 0 \quad (3.6)$$

$$\implies \log \sqrt[n]{x} = \frac{1}{n} \log x \quad (3.7)$$

### Ejemplo 3.22 (Numeración selectiva)

El texto

---

```
% Numeración selectiva >>
\begin{eqnarray}
y=\sqrt[n]{x} & \Longrightarrow & y^n = x \quad \text{\textbackslash nonumber}\\
& \Longrightarrow & n\log y = \log x, \text{ si } x>0, y>0 \\
& \Longrightarrow & \log \sqrt[n]{x}=\frac{1}{n}\log x
\end{eqnarray}
```

---

produce:

$$y = \sqrt[n]{x} \implies y^n = x \quad (3.8)$$

$$\implies n \log y = \log x, \text{ si } x > 0, y > 0 \quad (3.9)$$

$$\implies \log \sqrt[n]{x} = \frac{1}{n} \log x \quad (3.9)$$

### Entornos “align” y “multline”

El entorno ‘eqnarray’ no se recomienda porque tiene algunos inconvenientes: Produce un espacio irregular en los signos de igualdad y no hacen ningún intento para evitar la sobreimpresión del cuerpo de la ecuación y número de la ecuación. Si usamos el paquete `amsmath` (como lo asumimos aquí) podemos acceder a los entornos “align” y “multline”

El entorno ‘align’ es similar a ‘eqnarray’ pero sin los problemas indicados, y también permite ecuaciones numeradas o sin numerar (usando `align*`).

En el entorno `align`:

- `\&=` establece una igualdad en una misma columnas mientras que `\&` establece un cambio de columna.

- El comando `\intertext{texto}` intercala texto entre filas mientras se mantiene las columnas alineadas.

### Ejemplo 3.23

El código

```

\begin{align*}
\intertext{Agrupamos,}
\frac{a+ay+ax+y}{x+y} &= \frac{ax+ay+x+y}{x+y} & \text{\textsf{Agrupar}} \\
\intertext{sacamos el factor común,}
&= \frac{a(x+y)+x+y}{x+y} & \text{\textsf{Factor común}} \\
&= \frac{(x+y)(a+1)}{x+y} & \text{\textsf{Simplificar}} \\
&= a+1
\end{align*}

```

produce:

Agrupamos,

$$\frac{a + ay + ax + y}{x + y} = \frac{ax + ay + x + y}{x + y} \quad \text{Agrupar}$$

sacamos el factor común,

$$\begin{aligned}
 &= \frac{a(x+y) + x + y}{x+y} && \text{Factor común} \\
 &= \frac{(x+y)(a+1)}{x+y} && \text{Simplificar} \\
 &= a+1
 \end{aligned}$$

### Ejemplo 3.24

El código

```
\begin{align*}
a &= b + c - d \\
&\quad \& \quad e - f \\
&= m
\end{align*}
```

```
\begin{multiline*}
    a+b+c+d+e+f+t+x+y\\
    +m+n+r+t+y
\end{multiline*}
```

produce:

$$\begin{aligned} a &= b + c - d \\ &\quad + e - f \\ &= m \end{aligned} \qquad \begin{aligned} a + b + c + d + e + f + t + x + y \\ + m + n + r + t + y \end{aligned}$$

## 3.16 Tablas de símbolos matemáticos frecuentes

---

### 3.16.1 Letras griegas

$\alpha$	\alpha	$\kappa$	\kappa	$\varsigma$	\varsigma	$\Lambda$	\Lambda
$\beta$	\beta	$\lambda$	\lambda	$\tau$	\tau	$\Xi$	\Xi
$\gamma$	\gamma	$\mu$	\mu	$\upsilon$	\upsilon	$\Pi$	\Pi
$\delta$	\delta	$\nu$	\nu	$\phi$	\phi	$\Sigma$	\Sigma
$\epsilon$	\epsilon	$\xi$	\xi	$\varphi$	\varphi	$\Upsilon$	\Upsilon
$\varepsilon$	\varepsilon	$\circ$	\circ	$\chi$	\chi	$\Phi$	\Phi
$\zeta$	\zeta	$\pi$	\pi	$\psi$	\psi	$\Psi$	\Psi
$\eta$	\eta	$\varpi$	\varpi	$\omega$	\omega	$\Omega$	\Omega
$\theta$	\theta	$\rho$	\rho	$\Gamma$	\Gamma		
$\vartheta$	\vartheta	$\varrho$	\varrho	$\Delta$	\Delta		
$\iota$	\iota	$\sigma$	\sigma	$\Theta$	\Theta		

### 3.16.2 Operadores binarios

$\pm$	\pm	$\circ$	\circ	$\triangleleft$	\triangleleft	$\oplus$	\oplus
$\mp$	\mp	$\bullet$	\bullet	$\triangleright$	\triangleright	$\ominus$	\ominus
$\setminus$	\setminus	$\div$	\div	$\wr$	\wr	$\otimes$	\otimes
$\cdot$	\cdot	$\cap$	\cap	$\bigcirc$	\bigcirc	$\oslash$	\oslash
$\times$	\times	$\cup$	\cup	$\bigtriangleup$	\bigtriangleup	$\odot$	\odot
$*$	\ast	$\uplus$	\uplus	$\bigtriangledown$	\bigtriangledown	$\dagger$	\dagger
$\star$	\star	$\sqcap$	\sqcap	$\vee$	\vee	$\ddagger$	\ddagger
$\diamond$	\diamond	$\sqcup$	\sqcup	$\wedge$	\wedge	$\amalg$	\amalg

### 3.16.3 Relaciones

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

### 3.16.4 Negación de relaciones

En general, cualquier negación se puede hacer anteponiendo la instrucción \not a cualquier relación anterior, algunos ejemplos se muestran en la tabla siguiente.

$\not\leq$	\not\leq	$\not\subseteq$	\not\subseteq
$\not\geq$	\not\geq	$\not\sqsubseteq$	\not\sqsubseteq
$\not\succ$	\not\succ	$\not\supset$	\not\supset
$\not\succeq$	\not\succeq	$\not\supseteq$	\not\supseteq

$\not\geq$	<code>\not\geq</code>	$\neq$	<code>\not=</code>
$\not\succ$	<code>\not\succ</code>	$\not\equiv$	<code>\not\equiv</code>
$\not\mathrel{\succcurlyeq}$	<code>\not\mathrel{\succcurlyeq}</code>	$\not\sim$	<code>\not\sim</code>
$\not\supset$	<code>\not\supset</code>	$\not\mathrel{\simeq}$	<code>\not\mathrel{\simeq}</code>
$\not\supseteq$	<code>\not\supseteq</code>	$\not\approx$	<code>\not\approx</code>
$\not\sqsupseteq$	<code>\not\sqsupseteq</code>	$\not\cong$	<code>\not\cong</code>

### 3.16.5 Otros símbolos

$\aleph$	<code>\aleph</code>	$\partial$	<code>\partial</code>	$\parallel$	<code>\parallel</code>	$\natural$	<code>\natural</code>
$\hbar$	<code>\hbar</code>	$\infty$	<code>\infty</code>	$\angle$	<code>\angle</code>	$\sharp$	<code>\sharp</code>
$\imath$	<code>\imath</code>	$\prime$	<code>\prime</code>	$\triangle$	<code>\triangle</code>	$\clubsuit$	<code>\clubsuit</code>
$\jmath$	<code>\jmath</code>	$\emptyset$	<code>\emptyset</code>	$\backslash$	<code>\backslash</code>	$\diamondsuit$	<code>\diamondsuit</code>
$\ell$	<code>\ell</code>	$\nabla$	<code>\nabla</code>	$\forall$	<code>\forall</code>	$\heartsuit$	<code>\heartsuit</code>
$\wp$	<code>\wp</code>	$\surd$	<code>\surd</code>	$\exists$	<code>\exists</code>	$\spadesuit$	<code>\spadesuit</code>
$\Re$	<code>\Re</code>	$\top$	<code>\top</code>	$\neg$	<code>\neg</code>		
$\Im$	<code>\Im</code>	$\bot$	<code>\bot</code>	$\flat$	<code>\flat</code>		

### 3.16.6 Especiales

$$\widetilde{A} \quad \$\widetilde{A}\$ \quad \overrightarrow{v} \quad \$\overrightarrow{v}\$$$

$$\widehat{A} \quad \$\widehat{A}\$$$

### 3.16.7 Símbolos del paquete amssymb

El paquetes `amssymb` se carga si usamos el preámbulo propuesto al inicio del capítulo.

$\mathbb{R}$	<code>\mathbb{R}</code>	$\triangleright$	<code>\gtrdot</code>	$\trianglerighteq$	<code>\lesseqgtr</code>	$\supseteq$	<code>\Supset</code>
$\mathbb{Q}$	<code>\mathbb{Q}</code>	$\trianglerighteq$	<code>\gtrless</code>	$\doteqdot$	<code>\doteqdot</code>	$\sqsubset$	<code>\sqsubset</code>
$\mathbb{Z}$	<code>\mathbb{Z}</code>	$\lessgtr$	<code>\eqslantless</code>	$\cdot\cdot\cdot$	<code>\risingdotseq</code>	$\sqsupset$	<code>\sqsupset</code>
$\mathbb{I}$	<code>\mathbb{I}</code>	$\lessapprox$	<code>\lessim</code>	$\cdot\cdot\cdot$	<code>\fallingdotseq</code>	$\succcurlyeq$	<code>\succcurlyeq</code>
$\therefore$	<code>\therefore</code>	$\approx\approx$	<code>\lessapprox</code>	$\circ\circ$	<code>\circeq</code>	$\preccurlyeq$	<code>\preccurlyeq</code>
$\because$	<code>\because</code>	$\approx\approx$	<code>\approxeq</code>	$\triangle\triangle$	<code>\triangleq</code>	$\curlyeqprec$	<code>\curlyeqprec</code>
$\leq$	<code>\leq</code>	$\lessdot$	<code>\lessdot</code>	$\sim$	<code>\thicksim</code>	$\curlyeqsucc$	<code>\curlyeqsucc</code>
$\geq$	<code>\geq</code>	$\gtreqless$	<code>\gtreqless</code>	$\approx$	<code>\thickapprox</code>	$\precsim$	<code>\precsim</code>
$\leqslant$	<code>\leqslant</code>	$\gtreqqless$	<code>\gtreqqless</code>	$\backsim$	<code>\backsimeq</code>	$\succsim$	<code>\succsim</code>
$\geqslant$	<code>\geqslant</code>	$\lll$	<code>\lll</code>	$\backsim$	<code>\backsimeq</code>	$\preccurlyeq$	<code>\preccurlyeq</code>
$\leqslantgtr$	<code>\leqslantgtr</code>	$\ggg$	<code>\ggg</code>	$\subsetneqq$	<code>\subsetneqq</code>	$\succcurlyeq$	<code>\succcurlyeq</code>
$\gtreqsim$	<code>\gtreqsim</code>	$\lessgtr$	<code>\lessgtr</code>	$\supseteqq$	<code>\supseteqq</code>	$\vartriangleright$	<code>\vartriangleright</code>
$\gtreqapprox$	<code>\gtreqapprox</code>	$\lesseqgtr$	<code>\lesseqgtr</code>	$\Subset$	<code>\Subset</code>	$\trianglerighteq$	<code>\trianglerighteq</code>

## 3.17 Cómo hacer nuevos Comandos.

---

Podemos abreviar el código de los comandos creando comandos propios. Para esto usamos

- `\newcommand{\nuevo_nombre}{\comando_original}`
- `\newcommand{\nombre}[n]{\f{\#1}... \h{\#n}}`.  $n$  es el número de parámetros. Cada parámetro es recibido por un comando existente.

Las definiciones de los nuevos comandos se ponen en el *preámbulo* (para comodidad de otros usuarios).

Una práctica muy recomendada es hacerse un archivo aparte con estas definiciones, este archivo debe ir **sin** preámbulo **ni** `\begin{document}... \end{document}`. Si el archivo es “`miscmds.tex`”, éste se invoca en el preámbulo con `\input{miscmds.tex}`. Este archivo puede estar en el directorio de trabajo preferiblemente.

Vamos a ver algunos ejemplos de abreviaciones

```
\newcommand{\bc}{\begin{center}}
\newcommand{\ec}{\end{center}}
\newcommand{\ds}[1]{\displaystyle{#1}} %un parámetro
\newcommand{\sii}{\Leftrightarrow}
\newcommand{\gfrac}[2]{\ds{\frac{#1}{#2}}} % dos parámetros. Usamos \ds{}
\newcommand{\imp}{\Longrightarrow}
\newcommand{\wvec}[1]{\overrightarrow{#1}}
\newcommand{\wvecb}[1]{\color{blue}\overrightarrow{#1}} % azul
\newcommand{\proy}[2]{\ds{\mbox{\rm proy}}_{\sum_{k=1}^{#1} u_k}} % dos parámetros. Usamos \ds{}
\newcommand{\sumauk}[2]{\ds{\sum_{k=1}^{#1} u_k}} % dos parámetros. Usamos \ds{}
```

Ahora podemos escribir, por ejemplo:

```
$S_{\color{red}N}=\sumauk{1}{\color{red}N} \, , \Longleftrightarrow \, S_{N+1}=S_N+u_{N+1} \$
```

para producir

$$\text{Si } S_{\color{red}N} = \sum_{k=1}^N u_k \implies S_{N+1} = S_N + u_{N+1}$$

Y también, el código:

```
$\proy{v}{w}=\gfrac{\wvec{v} \cdot \wvec{w}}{||\wvec{w}||^2}, \wvec{w} \$
```

$$\text{va a producir: } \proy{\vec{v}}{\vec{w}} = \frac{\vec{v} \cdot \vec{w}}{||\vec{w}||^2} \vec{w}$$

Podemos abreviar otros comandos con ayuda de parámetros, por ejemplo matrices, `minipage`, etc. Se debe especificar el número de parámetros del comando con `[n]`, Se usa `#k` para hacer referencia al parámetro  $k$ -ésimo.

**Ejemplo 3.25**


---

```
%Un comando para el ambiente minipage
\newcommand{\mpage}{[2]{\begin{minipage}[t]{0.5 \textwidth}
#1
\end{minipage} \hfill \begin{minipage}[t]{0.5 \textwidth}
#2
\end{minipage}}}

%Un comando para límites
\newcommand{\limite}{[2]{\lim_{#1} \rightarrow #2}}
```

---

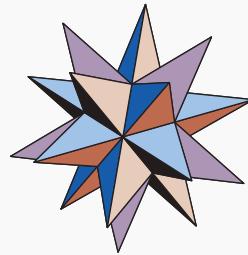
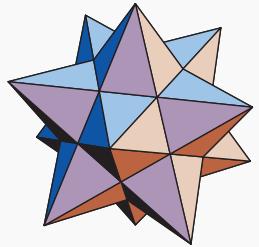
Así, el texto:

---

```
\mpage{
\centering
\includegraphics{images/ML_fig13.pdf}
}
\centering
\includegraphics{images/ML_fig14.pdf}
```

---

produce (compilando con PDF<sub>L</sub>A<sub>T</sub>E<sub>X</sub>):



El texto:

```
\[\limite{n}{\infty} \arctan(n)\]
```

produce:

$$\lim_{n \rightarrow \infty} \arctan(n)$$

Otros ejemplos son

---

```
\newcommand{\colr}{\color{red}}
\newcommand{\fhv}[1]{\fontfamily{hv}\fontsize{9}{1}\selectfont{#1}}%Uso \fhv{texto}
\newcommand{\bc}{\begin{center}}
\newcommand{\ec}{\end{center}}
\newcommand{\be}{\begin{enumerate}}
\newcommand{\ee}{\end{enumerate}}
\newcommand{\bt}{\begin{tabular}} % Uso: \bt{cccc}...\et
\newcommand{\et}{\end{tabular}}
```

---

## 3.18 Comandos con opciones

Podemos agregar opciones a nuestros comandos dejando algunos valores por default. Esto lo podemos hacer con el paquete `xargs`: Ponemos `\usepackage{xargs}` en el *preámbulo*.

Un ejemplo clásico es el de crear un comando para abbreviar una sucesión:  $x_0, x_1, \dots, x_n$ . En este caso, es deseable que podamos tener un comando flexible que nos permita iniciar en 0 o en 1 y terminar en  $n$  o en  $k$  y cambiar  $x_i$  por  $u_i$ , etc.

La sucesión por default será  $x_0, x_1, \dots, x_n$ .

### Ejemplo 3.26

El comando

```
\newcommandx*\coord[3][1=0, 3=n]{#2_{#1}, \ldots, #2_{#3}}
```

recibe tres argumentos, el primero y el tercero son opcionales y tienen valor default 0 y  $n$  respectivamente, por eso aparece `[1=0, 3=n]`. El parámetro `#2` permite elegir  $x, u$ , etc.

El código:

```
$\coord{x}$
```

```
$\coord[0]{y}$
```

```
$\coord{z}[m]$
```

```
$\coord[0]{t}[m]$
```

produce:

$(x_1, \dots, x_n)$

$(y_0, \dots, y_n)$

$(z_1, \dots, z_m)$

$(t_0, \dots, t_m)$

### Ejemplo 3.27

Podemos hacer más flexible nuestro comando ‘`\mpage`’ de la siguiente manera

```
\newcommandx*\mpage[4][1=0.45,2=0.45]{ %#1 y #2 definen el porcentaje de página
\begin{minipage}[t]{#1\textwidth}
#3
\end{minipage} \hfill \begin{minipage}[t]{#2\textwidth}
#4
\end{minipage}}}
```

de tal manera que lo podemos usar como `\mpage[0.7][0.2]{...}{...}` tanto como `\mpage{...}{...}`



# 4 TABLAS

Las tablas se editan en forma similar a las matrices pero en las tablas se pueden poner líneas verticales y horizontales. El modo matemático debe especificarse en una tabla.

- Para agregar líneas verticales se ponen marcas como `|` o `||` en la parte que corresponde al alineamiento de columnas.
- Para agregar líneas horizontales, al final de cada fila se especifica
  - `\hline`: línea tan larga como la tabla
  - `\cline{i-j}`: línea de columna  $i$  a columna  $j$

### Ejemplo 4.1 (Usando `tabular`)

El texto:

produce:

```
\begin{tabular}{|c|c|c|} \hline
$p$ & $q$ & $p \rightarrow q$ \\ \hline
0 & 0 & 1 \\ \hline
0 & 1 & 1 \\ \hline
1 & 0 & 0 \\ \hline
1 & 1 & 1 \\ \hline
\end{tabular}
```

$p$	$q$	$p \rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

## 4.1 Los ambientes `figure` y `table`

Un objeto (gráfico o una tabla) debe aparecer en el lugar más cercano al texto que hace referencia a él. Al ir haciendo cambios en el texto, los objetos pueden desplazarse de manera no apropiada. L<sup>A</sup>T<sub>E</sub>X resuelve (y a veces complica) este problema manipulando las figuras como objetos flotantes en el documento.

$\text{\LaTeX}$  nos ofrece dos comandos (ambientes) para indicarle nuestras preferencias sobre el desplazamiento del objeto.

```
\begin{figure}[h]....\caption{...}\label{figure:nombre} \end{figure}
\begin{table}[h].... \caption{...}\label{table:nombre} \end{table}
```

- [h] le indica a  $\text{\LaTeX}$  que queremos la figura o la tabla, exactamente en ese lugar (h=here, esto no es tan exacto, ya que  $\text{\LaTeX}$  en realidad lo acomoda lo más cerca posible de ese lugar). Otras opciones son [t]=top, [b]=bottom.
- \caption{ texto} es la etiqueta de cada objeto (numerándolo automáticamente). Se puede omitir.
- \label es la identificación del objeto. En el texto podemos hacer referencia a la tabla o a la figura, poniendo

‘En la figura \ref{fig:nombre}...’ o ‘En la tabla \ref{nombre}...’.

Si no vamos a hacer referencia, podemos omitir este comando.

#### Ejemplo 4.2 (Entorno `table`)

El texto:

---

```
\begin{table}[h]
\begin{tabular}{|c|c|c|} \hline
$p \& $q & $p \rightarrow q$ \\ \hline
0 & 0 & 1 \\
0 & 1 & 1 \\
1 & 0 & 0 \\
1 & 1 & 1 \\ \hline
\end{tabular}
\caption{Tabla de verdad para $p \rightarrow q$}
\end{table}
```

---

produce:

$p$	$q$	$p \rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

**Tabla 4.1** Tabla de verdad para  $p \rightarrow q$

**Nota:** Si va a poner una figura o una tabla en el ambiente `minipage`, debería usar el siguiente formato

---

```
\begin{figure}[h]
\begin{minipage}{...} ... \end{minipage}
\end{figure}
```

---



---

```
\begin{table}[h]
\begin{minipage}{...} ... \end{minipage}
\end{table}
```

---

## 4.2 Fuentes en tabular.

A veces es conveniente cambiar la fuente en un ambiente tabular, por ejemplo si tenemos una tabla de números sería bueno cambiar a una fuente `cmr10` para que todo quede en modo matemático sin tener que hacer esto número por número,

**Ejemplo 4.3 (Cambio de fuente)**

El texto:

```
\fontfamily{cmr10}\selectfont{
\begin{tabular}{c|c}
x$ & $x^2+3$ \\ \hline
2 & 7 \\
4 & 19 \\
\end{tabular} } }\fontfamily
```

produce:

$$\begin{array}{c|c} x & x^2 + 3 \\ \hline 2 & 7 \\ 4 & 19 \end{array}$$

**4.3 Color en tablas.**

Para poner color en las filas o las columnas de una tabla podríamos usar el paquete `xcolor` agregamos al *preámbulo* `\usepackage[x11names,table]{xcolor}`. La opción `x11names` habilita un conjunto de colores que podemos llamar por su nombre. La lista completa la puede obtener en `xcolor.pdf`. Algunos nombres se indican en la tabla que sigue,

<code>\color{RoyalBlue1} RoyalBlue1</code>	<code>RoyalBlue1</code>	<code>\color{LightSteelBlue1} LightSteelBlue1</code>	<code>Col</code>
<code>\color{RoyalBlue2} RoyalBlue2</code>	<code>RoyalBlue2</code>	<code>\color{LightSteelBlue2} LightSteelBlue2</code>	<code>Col</code>
<code>\color{RoyalBlue3} RoyalBlue3</code>	<code>RoyalBlue3</code>	<code>\color{LightSteelBlue3} LightSteelBlue3</code>	<code>Col</code>
<code>\color{RoyalBlue4} RoyalBlue4</code>	<code>RoyalBlue4</code>	<code>\color{LightSteelBlue4} LightSteelBlue4</code>	<code>Col</code>
<code>\color{Tan1} Tan1</code>	<code>Tan1</code>	<code>\color{Tan2} Tan2</code>	<code>Col</code>
<code>\color{Tan3} Tan3</code>	<code>Tan3</code>	<code>\color{Tan4} Tan4</code>	<code>Col</code>
<code>\color{Yellow1} Yellow1</code>	<code>Yellow1</code>	<code>\color{Yellow2} Yellow2</code>	<code>Col</code>
<code>\color{Yellow3} Yellow3</code>	<code>Yellow3</code>	<code>\color{Yellow4} Yellow4</code>	<code>Col</code>
<code>\color{Thistle1} Thistle1</code>	<code>Thistle1</code>	<code>\color{Thistle2} Thistle2</code>	<code>Col</code>
<code>\color{Thistle3} Thistle3</code>	<code>Thistle3</code>	<code>\color{Thistle4} Thistle4</code>	<code>Col</code>
<code>\color{Orange1} Orange1</code>	<code>Orange1</code>	<code>\color{Orange2} Orange2</code>	<code>Col</code>
<code>\color{Orange3} Orange3</code>	<code>Orange3</code>	<code>\color{Orange4} Orange4</code>	<code>Col</code>

**Tabla 4.2** Algunos nombres de colores vía la opción `x11names` de `xcolor`

Para colorear una fila se debe poner `\rowcolor{nombre-color}` al inicio de la fila. Para colorear de manera alternada se agrega `\rowcolors{nfila}{color fila-impar}{color fila-par}` antes del inicio de la tabla. Aquí, `nfila` es el número de fila de la primera fila en ser coloreada. Los

`rowcolor`

columncolor,  
cellcolor

colores de fila par e impar se pueden dejar en blanco (no se pondrá color en esa fila).

Los comandos `\columncolor` y `\cellcolor` se usan para colorear las columnas y celdas, respectivamente.

**Nota:** Si hay conflictos (“clash”) con otros paquetes, como por ejemplo con `Beamer`, para evitar el problema podríamos poner la opción al inicio, en la clase de documento,

```
\documentclass[xcolor=table, x11names]{...}
```

#### Ejemplo 4.4

En este ejemplo la fila inicial se colorea con el color `LightBlue2` de la opción `x11names` de el paquete `xcolor`. Esto se hace agregando `\rowcolor{LightBlue2}` al inicio de la primera fila. También se colorean con un gris degradado a un 20% (`gray!20`) las filas pares y las impares se dejan con fondo blanco. Esto se hace agregando, antes del inicio de la tabla, la instrucción `\rowcolors{1}{}{gray!20}`.

El código:

---

```
\begin{table}[h]
\centering
{\fontfamily{ptm}\selectfont
\rowcolors{1}{}{gray!20}
\begin{tabular}{ll}
\rowcolor{LightBlue2} $x_{n+1}$ & $|x_{n+1}-x_n|$\hline
1.20499955540054 & 0.295000445\\
1.17678931926590 & 0.028210236\\
1.17650193990183 & 3.004\times10^{-8}\\
1.17650193990183 & 4.440\times10^{-16}\hline
\end{tabular}}\%font
\caption{Iteración de Newton para $x^2-\cos(x)-1=0$ con $x_0=1.5$.}
\end{table}
```

---

produce:

$x_{n+1}$	$ x_{n+1} - x_n $
1.20499955540054	0.295000445
1.17678931926590	0.028210236
1.17650193990183	$3.004 \times 10^{-8}$
1.17650193990183	$4.440 \times 10^{-16}$

**Tabla 4.3** Iteración de Newton para  $x^2 - \cos(x) - 1 = 0$  con  $x_0 = 1.5$ .

**Ejemplo 4.5**

En este ejemplo se colorean con un 20% gris dos celdas: Simplemente agregamos `\cellcolor[gray]{0.80}` en las celdas que queremos.

El código:

---

```
\begin{table}[h]
\centering
{\fontfamily{ptm}\selectfont
\begin{tabular}{ll}
\rowcolor{LightBlue2} $x_{n+1}$ & $|x_{n+1}-x_n|$\\ \hline
\cellcolor[gray]{0.80} 1.20499955540054 & 0.295000445\\
1.17678931926590 & 0.028210236\\
1.17650196994274 & 0.000287349\\
1.17650193990183 & 3.004$\times10^{-8}$\\
\cellcolor[gray]{0.80} 1.17650193990183 & 4.440$\times10^{-16}$\\ \hline
\end{tabular}
\caption{Iteración de Newton para $x^2-\cos(x)-1=0$ con $x_0=1.5$.}
}}\%font
```

---

produce:

$x_{n+1}$	$ x_{n+1} - x_n $
1.20499955540054	0.295000445
1.17678931926590	0.028210236
1.17650196994274	0.000287349
1.17650193990183	$3.004 \times 10^{-8}$
1.17650193990183	$4.440 \times 10^{-16}$

**Tabla 4.4** Iteración de Newton para  $x^2 - \cos(x) - 1 = 0$  con  $x_0 = 1.5$ .

## 4.4 Rotación de texto en celdas.

Para rotar una tabla completa o simplemente el texto en las celdas. se usa el entorno

```
\begin{sideways} ... \end{sideways}
```

aplicado directamente a la tabla o a la(s) celda(s). Necesitamos agregar en el preámbulo

```
\usepackage{rotating}
```

### Ejemplo 4.6

El código:

```
\begin{sideways}
\begin{tabular}{lc}
$x_{n+1}$ & $|x_{n+1}-x_n|$ \\
\hline
\cellcolor{gray}{0.80} 1.17 & $3.\times 10^{-8}$ \\
& $1.17 \times 10^{-16}$ \\
\hline
\end{tabular}
\end{sideways}
```

produce:

$x_{n+1}$	$ x_{n+1} - x_n $
1.17	$3 \times 10^{-8}$
1.17	$4 \times 10^{-16}$

El código:

```
\begin{tabular}{lc}
$x_{n+1}$ & $|x_{n+1}-x_n|$ \\
\hline
\begin{sideways} $|x_{n+1}-x_n|$ \end{sideways} \\
\hline
\cellcolor{gray}{0.80} 1.17 & $3.\times 10^{-8}$ \\
1.173 & $4.\times 10^{-16}$ \\
\hline
\end{tabular}
```

produce:

$x_{n+1}$	$ x_{n+1} - x_n $
1.17	$3 \times 10^{-8}$
1.173	$4 \times 10^{-16}$

## 4.5 Unir celdas.

A veces es conveniente unir dos o más celdas para poner una leyenda un poco extensa. Para hacer esto usamos

```
\multicolumn{columnas}{Alin}{texto}
```

columnas : Número de columnas que abarcará la celda.  
 Alin : Indica la alineación del texto.

### Ejemplo 4.7

El código:

```
\begin{table}[h]
\centering
\begin{tabular}{lll}
& \multicolumn{2}{l}{Estimación del error} \\
& \multicolumn{2}{l}{absoluto y relativo} \\
\rowcolor{LightBlue2} $x_n$ & $x_{n+1}$ & $|x_{n+1}-x_n|/|x_{n+1}|$ \\ \hline
-3.090721649 & 2.990721649 & 1.6717 \\
-2.026511552 & 1.064210097 & 0.525143859 \\
-1.205340185 & 0.821171367 & 0.681277682 \\ \hline
\end{tabular}
\caption{}
\end{table}
```

Produce:

Estimación del error absoluto y relativo		
$x_n$	$x_{n+1}$	$ x_{n+1}-x_n / x_{n+1} $
-3.090721649	2.990721649	1.6717
-2.026511552	1.064210097	0.525143859
-1.205340185	0.821171367	0.681277682

Tabla 4.5

## 4.6 Escalar una tabla

A veces tenemos tablas muy grandes. Las podemos escalar en un porcentaje y también usar unión de celdas para lograr un efecto decente. El escalamiento lo podemos hacer con el comando `\scalebox{0.h}{0.v}{...}`. Aquí, 0.h y 0.v es el porcentaje de escalamiento horizontal

`\scalebox`

y vertical. Si ponemos solamente `\scalebox{0.h}{...}` se escala igual en cada dirección.

### Ejemplo 4.8

El código que sigue escala un 80% una tabla,

```
\begin{table}[h] \label{ML:tabla_escalada}
\centering
\scalebox{0.8}{\begin{tabular}{ccccccccccccc}
Est.&P.16(a)&Pr.14&Pr.16(b)&Pr.1&Pr.9&Pr.5&Pr.4&Pr.15&Pr.3&Pr.13&Pr.11&Pr.7&Cal.\hline
L & 0 & 0 & 0 & 2 & 2 & 2 & 1 & 2 & 2 & 2 & 54\\ \hline
S & 0 & 2 & 2 & 0 & 1 & 1 & 0 & 2 & 0 & 2 & 42\\ \hline
R & 0 & 1 & 0 & 0 & 1 & 1 & 2 & 0 & 2 & 2 & 38\\ \hline
Total& 0 & 1 & 2 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 10 & 14 & \\
\multicolumn{12}{l}{\{Abreviaturas. Est.: Estudiante, Pr.: Pregunta,Cal.: Calificación\}}\\
\end{tabular}}
\caption{Resultados del cuestionario diagnóstico}
\end{table}
```

produce:

Est.	P.16(a)	Pr.14	Pr.16(b)	Pr.1	Pr.9	Pr.5	Pr.4	Pr.15	Pr.3	Pr.13	Pr.11	Pr.7	Cal.
L	0			0	0	2	2	2	1	2	2	2	54
S	0			2	2	0	1	1	0	2	0	2	42
R	0	1	0	0	0	1	1	1	2	0	2	2	38
Total	0	1	2	2	3	4	5	6	7	8	10	14	

Abreviaturas. Est.: Estudiante, Pr.: Pregunta,Cal.: Calificación

**Tabla 4.6** Resultados del cuestionario diagnóstico

### Expresiones @{}.

En un ambiente tabular el separador de columnas se puede cambiar con una instrucción del tipo `@{txt}`. Este comando elimina la separación automática entre columnas y la reemplaza con el texto txt.

### Ejemplo 4.9

En el código que sigue, `r@{.}1` sustituye la columna central por un punto,

El texto:

produce:

```
\begin{tabular}{r@{.}1}
3 & 14159 \\
2 & 7182818\\
0 & 577216 \\
\end{tabular}
```

3.14159
2.7182818
0.577216

## 4.7 Espaciado en celdas.

A veces el texto matemático queda muy pegado a alguno de los bordes de las celdas, necesitamos hacer un poco de espacio hacia arriba, hacia abajo o variar el ancho de la celda. Para hacer esto usamos una instrucción del tipo `@{ }{ }`. En estas instrucciones también se puede incluir comandos, por ejemplo para agregar espacio horizontal se usa `@{\hspace{5cm}}{ }`.

La instrucción `@{\vrule height xpt depth ypt width zpt}` agrega espacio vertical: `height xpt`, espacio en el fondo: `depth ypt` y ancho: `width zpt`.

Aquí, la unidad de medida que usamos es  $1 \text{ pt} = \frac{1}{72} \text{ pulgada}$ . En el ejemplo que sigue se agrega 15pt de espacio vertical y 10pt en el fondo.

### Ejemplo 4.10

En este ejemplo tenemos una tabla problemática: el texto matemático está muy ajustado.

```
\begin{tabular}{l l l}\hline
\$\displaystyle\frac{x}{x+1} & \$\sqrt{x} & \$x^{2^n} \\ \hline
\end{tabular}
```

$$\begin{array}{ccc} \frac{x}{x+1} & \sqrt{x} & x^{2^n} \end{array}$$

Una solución es crear espacio vertical y en el fondo, en la tercera columna.

```
\begin{tabular}{l l l@{\vrule height 15pt depth 10pt width 0pt}}\hline
\$\displaystyle\frac{x}{x+1} & \$\sqrt{x} & \$x^{2^n} \\ \hline
\end{tabular}
```

$$\begin{array}{ccc} \frac{x}{x+1} & \sqrt{x} & x^{2^n} \end{array}$$

Otra solución para este ejemplo es incluir una línea ‘invisible’ de altura 0.8cm y ‘baseline’ -0.3:

```
\rule[-0.3cm]{0cm}{0.8cm}
```

```
\begin{tabular}{ccc}\hline
\$\displaystyle\frac{x}{x+1} & \$\sqrt{x} & \$x^{2^n} \rule[-0.3cm]{0cm}{0.8cm} \\ \hline
\end{tabular}
```

$$\begin{array}{ccc} \frac{x}{x+1} & \sqrt{x} & x^{2^n} \end{array}$$

## 4.8 Ancho de las columnas

Se puede controlar el ancho de las columnas con la instrucción `p{xcm}`,

### Ejemplo 4.11

El código:

```
\begin{tabular}{|p{3cm}|p{10cm}|}\hline
\textit{Representación} & \textit{Notación} \\ \hline
$R_{4-4,2}(O_6)$ & \par $R_{4-4,2}(O_6)$ &
Representación 3, en registro algebraico ($R^4$) en $\mathbb{R}$,  

interpretación de la letra como incógnita (2), de la relación  

entre volumen-altura-radio del vaso unidad ($O_6$):; $U = \pi 2r^2h$ \par
Representación 4, en registro algebraico ($R^4$) en el conjunto de  

los números reales (4), interpretación de la letra como incógnita (2),  

de la relación entre volumen-altura-radio del vaso unidad  

($O_6$): $h = U/\pi 2r^2$. \\ \hline
$R_{4-1,1}(O_8)$ & \\
Representación 6, en registro algebraico ($R^4$)  

en el conjunto de los números naturales (1), interpretación  

de la letra como evaluada (1), altura 10 y radio de los recipientes  

$3, 6, 9, 12$ $(O_8)$ \\ \hline
\end{tabular}
```

produce:

<i>Representación</i>	<i>Notación</i>
\$R_{4-4,2}(O_6)\$ \$R_{4-4,2}^4(O_6)\$	Representación 3, en registro algebraico (\$R^4\$) en \$\mathbb{R}\$, interpretación de la letra como incógnita (2), de la relación entre volumen-altura-radio del vaso unidad (\$O_6\$): \$U = \pi 2r^2h\$ Representación 4, en registro algebraico (\$R^4\$) en el conjunto de los números reales (4), interpretación de la letra como incógnita (2), de la relación entre volumen-altura-radio del vaso unidad (\$O_6\$): \$h = U/\pi 2r^2\$.
\$R_{4-1,1}(O_8)\$	Representación 6, en registro algebraico (\$R^4\$) en el conjunto de los números naturales (1), interpretación de la letra como evaluada (1), altura 10 y radio de los recipientes 3, 6, 9, 12 (\$O_8\$)

El siguiente ejemplo es un poco más elaborado,

**Nota:** Si las imágenes están en la primera columna es probable que éstas queden alineadas arriba de la caja y el texto quede alineado en el fondo de su caja. Una manera de resolver esto consiste en poner un nuevo comando en el preámbulo,

```
\def\imagetop#1{\vtop{\null\hbox{#1}}}
```

`\vtop` crea una caja vertical. Esta caja contiene dos elementos: una caja vacía horizontal (`\null`) y una caja horizontal con la imagen. También este caja se alinea con la primera caja horizontal vacía. Luego en el código, los gráficos los agregamos con `\imagetop{\includegraphics{...}}`

**Ejemplo 4.12 (Texto e imágenes en columnas)**

El código:

```
\begin{table}[h]
\centering
\begin{tabular}{|p{6.5cm}|p{6.5cm}|}\hline
\begin{center}
\includegraphics[width=6.5cm]{images/Utilizacion4}
\par\textrbf{PROPOSICI\'ON I}
\par\textrbf{Problema}
\end{center}
4. TOMAR la diferencia de varias cantidades sumadas,
o sustra\'idas ... & %Cambio de columna
\begin{center}
\includegraphics[width=6.5cm]{images/Utilizacion5}
\par\textrbf{REGLA I}
\par\textrbf{\textit{Para las cantidades sumadas o sustra\'idas}}
\end{center}
Tomemos la diferencia de cada t\'ermino de la cantidad
propuesta, y ...
\hline
\end{tabular}
\caption{La tabla muestra el modelo:...}\label{ML:tabla_escalada2}
\end{table}
```

produce:

<p><b>PROPOSITION I.</b> Problème.</p> <p>4. P <b>E</b>NTRER la différence de plusieurs quantités ajoutées ensemble, ou soustraites les unes des autres. Soit <math>a + x + y - z</math>, dont il faut prendre la différence. Si l'on suppose que <math>x</math> soit augmentée d'une portion infinitement petite; c'est-à-dire qu'elle devienne <math>x + dx</math>; <math>y</math> devient <math>y - dy</math>.</p> <p><b>PROPOSICIÓN I</b> <b>Problema</b></p> <p>4. TOMAR la diferencia de varias cantidades sumadas, o sustraídas ...</p>	<p><b>REGLE I.</b> <i>Pour les quantités ajoutées, ou soustraites.</i></p> <p>On prendra la différence de chaque terme de la quantité proposée, &amp; retenant les mêmes signes, on en composera une autre quantité qui sera la différence cherchée.</p> <p><b>REGLA I</b> <i>Para las cantidades sumadas o sustraídas</i></p> <p>Tomemos la diferencia de cada término de la cantidad propuesta, y ...</p>
--	---

**Tabla 4.7** La tabla muestra el modelo:...

En el ejemplo que sigue, la imagen en la primera columna se alinea en la parte superior de la caja con el comando `\imagetop` previamente definido en el preámbulo.

### Ejemplo 4.13 (Figuras en el margen izquierdo)

El código:

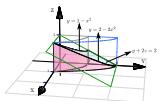
```
\hspace*{-2.8cm}
\begin{tabular}{p{2cm}p{13cm}} %\imagetop es un comando personal, ver más arriba.
\imagetop{\includegraphics[width=2cm]{images/exersolido21.pdf}}& Sólido $Q$\\
limitado por las superficies $y = 2 - 2x^2; $ $y = 1 - x^2; $ $y + 2z = 2; $\\
$; $ $x = 0 $ $y $ $z = 0; $ en el I octante.\\
\end{tabular}

\hspace*{-2.8cm}
\begin{tabular}{p{2cm}p{13cm}}
\imagetop{\includegraphics[width=2cm]{images-cap2/exersolido19.pdf}}& Sólido $Q$\\
limitado por las superficies $z = 4 - x^2/4; $ $y = 6 - x; $ $y $ $y = 4, $ en el primer\\
octante.\\
\end{tabular}

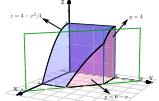
\hspace*{-2.8cm}
\begin{tabular}{p{2cm}p{13cm}}
\imagetop{\includegraphics[width=2cm]{images-cap2/exersolido20.pdf}}& Sólido $Q$\\
limitado por las superficies $z=4-x^2; $ $x+2y=4; $ $z=4 $ $y $ $z=y=0. $\\
\end{tabular}
```

produce (compilado con PDFLaTeX):

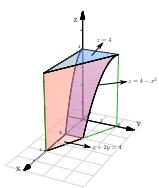
Sólido  $Q$  limitado por las superficies  $y = 2 - 2x^2$ ;  $y = 1 - x^2$ ;  $y + 2z = 2$ ;  $x = 0$  y  $z = 0$ ; en el I octante.



Sólido  $Q$  limitado por las superficies  $z = 4 - x^2/4$ ;  $y = 6 - x$ ;  $y$   $y = 4$ , en el primer octante.



Sólido  $Q$  limitado por las superficies  $z = 4 - x^2$ ;  $x + 2y = 4$ ;  $z = 4$  y  $z = y = 0$ .



## 4.9 Modo matemático en tablas con `tabularx`

A veces es conveniente usar el ambiente `tabular` habilitado para texto matemático. Esto se puede hacer con el paquete `tabularx`. Debemos poner en el preámbulo

```
\usepackage{tabularx}
```

Este paquete habilita el comando `newcolumntype` para definir columnas con contenido en modo matemático.

### Ejemplo 4.14

En este ejemplo se habilita el ambiente `tabular` para texto matemático. Observe que se usa una línea ‘invisible’ de altura `1cm` para crear espacio vertical en la columna donde el texto matemático quedaría algo ajustado. También se hace espacio hacia abajo con `-0.3cm`. El código:

```
\newcolumntype{D}{>{$\displaystyle c<{$}}%Las columnas M aceptan texto matemático |c|
\newcolumntype{M}{>{$1<{$}}.....%Las columnas N aceptan texto matemático |l|
% se usa 'tabular' normal.
\begin{tabular}{|D|D|D|D|M|M|r|}\hline
n &-1& 0 & 1 & 2 & 3 & 5 & 5 \\
a_n & 8 & 5 & 2 & 2 & 4 & n & n \\
r_n & 0 & 1 & 2 & 5 & 22 & 444& 444 \\
s_n & 1 & 0 & 1 & 2 & 9 & 7 & 7 \\
F & 0&-2&\frac{n}{4}& \frac{n}{4}& n^2& 2 & 2 \\
\end{tabular}
```

produce:

$n$	-1	0	1	2	3	5	5
$a_n$	8	5	2	2	4	$n$	$n$
$r_n$	0	1	2	5	22	444	444
$s_n$	1	0	1	2	9	7	7
$F$	0	-2	$\frac{n}{4}$	$\frac{n}{4}$	$n^2$	2	2

## 4.10 Problemas con los objetos flotantes: Paquete `float`

Es común tener problemas en la manera como LaTeX acomoda los gráficos. Una manera de tomar control sobre la ubicación de los gráficos es usar el paquete `float`; para esto, agregamos en *preámbulo*, `\usepackage{float}`. Ahora, en vez de digitar `\begin{table}[h]` o `\begin{tabular}[h]`, digitamos `\begin{table}[H]` o `\begin{tabular}[H]` (con H): El gráfico o la tabla quedará donde está haya o no haya espacio.

## 4.11 Tablas sofisticadas con $\text{\texttt{TikZ}}$

---

Podemos usar el paquete  $\text{\texttt{TikZ}}$  para agregar sofisticación a nuestras tablas. Por ejemplo, consideremos la siguiente tabla (idea de [Stefan Kottwitz en "TiKz examples"](#)),

Puesto	Distribución	Visitas diarias
1	Ubuntu	2114 ↓
2	Fedora	1451 ↑
3	Mint	1297 —
4	OpenSUSE	1228 ↑
5	Debian	910 ↓
6	Mandriva	907 ↑
7	PCLinuxOS	764 ↑
8	Puppy	738 ↑
9	Sabayon	671 ↑
10	Arch	625 ↓

Datos de DistroWatch.com. Abarcan los últimos 6 meses. Visitas por día

---

En el código reconocerá algunos comandos que ya hemos usado pero requiere conocer un poco del paquete  $\text{\texttt{TikZ}}$ . Sin embargo no hay problema en usar este código como plantilla, siempre y cuando se incluya los paquetes que se indican!.

```
\documentclass[xcolor=pdftex, x11names,table]{article}
\usepackage[latin1]{inputenc}
\usepackage{amsmath, amssymb, amsfonts, latexsym, stmaryrd}
\usepackage{tabularx}
\usepackage{colortbl}
\usepackage{tikz}
\usetikzlibrary{calc}
\pgfdeclarelayer{background}
\pgfdeclarelayer{foreground}
\pgfsetlayers{background,main,foreground}
%comandos especiales
\newcommand*\up{\textcolor{black}{\%
  \ensuremath{\uparrow}}}
\newcommand*\down{\textcolor{red}{\%
  \ensuremath{\downarrow}}}
\newcommand*\const{\textcolor{darkgray}{\%
  \textbf{--}}}

\begin{document}

\textbf{\large \color{Tan4} Ranking de distribuciones Linux.}
\textbf{\large \color{Tan4} Agosto 26, 2009.}

\begin{tikzpicture}
```

```

\node (tbl) { \rowcolors{1}{}{gray!5}
\begin{tabularx}{.8\textwidth}{cXrcc}
\textbf{\color{white} Puesto} & \textbf{\color{white} Distribución} \\
& \textbf{\color{white} Visitas diarias} & \\
1 & Ubuntu & 2114 & \downarrow \\
2 & Fedora & 1451 & \uparrow \\
3 & Mint & 1297 & \const \\
4 & OpenSUSE & 1228 & \uparrow \\
5 & Debian & 910 & \downarrow \\
6 & Mandriva & 907 & \uparrow \\
7 & PCLinuxOS & 764 & \uparrow \\
8 & Puppy & 738 & \uparrow \\
9 & Sabayon & 671 & \uparrow \\
10 & Arch & 625 & \downarrow \\
\end{tabularx}};

\begin{pgfonlayer}{background}
\draw[rounded corners,top color=red,bottom color=black,
    draw=white] ($(tbl.north west)+(0.14,0)$)
    rectangle ($(tbl.north east)-(0.13,0.9)$);
\draw[rounded corners,top color=white,bottom color=black,
    middle color=red,draw=blue!20] ($(tbl.south west)
    +(0.12,0.5)$) rectangle ($(tbl.south east)-(0.12,0)$);
\draw[top color=blue!1,bottom color=blue!20,draw=white]
    ($(tbl.north east)-(0.13,0.6)$)
    rectangle ($(tbl.south west)+(0.13,0.2)$);
\end{pgfonlayer}
\end{tikzpicture}
\end{center}
\small Datos de DistroWatch.com. Abarcan los últimos 6 meses.
Visitas por día
\normalsize
\end{document}

```

---





# 5

# INSERTAR GRÁFICOS Y FIGURAS EN DOCUMENTOS LATEX

## 5.1 Introducción

Además de texto corriente y texto en modo matemático, podemos insertar figuras tales como gráficos y figuras en los formatos ".pdf", ".png", ".jpg", ".gif", etc., o figuras nativas, generadas directamente con algún paquete LATEX (Tikz, PStricks, TeXDraw,...).

### ¿Cuál es el formato de imagen adecuado?

Los formatos .eps (Encapsulated PostScript) y .pdf son adecuados para las figuras usuales en matemáticas. Se ven bien en pantalla y son adecuados para la impresión. Para el manejo de imágenes generales (capturas de pantalla, fotografía, etc.) es conveniente usar el formato .png o .jpg.

## 5.2 ¿Cómo insertar las figuras?

Aquí vamos a describir la manera fácil de insertar figuras. Todo el manejo gráfico lo vamos a hacer usando el paquete `graphicx`.

En lo que sigue, vamos a considerar las siguientes tareas,

- ① Insertar figuras .eps (PostScript Encapsulado): Este es un formato de alta calidad y el de mayor soporte en LATEX (aunque el formato .pdf ha ganado mucho terreno).
- ② Insertar figuras .jpg, .png, .gif, .pdf, etc.
- ③ Insertar figuras cuando compilamos con `PDFLaTeX`
- ④ Convertir imágenes a otro formato con Software libre.
- ⑤ Extraer figuras de libros o de Internet.
- ⑥ Crear figuras nativas con [Tikz](#), [LaTeXDraw](#).

Si compilamos con **LaTeX**, obtendrá un DVI el cual podrá ver con un visualizador para DVI ([Yap](#) en Windows, [Okular](#) en Ubuntu, por ejemplo).

**LATEX** tiene un gran soporte para imágenes .eps. En muchos programas podemos guardar (o convertir) nuestros gráficos en este formato: [Mathematica](#), [MatLab](#), [QtOctave](#), [WinPlot](#), [Geogebra](#), etc. Este formato no es adecuado para incluir ‘pantallazos’ en un pdf.

### Convertir otras imágenes a formato .eps

Podemos convertir imágenes en formato .pdf, .png, .jpg, .gif, etc. a formato .eps ; esto se puede hacer con [Inkscape](#) (ver A.1) o con [Gimp](#) (software libre Ubuntu-Windows), solo debe abrir los archivos con alguno de estos programas (posiblemente editar algo adicional) y guardar como .eps

### Incluir las figuras

Para incluir las figuras en su documento, se debe agregar el paquete [graphicx](#) en el *preámbulo* y los gráficos se incluyen con el comando `\includegraphics`.

```
\includegraphics[opciones]{...}
```

Como es natural, los gráficos se deben *escalar* para que se acomoden al texto. El escalamiento se puede hacer especificando el ancho: `width =xcm` o especificando un porcentaje: `scale=x`.

#### Ejemplo 5.1

En este ejemplo vamos a incluir el gráfico `ubuntu.eps` que está en la subcarpeta `images` de la carpeta en la que está nuestro archivo `pruebal.tex`.

El texto:

---

```
\documentclass{article} %Compilar -> LaTeX
\usepackage{graphicx}
\begin{document}
Logo de Ubuntu: \includegraphics[width=1cm]{images/ubuntu.eps} \\
\end{document}
```

---

produce:



### Opciones adicionales

Cuando compilamos con **LaTeX** el comando `includegraphics` tiene varias opciones,

### Ejemplo 5.2

Continuando con el ejemplo anterior, vamos a incluir el gráfico ubuntu, que esta en la subcarpeta images de la carpeta en la que está nuestro archivo pruebal.tex, con un par de opciones más,

produce:

El texto:

```
\documentclass{article} %Compilar -> LaTeX
\usepackage{graphicx}
\begin{document}

Logo centrado y escalado a {\tt 3cm}

\begin{center}
\includegraphics[width=3cm]{images/ubuntu.eps}
\end{center}

En la figura \ref{fig:ubuntu} se
muestra el logo de Ubuntu, centrado y
escalado, en un ambiente {\tt figure}\backslash\backslash

\begin{figure}[h]
\centering
\includegraphics[scale=0.2]{images/ubuntu.eps}
\caption{Logo de Ubuntu}\label{fig:ubuntu}
\end{figure}
\end{document}
```

Logo centrado y escalado



En la figura 5.1 se muestra el logo de Ubuntu, centrado y escalado, en un ambiente figure



**Figura 5.1** Logo de Ubuntu

#### Opción

`width=xcm`

Escala la imagen especificando el ancho deseado y manteniendo la proporción.

`height=xcm`

Escala la imagen especificando el alto deseado y manteniendo la proporción.

`scale=x`

Escala la imagen según una proporción: `scale=0.5` indica escalar la imagen a la mitad y `scale=0.5` escala la imagen al doble.

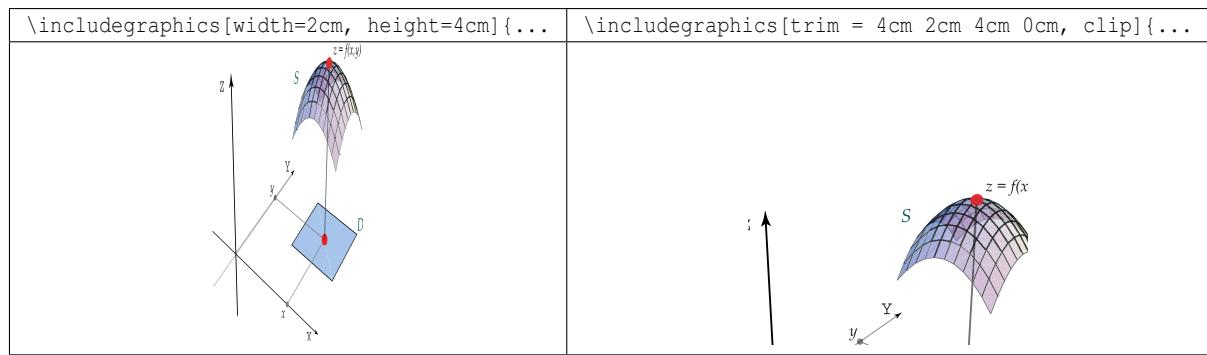
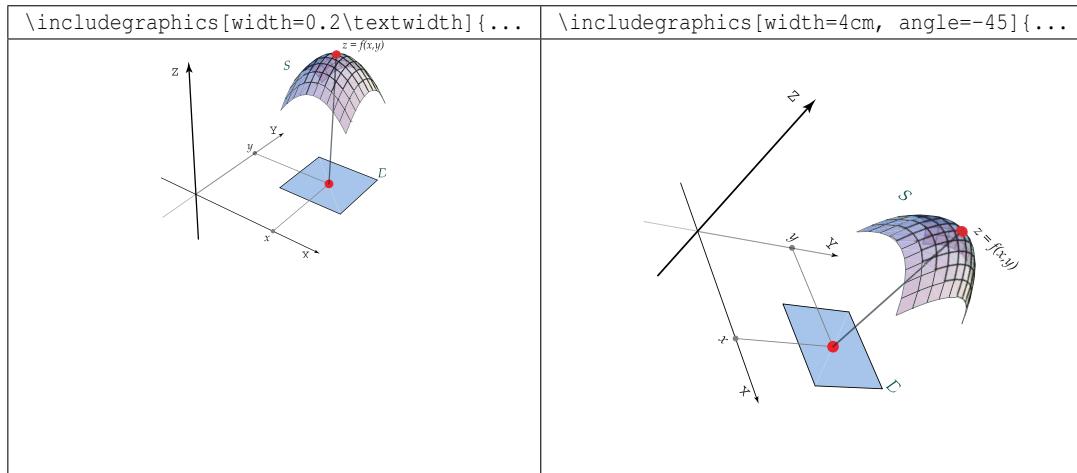
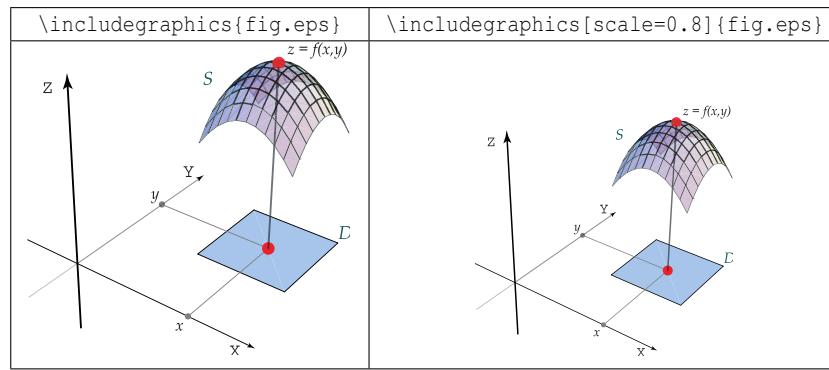
`trim=lcm bcm rcm tcm`

Esta opción recortar la imagen: `lcm` a la izquierda, `bcm` en la parte inferior, `rcm` a la derecha, y la `tcm` en la parte superior. `l`, `b`, `r` y `t` son longitudes.

`clip`

Para que la opción `trim` trabaje se debe establecer `clip=true`.

Veamos algunas de las opciones aplicadas a la figura sólido,

Compilando con  
**PDFLaTeX**

Este libro fue compilado con **PDFLaTeX**. Usa figuras .png, .pdf, .jpg y las figuras .eps fueron convertidas a .pdf con el paquete `epstopdf` (agregar `\usepackage{epstopdf}` en el preámbulo).

Si compilamos con **PDFLaTeX**, obtendrá un PDF el cual podrá ver con un visor para PDF ([Adobe Reader](#), por ejemplo). **PDFLaTeX** soporta los formatos .pdf, .png, .jpg, .gif. Para incluir figuras, sin reparar en la extensión, se debe poner en el *preámbulo*

```
\usepackage[pdftex]{graphicx}
\usepackage{epstopdf} % Convertir .eps a .pdf (si fuera necesario)
\DeclareGraphicsExtensions{.pdf,.png,.jpg,.eps} % busca en este orden!
```

Ahora, las figuras se pueden insertar como antes y con las mismas opciones,

<code>\includegraphics{images/geometrial}</code>	<code>\includegraphics[scale=0.8]{images/geometrial}</code>
<code>\includegraphics[width=0.2\textwidth]{...}</code>	<code>\includegraphics[width=4cm, angle=-45]{...}</code>
<code>\includegraphics[width=2cm, height=4cm]{...}</code>	<code>\includegraphics[trim = 4cm 2cm 4cm 0cm, clip]{...}</code>

### Ejemplo 5.3

En este ejemplo insertamos las figuras `geometrial.pdf`, `gausspng.png`, `liouvillejpg.jpg` y la figura `coedesfericas`. Observe que el comando `\DeclareGraphicsExtensions` requiere **no** especificar la extensión.

El código:

---

```

\documentclass{article}
\usepackage[pdftex]{graphicx}
\DeclareGraphicsExtensions{.pdf,.png,.jpg,.gif}
\usepackage{epstopdf}

\begin{document}

```

---

**Ejemplo 5.3 (continuación).**

```
\begin{center}
\begin{tabular}{|c|c|}\hline
\verb+\includegraphics[width=4cm]{images/geometrial}+ & \\
& \verb+\includegraphics[width=4cm]{images/gausspng}+\hline
\verb+\includegraphics[width=4cm]{images/geometrial}+ & \\
& \verb+\includegraphics[width=4cm]{images/gausspng}\hline
\verb+\includegraphics[width=4cm]{images/liouvillejpg}+ & \\
& \verb+\includegraphics[width=4cm]{images/coodesfericas}\hline
\verb+\includegraphics[width=4cm]{images/liouvillejpg}+ & \\
& \verb+\includegraphics[width=4cm]{images/coodesfericas}\hline
\end{tabular}
\end{center}
\end{document}
```

produce:

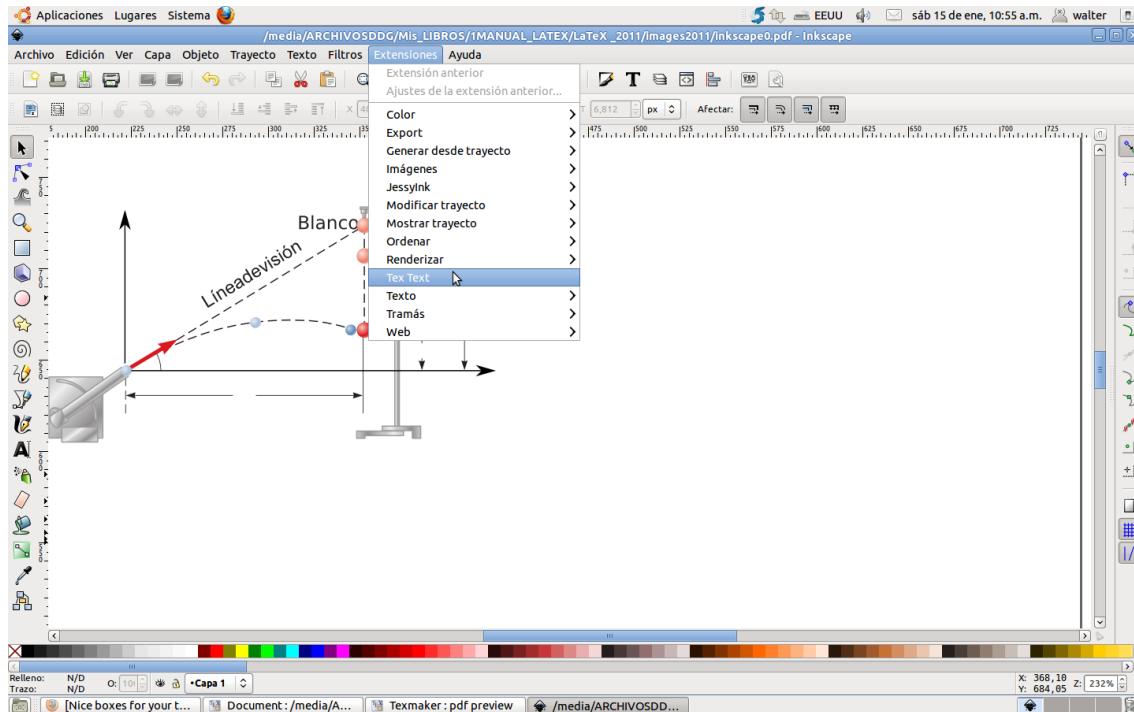
<code>\includegraphics[width=4cm]{images/fpdf}</code>	<code>\includegraphics[width=4cm]{images/fpng}</code>
<code>\includegraphics[width=4cm]{images/fjpg}</code>	<code>\includegraphics[width=4cm]{images/feps}</code>

## 5.3 Edición adicional de figuras con Inkscape

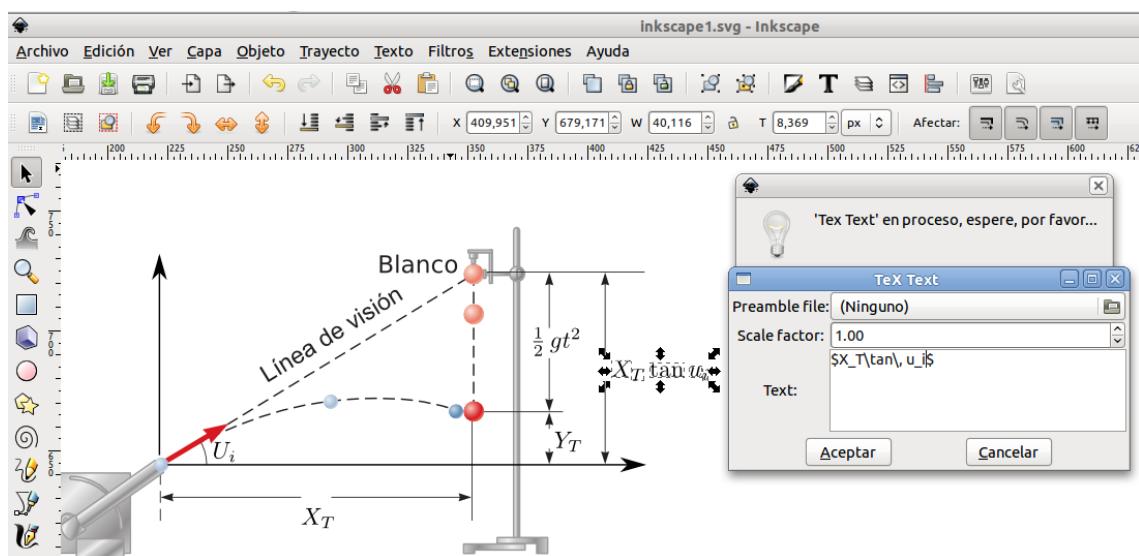
Los gráficos y figuras se pueden editar usando [Inkscape](#).

### Agregando texto matemático con TeXtext

[TeXText](#) es una extensión de [Inkscape](#). Una vez instalada (ver A.1) se accede a ella con [Herramientas-Tex text](#). Se puede usar para agregar texto [LaTeX](#).



La figura que sigue fue elaborada con [Inkscape](#), el texto matemático adicional se agregó con la extensión [TeXtext](#),



**Ejemplo 5.4**

La figura (5.2) fue editada con **Inkscape** y guardada en formato **.pdf** (este documento se compiló con **PDFLaTeX**)

```
\begin{figure} [h]
\begin{minipage} [b] {0.5\linewidth}
%newtheorem{teo}{Teorema} está en el preámbulo
\begin{teo}[Teorema del valor Medio]
Sea $f(x)$ continua en $[a,b]$ y derivable en $]a,b[$, $\\$ entonces $\exists \xi \in ]a,b[$ tal que

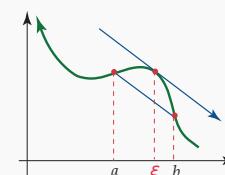
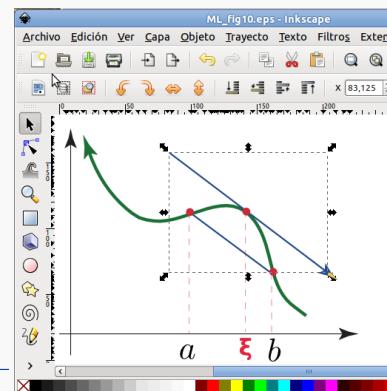
$$ f(b) - f(a) = f'(\xi)(b-a). \\ $\\ \end{teo}
\end{minipage}
\hfill\begin{minipage} [b] {0.45\linewidth}
\includegraphics[scale=0.7]{images/ML_fig10}.pdf
\caption{{\small Teorema del valor medio}}
\label{Calculo:fig...}
\end{minipage}
\end{figure}
```

produce:

**Teorema 5.1 (Teorema del valor Medio)** *Sea  $f(x)$  continua en  $[a,b]$  y derivable en  $]a,b[$ , entonces  $\exists \xi \in ]a,b[$  tal que*

$$f(b) - f(a) = f'(\xi)(b-a)$$

En particular, siendo  $f(x) = 6 - (x-2)^3 + x$ ,  $a = 2$  y  $b = 4$   $\Rightarrow \xi = \frac{2}{3}(3 + \sqrt{3})$ .

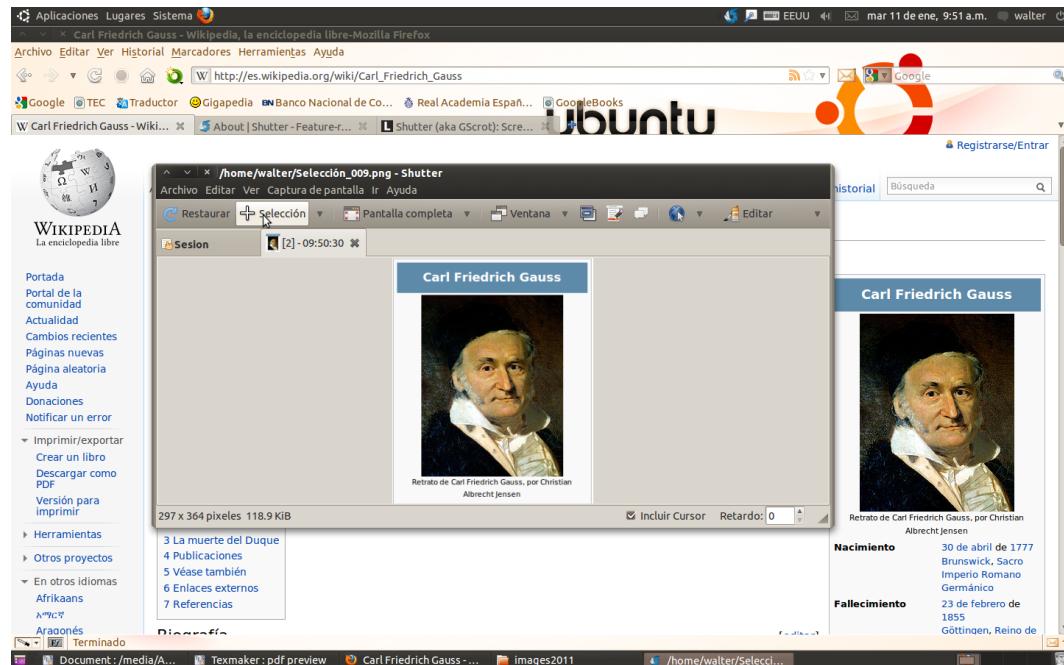


**Figura 5.2** Teorema del valor medio

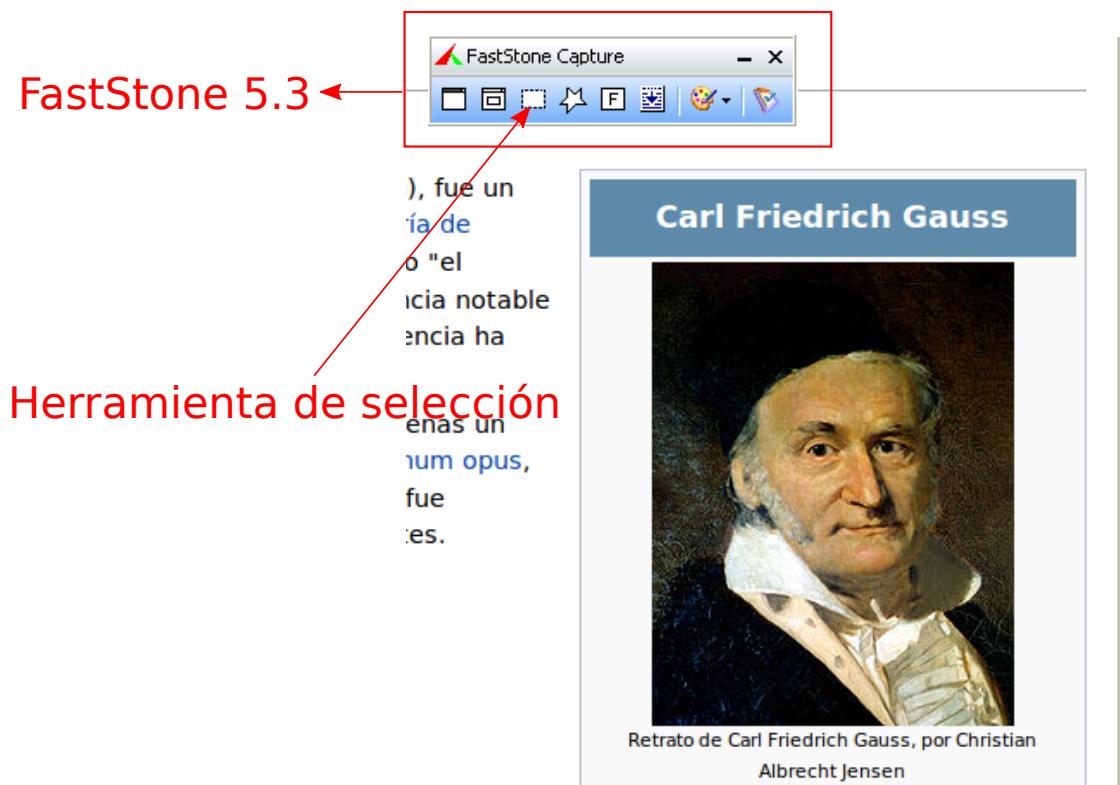
**Recortando figuras de la pantalla**

Entre las opciones que tenemos para recortar figuras en la pantalla están (ver A.1),

- **Shutter** para **Ubuntu**: **Shutter** es un programa de captura de pantalla con muchas características. Se puede hacer una captura de pantalla de un área específica, ventana, pantalla completa, o incluso de un sitio web y luego aplicar diferentes efectos a la misma.



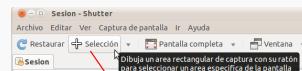
- FastStone 5.3 (la versión libre se puede buscar en Google exactamente con este nombre) para Windows: FastStone Capture es un programa de captura y edición de pantallas, similar a Shutter. La versión actual es la 6.7 pero la versión 5.3 es libre para uso personal y viene con la mayoría de las funciones que nos interesan.



### Ejemplo 5.5 (Usando Shutter para recortar)

He aquí un ejemplo de dos imágenes tomadas de Wikipedia ([www.wikipedia.org](http://www.wikipedia.org)). Para recortarlas de la pantalla de la PC se usó **Shutter** (estamos en Ubuntu!) y luego se guardó en formato .pdf (estamos compilando con **PDFLaTeX**),

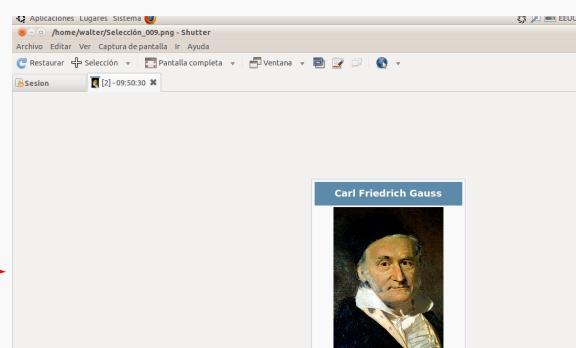
**Shutter**



Recorte con la herramienta selección [Enter]



La selección en Shutter



### Errores relacionados con "BoundingBox".

En la compilación, ya sea **LaTeX** o **PDFLaTeX**, a veces se observa el mensaje de error:

Error: Cannot determine size of graphic (no BoundingBox)

Esto sucede cuando una imagen no viene con las dimensiones (BoundingBox) de la caja.

La manera fácil de resolver este problema es abrir la imagen, digamos con **Inkscape** y guardar la imagen de nuevo.

Si abrimos la imagen en Adobe Illustrator, por ejemplo, para aplicar las dimensiones correctas, se debe ir a **File-Document Setup** y poner las dimensiones adecuadas para que la figura se ajuste a la región.

### Resolución de la imagen.

Los gráficos se ven bien en los formatos .pdf pero las imágenes se ven mejor en formato .png o .pdf. En general, la resolución mejora si las figuras originales son lo suficientemente grandes. Cuando se gana en resolución el tamaño del PDF aumenta.

## 5.4 De nuevo: Paquete **float**

Es común tener problemas en la manera como LaTeX acomoda los gráficos. Una manera de tomar control sobre la ubicación de los gráficos es usar el paquete **float**; para esto, agregamos en **preámbulo**, `\usepackage{float}`. Ahora, en vez de digitar `\begin{table}[h]` o `\begin{tabular}[h]`, digitamos `\begin{table}[H]` o `\begin{tabular}[H]` (con H): El gráfico o la tabla quedará donde está.

## 5.5 Paquete `subfigure`

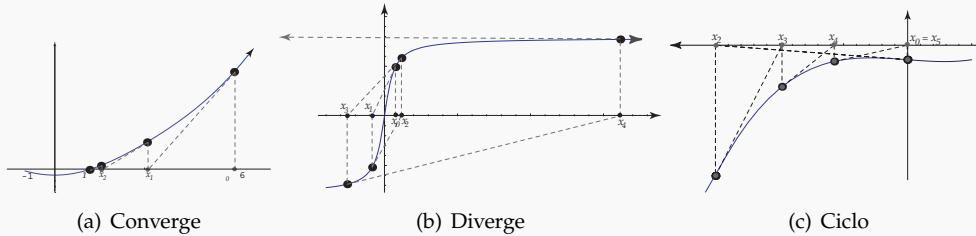
A veces tenemos varias figuras y nos encantaría poner un `\caption` a cada una en un mismo ambiente `figure`. Esto lo podemos hacer si usamos el paquete `\usepackage{subfigure}`. El siguiente ejemplo ilustra su uso.

### Ejemplo 5.6

El código:

```
\begin{figure}[h]
\centering
\subfigure[Converge]{\includegraphics[scale=0.5]{images/newton6}}
\subfigure[Diverge]{\includegraphics[scale=0.5]{images/newton5}}
\subfigure[Ciclo]{\includegraphics[scale=0.5]{images/newton4}}
\caption{Iteración de Newton}
\end{figure}
```

produce:



**Figura 5.3** Iteración de Newton

## 5.6 Los ambientes `wrapfigure` y `floatflt`

Otros ambientes flotantes muy útiles son `\wrapfigure` y `\floatflt`, para poderlos utilizar se deben cargar en el preámbulo sus paquetes respectivos con las instrucciones

```
\usepackage{wrapfig} %Figuras al lado de texto
\usepackage[rflt]{floatflt} %Figuras flotantes entre el texto
```

### `wrapfigure`

El ambiente `wrapfigure` permite incluir gráficos o texto en un recuadro al lado del documento, L<sup>A</sup>T<sub>E</sub>X se encarga de acomodar el texto del documento alrededor del recuadro introducido. Con este ambiente se introdujo la foto de D. Knuth al inicio de este documento.

**Ejemplo 5.7**

El código

---

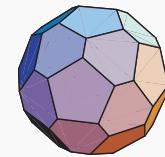
```
\begin{wrapfigure}{r}{2.5cm}
\includegraphics{images/ML_fig11}
\end{wrapfigure}
```

Al incluir un recuadro con ...

---

produce:

Al incluir un recuadro con `\wrapfigure` se debe tomar algunas cosas en cuenta: En la definición `{r}` significa que el recuadro se introducirá a la derecha del texto, también se puede utilizar `{l}` para que sea a la izquierda. El ambiente se debe iniciar entre párrafos, es decir, es problemático escribir un ambiente `wrapfigure` en medio de un párrafo.



El recuadro será introducido justo al lado del párrafo siguiente de la definición del ambiente.

La separación del recuadro con el texto está dado por la instrucción `\columnsep` del preámbulo.

En realidad este ambiente no es “flotante”, es decir, en este caso el recuadro se introduce justo en el párrafo donde uno quiere, por lo tanto, es nuestra responsabilidad que el recuadro se “vea” bien (que no quede cortado entre páginas o cosas de este estilo); se recomienda revisar todos los gráficos o texto introducido con este comando al obtener la versión final del documento.

Este comando es frágil, por lo que no se puede utilizar dentro de otros ambientes, sin embargo, sí se puede utilizar en párrafos con multicolumnas.

Por último, el ambiente puede presentar problemas cuando el texto alrededor del recuadro no lo cubre por completo, en estos casos es mejor utilizar `\parbox` o `minipage`.

Note que la gran ventaja que tiene este ambiente (sobre `\parbox`, por ejemplo) es que no hay que preocuparse por la cantidad de texto que hay en cada columna, LATEX se encarga de la distribución de manera automática.

**floatflt**

El ambiente `floatflt` es muy similar a `wrapfigure` ya que permite la inserción de un objeto flotante rodeado de texto; en este caso LATEX se encarga de acomodar el texto alrededor de él.

Para poder utilizar este ambiente se necesita incluir la librería, para esto, se coloca en el preámbulo la instrucción

```
\usepackage[rflt]{floatflt}
```

En donde el argumento opcional `rflt` indica que, por defecto, los gráficos se colocarán a la derecha del texto, también se puede escribir `lflt` para la izquierda o `vflt` que indica que el gráfico saldrá a la derecha en páginas impares y a la izquierda en páginas pares.

Por ejemplo, el texto:

---

```
\begin{floatingfigure}[r]{4.5cm}
\includegraphics{images/ML_fig12}
\caption{Un poliedro}
\end{floatingfigure}
```

Este ambiente sólo funciona ...

---

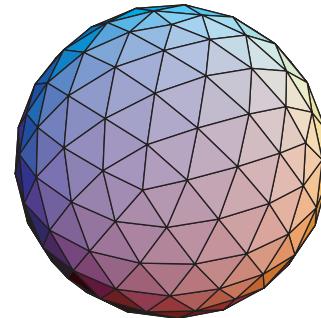
produce:

Este ambiente sólo funciona si se pone antes de un párrafo, la figura aparecerá lo más cerca del lugar en donde se haya escrito, esto quiere decir que L<sup>A</sup>T<sub>E</sub>X primero intenta poner la figura en la página actual, si no encuentra suficiente espacio vertical entonces la coloca en la página siguiente.

El argumento `[r]` es un argumento opcional que hace que el gráfico salga a la derecha del texto (no importa lo que se haya puesto al cargar la librería).

Aunque el ambiente `floatflft` sí trabaja en páginas a doble columna se debe tener cuidado si el gráfico sobrepasa el ancho de la columna porque sino el gráfico quedará encimado sobre la columna contigua. Tampoco se debe usar el ambiente muy cerca del final de una sección, sino el gráfico quedará encimado.

en la sección siguiente.  
Si se escribe el ambiente en el primer párrafo de una página es posible que el gráfico aparezca más abajo, es decir, el ambiente no coloca figuras al inicio de la página y, en el peor de los casos, la figura nunca aparecerá.



**Figura 5.4** Un poliedro

## 5.7 Crear figuras nativas con `Tikz`

---

El ambiente `picture` de L<sup>A</sup>T<sub>E</sub>X es un ambiente especial para insertar figuras implementados con comandos relativamente simples. Las figuras generadas en el ambiente `picture` de L<sup>A</sup>T<sub>E</sub>X quedan insertadas de manera automática en el documento. Programar los gráficos permite tener un control absoluto y preciso sobre todos los detalles, realizar gráficos sencillos es también muy rápido. Por otra parte, hay nuevo lenguaje que aprender, no tiene una interfaz gráfica y el código (por más sencillo que sea) no permite ver inmediatamente como se verá finalmente el gráfico. Existen varios editores que permiten hacer figuras y generan el código L<sup>A</sup>T<sub>E</sub>X, listo para introducirlo en nuestro documento.

### Paquete `Tikz`

Este es un paquete para crear gráficos para documentos L<sup>A</sup>T<sub>E</sub>X usando un ambiente '`tikzpicture`' y comandos especiales para dibujar líneas, curvas, rectángulos, etc. Muy adecuado para trabajar con presentaciones Beamer. Si desea hacer un documento PDF, tenga en cuenta que los gráficos

permanecen si compila con **PDFLaTeX** (se puede usar con Beamer) no así con  $\text{dvi} \leftrightarrow \text{pdf}$ .

Para usar el paquete se debe poner en el *preámbulo*

```
\usepackage{tikz}
```

La versatilidad de este paquete le permite crear gráficos hasta en el mismo texto usando el comando `\tikz`. Por ejemplo, podemos crear un círculo anaranjado como este: ● con el código

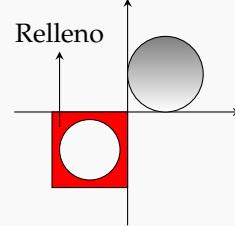
...como este:`\tikz \fill[orange] (1ex,1ex) circle (1ex);` con.... Aquí el ";" es necesario.

Para crear figuras complejas podemos usar el ambiente "tikzpicture".

Para dibujar líneas, rectángulo, círculos, etc., se usa `\draw` con las especificaciones respectivas. En el ejemplo que sigue se dibuja un par de ejes, con una flecha, usando dos líneas, una de  $(-1.5,0)$  a  $(1.5,0)$  y la otra  $(0,-1.5)$  a  $(0,1.5)$ . La flecha se agrega poniendo la opción " $[->]$ ". También vamos a dibujar un círculo (con un efecto de sombra) con centro en  $(0.5,0.5)$  de radio 0.5 y un rectángulo, con relleno rojo, con extremo inferior izquierdo en  $(-1,-1)$  y extremo superior derecho en  $(0,0)$  y un círculo con centro en  $(-0.5,-0.5)$ .

### Ejemplo 5.8

```
\begin{tikzpicture} [->]
\draw [->] (-1.5,0) -- (1.5,0);
\draw [->] (0,-1.5) -- (0,1.5);
\shadedraw [radius=0.5cm] (0.5,0.5) circle;
%Relleno
\filldraw [fill=red,even odd rule]
    (-1,-1) rectangle (0,0)
    (-0.5,-0.5) circle [radius=0.4cm];
\draw[->] (-0.9,-0.2) -- +(0,1)
    [above] node{Relleno};
\end{tikzpicture}
```



## Representación gráfica de una función con TikZ

Para graficar funciones usamos el comando `\plot`. La variable  $x$  se escribe en el código como `\x` y el dominio  $[a,b]$  se especifica con la opción `domain=a:b` del comando `draw`.

### Ejemplo 5.9

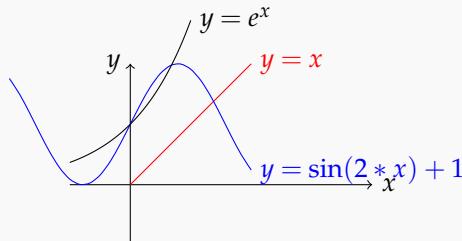
El código:

---

```
\begin{tikzpicture}[scale=0.8]
\draw[->] (-1,0) -- (4,0) node[right] {$x$}; % segmento de (-1,0) a (4,0)
\draw[->] (0,-1) -- (0, 2) node[above] {$y$};
% Dominio: domain = a:b?
\draw[smooth, domain = 0:2, color=red]
plot (\x, \x) node[right] {$y = x$};
\draw[smooth, domain = -2:2, color=blue]
% \x r indica que x se mide en radianes?
plot (\x,{sin(2*\x r)+1}) node[right] {$y = \sin(2*x)+1$};
\draw[smooth, domain = -1:1, color=black]
plot (\x,{exp(\x)}) node[right] {$y = e^x$};
\end{tikzpicture}
```

---

produce:



### Bolas 3D

En la sección de listas enumeradas (2.10) ya se indicó como crear listas con ‘bolas’ 3D usando comandos **TikZ**. Podemos agregar puntos individuales a los ítems, agregando al preámbulo los nuevos comandos

---

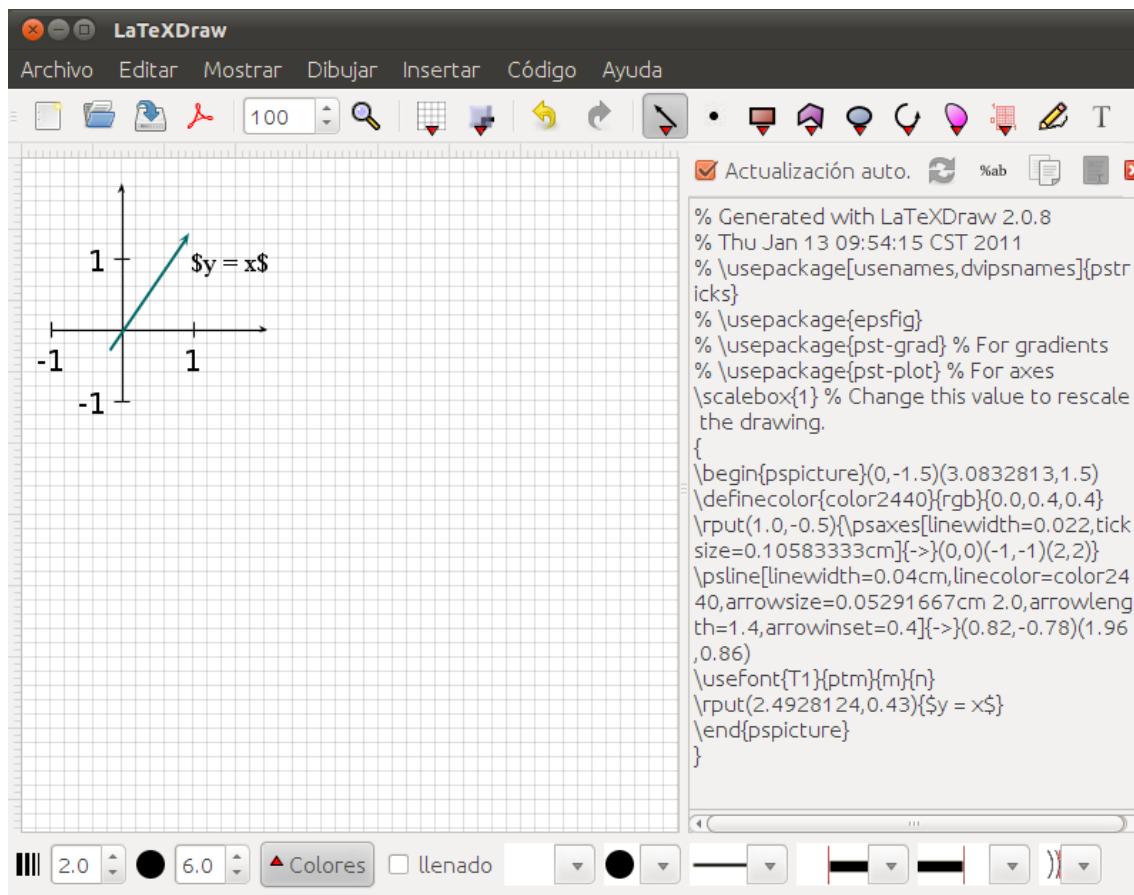
```
%Uso \item[\tpto]
\newcommand{\tpto}{\tikz \shadedraw [shading=ball] (0,0) circle (.1cm);}
%Uso \item[\ttpto{1}]
\newcommand{\ttpto}[1]{%
\begin{tikzpicture}
\node[scale=.5, circle, shade, ball color=blue] {\color{white}\Large\textrm{\bf #1}};
\end{tikzpicture}}
```

---

Un manual muy detallado de **TikZ** se puede obtener en <http://www.ctan.org/tex-archive/help/Catalogue/entries/pgf.html>

### Otras programas

- **PStricks**. Paquete parecido, en general, a **TikZ**
- **LaTeXDraw**. Este es un editor visual gratuito y multiplataforma (implementado en Java) basado en **PStricks**. **LaTeXDraw** genera el código **LaTeX** de las figuras.  
LaTeXDraw se puede obtener en <http://latexdraw.sourceforge.net/download.html> (debe tener habilitado Java <http://www.java.com/es/download/>).



**Figura 5.5** Editor LaTeXDraw

El ambiente es similar a ambientes comerciales de dibujo. Las opciones de dibujo se obtiene con el clic derecho. El código LaTeX de la figura (para pegar en nuestro documento) se puede obtener con el menú Code – copy all the code.

**Nota:** Para generar un PDF correcto se debe hacer primero dvi → ps y luego ps → pdf.

Para usar [LaTeXDraw](#) debemos poner en el *preámbulo*

```
\usepackage[usenames,dvipsnames]{pstricks}
\usepackage{epsfig}
\usepackage{pst-grad} % Para gradientes
\usepackage{pst-plot} % Para ejes
```

- [Inkscape](#). Puede exportar sus figuras en formato [PsTricks](#).
- [PiCTex](#). Es un paquete con una colección de macros [TEX](#) para gráficos.  
Un documento detallado lo puede encontrar en [La revista digital de Matemática](#).

### Ejemplo 5.10

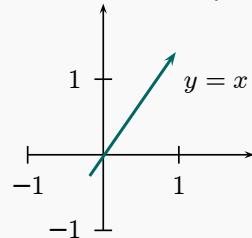
De a figura que se ve en el editor [LaTeXDraw](#), en el ejemplo (5.7), se obtiene el código

---

```
\fboxsep 12pt \fboxrule 1pt%
\scalebox{1} % Change this value to rescale the drawing.
{
\begin{pspicture}(0,-1.5)(3.101875,1.5)
\definecolor{color2440}{rgb}{0.0,0.4,0.4}
\rput(1.0,-0.5){\psaxes[linewidth=0.022,
    ticksize=0.10583333cm]{->}(0,0)(-1,-1)(2,2)}
\psline[linewidth=0.04cm, linecolor=color2440,
    arrowsize=0.05291667cm 2.0,
    arrowlength=1.4, arrowinset=0.4]{->}(0.82,-0.78)(1.96,0.86)
\usefont{T1}{ptm}{m}{n}
\rput(2.5114062,0.43){$y = x$}
\end{pspicture}
}
```

---

Ahora este código lo pegamos en nuestro documento y compilamos (con [LaTeX](#)) y obtenemos,





# 6 DISEÑO EDITORIAL

---

Cuando escribimos un documento  $\text{\LaTeX}$  no hay que preocuparse, en general, por el diseño del documento,  $\text{\LaTeX}$  se encarga y aplica el diseño establecido según el ``\documentclass''. Una vez que tenemos resuelto el contenido tenemos que fijarnos en el estilo y en el diseño editorial. Es un hecho comprobado que una representación que funciona mejor es también percibida como más bella. La sensación de belleza viene de la facilidad de uso, es decir; lo fácil de usar, generalmente, es placentero. Uno no quiere leer una página mal organizada en el que no haya una jerarquía clara, queremos leer páginas placenteras que se vean claras y bien arregladas. Ahora podemos analizar si estamos comunicando de manera agradable y efectiva la información. A los ojos les gusta ver orden, esto crea una sensación de calma y seguridad. También les gusta ver contraste: Frecuentemente en una misma página contamos varias historias y esto puede suceder hasta en un solo párrafo o una fórmula; el contraste ayuda al lector a ver la lógica y el flujo de las ideas y a organizar la información y crea de paso un interés. Hay algunos principios que nos ayudan a definir la manera en que vamos a organizar y presentar la información, de eso se trata este capítulo.

## 6.1 Bajar la carga cognitiva

---

La carga cognitiva está conformada por las demandas que se imponen a la memoria de trabajo durante una observación o aprendizaje. Aquí hay dos conceptos que nos interesan, la carga cognitiva intrínseca y la carga cognitiva extrínseca.

**Carga intrínseca:** Está determinada “por la naturaleza del material y la experticia del aprendiz”

**Carga extrínseca:** La carga cognitiva extrínseca está asociada con procesos que no tienen relación directa con el aprendizaje del concepto mismo sino más bien con la lectura del material.

En principio no podemos hacer mucho en lo que respecta a la carga intrínseca pero si podemos hacer algo para reducir la carga cognitiva extrínseca. El diseño editorial que nos ocupa esta orientado a reducir la carga cognitiva extrínseca presentando la información de un modo acorde con el modo en que los seres humanos percibimos el mundo.

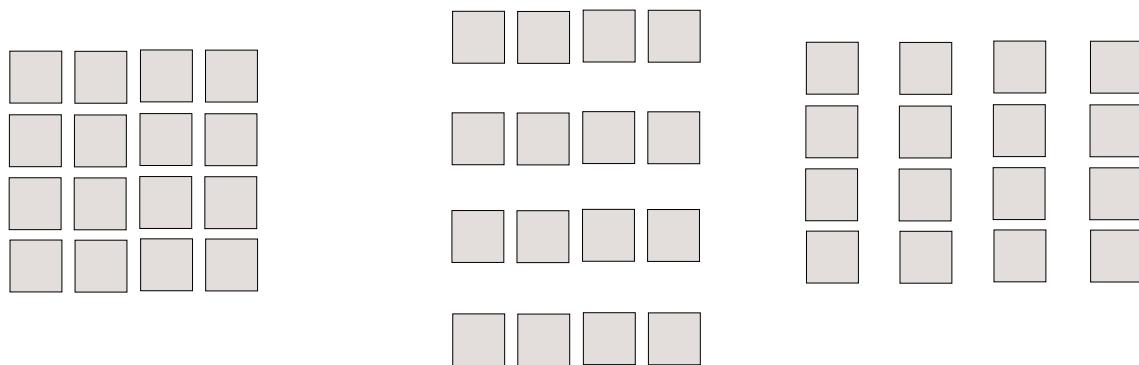
### Las leyes Gestalt

Leyes de percepción Gestalt ('forma') tratan de explicar cómo el cerebro humano percibe e inter-

preta el mundo y forman parte del conglomerado básico de la teoría del diseño. Dos leyes nos interesan en particular, la ley de proximidad y la ley de semejanza.

**Ley de la proximidad:** El principio de proximidad dicta que los objetos más cercanos (en tiempo o espacio) se perciben como un grupo.

Por ejemplo, en la figura que sigue el arreglo de la izquierda no presenta una pertenencia clara, en el centro se acentúa la pertenencia por filas y a la derecha la pertenencia por columnas.



Ahora veamos un ejemplo en el contexto de las matemáticas: El siguiente texto presenta ambigüedad en la pertenencia,

$$\begin{aligned} \left(x + \frac{b}{2}\right)^2 - \frac{b^2}{4} + c &= \left[x^2 + 2(x)\left(\frac{b}{2}\right) + \left(\frac{b}{2}\right)^2\right] - \frac{b^2}{4} + c \\ &= \left[x^2 + bx + \frac{b^2}{4}\right] - \frac{b^2}{4} + c \\ &= x^2 + bx + c \end{aligned}$$

Para mejorar la pertenencia horizontal creamos espacio,

$$\begin{aligned} \left(x + \frac{b}{2}\right)^2 - \frac{b^2}{4} + c &= \left[x^2 + 2(x)\left(\frac{b}{2}\right) + \left(\frac{b}{2}\right)^2\right] - \frac{b^2}{4} + c \\ &= \left[x^2 + bx + \frac{b^2}{4}\right] - \frac{b^2}{4} + c \\ &= x^2 + bx + c \end{aligned}$$

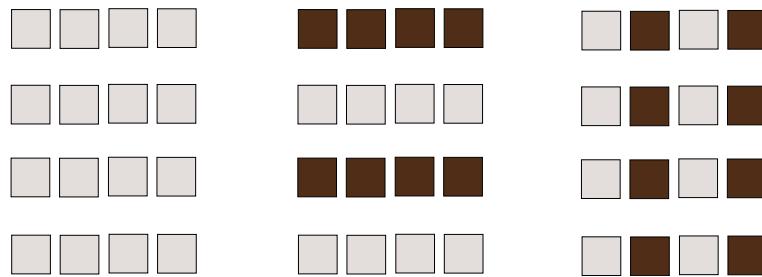
**Ley de la semejanza:** El principio de semejanza dicta que los objetos que poseen características similares de forma, color, luminosidad o tamaño aparentan pertenencia.

En la figura de abajo, el grupo de cuadrados aparece dividido en dos grupos: Los oscuros y los claros.



**Proximidad versus semejanza:** Las leyes gestalt pueden reafirmarse o balancearse si se combinan. En la figura que sigue, se tiene una combinación de la ley de semejanza versus la ley de proximidad: Los cuadrados tienen una relación hacia las líneas (horizontales) más que a las columnas (verticales), en el segundo esquema esta relación se reafirma con color (ley de semejanza), sin embargo, en el tercer esquema a pesar de que la relación de proximidad no ha cambiado las

columnas operan con más fuerza que las líneas, es decir la similaridad actúa por encima de la proximidad.



Como un ejemplo, en el siguiente texto se usa lo que se llama un ‘código cromático’ (provocar impacto visual mediante la manipulación del color), que permite al lector tener una clara distinción de los tipos de elementos sin ni siquiera pensar en ello.

$$\begin{aligned}
 4a^2x + 3bm - 4ab - 3max &= (4a^2x - 4ab) + (3mb - 3max) \\
 &= 4a(ax - b) + 3m(b - ax) \\
 &= 4a(ax - b) - 3m(ax - b) \\
 &= (4a - 3m)(ax - b)
 \end{aligned}$$

## 6.2 Amenidad: Los Cuatro Principios Básicos

---

### 6.2.1 Proximidad.

El propósito básico de la proximidad es organizar. La idea es agrupar varios items relacionados de tal manera que se visualicen como una sola unidad. Tal vez, el ejemplo más sencillo es una tarjeta de presentación. En la figura 6.1-(a), se muestra una tarjeta con varios elementos sin agrupar, en la parte (b) se muestra con los elementos agrupados según su proximidad.



Figura 6.1

### 6.2.2 Alineamiento.

El propósito del alineamiento es unificar y organizar cada página. Nada se debe colocar de manera arbitraria, los elementos deben tener conexión visual con los otros elementos.

### Alineamiento incorrecto,

Por ejemplo, para construir una tabla parcial en base  $b = 2$  módulo 13, calculamos las potencias de 2 módulo 13.

$$\begin{aligned} 2 &\equiv 2^1 \pmod{13}, & 11 &\equiv 2^7 \pmod{13}, \\ 4 &\equiv 2^2 \pmod{13}, & 9 &\equiv 2^8 \pmod{13}, \\ 8 &\equiv 2^3 \pmod{13}, & 5 &\equiv 2^9 \pmod{13}, \\ 3 &\equiv 2^4 \pmod{13}, & 10 &\equiv 2^{10} \pmod{13}, \\ 6 &\equiv 2^5 \pmod{13}, & 7 &\equiv 2^{11} \pmod{13}, \\ 12 &\equiv 2^6 \pmod{13}, & 1 &\equiv 2^{12} \pmod{13}. \end{aligned}$$

Luego, ponemos la información en una tabla,

$a$	1	2	3	4	...
$\text{Ind}_2(a)$	12	1	4	2	...

### Alineamiento correcto,

Por ejemplo, para construir una tabla parcial en base  $b = 2$  módulo 13, calculamos las potencias de 2 módulo 13.

$$\begin{aligned} 2 &\equiv 2^1 \pmod{13}, & 11 &\equiv 2^7 \pmod{13}, \\ 4 &\equiv 2^2 \pmod{13}, & 9 &\equiv 2^8 \pmod{13}, \\ 8 &\equiv 2^3 \pmod{13}, & 5 &\equiv 2^9 \pmod{13}, \\ 3 &\equiv 2^4 \pmod{13}, & 10 &\equiv 2^{10} \pmod{13}, \\ 6 &\equiv 2^5 \pmod{13}, & 7 &\equiv 2^{11} \pmod{13}, \\ 12 &\equiv 2^6 \pmod{13}, & 1 &\equiv 2^{12} \pmod{13}. \end{aligned}$$

Luego, ponemos la información en una tabla,

$a$	1	2	3	4	...
$\text{Ind}_2(a)$	12	1	4	2	...

### 6.2.3 Repetición.

La repetición es una poderosa manera de ser *consistente*. El propósito de la repetición es unificar y agregar interés visual. Los elementos repetitivos pueden ser fuentes en negrita, líneas delgadas, viñetas, encabezados, márgenes, color, fuentes, etc.

### 6.2.4 Contraste.

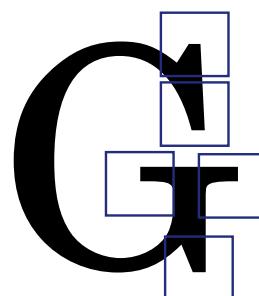
El propósito del contraste es crear interés en la página y, al mismo tiempo, ayudar en la organización. El contraste puede ser creado de varias maneras: Fuentes grandes con fuentes pequeñas, color, elementos horizontales o verticales, etc.

1. Divisibilidad por 9 : 9 divide a  $a$  si y sólo si 9 divide la suma de sus dígitos, es decir,  $9|a \iff 9|\sum_{i=0}^n a_i$
2. Divisibilidad por 3 : 3 divide a  $a$  si y sólo si 3 divide la suma de sus dígitos.
3. Divisibilidad por 2 y por 5 : tanto 2 como 5 dividen a  $a$  si y sólo si dividen  $a_0$ .

1. **Divisibilidad por 9**: 9 divide a  $a$  si y sólo si 9 divide la suma de sus dígitos, es decir,  $9|a \iff 9|\sum_{i=0}^n a_i$
2. **Divisibilidad por 3**: 3 divide a  $a$  si y sólo si 3 divide la suma de sus dígitos.
3. **Divisibilidad por 2 y por 5**: tanto 2 como 5 dividen a  $a$  si y sólo si dividen  $a_0$ .

## 6.3 Legibilidad: Cómo escoger las fuentes.

Los cuatro principios básicos de los que hablamos anteriormente son una guía para la *amenidad*. Ahora nos interesa la legibilidad. La legibilidad nos debe guiar en la selección de tipo de letra. Hay tres tipos de letra que podemos usar: **Serif** (letras con serifas o ‘remates’, como Times o Palatino), **sans serif** (letras sin serifas ‘remates’ como Helvetica o Arial) y **decorativa**. El texto **serif** se considera el más fácil de leer en los textos impresos y cansa menos la vista cuando se trata de grandes bloques, pues está diseñado para ayudar al lector a identificar y discriminar entre las letras específicas, por eso se usa en cuerpo del texto. Proporciona efecto de tranquilidad, modifica el significado del texto y añade connotaciones específicas a lo expresado.



El texto `sans serif` es el segundo tipo más leíble de texto y por lo general se utiliza para los títulos y subtítulos de las secciones y se aplica también a textos con fuente muy pequeña (manuales de artefactos electrónicos, medicinas, etc.) También es un texto adecuado para textos cortos en pantalla. Este tipo de fuente crea el efecto de modernidad, sobriedad, alegría y seguridad. Hay que tomar en cuenta que diferentes tipografías atraen audiencias diferentes, tanto de manera subliminal como abiertamente (piense en una tipografía para niños).



Cuando se trata de seleccionar tipos de letras para un folleto o un libro, una regla general que se menciona es la siguiente,

- ➊ Utilice un tipo de letra serif simple, fácilmente reconocido para el cuerpo del trabajo (Times, Times New Roman, Palatino,...).
- ➋ Divida el texto con títulos y subtítulos en tipo de letra sans serif (puede ser Arial o helvetica, aunque hay otras).

`LATEX` no usa las fuentes del sistema sino que tiene sus propias fuentes. La fuente default de `LATEX` es Computer Modern, tal vez por esto sea una fuente usada en exceso. Para optimizar la calidad de impresión y también de visualización en pantalla (vía PDF), es adecuado forzar `LATEX` para que use fuentes "postscript" (que vienen con las distribuciones actuales, por ejemplo MikTeX 2.x o TeXLive 2010). Esto se puede hacer, por ejemplo, usando algún paquete. Por ejemplo, el paquete `pslatex` o la familia de fuentes `PSNFSS`. En la documentación de cada paquete se encuentran algunos detalles adicionales relacionados con la codificación u otra consideraciones. Algunos ejemplos son,

- El paquete `pslatex`: La fuente default es "Times".  
Agregamos en el preámbulo `\usepackage{pslatex}`
- El paquete `mathpazo`: La fuente default es "Palatino".  
Agregamos en el preámbulo `\usepackage{mathpazo}`

¿Qué significa "tomar un número natural al azar"? Los naturales son un conjunto infinito, así que no tiene sentido decir que vamos a tomar un número al azar. Lo que si podemos es tomar un número de manera aleatoria en un conjunto finito  $\{1, 2, \dots, n\}$  y luego (atendiendo a la noción frecuencista de probabilidad) ver que pasa si  $n$  se hace grande (i.e.  $n \rightarrow \infty$ ).

- El paquete `mathptmx`: La fuente default es "Times".  
Agregamos en el preámbulo `\usepackage{mathptmx}`

¿Qué significa "tomar un número natural al azar"? Los naturales son un conjunto infinito, así que no tiene sentido decir que vamos a tomar un número al azar. Lo que si podemos es tomar un número de manera aleatoria en un conjunto finito  $\{1, 2, \dots, n\}$  y luego (atendiendo a la noción frecuencista de probabilidad) ver que pasa si  $n$  se hace grande (i.e.  $n \rightarrow \infty$ ).

- El paquete `bookman`: La fuente default es "Bookman".  
Agregamos en el preámbulo `\usepackage{bookman}`

---

¿Qué significa “tomar un número natural al azar”? Los naturales son un conjunto infinito, así que no tiene sentido decir que vamos a tomar un número al azar. Lo que si podemos es tomar un número de manera aleatoria en un conjunto finito  $\{1, 2, \dots, n\}$  y luego (atendiendo a la noción frecuencista de probabilidad) ver que pasa si  $n$  se hace grande (i.e.  $n \rightarrow \infty$ ).

---

- El paquete `newcent`: La fuente default es ‘New Century Schoolbook’. Agregamos en el preámbulo `\usepackage{newcent}`

---

¿Qué significa “tomar un número natural al azar”? Los naturales son un conjunto infinito, así que no tiene sentido decir que vamos a tomar un número al azar. Lo que si podemos es tomar un número de manera aleatoria en un conjunto finito  $\{1, 2, \dots, n\}$  y luego (atendiendo a la noción frecuencista de probabilidad) ver que pasa si  $n$  se hace grande (i.e.  $n \rightarrow \infty$ ).

---

## 6.4 Color

---

Esta podría ser la parte más difícil del diseño. El color es una parte inseparable de nuestras vidas y es una parte de todo lo que percibimos. El color tiene un fuerte impacto en nuestras emociones y sentimientos y se puede considerar como un elemento de diseño que se puede utilizar para crear ambientes de aprendizaje mejorados.

Si vamos a usar color, lo mejor es seguir un “esquema de color”. En principio usamos un esquema de color simple: Fondo blanco con letras negras. Esquemas más avanzados involucran la combinación de varios colores. En internet podemos obtener esquemas ya hechos, como el de la figura (6.2).

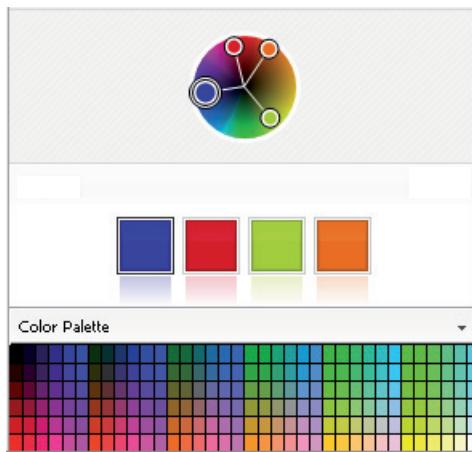


**Figura 6.2** Esquema generado con “Color Scheme designer”

Podemos también crear esquemas personalizados basados en varios esquemas, por ejemplo

- ① Esquema acromático: Utiliza sólo el negro, el blanco, y los grises.
- ② Esquema análogo: Utiliza cualquiera de tres tonos consecutivos o cualquiera de sus tintes y matices del círculo cromático.
- ③ Esquema complementario: Usa los opuestos directos del círculo cromático.
- ④ Esquema complementario dividido: Consta de un tono y los dos tonos a ambos lados de su complemento.

Hay algunos sitios en Internet con esquemas de color ya hechos, como “Color Schemer Studio”, y también hay aplicaciones “online”, que nos permiten crear esquemas, como ‘Colors on the Web’ (<http://www.colorsontheweb.com>) o ‘Color Scheme designer’ (<http://colorschemedesigner.com/>)



**Figura 6.3** Color Scheme Studio

Cuando agregamos color a una imagen, es usual observar la codificación del color (tres números) en varios modelos de color. Aquí son de interés dos modelos: RGB (acrónimo de red, green y blue) y CMYK (acrónimo de Cyan, Magenta, Yellow y Key). RGB es la elección si el material se va visualizar en pantalla y CMYK es la elección si es para impresión.

## 6.5 Editar un PDF con Inkscape

---

Después de generar una archivo PDF, se puede agregar detalles de diseño (color, figuras, imágenes, retoques, etc.) abriendo cada página del archivo PDF en [Inkscape](#). Hay otro tipo de software, como Adobe Pro Extended, PitStop, etc., que permite agregar ligas, corrección de errores menores, agregar anotaciones, agregar video (por ejemplo video flash), etc.

El primer problema es el de las fuentes, [Inkscape](#) no puede acceder a las fuentes del documento y lo que hace es una sustitución de fuentes; esto significa que cuando [Inkscape](#) importa un PDF, lee los nombres de las fuentes presentes en el documento (eso si se puede hacer) y sustituye estos nombres con los nombres (lo más parecido que encuentre) de las fuentes instaladas en su sistema (ver '[Inkscape: PDF import](#)').

Por ejemplo, si el PDF usa la fuente “TimesNewRomanPSMT” y lo más cercano que tenemos a este nombre es “Times New Roman”, entonces esta será la fuente que se usará.

En general esta sustitución mejora si instalamos algunas fuentes adicionales en el sistema tal y como se recomienda en [A.1](#); pero *esto no es perfecto*. En todo caso se pueden instalar las extensiones `TeXtext` y `Replace Font`, de esta manera podemos generar fragmentos de texto `LATeX` que se pierdan o reemplazar fuentes por la fuente correcta.

En la figura que sigue se muestra un PDF generado con `PDFLaTeX` y cómo lo levanta [Inkscape](#). Se usó el paquete de fuentes `mathpazo` porque las fuentes `PazoMath.ttf` son fuentes TrueType y están instaladas en mi sistema e [Inkscape](#) las reconoce bien. Este paquete cambia la fuente default a `Adobe Palatino` y usa las fuentes `mathpazo` para las matemáticas (no todo, pero bastante). Si faltan algunos símbolos, se pueden reeditar con la extensión `TeXtext`. El documento fue generado con el preámbulo

```
\documentclass[xcolor=pdftex, x11names,table]{book}
\usepackage{amsmath, amssymb, amsfonts, latexsym, stmaryrd}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
```

```
\usepackage{xcolor}
\usepackage{mathpazo}
\begin{document}

....
```

$0^0$  es una expresión indefinida. Si  $a > 0$  entonces  $a^0 = 1$  pero  $0^a = 0$ . Sin embargo, convenir en que  $0^0 = 1$  es adecuado para que algunas fórmulas se puedan expresar de manera sencilla, sin recurrir a casos especiales, por ejemplo

$$\begin{aligned} e^x &= \sum_{n=0}^{\infty} \frac{x^n}{n!} \\ (x+a)^n &= \sum_{k=0}^n \binom{n}{k} x^k a^{n-k} \\ \prod_{\substack{i=0 \\ i \neq k}}^n \frac{w_i}{(w_i - w_k)} \end{aligned}$$

Expresión	Código
$a \xrightarrow{f} b$	
$\lim_{x \rightarrow 0} f(x)$	
$\binom{a}{b}$	
$\sum_{\substack{0 < i < m \\ 0 < j < n}} a_i b_j$	
$\prod_{\substack{i=0 \\ i \neq k}}^n \frac{w_i}{(w_i - w_k)}$	

Figura 6.4 PDF original con fuentes MathPazo.ttf

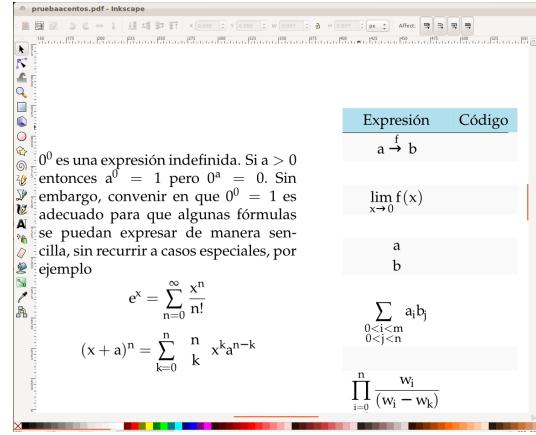


Figura 6.5 PDF en Inkscape.

Utilizando otros paquetes de fuentes los resultados más bien pueden ser desalentadores. En todo caso, una vez que hemos decidido levantar el PDF con [Inkscape](#), es conveniente dividirlo en páginas individuales y editar y guardar cada página individual, luego se vuelven a unir; todo esto se hace con [Pdfsam](#) (ver A.1).

### Otra solución: Convertir a contornos.

Otra opción que obvia el problema de las fuentes es convertir el texto en contornos. Una manera de hacer esto es generar un archivo .ps y luego aplicar una instrucción postscript para generar el archivo de contornos. Este archivo se puede convertir a formato PDF y lo podemos editar con [Inkscape](#) sin ningún problema. En general, el archivo resultante es de buena calidad y de tamaño similar al PDF original.

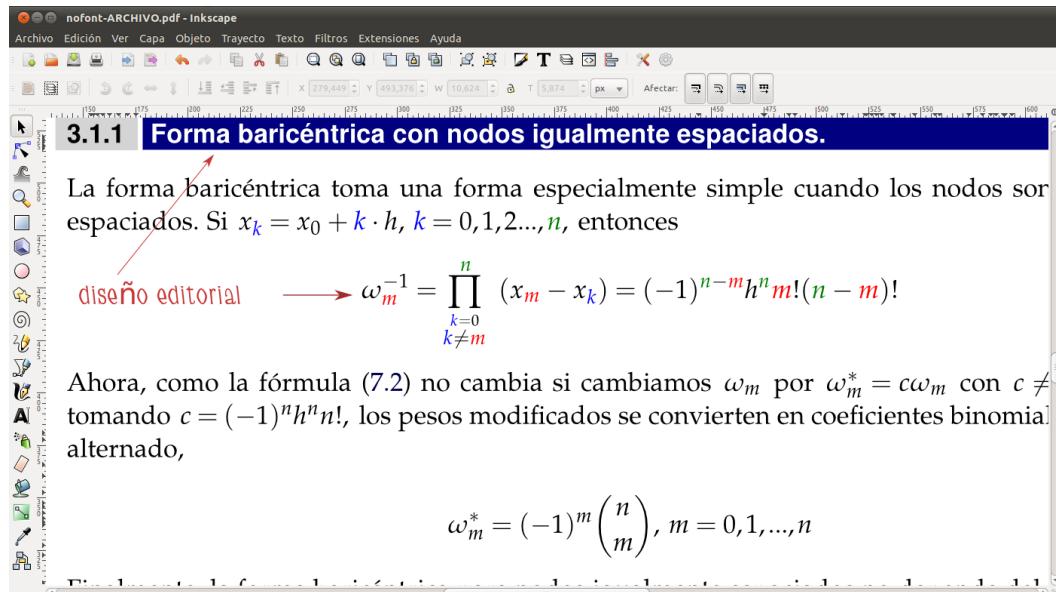


Figura 6.6 PDF con contornos en Inkscape

Para hacer esto, seguimos los siguientes pasos ([usando software libre](#)),

- 1 Si compilamos con [LaTeX](#) se genera un archivo .dvi. Este se convierte a .ps usando DVI->PS (presente en el menú del editor). Luego generamos el PDF con PS->PDF. Ahora aplicamos una instrucción [PostScript](#) que convierte todo el texto en contornos. Suponiendo que tenemos los dos archivos ARCHIVO.ps y ARCHIVO.pdf; desde una *terminal* nos vamos a la carpeta que contiene los archivos y se ejecuta el código:

```
gs -sDEVICE=pswrite -dNOCACHE -sOutputFile=nofont-ARCHIVO.ps -q -dbatch -dNOPAUSE
ARCHIVO.pdf -c quit
```

Esto genera un nuevo archivo [nofont-ARCHIVO.ps](#). Si lo queremos pasar a PDF, en la misma terminal ejecutamos,

```
ps2pdf nofont-ARCHIVO.ps nofont-ARCHIVO.pdf
```

Ahora el PDF se divide en páginas individuales (con [PDFsam](#) por ejemplo) y se edita cada página por separado.

- 2 Si compilamos con [PdfLaTeX](#) se genera un archivo .pdf. Para convertirlo a un archivo PDF, abrimos una terminal y usamos el programa pdftops. Suponiendo que el archivo se llama ARCHIVO.pdf, nos vamos a la carpeta que contiene el archivo y se ejecuta el código:

```
pdftops ARCHIVO.pdf
```

Esto genera un nuevo archivo ARCHIVO.ps. Ahora aplicamos el código [PostScript](#) que convierte todo el texto en contornos. Desde una *terminal* nos vamos a la carpeta que contiene los archivos y se ejecuta el código:

```
gs -sDEVICE=pswrite -dNOCACHE -sOutputFile=nofont-ARCHIVO.ps -q -dbatch -dNOPAUSE
ARCHIVO.pdf -c quit
```

Esto genera un nuevo archivo [nofont-ARCHIVO.ps](#). Este archivo ya está listo para editar en [Inkscape](#).

**Nota.** Esto funciona muy bien en [Ubuntu](#) sin hacer algo adicional. En [Windows](#) es más complicado. Es probable que haya que agregar la ubicación de gswin32c.exe en el PATH: Ir a Panel de Control-Sistema-Propiedades del sistema-Opciones avanzadas-Variables de Entorno. Una vez aquí, presiona el botón Añadir; la variable es GS\_PROG y la dirección es usualmente C:/Archivosdeprograma/gs/gs.../bin/gswin32c.exe.

# 7

# PERSONALIZAR EL DOCUMENTO

## 7.1 Numeración automática de definiciones, teoremas y ejemplos.

Para que L<sup>A</sup>T<sub>E</sub>X numere automáticamente definiciones, teoremas, axiomas, ejemplos, etc.; se pone en el preámbulo (por comodidad) el comando

```
\newtheorem{abreviación}{ambiente}
```

Por ejemplo, para numerar ejemplos y definiciones de tal manera que la numeración incluya el número de capítulo, se podría poner en el *preámbulo*:

```
\newtheorem{ejemplo}{\it Ejemplo}[chapter]
\newtheorem{defi}{\it Definición}[chapter]
```

### Ejemplo 7.1

El Texto:

```
\begin{defi}
Si $y=f(x)$ es derivable, $dx \wedge, $ es cualquier
número real no nulo, mientras que
$dy = f' \prime (x) \wedge, dx \\$\\
\end{defi}
```

produce:

**Definición 7.1** Si  $y = f(x)$  es derivable,  $dx$  es cualquier número real no nulo, mientras que

$$dy = f'(x) dx$$

- El texto de la definición está, por default, en modo “enfático”. Podemos Cambiar el modo enfático a modo normal poniendo el cuerpo en el entorno `\rm ...`

```
\begin{defi}
{\rm ...text...}
\end{defi}
```

## 7.2 El paquete `ntheorem`

---

Este paquete es una extensión del enumerado automático de teoremas que ofrece L<sup>A</sup>T<sub>E</sub>X con `\newtheorem`, el paquete permite definir aspectos globales de estos ambientes.

Para poder utilizar este paquete se tiene que cargar en el preámbulo con la instrucción.

```
\usepackage{ntheorem}
```

En el preámbulo se pueden escribir los comandos

```
\setlength{\theorempreskipamount}{xmm}
\setlength{\theorempostskipamount}{xmm}
```

el primer comando define un espacio de  $x$  milímetros entre el texto anterior al ambiente y el título del mismo. El segundo define el espacio entre el final del ambiente y el texto que le sigue.

El comando

```
\theoremstyle{estilo}
```

define el estilo que van a tener los teoremas, entre los estilos posibles están:

Estilo	
<code>plain</code> :	Este es idéntico al estilo por defecto de L <sup>A</sup> T <sub>E</sub> X.
<code>break</code>	El título es seguido por un paso de línea.
<code>margin</code>	El número del teorema se coloca antes del título.
<code>marginbreak</code>	Igual a 'margin' pero el título va seguido por un paso de línea.

Para cambiar el tipo de letra para el texto del ambiente se utiliza el comando

```
\theorembodyfont{fuente}
```

Y para cambiar el tipo de letra del título del ambiente se utiliza

```
\theoremheaderfont{fuente}
```

Entre las fuentes se puede utilizar `\rmfamily`, `\sffamily`, `\ttfamily`, `\mdfamily`, `\bfseries`, `\upshape`, `\itshape`, `\slshape`, `\scshape`, `\normalfont` y se pueden combinar con los tamaños de letra.

En la primera versión de este libro se utilizó

---

```
\setlength{\theorempreskipamount}{7mm}
\setlength{\theorempostskipamount}{7mm}
\theoremstyle{break}
\theorembodyfont{\normalfont}
\theoremheaderfont{\scshape\large}
\newtheorem{ejemplo}{Ejemplo}
\newtheorem{defi}{Definición}
```

---

En la sección que sigue se indica cómo cambiar el color.

### Ejemplo 7.2

El texto:

---

```
\begin{defi}[Polinomio característico]
Si  $A$  es una matriz cuadrada de orden  $n$ , al polinomio
 $p$  definido por  $p(x) = |A - \lambda \cdot I_n|$ , se le
llama polinomio característico de  $A$  y la ecuación
 $p(x) = 0$  se llama ecuación característica de  $A$ .
\end{defi}
```

---

produce:

**Definición 7.2 (Polinomio característico)** Si  $A$  es una matriz cuadrada de orden  $n$ , al polinomio  $p$  definido por  $p(x) = |A - \lambda \cdot I_n|$ , se le llama **polinomio característico** de  $A$  y la ecuación  $p(x) = 0$  se llama **ecuación característica** de  $A$ .

## 7.3 Personalización de teoremas, definiciones, etc.

Si usamos el estilo plain (este es el default) del paquete `ntheorem`, podemos cambiar el color de ‘Teorema’, ‘Definición’, etc. y la numeración. Por ejemplo si queremos cambiar el nombre a azul y la numeración a rojo (tal y como se usa en este libro), agregamos en el preámbulo

---

```
%\usepackage{ntheorem}
\makeatletter
\renewtheoremstyle{plain}%
{\item[\hspace{\labelsep}\textcolor{red}{\textbf{theorema}}\#1\textcolor{blue}{\textbf{theorema}}\#2\textcolor{red}{\textbf{theorema}}\#3]}%
{\item[\hspace{\labelsep}\textcolor{red}{\textbf{theorema}}\#1\textcolor{blue}{\textbf{theorema}}\#2\textcolor{blue}{\textbf{theorema}}\#3]}%
{\textcolor{blue}{\textbf{theorema}}\#3}
\makeatother
```

---

## 7.4 Personalización de secciones, subsecciones, etc.

En el sitio de [Vincent Zoonekynd](#), en la sección “Exemples”, se pueden encontrar ejemplos de cómo se puede rediseñar los títulos, la página para cada capítulo y las secciones. Como lo advierte el autor, *es posible que se deban hacer algunos ajustes*.

Por ejemplo, para el diseño de los capítulos de este libro se usa una figura `ImagenCapitulo.pdf` que aparece a la izquierda del nombre del capítulo (recuerde que este documento se compiló con **PDFLaTeX**). Esta imagen se supone que está en la subcarpeta `images`. Para obtener este efecto se agrega en el preámbulo,

---

```
\usepackage{pstricks}
\makeatletter
\def\@makechapterhead#1{%
\parindent \z@ \raggedright \reset@font
\hbox to \hsize{%
```

```
\rlap{\raisebox{-2.5em}{\raisebox{\depth}{% Imagen a la izquierda del título
  \IfFileExists{imagencapítulo.pdf}{\includegraphics{imagencapítulo.pdf}}{}%
  \IfFileExists{images/imagencapítulo.pdf}{\includegraphics{images/imagencapítulo.pdf}}{}%
}}}
\rlap{\hbox to 6em{\hss
  \reset@font\fontsize{12em}{12em}\selectfont\black %Color
  \thechapter\hss} \%}
\hspace{10em} \% 
\vbox{%
  \advance\hsize by -10em
  \reset@font\bfseries\Huge\selectfont#1    %Texto del título
  \par
} \% 
} } \% 
\vskip 5pt
\hrulefill
\vskip 50pt
}
\makeatother
```

---

## 7.5 El paquete `todonotes`

Este paquete viene incluido en las distribuciones y se usa para agregar notas en el margen. Hay varias opciones, por ejemplo podríamos agregar en el *preámbulo*,

```
\usepackage[colorinlistoftodos, textwidth=2cm, shadow]{todonotes}
```

Con esto obtendríamos texto en cajas sombreadas de 2cm. Por ejemplo,

“... la cónica  $3x^2 - \sqrt{36}xy + 3y^2 - 10x - 10y - 4 = 0$  corresponde a una parábola pues, usando invariantes,  $B^2 - 4AC = 0$  y  $4ACF + BDE - AE^2 - CD^2 - FB^2 = -1200 \neq 0$ . Como  $A = C$  entonces  $\theta = \pi/4$ . Al aplicar la rotación nos queda la ecuación  $6y'^2 - 10\sqrt{2}x' - 4 = 0$ .”

En general, no basta con calcular  $\Delta$

se produce con el código,

---

```
... la cónica $3 x^2 - \sqrt{36} xy + 3y^2 - 10 x - 10 y - 4=0$ corresponde a una
parábola pues,
usando invariantes, \todo[color={red!100!green!33},size=\small]{En general, no
  basta con calcular $\Delta$} $B^2-4AC=0$ y $4ACF+BDE-AE^2-CD^2-FB^2=-1200\negq 0.$
Como $A=C$ entonces $\theta=\pi/4$. Al aplicar la rotación nos queda la ecuación
$6 y'^2- 10 \sqrt{2} x' - 4=0.$
```

---

El manual de este paquete se puede obtener en ‘Manual `todonotes`’.

## 7.6 El paquete `boibootes`

Alexis Flesch ha creado un bonito paquete, basado en `TikZ`, que permite personalizar los entornos de definiciones, teoremas, ejemplos, etc. Los ejemplos de este libro usan este paquete. El ejemplo que aparece en la página de Alexis es

**Theorem 1** (Law of Large Numbers).

Let  $(X_n)_{n \in \mathbb{N}}$  be an infinite sequence of i.i.d. variables with finite expected value. Then:

$$\frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{\text{a.s.}} \mathbb{E}(X_1).$$

**Instalación.** Para usar este paquete, debemos primero instalarlo pues, por ahora, no viene en la distribución de MikTeX ni de TeXLive. Primero descargamos el archivo `boiboites.sty` en

<http://snouffy.free.fr/blog-en/index.php/post/2010/01/30/Nice-boxes-for-your-theorems-with-tikz>

**TeXLive - Ubuntu.** El archivo se pega en la subcarpeta `base`: Abrimos una terminal y nos vamos a la carpeta donde se descarga el archivo y lo pegamos en la subcarpeta `base`,

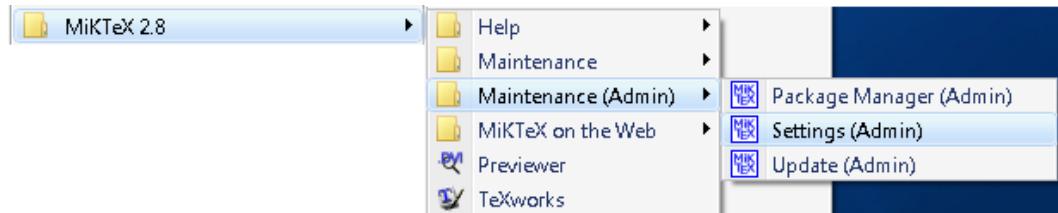
```
sudo cp -a boiboites1.sty /usr/share/texmf-texlive/tex/latex/base
```

luego actualizamos el `TeXLive`,

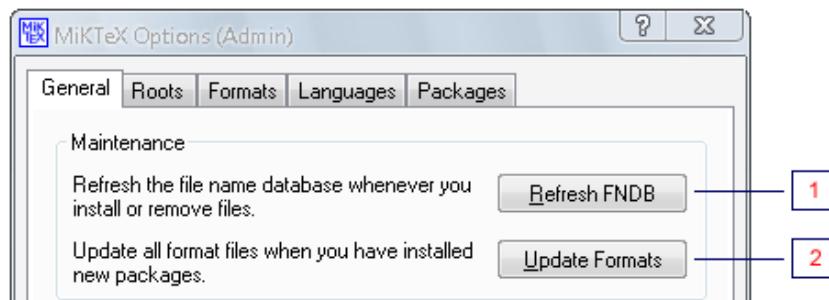
```
sudo texhash
```

**MiKTeX - Windows.** Hay que pegar el archivo en `C:/Archivosdeprograma/MiKTeX2.x/tex/latex/base` y luego debemos actualizar y refrescar la base de datos para que los cambios tengan efecto,

- Levantar las opciones de MiKTeX: Inicio-MiKTeX2.x-Maintenance (Admin)-Settings



- Hacer clic primero sobre el botón Refresh FNBD y luego sobre el botón Update Formats



Ahora agregamos en el preámbulo `\usepackage{boiboites}`. Luego hay que personalizar los entornos. Por ejemplo, para generar el ejemplo de Alexis usamos el código,

---

```
\documentclass{article}
\usepackage{amsmath}
\usepackage{boiboites}
%Definiendo colores para los entornos 'bteo' y 'bdefi'
\newboxedtheorem[boxcolor=orange, background=blue!5, titlebackground=blue!20,
```

```

titleboxcolor = black}{ bteo}{Theorem}{thCounter}
\newboxedtheorem[boxcolor=orange, background=blue!5, titlebackground=blue!20,
titleboxcolor = black]{ bdefi}{Definición}{thCounter}

\begin{document}
\begin{bteo}[Law of Large Numbers]
Sea $(X_n)_{n \in \mathbb{N}}$ una sucesión infinita de variables i.i.d. con valor esperado finito. Entonces:

$$\frac{1}{n} \sum_{i=1}^n X_i \xrightarrow{\text{a.s.}} \mathbb{E}(X_1).$$

\end{bteo}
\end{document}

```

---

## 7.7 Paquete `algorithm2e`

---

Este es un paquete adecuado para describir claramente algoritmos de programación. Debemos poner en el *preámbulo*

```
\usepackage[ruled,vlined,lined,linesnumbered,algochapter,portugues]{algorithm2e}
```

Los comandos más frecuentes son

---

```

\If{ condición }{ código }
\Else{ condición }{ código }{ else ... código }
\For{ condición }{ código }
\While{ condición }{ código }
\Repeat{ condición ("Until") }{ código }

```

---

\; se usa para el cambio de línea.

En los siguientes ejemplos se muestra cómo usar If, For, While, etc.

El código:

---

```

\begin{algorithm}[h]
\caption{Máximo común divisor}\label{ML:Algorithm1}
\SetLine
\KwData{$a, b \in \mathbb{N}$}
\KwResult{$\text{MCD}(a, b)$}
\linesnumbered
\SetVline
$c = |a|; d = |b|;$;
\While{$d \neq 0$}{
    $r = \text{rem}(c, d)$;;
    $c = d$;;
    $d = r$;;
}
\Return{$\text{MCD}(a, b) = |c|$};
\end{algorithm}

```

---

produce:

---

**Algoritmo 7.1:** M\'aximo com\'un divisor

---

**Datos:**  $a, b \in \mathbb{N}$ .  
**Salida:**  $\text{MCD}(a, b)$

```

1  $c = |a|, d = |b|;$ 
2 while  $d \neq 0$  do
3    $r = \text{rem}(c, d);$ 
4    $c = d;$ 
5    $d = r;$ 
6 return  $\text{MCD}(a, b) = |c|;$ 
```

---

El c\'odigo:

---

```

\begin{algorithm}[h]
\caption{Inverso Multiplicativo mod $m$.}
\SetLine
\KwData{$a \in \mathbb{Z}_m$}
\KwResult{$a^{-1} \bmod m$, si existe.}
\linesnumbered
\SetVline
Calcular $x, t$ tal que $xa + tm = \text{MCD}(a, m)$;
\If{$\text{MCD}(a, m) > 1$}{$a^{-1} \bmod m$ no existe}
\Return {$\text{rem}(x, m)$}
\end{algorithm}

```

---

produce:

---

**Algoritmo 7.2:** Inverso Multiplicativo mod  $m$ .

---

**Datos:**  $a \in \mathbb{Z}_m$   
**Salida:**  $a^{-1} \bmod m$ , si existe.

```

1 Calcular $x, t$ tal que $xa + tm = \text{MCD}(a, m)$;
2 if $\text{MCD}(a, m) > 1$ then
3   $a^{-1} \bmod m$ no existe
4 else
5   return $\text{rem}(x, m)$.
```

---

El c\'odigo:

---

```

\begin{algorithm}[h]
\caption{Criba de Erat\'ostenes} \label{CribaEraostenes}
\SetLine
\KwData{$n \in \mathbb{N}$}
\KwResult{Primos entre $2$ y $n$}
\linesnumbered
max$ = \left\lfloor \frac{(n-3)}{2} \right\rfloor$;
boolean esPrimo$[i]$, $i=1, 2, \dots, max$;

\SetVline
\For{$j=1, 2, \dots, max$}{esPrimo$[j]=True$}
\$i=0$;
\While{$(2i+3) \leq n$}{
```

---

```

\$k=i+1\$;
\If{\rm esPrimo[i]}{
\While{((2k+1)(2i+3) \leq n} {
{
    esPrimo[((2k+1)(2i+3)-3)/2]=False;
    \$k=k+1\$;
}
}
\$i=i+1\$;
}
Imprimir;
\For{j=1,2,...,max} {
\If{\rm esPrimo[j]=True}{Imprima j}
}
\end{algorithm}

```

---

produce:

---

### Algoritmo 7.3: Criba de Eratóstenes

---

**Datos:**  $n \in \mathbb{N}$

**Salida:** Primos entre 2 y  $n$

```

1 max=  $\left\lceil \frac{(n-3)}{2} \right\rceil$ ;
2 boolean esPrimo[i], i = 1,2,...,max;
3 for j = 1,2,...,max do
4   esPrimo[j] = True;
5 i = 0;
6 while (2i + 3)(2i + 3) ≤ n do
7   k = i + 1;
8   if esPrimo[i] then
9     while (2k + 1)(2i + 3) ≤ n do
10      esPrimo[((2k + 1)(2i + 3) - 3)/2] = False;
11      k = k + 1;
12   i = i + 1;
13 Imprimir;
14 for j = 1,2,...,max do
15   if esPrimo[j] = True then
16     Imprima j

```

---

**Nota:** Se puede cambiar el lenguaje modificando el archivo `algorithm2e.sty` localizado en `/usr/share/texmf-texlive/tex/latex/algorithm2e` en `Ubuntu` o en `C:/Archivosdeprograma/MiKTeX2.x/tex/latex/algorithm2e` en `Windows`.

## 7.8 Código de lenguajes de programación en color

---

Una aplicación obvia del entorno `verbatim` es editar código de programación. Para agregar jerarquía al código se usa identación y color. Hay varios paquetes para agregar color al código: Por ejemplo los paquetes `minted`<sup>4</sup>, `verbments` y `listings`.

<sup>4</sup>Windows 7 requiere una instalación asistida, ver 'Minted en Windows 7'

Con estos paquetes se elige el lenguaje (C++, java, latex, etc.) y permiten también, entre otras opciones, usar código L<sup>A</sup>T<sub>E</sub>X para generar texto matemático para documentar el código.

### 7.8.1 Paquete `minted`

Para usar este paquete debemos poner en el preámbulo `\usepackage{minted}`. Adicionalmente se puede cambiar el *tema* (que es el que define el esquema de color). Por ejemplo, en este libro usamos en el tema vs. Para cambiar el tema default y usar este tema, se pone en el preámbulo `\usemintedstyle{vs}`. En general tendríamos algo como,

---

```
\documentclass{article}
  \usepackage{minted}
  \usemintedstyle{vs}
\begin{document}

\begin{minted}[opiones]{lenguaje}
código
\end{minted}
...

```

---

#### Ejemplo 7.3 (Usando `minted`).

El texto:

---

```
\begin{minted}[mathescape, frame=lines,
    rulecolor=\color{gray}]{python}
# Returns $\displaystyle\sum_{i=1}^n i$
def sum_from_one_to(n):
    r = range(1, n + 1)
    return sum(r)
\end{minted}

```

---

produce:

---

```
# Returns  $\sum_{i=1}^n i$ 
def sum_from_one_to(n):
    r = range(1, n + 1)
    return sum(r)
```

---

En el ejemplo anterior el lenguaje que se declara es `python`. La opción `mathescape` permite aplicar texto en modo matemático en el entorno (debe ir precedido por el símbolo de comentario del lenguaje). El paquete `minted` usa la librería Pygments así que si quiere saber más de `minted` hay que ver la documentación del paquete `minted` y la documentación de Pygments.

### 7.8.2 Paquete `verbments` para TeXLive

Este paquete es parecido a `minted` pues está basado también en Pygments, así que las opciones serán familiares. Tiene la ventaja de que el código se divide si queda entre dos páginas y se puede poner el tema directamente en las opciones. También las opciones se pueden declarar al inicio del documento. Para usar el paquete debemos poner en el preámbulo `\usepackage{verbments}`. El entorno sería algo como,

---

```
\begin{pyglist}[language=java, caption={Nombre del programa}, otras opciones]
...
código
\end{pyglist}
```

---

Cuando se usan muchas opciones, como en el ejemplo que sigue, es mejor declararlas antes con los comandos `fvset` y `plset`.

#### Ejemplo 7.4 (Usando `verbments`).

El texto:

```
\documentclass{article}
\usepackage{verbments}
\definecolor{verbmentsbgcolor}{rgb}{0.9764, 0.9764, 0.9762}
\begin{document}
%opciones verbments
\fvset{frame=bottomline,framerule=4pt,rulecolor=\color{verbmentscaptionbgcolor}}
\plset{language=java,texcl=true,style=vs,%
       listingnamefont=\sffamily\bfseries\color{white},%
       bgcolor=verbmentsbgcolor,captionfont=\sffamily\color{white},%
       captionbgcolor=verbmentscaptionbgcolor, listingname=\textbf{Programa}}
%Entorno
\begin{pyglist}[caption={Imprimir.java}]
import javax.swing.*;
import java.awt.*;
public class app_prg1 extends JApplet
{ public void init(){}
  public void paint ( Graphics g ){
    g.drawString(" 3 +46 = "+(3+46),30, 30 );
  }
\end{pyglist}
\end{document}
```

produce:

**Programa 1: Imprimir.java**

```
import javax.swing.*;
import java.awt.*;
public class app_prg1 extends JApplet
{ public void init(){}
  public void paint ( Graphics g ){
    g.drawString(" 3 +46 = "+(3+46),30, 30 );
  }
```

**Figura 7.1** Código con `verbments` de TeXLive

**Nota:** Igual que `minted`, puede agregar texto en modo matemático. El texto matemático debe de ir precedido por el símbolo usado para los comentarios, en el caso de Java sería `// $ ....$`. Debe agregar la opción `mathescape=true`.

**Nota:** Este paquete viene con la versión `TeXLive` más reciente. Si tiene una versión anterior, debe obtener la versión del archivo `verbments.sty` del 2011 o la más reciente. El archivo `verbments.sty` se puede obtener en <http://texcatalogue.sarovar.org/entries/verbments.html>

## 7.9 Cómo hacer listas de ejercicios con solución

Para hacer listas de ejercicios con solución se puede usar el paquete `answers` (hay un paquete para exámenes en CTAN-exam).

En el código que sigue se muestra un ejemplo de cómo se podría usar y la salida.

### Ejemplo 7.5 (Paquete `answers`)

```
\documentclass[12pt,a4paper]{book}
\usepackage{amsmath,amssymb,amsfonts,latexsym}
\usepackage[spanish]{babel}
\usepackage[latin1]{inputenc} %acentos desde el teclado
\usepackage[T1]{fontenc}
\setcounter{chapter}{1}
%paquete
\usepackage{answers}
\Newassociation{sol}{Solution}{ans}
\newtheorem{ex}{}{}

\begin{document}

\Opensolutionfile{ans}[ans1]
\section{Ejercicios}

\begin{ex} \rm Primer ejercicio.
\begin{sol}
Solución del ejercicio 1
\end{sol}
\end{ex}

\begin{ex} \rm Segundo ejercicio.
\begin{sol}
Solución del ejercicio 2
\end{sol}
\end{ex}

\Closesolutionfile{ans}

% Imprimir las soluciones al final del capítulo, por ejemplo
\appendix
\section*{Solución de los ejercicios del capítulo 1}
\input{ans1} % soluciones de la primera lista de ejercicios
\end{document}
%continúa...
```

**Ejemplo 7.5 (continuación).**

La salida sería:

### 1.1. Ejercicios

- 1** Primer ejercicio.
- 2** Segundo ejercicio.

### Solución de los ejercicios del capítulo 1

- 1** Solución del ejercicio 1
- 2** Solución del ejercicio 2

**Figura 7.2** Paquete answers

## 7.10 Cabeceras

Se puede usar el paquete `fancyhdr` para personalizar las cabeceras y los pie de página. Por ejemplo, podemos poner cabeceras en minúscula y helvética (como en la figura que sigue),

<u>1.2 Dejar la línea con grosor 0.5pt</u>	3
<b>1.2. Dejar la línea con grosor 0.5pt</b> <pre>\renewcommand{\headrulewidth}{0.5pt} \addtolength{\headheight}{0.5pt} % espacio para la raya</pre>	

**Figura 7.3** Cabecera en helvética y con línea

2	Cabeceras con el paquete <code>fancyhdr</code>
<b>1.1. Eliminar la línea</b> <pre>\renewcommand{\headrulewidth}{0pt}</pre>	

**Figura 7.4** Cabecera en helvética y sin línea

El código que se usó para generar estas cabeceras es el siguiente,

---

```
\documentclass{book}
\usepackage[spanish]{babel}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
```

```

\usepackage{helvet}
    %Helvética
\newcommand{\helv}{\fontfamily{phv}\fontsize{9}{11}\selectfont}
\usepackage{fancyhdr}
\pagestyle{fancy}

\renewcommand{\chaptermark}[1]{\markboth{\#1}{}}
\renewcommand{\sectionmark}[1]{\markright{\thesection \ #1}}
\fancyhf{} % borra cabecera y pie actuales
\fancyhead[LE,RO]{\helv\thepage} %Left Even page - Right Odd page
\fancyhead[LO]{\helv\rightmark}
\fancyhead[RE]{\helv\leftmark}
\renewcommand{\headrulewidth}{0pt} % Sin raya. Con raya?: cambiar {0} por {0.5pt}
\renewcommand{\footrulewidth}{0pt}
\addtolength{\headheight}{0.5pt} % espacio para la raya
\fancypagestyle{plain}{%
\fancyhead{} % elimina cabeceras y raya en páginas "plain"
\renewcommand{\headrulewidth}{0pt} }

\begin{document}
\chapter{Cabeceras con el paquete {\tt fancyhdr}}
\newpage
\section{Eliminar la línea}
\verb+\renewcommand{\headrulewidth}{0pt}+
\newpage
\section{Dejar la línea con grosor {\tt 0.5pt}}
\verb+\renewcommand{\headrulewidth}{0.5pt} % grosor 0.5pt+\
\verb+\addtolength{\headheight}{0.5pt} % espacio para la raya+
\end{document}

```

---

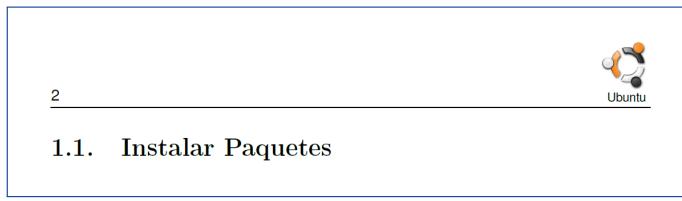
Usando este mismo paquete también se podría incluir una figura en todas las cabeceras, digamos a la derecha, con el código,

```

\rhead{\setlength{\unitlength}{1mm}}
\begin{picture}(0,0)
    % Figura
    \put(-10,2){\includegraphics[width=10mm]{images/ubuntu.png}}
    % Texto
    \put(-92,5){
        \parbox[t]{90mm}{
            \begin{flushright}
            \begin{scriptsize}
                \textsf{Ubuntu}
            \end{scriptsize}
            \end{flushright}
        }
    }
\end{picture}

```

---





# CITAS BIBLIOGRÁFICAS CONSISTENTES CON BibTeX

La bibliografía es una de las partes más importantes de un documento, esta permite hacer referencia a trabajos realizados anteriormente por otros autores. L<sup>A</sup>T<sub>E</sub>X ofrece dos formas de realizar bibliografías en un trabajo: El entorno `thebibliography` y el uso de BibT<sub>E</sub>X.

## 8.1 Entorno `thebibliography`

Para utilizar el entorno `thebibliography` se deben poner las referencias entre los comandos `\begin{thebibliography}{99} . . . \end{thebibliography}`. Cada una de las entradas de la bibliografía se pone con un comando `\bibitem{llave}`, la llave se utiliza para hacer la referencia dentro del texto.

### Ejemplo 8.1

El texto:

```
\begin{thebibliography}{99}
\bibitem{Goossens} M. Goossens; F, Mittelbach; A. Samarin.  

    {\it The \LaTeX Companion}. Addison-Wesley. 1993.
\bibitem{Lamport} L. Lamport. {\it \LaTeX}. Addison-Wesley. 1996.
\end{thebibliography}
```

produce la bibliografía

### Bibliografía

- [1] M. Goossens; F, Mittelbach; A. Samarin. *The \LaTeX Companion*. Addison-Wesley. 1993.
- [2] L. Lamport. *\LaTeX*. Addison-Wesley. 1996.

### Ejemplo 8.2

El texto:

---

En `\cite{Goossens}` y `\cite{Lamport}` se muestra como...

---

produce las referencias

En [1] y [2] se muestra como... 

Este entorno tiene la ventaja que las referencias se escriben directamente en el documento, la desventaja es que si se tiene otro documento que hace la misma referencia entonces hay que escribir la entrada en ambos documentos.

## 8.2 Bib<sub>TEX</sub>

---

La segunda opción para realizar bibliografías es utilizar Bib<sub>TEX</sub>, para este caso lo que se realiza es una “base de datos” de los libros en un archivo de texto aparte, este archivo se debe guardar en la misma carpeta del documento con extensión .bib. Este archivo se puede realizar con el Bloc de Notas en Windows o el Editor de Textos en Linux, en general funciona cualquier editor de texto plano. En el documento, donde se quiere que aparezca la bibliografía, se deben poner las instrucciones:

---

```
\bibliographystyle{ESTILO}
\bibliography{basededatos1[,basededatos2,...]}
```

---

El estilo define cómo se presentará la bibliografía, entre los estilos más populares están: plain, apalike, alpha, abrv, unsrt. Sin embargo, hay revistas o instituciones que manejan su propio estilo, en estos casos le brindan al usuario un archivo de estilo que se copia en la carpeta del documento.

Se pueden tener varias bases de datos de bibliografía separadas, por ejemplo, se puede tener una para los libros de computación y otra para los libros de álgebra; si en algún momento se está escribiendo un artículo sobre álgebra computacional, es probable que se quiera hacer referencia a libros de ambas bases de datos, entonces en el comando `\bibliography` se ponen ambas bases.

Una de las ventajas que tiene Bib<sub>TEX</sub> es que, aunque la base de datos tenga muchas referencias, en la bibliografía del documento sólo aparecen las referencias a las entradas que aparecen citadas en el texto. Si se quiere que aparezca alguna entrada aunque no se cite en el texto se agrega la línea `\nocite{Llave}` y si se quiere que todas las entradas se pongan aunque nunca se citen se debe agregar la línea `\nocite{*}`.

Cada entrada en el archivo .bib debe tener la siguiente estructura:

---

```
@tipo{Llave,
  propiedad1="valor1",
  propiedad2="valor2",
  ...
}
```

---

---

Donde el tipo se refiere al tipo de documento: artículo, libro, conferencia, etc. A continuación se presentan los tipos permitidos:

article	incollection	other
book	inproceedings	phdthesis
booklet	manual	proceedings
conference	mastersthesis	techreport
inbook	misc	unpublished

La llave es la que se utiliza dentro del texto para hacer las citas con la instrucción `\cite{Llave}`. Las propiedades se refieren a los datos que se toman en las referencias: autor, título, editorial, año, etc. Las propiedades permitidas se enuncian a continuación.

address	howpublished	number
abstract	institution	organization
author	ISBN	pages
booktitle	ISSN	publisher
chapter	journal	school
contents	key	series
copyright	keywords	title
crossref	language	url
edition	month	volume
editor	note	year

Las mismas entradas que se hicieron en la sección anterior, en BibTeX se pueden hacer en un archivo LaTeX.bib con los siguientes datos.

---

```
@book{Goossens,
  author="Michel Goossens and Frank Mittelbach and Alexander Samarin",
  title="The \LaTeX Companion",
  editor="Addison-Wesley",
  year="1993"
}

@book{Lamport,
  author="Leslie Lamport",
  title="\LaTeX",
  editor="Addison-Wesley",
  year="1996"
}
```

---

En el texto se citaría igual que en el entorno thebibliography:

---

En `\cite{Goossens}` y `\cite{Lamport}` se muestra como...

---

Y la bibliografía se mostraría con:

---

```
\bibliographystyle{apalike}
\bibliography{LaTeX.bib}
```

---



**Figura 8.1** Bibliografía con el estilo APA



**Figura 8.2** Bibliografía con el estilo Plain



**Figura 8.3** Bibliografía con el estilo Alpha

En este caso se está utilizando el estilo del APA (American Psychological Association) para citar, el resultado se muestra en la figura 8.1.

En las figuras 8.2 y 8.3 se muestran los resultados utilizando como estilo plain y alpha.

Existen muchos otros estilos con los que se puede trabajar, si se está editando un libro esta opción de cambiar de estilo es muy cómoda porque las revistas o las editoriales por lo general tienen su propio estilo y, con sólo cambiar el estilo de la bibliografía, ya el documento se adecúa.

## 8.3 JabRef

Existen algunos programas que facilitan la creación y el manejo de las bases de datos que utiliza BibTeX, uno de estos programas es JabRef, entre sus principales características es que es un programa gratuito y que se puede utilizar tanto en Windows como en Linux.

Al abrir el programa se nos presenta una ventana sin nada más que el menú y las barras de herramientas, al hacer una nueva base de datos se nos abre la base en blanco, tal como se muestra en la figura 8.4.

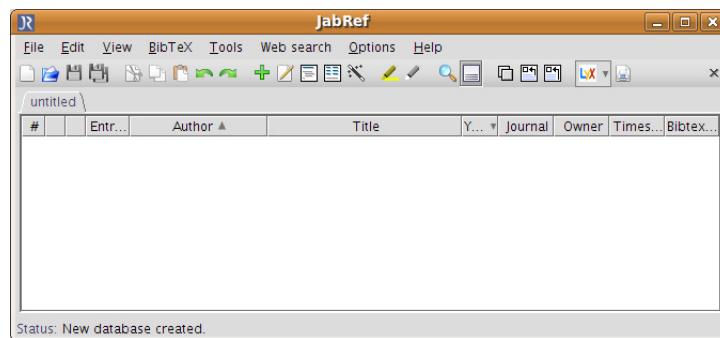


Figura 8.4 Pantalla principal de JabRef.

Para hacer una nueva entrada en la base de datos se utiliza el menú BibTeX->New Entry o el botón de la barra de herramientas, a continuación sale una ventana con las opciones que hay para la referencia, tal como se muestra en la figura 8.6

Ahora aparecerá una ventana donde se pueden ir llenando los campos de la entrada, en las lengüetas de arriba se puede ir pasando entre los campos requeridos, los opcionales, los de datos generales, el abstract (resumen) y el review (revisión), la última es para ver la entrada de BibTeX en modo texto. En general se llenan los campos de los que se disponga información. Esta ventana se puede observar en la figura 8.5.

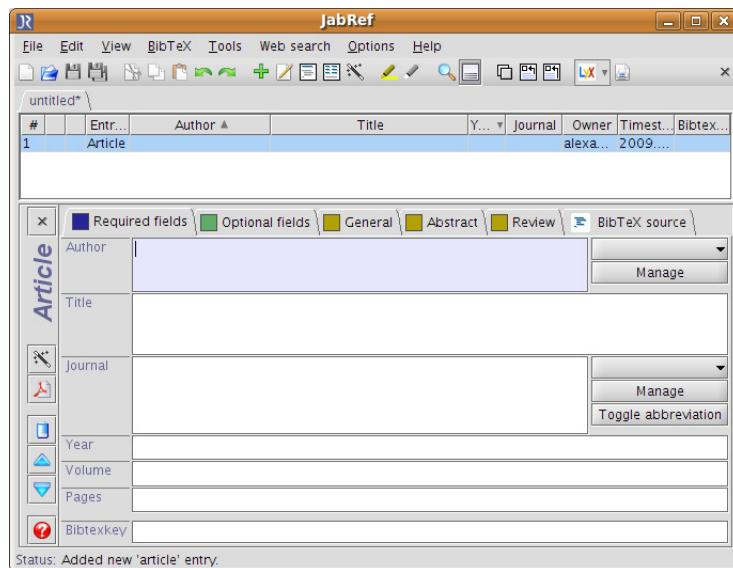


Figura 8.5 Nuevo Artículo.

Si se abre una base de datos ya existente, aparecen las entradas arriba y la ficha bibliográfica al lado abajo de la ventana, si se quiere editar alguna de las entradas existentes se debe utilizar en el menú BibTeX->Edit Entry o presionar el botón  de la barra de herramientas.

Al guardar el archivo lo que hace el programa es guardarlo en modo texto con extensión .bib tal como se vió en la sección 8.2



**Figura 8.6** Opciones para una nueva entrada en BibTeX.



# 9

# CÓMO HACER TRANSPARENCIAS CON LA CLASE BEAMER

## 9.1 Introducción

“Beamer” es una clase [LaTeX](#) (`\documentclass{beamer}`) que se usa para generar transparencias para presentaciones (à la Power Point). Se compila con [LaTeX+dvips](#) o [PDFLaTeX](#) y se usa código [LaTeX](#) estándar.

La versión actual (enero 2012) es Beamer 3.12. Beamer viene incluido en [TeXLive](#) y en [MikTeX](#). Si tiene una instalación completa, puede usar Beamer sin problemas. La documentación la puede encontrar en ‘The Beamer Class’.

Un documento Beamer consiste de una sucesión de marcos (*frames*). En el caso más simple, un marco solo contiene una transparencia. Un ejemplo de transparencia (usando el tema personalizado) se ve en la figura 9.1.

Instituto Tecnológico de Costa Rica  
Método de Newton.

Escuela de Matemática - CRV

CIDSE

① ¿Qué significa “suficientemente cercano” a  $x^*$ ?  
② Se puede aclarar en el sentido del teorema

**Teorema**

Sea  $x^*$  un cero simple de  $f$  y sea  $I_\varepsilon = \{x \in \mathbb{R} : |x - x^*| \leq \varepsilon\}$ . Asumamos que  $f \in C^2(I_\varepsilon)$  y que  $\varepsilon$  es lo suficientemente pequeño de tal manera que en este intervalo  $|f'(x)| \geq m > 0$  y existe  $M > 0$  tal que  $|f''(x)| \leq M$ . Si en el método de Newton, algún  $x_j$  satisface

$$x_j \in I_\varepsilon, \quad y \quad |x_j - x^*| < 2 \frac{m}{M} \quad (1)$$

entonces  $\lim_{n \rightarrow \infty} x_n = x^*$

W. Mora F.    Métodos Híbridos    3 / 13

Figura 9.1 Transparencia Beamer.

## 9.2 Un documento Beamer

---

La estructura general de un posible documento Beamer podría ser,

```
\documentclass{beamer}
\usepackage{amsmath,amssymb,amsfonts, latexsym, stmaryrd}
\usepackage[latin1]{inputenc}
\usepackage[T1]{fontenc}
    % Incluir figuras .pdf, .png, .jpg, .gif, .eps, etc. SIN extensión
%\usepackage{epstopdf}
%\DeclareGraphicsExtensions{.pdf,.png,.jpg,.gif,.eps}
\usefonttheme{professionalfonts} % fuentes de LaTeX
\usetitle{Warsaw} % tema escogido en este ejemplo
\setbeamercolor{background canvas}{white} % Velos

\newtheorem{Teorema}{Teorema}
\newtheorem{Ejemplo}{Ejemplo}
\newtheorem{Definicion}{Definición}
\newtheorem{Corolario}{Corolario}
\newtheorem{Prueba}{Prueba}

\begin{document}

\title{Factorización en  $\mathbb{Z}_p[x]$  y en  $\mathbb{Z}[x]$ .}
\subtitle{Parte I}
\author{\bf Prof. Walter Mora F.\\
        Escuela de Matemática, ITCR\\
        www.cidse.itcr.ac.cr/revistamate/}
\date{Junio 2007}

\frame{\titlepage}

\section{Primera sección}

\begin{frame}{Título de este marco}
    % texto
\end{frame}

\begin{frame}{Título de este marco}
    % texto
\end{frame}

\end{document}
```

---

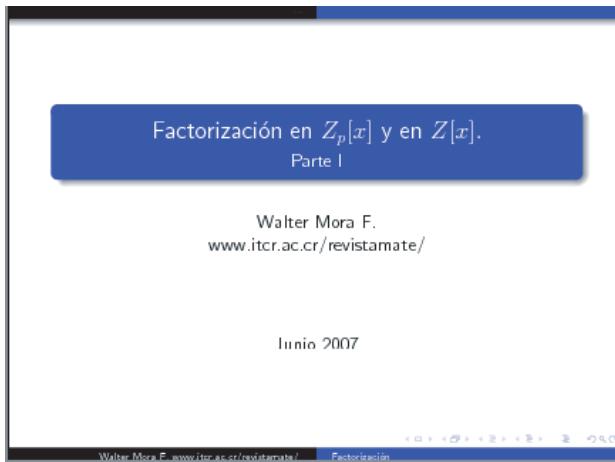
### Compilar

- Un documento Beamer se puede compilar directamente con **PDFLaTeX** o con **LaTeX-dvips**
  - Si hay figuras en distintos formatos, .eps, .pdf, .png, etc., podemos habilitar el paquete **epstopdf** y declarar las extensiones en el preámbulo de tal manera que podamos incluir las figuras *sin extensión* y compilar con **PDFLaTeX**. Habilitamos en el preámbulo,

```
\usepackage{epstopdf}
\DeclareGraphicsExtensions{.pdf,.png,.jpg,.gif, .eps}
```

- Si las figuras son todas .eps, .ps pues se incluyen con esa extensión, y por supuesto se compila con `LaTeX+dvips`.

La primera transparencia del código anterior es



**Figura 9.2** Transparencia Beamer. Tema Warsaw

Notas.

- Beamer carga (por default) ‘graphicx’ para el soporte de gráficos.
- La lista de temas que viene con Beamer es

---

\usetheme{Bergen}	\usetheme{Hannover}	\usetheme{Darmstadt}
\usetheme{Boadilla}	\usetheme{Luebeck}	\usetheme{Frankfurt}
\usetheme{Copenhagen}	\usetheme{AnnArbor}	\usetheme{Ilmenau}
\usetheme{Dresden}	\usetheme{Berkeley}	\usetheme{Madrid}
\usetheme{Warsaw}	\usetheme{Antibes}	\usetheme{Berlin}
\usetheme{CambridgeUS}	\usetheme{Malmoe}	\usetheme{PaloAlto}

---

- Una vista de los temas se puede encontrar en  
[http://www.math.umbc.edu/~rouben/beamer/quickstart-Z-H-30.html#node\\_sec\\_30](http://www.math.umbc.edu/~rouben/beamer/quickstart-Z-H-30.html#node_sec_30)
- En Internet hay otros tantos temas Beamer disponibles, usualmente temas de particulares o instituciones pero de distribución libre. Los temas se pueden personalizar (ver [15]).

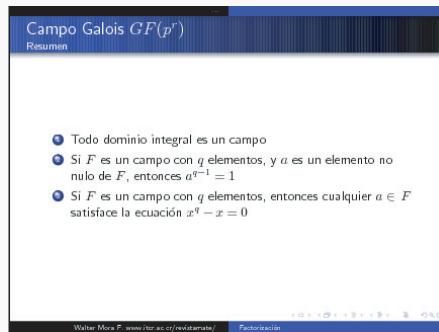
## 9.3 Marcos (frames)

---

El uso más común de un marco (frame) es poner una lista de items

**Ejemplo 9.1**

```
\begin{frame}
    \frametitle{Campo Galois $GF(p^r)$}
    \framesubtitle{Resumen}
    \begin{enumerate}
        \item Todo dominio integral (\emph{en finito}) es un campo\\
        \item Si  $F$  es un campo con  $q$  elementos, y  $a$  es un elemento no nulo de  $F$ , entonces  $a^{q-1}=1$ \\
        \item Si  $F$  es un campo con  $q$  elementos, entonces cualquier  $a \in F$  satisface la ecuación  $x^q-x=0$ 
    \end{enumerate}
\end{frame}
```



**Figura 9.3** Marco Beamer. Tema Warsaw

## 9.4 Velos (overlays)

En una presentación puede ser deseable que los ítems vayan apareciendo uno a al vez, mientras los otros permanecen con un *velo*. Esto se puede lograr agregando la opción [ $<+>$ ] a los entornos `enumerate` o `itemize`. Un ejemplo se muestra en la figura 9.4.

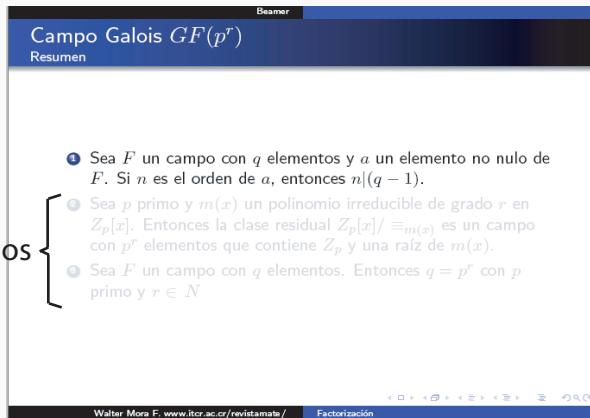
**Ejemplo 9.2**

```
\begin{frame}
    \frametitle{Campo Galois $GF(p^r)$}
    \framesubtitle{Resumen}
    \begin{enumerate}[<+>]
        \item Sea  $F$  un campo con  $q$  elementos y  $a$  un elemento no nulo de  $F$ . Si  $n$  es el orden de  $a$ , entonces  $n | (q-1)$ .
    \end{enumerate}

```

**Ejemplo 9.2 (continuación).**

```
\item Sea $p$ primo y $m(x)$ un polinomio irreducible de grado $r$ en $\mathbb{Z}_p[x]$.
Entonces la clase residual $\mathbb{Z}_p[x]/\langle m(x) \rangle$ es un campo con $p^r$ elementos que contiene $\mathbb{Z}_p$ y una raíz de $m(x)$.
\item Sea $F$ un campo con $q$ elementos.
Entonces $q=p^r$ con $p$ primo y $r \in \mathbb{N}$
\end{enumerate}
\end{frame}
```

**Figura 9.4** Marco Beamer con dos “velos”.**9.4.1 Opciones <i-> y \uncover<i->**

En vez de usar la opción [*<+>*] en el entorno `enumerate` (o `itemize`), se puede agregar un comportamiento un poco más dinámico usando las opciones *<i->* y `\uncover<i->{texto}`.

Con estas opciones podemos controlar la secuencia en la que se presentan distintas líneas en una transparencia.

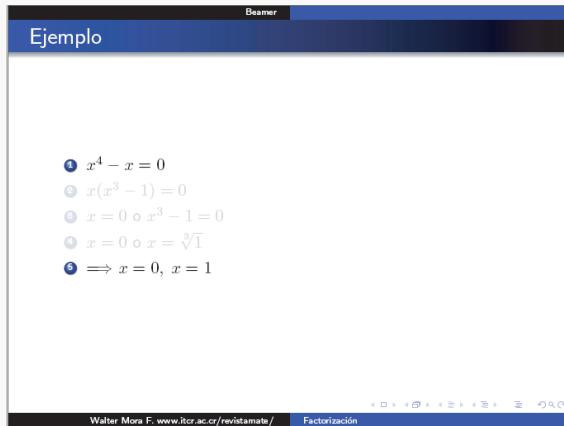
- `\item<i->` indica que este ítem se presenta en la *i*-ésima transparencia. En la práctica se puede ver como la misma transparencia con un velo menos.
- `\uncover<i->{texto}` indica que este `texto` se presenta en la *i*-ésima transparencia

Por ejemplo, en la siguiente transparencia, se quiere mostrar inicialmente una ecuación y la solución, ambos ítems se marcan con `<1->`. Después se muestra el procedimiento, que corresponde a los ítems `<2->, <3->, <4->`

- Un ejemplo del comando `\uncover` se presenta más adelante, en la sección 9.9.

**Ejemplo 9.3**

```
\begin{frame}{Ejemplo}
\begin{enumerate}
\item<1-> $x^4-x=0$ % <1->
\item<2-> $x(x^3-1)=0$ 
\item<3-> $x = 0 \cup x^3-1=0$ 
\item<4-> $x = 0 \cup x=\sqrt[3]{1}$ 
\item<1-> $\Longrightarrow x=0, \cup x=1$ % <1->
\end{enumerate}
\end{frame}
```



**Figura 9.5** Marco Beamer con overlays.

**9.4.2 Opción <i-|alert@ i>**

Esta opción se usa igual que en los casos anteriores. Lo nuevo es que la nueva transparencia descubre en color rojo el nuevo ítem.

**Ejemplo 9.4**

```
\begin{frame}{Ejemplo}
\begin{enumerate}
\item<1-|alert@1> $x^4-x=0$ % <-
\item<2-|alert@2> $x(x^3-1)=0$ 
\item<3-|alert@3> $x = 0 \cup x^3-1=0$ 
\item<4-|alert@4> $x = 0 \cup x=\sqrt[3]{1}$ 
\item<1-|alert@1> $\Longrightarrow x=0, \cup x=1$ % <-
\end{enumerate}
\end{frame}
```

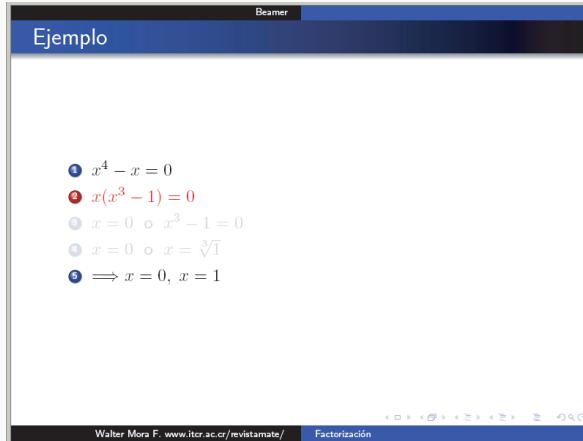
**Ejemplo 9.4 (continuación).**

Nota: Si solo queremos el efecto de ‘alert’ en cada ítem, podemos poner

---

```
\begin{frame}{Ejemplo}
\begin{enumerate}[<+| alert@+>] % <- opción
\item $x^4-x=0$ 
\item $x(x^3-1)=0$ 
\item $x = 0 \text{ o } x^3-1=0$ 
\item $x = 0 \text{ o } x=\sqrt[3]{1}$ 
\item $\Longrightarrow x=0, x=1$ 
\end{enumerate}
\end{frame}
```

---



**Figura 9.6** Marco Beamer con opciones `<i-|alert@ i>`

## 9.5 Comando `pause`.

Para crear un velo, se puede usar `\pause`. Si solo se usa una vez, se cubre la parte del marco que está después de `\pause` y se puede usar varias veces en el marco si queremos fragmentar los velos. Los ejemplos de cómo se podría usar `pause` están en la sección que sigue.

## 9.6 Entornos para teoremas, definición, etc.

Ya habíamos puesto en el preámbulo nuestras definiciones para los entornos de Teorema, Definición, etc. Estos entornos se usan igual que `enumerate`

Nota: Beamer tiene su propio entorno para ejemplos, teoremas, definiciones, etc.

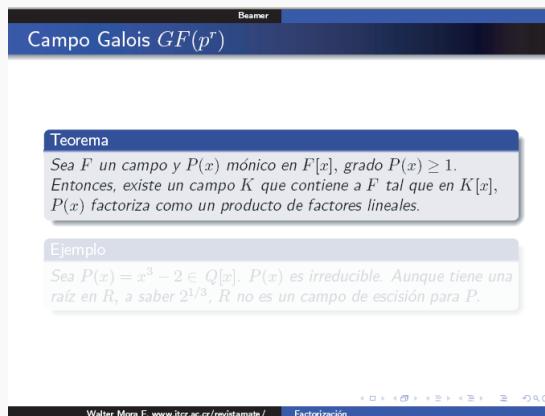
El idioma se puede cambiar en el archivo (el paquete `babel` podría presentar choques con Beamer) ...`\tex\latex\beamer\base\beamertemplatethemes.sty`

**Ejemplo 9.5**

```
\begin{frame}{Campo Galois $GF(p^r)$}
\begin{teorema}
Sea $F$ un campo y $P(x)$ mónico en $F[x]$, grado $P(x) \geq 1$. Entonces, existe un campo $K$ que contiene a $F$ tal que en $K[x]$, $P(x)$ factoriza como un producto de factores lineales.
\end{teorema}

\pause % ----- PAUSA
\begin{ejemplo} % Entorno definido en el preámbulo
Sea $P(x)=x^3-2 \in Q[x]$. $P(x)$ es irreducible. Aunque tiene una raíz en $R$, a saber $2^{1/3}$, $R$ no es un campo de escisión para $P$.
\end{ejemplo}

\end{frame}
```

**Figura 9.7** Entornos Teorema y Ejemplo.**9.7 Blocks.**

La numeración y la etiqueta de los entornos teorema, definición, etc., son adecuados para notas de clase, libros o artículos, es decir, en contextos donde el público puede acceder al material escrito porque la numeración de teoremas es algo complicado de recordar (recuerden el sufrido libro de Jean Dieudonne, “Foundations of Modern Analysis”). En una presentación es mejor indicar las cosas con nombre fáciles de recordar, como “Teorema de Rolle”, etc. El entorno Block es solo un caja sin etiqueta de entorno y sin numeración; la ventaja es que nos permite poner nombres arbitrarios a las cajas. Veamos un ejemplo,

**Ejemplo 9.6**

```
\begin{frame}{Nodos igualmente espaciados}
\begin{block}{Diferencias hacia adelante}
\begin{eqnarray*}
\Delta^0 y_k &:=& y_k, \\
\Delta^1 y_k &=& y_{k+1} - y_k, \\
\Delta^2 y_k &=& \Delta(y_{k+1} - y_k) = y_{k+2} - y_{k+1} - y_{k+1} + y_k \\
&=& y_{k+2} - 2y_{k+1} + y_k, \\
&\dots& \\
\Delta^n y_k &=& \sum_{j=0}^n (-1)^j \binom{n}{j} y_{k+n-j}
\end{eqnarray*}
\end{block}
\end{frame}
```

**Nodos igualmente espaciados****Diferencias hacia adelante**

$$\begin{aligned}
\Delta^0 y_k &:= y_k, \\
\Delta^1 y_k &= y_{k+1} - y_k, \\
\Delta^2 y_k &= \Delta(y_{k+1} - y_k) = y_{k+2} - y_{k+1} - y_{k+1} + y_k = y_{k+2} - 2y_{k+1} + y_k, \\
&\dots \\
\Delta^n y_k &= \sum_{j=0}^n (-1)^j \binom{n}{j} y_{k+n-j}
\end{aligned}$$

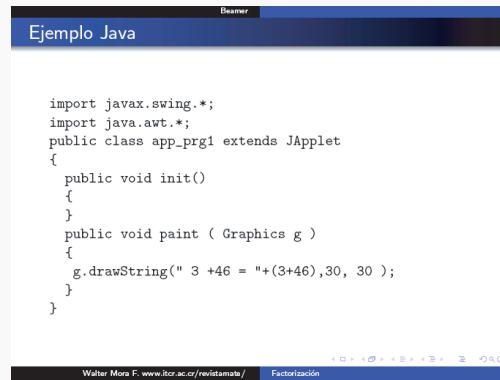
**Figura 9.8** Block.

## 9.8 Opción `fragile`

Se debe usar la opción `fragile` en un marco que contiene `verbatim`, `algorithm2e`, etc. Un ejemplo típico, es presentar el código de un programa (ver figura 9.9),

### Ejemplo 9.7

```
\begin{frame}[fragile]
\frametitle{Ejemplo Java}
\begin{verbatim}
import javax.swing.*;
import java.awt.*;
public class app_prg1 extends JApplet
{public void init(){}
public void paint ( Graphics g )
{g.drawString(" 3 +46 = "+(3+46),30, 30);}
}
\end{verbatim}
\end{frame}
```



**Figura 9.9** Opción `fragile`.

## 9.9 Entornos para código de programas

### 9.9.1 Entorno `semiverbatim`

A veces es adecuado mostrar el código de un programa en bloques de una manera no necesariamente lineal. Para esto usamos recubrimientos y un ambiente `semiverbatim`.

- `\alert<i>{texto}` para poner el texto en rojo.
- `\uncover<i>{...}` para controlar la secuencia de recubrimientos dentro del entorno `semiverbatim`

- `\visible<i>{texto}` para mostrar `texto` en la transparencia *i* pero fuera del entorno `semiverbatim`. Se podrían usar los siguientes comandos,

### Ejemplo 9.8

```
\begin{frame}[fragile]
\frametitle{Un algoritmo para buscar números primos}
\begin{semiverbatim}
\uncover<1>{\alert<0>{int main (void) {}}
\uncover<1>{\alert<0>{\{\}}
\uncover<1>{\alert<1>{ \alert<4>{std::}vector is_prime(100,true) {}}
\uncover<1>{\alert<1>{ for (int i = 2; i < 100; i++) {}}
\uncover<2>{\alert<2>{ if (is_prime[i]))}}
\uncover<2>{\alert<0>{ \{\}}
\uncover<3>{\alert<3>{ \alert<4>{std::}cout << i << " ";}
\uncover<3>{\alert<3>{ for (int j = 1; j < 100;)}
\uncover<3>{\alert<3>{ is_prime [j] = false, j+=i);}}
\uncover<2>{\alert<0>{ \{\}}
\uncover<1>{\alert<0>{ return 0;}}
\uncover<1>{\alert<0>{ \{\}}
\end{semiverbatim}
\visible<4>{Notar el uso de \alert{\texttt{std::}}.}
\end{frame}
```

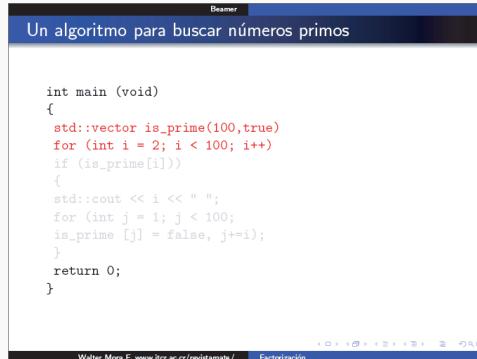


Figura 9.10 semiverbatim

### 9.9.2 Entorno `minted`

Para agregar color de manera automática al código de un lenguaje de programación se puede usar el paquete `minted`<sup>5</sup> (ver detalles en la sección 7.8). Para usar `minted` ponemos en el preámbulo `\usepackage{minted}`. Recordemos que el entorno `minted` es algo como,

```
\begin{minted}[opciones]{lenguaje}
... código
\end{minted}
```

<sup>5</sup>En Windows 7 hay que hacer algunos arreglos para instalar este paquete.

### Ejemplo 9.9

```
\begin{frame}[fragile]{Programación java} % fragile

\begin{minted}[frame=lines, rulecolor=\color{gray}]{java}

import javax.swing.*;
import java.awt.*;
public class app_prg1 extends JApplet
{ public void init(){}
  public void paint ( Graphics g ){
    g.drawString(" 3 +46 = "+(3+46),30, 30 );
  }

}\end{minted}

\end{frame}
```

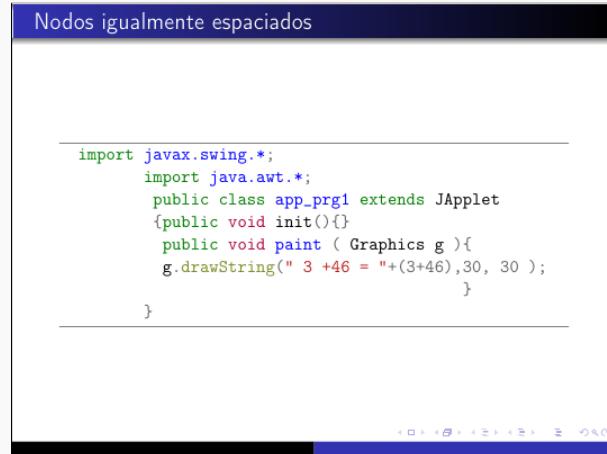


Figura 9.11 Paquete minted en Beamer

## 9.10 Beamer y el paquete algorithm2e

En esta sección vamos a ver un ejemplo en el que se usa el paquete algorithm2e (ver sección 7.7) en un entorno frame. Para este ejemplo, se puso en el preámbulo

---

```
%preámbulo
\usepackage[ruled,,vlined,lined,linesnumbered,algosection,portugues]{algorithm2e}
```

---

Observe que necesitamos la opción `fragile` para `frame` y la opción `[h]` para `algorithm`.

**Ejemplo 9.10**

```
\begin{frame}[fragile]
\begin{algorithm}[h] % <- necesario
\SetLine
\KwData{$(x_0, y_0), \dots, (x_m, y_m)$}
\KwResult{Coeficientes $\alpha_0, \alpha_1, \dots, \alpha_m$ en la base $\{X^{(0)}, \dots, X^{(n)}\}$}
\linesnumbered
$a_0=y_0$;
$s=\alpha_j-\alpha_0$;
$f=x_j-x_0$;
\SetVline
...
\Return $\alpha_j=s/f$ ;
}
\end{algorithm}
\end{frame}
```

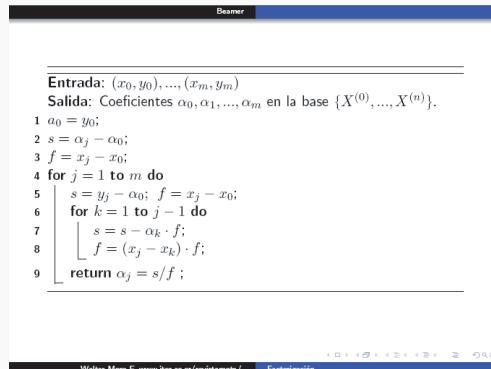


Figura 9.12 Beamer con algorithm2e

## 9.11 Gráficos

La inclusión de gráficos se hace de la misma manera que un documento **LaTeX** usual (ver capítulo 5). Como dijimos antes,

- ➊ Si hay figuras en distintos formatos, .eps, .pdf, .png, etc., podemos habilitar el paquete `epstopdf` y declarar las extensiones en el preámbulo de tal manera que podamos incluir las figuras *sin extensión* y compilar con **PDFLaTeX**. Habilitamos en el preámbulo,

```
\usepackage{epstopdf}
\DeclareGraphicsExtensions{.pdf,.png,.jpg,.gif,.eps}
```

- ➋ Si las figuras son todas .eps o .ps, pues se incluyen con esa extensión, y por supuesto se compila con **LaTeX+dvips**.

### Ejemplo 9.11

```
\begin{frame}{Superficie $$$ y proyección}
\begin{figure}[h]
\centering
\includegraphics{images/B_fig12.pdf} %compilando PDFLaTeX
\caption{Superficie $$$}\label{Beamer:fig12}
\end{figure}
\end{frame}
```

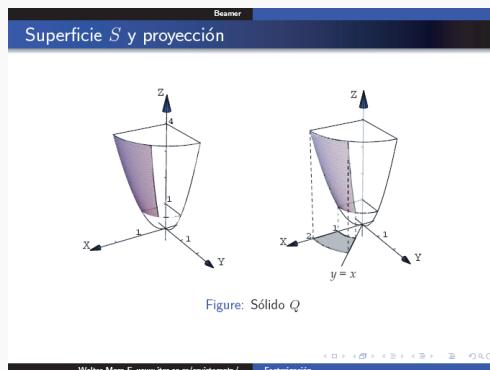


Figura 9.13 Incluir un gráfico

## 9.12 Ligas y botones.

Digamos que queremos poner un botón para ir a la transparencia  $j$  desde la transparencia  $i$  y, además, poner un botón en la transparencia  $j$  de retorno. Para esto usamos ligas simples o dos botones **Beamer**: \beamergotobutton y \beamerreturnbutton. Cada marco debe tener una identificación de marco y un marco destino.

```
\hyperlink{identificación del marco}{botón + texto}
```

```
\hypertarget<2>{identificación del destino}{}<!--</pre>

```

- También se puede incorporar ligas *sin* botones,

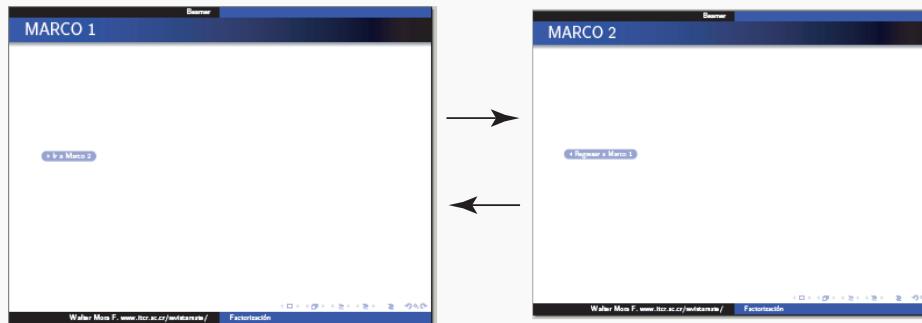
```
\begin{frame}{MARCO 3}
\hyperlink{MARCO3}{$\$ Ir a Marco 4}
\hypertarget<2>{MARCO4}{$<\$}
\end{frame}
```

```
\begin{frame}{MARCO 4}
\hyperlink{MARCO4}{Regresar a Marco 3}
\hypertarget<2>{MARCO3}{}<!--</pre>

```

**Ejemplo 9.12**

```
\begin{frame}{MARCO 1}
\hyperlink{MARCO1}{\beamergotobutton{Ir a Marco 2}}
\hypertarget<2>{MARCO2}{} 
\end{frame}
\begin{frame}{MARCO 2}
\hyperlink{MARCO2}{\beamerreturnbutton{Regresar a Marco 1}}
\hypertarget<2>{MARCO1}{} 
\end{frame}
```

**Figura 9.14** Ligas

## 9.13 Efectos de Transición. Color

Un efecto de transición de una transparencia A a una transparencia B, se puede lograr poniendo el comando respectivo en cualquier parte de la transparencia B. El efecto se logra ver solo a pantalla completa.

### Otros efectos

- \transblindshorizontal
- \transblindsvertical<2,3>
- \transboxin
- \transboxout
- \transglitter<2-3>[direction=90]
- Se pueden incluir películas, animaciones, etc. con \usepackage{multimedia}
- etc.

En el ejemplo que sigue, además de poner un efecto de transición vamos a crear una entorno tabular con las filas con color azul, específicamente ZurichBlue. Necesitamos hacer dos cosas en el preámbulo para que todo esto funcione,

- \documentclass[xcolor=pdftex,table]{beamer}. La opción “table” le informa a Beamer que el paquete colortbl debe ser cargado para poder usar la opción \rowcolors

- `\definecolor{ZurichBlue}{rgb}{.255,.41,.884}`. Con esto definimos lo que será nuestro ZurichBlue. En el código que sigue, el color se pone en distintos porcentajes.

### Ejemplo 9.13

```
%\documentclass[xcolor=pdftex,table]{beamer}
%\definecolor{ZurichBlue}{rgb}{.255,.41,.884}
\begin{frame}{Marco B}
\transdissolve % <--- Efecto de transición

\begin{center}
\rowcolors{1}{ZurichBlue!20}{ZurichBlue!5} %Porcentaje de color

\begin{tabular}{|l|l|c|}\hline
Enteros &\tt long & de $-2^{63}$ a $2^{63}-1$\\\hline
&\tt int & de $-2^{31}$ a $2^{31}-1$\\\hline
&\tt short & de $-2^{16}$ a $2^{16}-1$\\\hline
&\tt byte & de $-2^7$ a $2^7-1$\\\hline
&&\\\hline
Coma flotante&\tt float & de $3.4 \times 10^{-33}$ a $3.4 \times 10^{38}$\\\hline
&\tt double & de $1.7 \times 10^{-308}$ a $1.7 \times 10^{308}$\\\hline
Caracteres&\tt char &\\\hline
boolean&\tt true o false &\\\hline
\end{tabular}

\end{center}
\end{frame}
```

Enteros	<code>long</code>	de $-2^{63}$ a $2^{63}-1$
	<code>int</code>	de $-2^{31}$ a $2^{31}-1$
	<code>short</code>	de $-2^{16}$ a $2^{16}-1$
	<code>byte</code>	de $-2^7$ a $2^7-1$
Coma flotante	<code>float</code>	de $3.4 \times 10^{-33}$ a $3.4 \times 10^{38}$
	<code>double</code>	de $1.7 \times 10^{-308}$ a $1.7 \times 10^{308}$
Caracteres	<code>char</code>	
	<code>boolean</code>	<code>true o false</code>

Figura 9.15 Transición y Color

## 9.14 Ligas a Documentos Externos

Para hacer ligas a documentos externos podemos usar el comando

```
\href{http://...}{ texto}
```

### Ejemplo 9.14

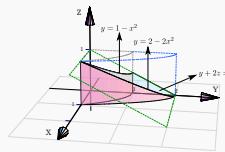
El código:

---

```
\parbox{3cm} {
\href{http://www.cidse.itcr.ac.cr/cursos-linea/3D-Web/exersolido21.html} {
\includegraphics[width=3cm]{images/exersolido21}}
}\parbox{12cm}{Sólido $Q_{14}$ limitado por las
superficies $y = 2 - 2x^2$; $y = 1 - x^2$; $y + 2z = 2$; $x = 0$ y $z = 0$; en el
I octante.}\\"
```

---

produce: (puede hacer clic sobre el gráfico)



Sólido  $Q_{14}$  limitado por las superficies  $y = 2 - 2x^2$   $y = 1 - x^2$ ;  $y + 2z = 2$ ;  
 $x = 0$  y  $z = 0$ ; en el I octante.

### 1 Liga a un documento en el disco duro, por ejemplo

El código del programa está [aquí](#)

---

El código del programa está

---

```
\href{file:///C:/MiJava/programa1.java}{\underline{aquí}}
```

---

### 2 Otras ligas

---

Prof. Walter Mora F.

---

```
{\href{mailto:wmora2@gmail.com}{(wmora2@gmail.com)}}\\
```

---

En la revista digital Matemática, Educación e Internet  
encontrará

---

```
\href{http://www.cidse.itcr.ac.cr/revistamate/}{\underline{el Manual de LaTeX}},  

en la liga ''Libros''\\
```

---

## 9.15 Animaciones

---

Se puede preparar una animación simplemente desplegando una secuencia de gráficos, por ejemplo

---

```
\begin{frame}
\frametitle{Mi animación}
\begin{figure}[t]
\centering
\includegraphics[1>[scale=0.2]{images/picture_1.pdf}
\includegraphics<2>[scale=0.2]{images/picture_2.pdf}
\includegraphics<3>[scale=0.2]{images/picture_3.pdf}
\includegraphics<4->[scale=0.2]{images/picture_4.pdf}
\end{figure}
\end{frame}
```

---

**Lecturas recomendadas.** Se recomienda las lecturas [9] y [15].





# 10 DOCUMENTOS LATEX EN INTERNET

---

En este capítulo hay dos secciones principales: Traducir un archivo L<sup>A</sup>T<sub>E</sub>X directamente a HTML con “LaTeX2HTML Translator” y la otra: Usar [PDFScreen](#) para desplegar un PDF en modo ‘presentación en pantalla’

## 10.1 LaTeX2HTML Translator

---

‘LaTeX2HTML Translator’ es un conjunto de scripts en PERL. LaTeX2HTML convierte un documento L<sup>A</sup>T<sub>E</sub>X (un archivo \*.tex) en un documento adecuado para la internet. LaTeX2HTML fue creado por Nikos Drakos y Ross Moore

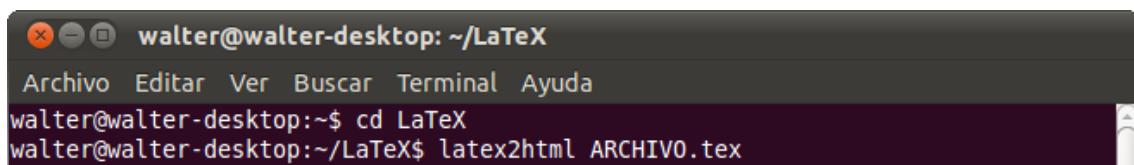
### La manera fácil

La conversión de un archivo L<sup>A</sup>T<sub>E</sub>X a HTML es sencilla en [Ubuntu](#): En el archivo .tex que desea convertir, debe agregar en el *preámbulo*

```
\usepackage{html,makeidx}
```

y luego abrir una terminal, ir a la carpeta donde está el archivo .tex, digamos que se llama ARCHIVO.tex, y ejecutar,

```
latex2html ARCHIVO.tex
```



```
walter@walter-desktop:~/LaTeX
Archivo Editar Ver Buscar Terminal Ayuda
walter@walter-desktop:~$ cd LaTeX
walter@walter-desktop:~/LaTeX$ latex2html ARCHIVO.tex
```

Esto genera una subcarpeta ARCHIVO con las imágenes y las páginas web (nodei.html). Eso es todo.

### La manera difícil

Para correr LaTeX2HTML Translator bajo Windows XP (no hemos hecho pruebas en Windows 7), se necesitan algunos programas: [Perl](#), [GhostScript](#) y [Netpbm](#) (biblioteca de conversión de imágenes).

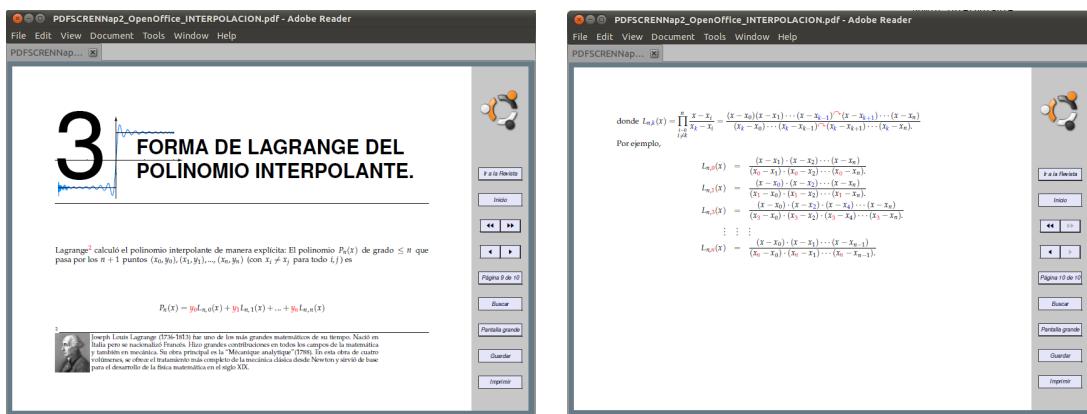
La mejor recomendación es: Si quiere traducir LaTeX a HTML con "LaTeX2HTML Translator", instale [Ubuntu](#) como huésped de [Windows](#) con [Wubi](#) (ver apéndice A) o en una ventana usando [VirtualBox](#) (necesitará el cd de instalación de [Ubuntu](#)), es fácil y le ahorrará mucho trabajo.

Si quiere probar en [Windows XP](#), pues deberá armarse de paciencia. Puede seguir las instrucciones en la página de [Jon Starkweather](#).

## 10.2 Otra Opción: PDFScreen

Hay otras opciones para poner texto matemático en Internet. Tal vez ya Ud. lo haya visto en [Google books](#) o en [Scribd \(ipaper\)](#). Bueno, aquí no vamos a hacer algo tan sofisticado, pero si algo parecido y muy eficiente.

Los navegadores tienen el plugin de Adobe Reader (sino, es fácil de instalar). Podemos convertir el texto [LATEX](#) en un PDF en modo 'presentación en pantalla', de tal manera que las páginas se carguen una a una. Existe un paquete muy eficiente que hace esto: [PdfScreen](#),



Para empezar, se podría agregar al *preámbulo*

```
\usepackage[spanish,screen,panelright,gray,paneltoc]{pdfscreen}
%Parámetros adicionales
%\emblema{images/logo.png}
\overlaysempty
\backgroundcolor{white}
\divname{Escuela de matem\'atica}
\margins{.75in}{.75in}{.75in}{.75in}
\screensize{6in}{9in}
\usepackage[epstopdf]
\DeclareGraphicsExtensions{.pdf,.png,.jpg,.gif, .eps}
```

% una imagen para el logo de la presentaci\'on.  
 % sin imagen de fondo  
 % color de fondo: blanco  
 % nombre de la Instituci\'on  
 % m\'argenes  
 % ancho y largo sugeridos  
 % Conversi\'on eps to PDF

Como es natural, se compila usando [PDFLaTeX](#) (se usa usar el paquete [epstopdf](#) para incluir cualquier tipo de imágenes como vimos en el capítulo 5). Algunas opciones del paquete son,

Opción	
screen	versión para pantalla
panelright	panel de navegación a la derecha
panelleft	panel de navegación a la izquierda
nopanel	sin panel de navegación
color	esquema de color: bluelace, blue, gray, orange, palegreen y chocolate
paneltoc	índice de contenidos en el panel

## Configuración adicional

Podemos modificar el archivo `pdfscreen.sty`. Este archivo está en la subcarpeta `/usr/share/texmf-texlive/tex/latex/base` en **Ubuntu** y en `C:/Archivos de programa/miktex2.x/tex/latex/base` en **Windows**. Aquí se puede redefinir los colores y agregar botones al menú, por ejemplo podemos agregar en la línea 624,

---

```
%botón de búsqueda
\Acrobatmenu{AcroSrch:Query}{\addButton{\buttonwidth}{\@Panelgobackname}}\\pfill
\Acrobatmenu{FullScreen}{\addButton{\buttonwidth}{\@Panelfullscreenname}}\\pfill
%botón
\Acrobatmenu{SaveAs}{\addButton{\buttonwidth}{\@Panelclosename}}\\pfill
```

---

La instrucción `\Acrobatmenu{AcroSrch:Query}...` agrega un botón para desplegar el menú de búsqueda en el PDF.

La instrucción `\Acrobatmenu{SaveAs}...` agrega un botón para desplegar el menú de Guardar como en el PDF.

Un manual del paquete se puede ver en <http://www.ctan.org/tex-archive/help/Catalogue/entries/pdfscreen.html>



## Bibliografía

---

- [1] Andrew Mertz y William Slough. *Beamer by Example*.  
En <http://www.tug.org/pracjourn/2005-4/mertz/mertz.pdf>
- [2] J. Bezos. *Tipografía*. <http://www.tex-tipografia.com/archive/tipos.pdf>
- [3] N. Drakos, R. Moore. *The LaTeX2HTML Translator*.  
En <http://cbl.leeds.ac.uk/>
- [4] F. Hernández. "Elementos de Infografía para la Enseñanza Matemática". Revista digital Matemática, Educación e Internet ([www.cidse.itcr.ac.cr/revistamate/](http://www.cidse.itcr.ac.cr/revistamate/)). Vol. 11, No 1. Agosto-Diciembre 2010.
- [5] Gilles Bertrand. "Preparing a presentation (Beamer)".  
En <http://www.rennes.enst-bretagne.fr/~gbertran/>
- [6] G. Grätzer *The New Standard LATEX*. Personal TeXInc. California. 1998.
- [7] I. Strizver. *Type rules! : the designer's guide to professional typography*. John Wiley & Sons, Second edition. 2006.
- [8] Keith Reckdahl. "Using Imported Graphics in LATEX2"  
<ftp://ftp.tex.ac.uk/tex-archive/info/epslatex.pdf>
- [9] Kijoo Kim. "Beamer v3.0 Guide".  
En <http://faq.ktug.or.kr/wiki/uploads/>
- [10] Hahn, J. *LATEX for everyone*. Prentice Hall, New Jersey, 1993.
- [11] L. Lamport. *LATEX*. Addison-Wesley. 1996.
- [12] M. Goossens; F. Mittelbach; A. Samarin. *The LATEX Companion*. Addison-Wesley. 1993.
- [13] R. Willians. *Non-Designer's Design Books*. Third Edition. Peachpit Press. 2008.
- [14] L. Seidel. "LaTeXtoHTML".  
En <http://apolo.us.es/CervanTeX/>
- [15] Till Tantau *User Guide to the Beamer Class, Version 3.07*  
En <http://latex-Beamer.sourceforge.net>, 2007.
- [16] The LaTeX Font Catalogue.  
En <http://www.tug.dk/FontCatalogue/utopia-md/>
- [17] Wichura M. "The Pictex Manual ". Universidad de Chicago, 1987.

MikTeX,  
TeXLive,  
MacTeX

## Apéndice A

### Instalar una distribución y un editor

---

#### A.0.1 Distribuciones $\text{\TeX}$

Hay varias distribuciones  $\text{\TeX}$  por ejemplo: TeXLive (Windows, Linux, Mac), MacTeX (Mac OS X) y MiK $\text{\TeX}$  (Windows). Las pruebas de este libro se hicieron con MiK $\text{\TeX}$ 2.9 (Windows XP, 32 bits) y con TeXLive 2010 ([Ubuntu](#) 10.10, 64 bits). Asumimos que el lector tiene la distribución respectiva *completa* a mano.

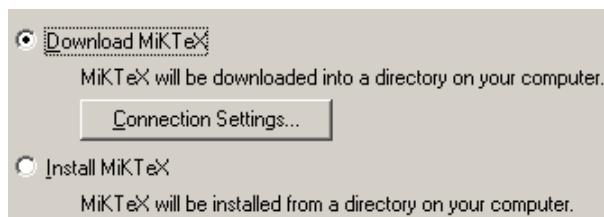


#### Obtener e Instalar MiK $\text{\TeX}$



Para obtener MiK $\text{\TeX}$ 2.9 se puede ir directamente al sitio web <http://miktex.org/2.9/setup> y descargamos “MiK $\text{\TeX}$  2.9 Net Installer”, esta opción descarga el ejecutable setup-2.9.3959.exe. Con este ejecutable se establece una conexión a Internet para *primero descargar* y luego instalar MiK $\text{\TeX}$  completo (hay una guía de instalación en <http://docs.miktex.org/2.9/manual/installing.html>). Los pasos básicos son:

- ① Ejecute setup-2.9.3959.exe y siga las instrucciones. En algún momento deberá elegir la opción ‘Download MiK $\text{\TeX}$ ’. Tenga en cuenta que esto puede tardar bastante tiempo (dependiendo de la velocidad de la conexión).

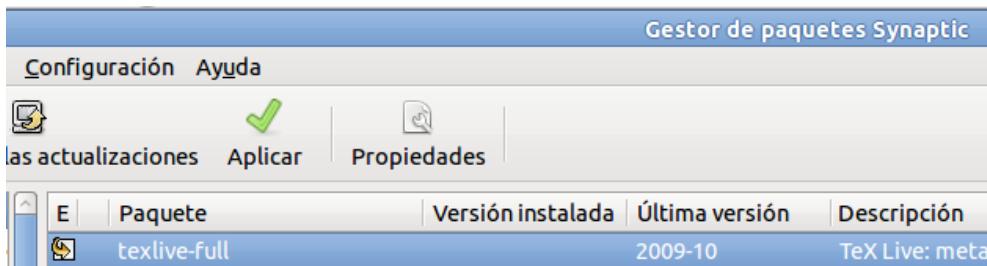


- ② Ejecute de nuevo setup-2.9.3959.exe y elija la opción ‘Install MiK $\text{\TeX}$ ’. Ahora se instalará MiK $\text{\TeX}$  desde la carpeta de descarga que eligió en el paso anterior.

## Obtener e Instalar TeXLive

En el apéndice B se indica cómo instalar [Ubuntu](#) 32 o 64 bits, desde Windows. [Ubuntu](#) 64 bits es recomendable para trabajar con tareas que requieren mucho trabajo gráfico o mucho cálculo (sin hablar de Mac), si no es este el caso, [Ubuntu](#) 32 bits es adecuado para el uso doméstico usual.<sup>1</sup>

En [Ubuntu](#), TeXLive se puede instalar con el '[Gestor de paquetes Synaptic](#)'. Se debe buscar la opción `texlive-full`, marcar y luego aplicar (también se puede instalar desde el 'Centro de Software de [Ubuntu](#)', buscando `tex live`).



### A.0.2 Un Editor

Después de la instalación de la distribución TeX instalamos un editor. Hay varios editores: [TeXmaker](#), [Texstudio](#), [Winshell](#), [Kile](#), etc. Los editores buscan la instalación TeX/LaTeX de manera automática. Luego se pueden configurar algunas cosas adicionales.



**Ubuntu:** Hay varios editores: [TeXMaker](#), [TeXMakerX](#), [Kile](#), etc.

**TeXMaker:** Se instala con el '[Gestor de paquetes Synaptic](#)'.

The screenshot shows the TexMaker interface. On the left is a large toolbar with various icons for file operations, text selection, and document structure. The main area contains a LaTeX code editor with several files open: `2011_Libro_LaTeX.tex`, `Formato.tex`, and `InstalarUbuntu.tex`. The code in `2011_Libro_LaTeX.tex` includes mathematical formulas and sections. To the right of the editor is a vertical sidebar titled 'Pages' which lists numbered pages from 1 to 14. At the bottom right of the editor window, there is a red header '1.1 ¿Qué es LaTeX?'.



Donald Knuth, 1938–2012  
“Añ Al Añumento  
TeX (diseñado y desarro  
un sofisticado programa  
ficos tales como artículos  
estándar para publicació  
computación, etc. LaTeX  
Leslie Lamport. LaTeX ne  
nos permite preparar auto  
dar y de alta calidad. Er  
comandos y LaTeX se enc  
a diferencia de un progra

### Configuración adicional

Aunque la instalación procede de manera automática, hay un par de cosas que podemos configurar.

- **Diccionario en español:** Vamos a [Opciones-Configurar TexMaker-Editor](#) y cambiamos a `/usr/share/myspell/dicts/es-CR.dic`

- **Búsqueda inversa (código↔visor):** Esta opción es muy útil para pasar del PDF, generado con L<sup>A</sup>T<sub>E</sub>X (DVI→PDF o P<sup>D</sup>F<sub>L</sub>A<sup>T</sup><sub>E</sub>X), al código T<sub>E</sub>Xy viceversa de tal manera que nos ubiquemos en el lugar actual del documento (funciona si usamos el visor default) Evince). Vamos a Opciones-Configurar TexMaker-Comandos. Modificamos la opción PdfLaTeX y digitamos

```
pdflatex -synctex=1 -interaction=nonstopmode %.tex
```

(solo agregamos -synctex=1). La búsqueda inversa se activa en el PDF con el botón derecho del ratón y luego haciendo clic.

**Nota:** A la fecha, esta faceta funciona bien con TeXMaker 2.1, asegúrese de instalar esta versión o una versión superior.

- **Visor PDF externo:** TeXMaker usa por defecto el visor Evince. Si queremos ver el documento en Adobe Reader como opción alternativa desde Evince, vamos a Opciones-Configurar TexMaker-VisorPDF-External Viewer y digitamos /usr/bin/acroread" %.pdf.
- **Visor Okular:** Un visor para archivos DVI más competente es Okular. Suponiendo que ya lo instalamos, vamos a Opciones-Configurar TexMaker-Comandos y modificamos la opción Visor DVI y digitamos okular --unique "file:%.dvisrc:@

Un manual muy instructivo y completo para TeXMaker se encuentra en <http://www.xmlmath.net/texmaker/doc.html>.

**TeXMakerX:** Es una derivación de TeXMaker con muchas facetas adicionales muy útiles. En particular el visor PDF es mejor (usa el visor de TeXworks) pero es un poco más lento para compilar. Se configura igual que TeXMaker en lo que tienen en común. Se descarga en <http://texmakerx.sourceforge.net/>.

**Kile:** Se instala con el 'Gestor de paquetes Synaptic'. Para configurar facetas adicionales de Kile se puede ir a <http://kile.sourceforge.net/Documentation/html/index.html>.

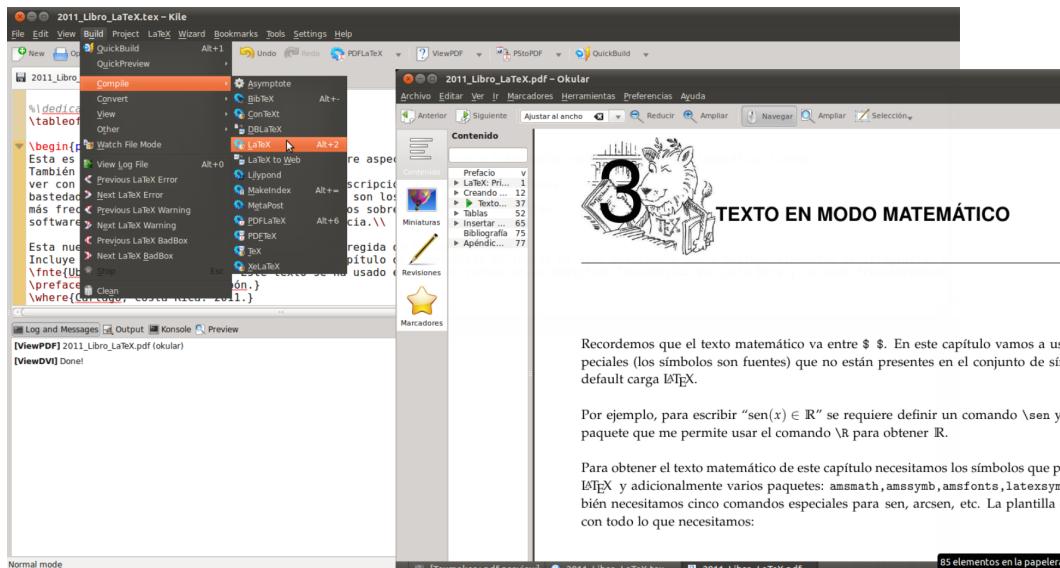


Figura A.1 Editor Kile en Ubuntu



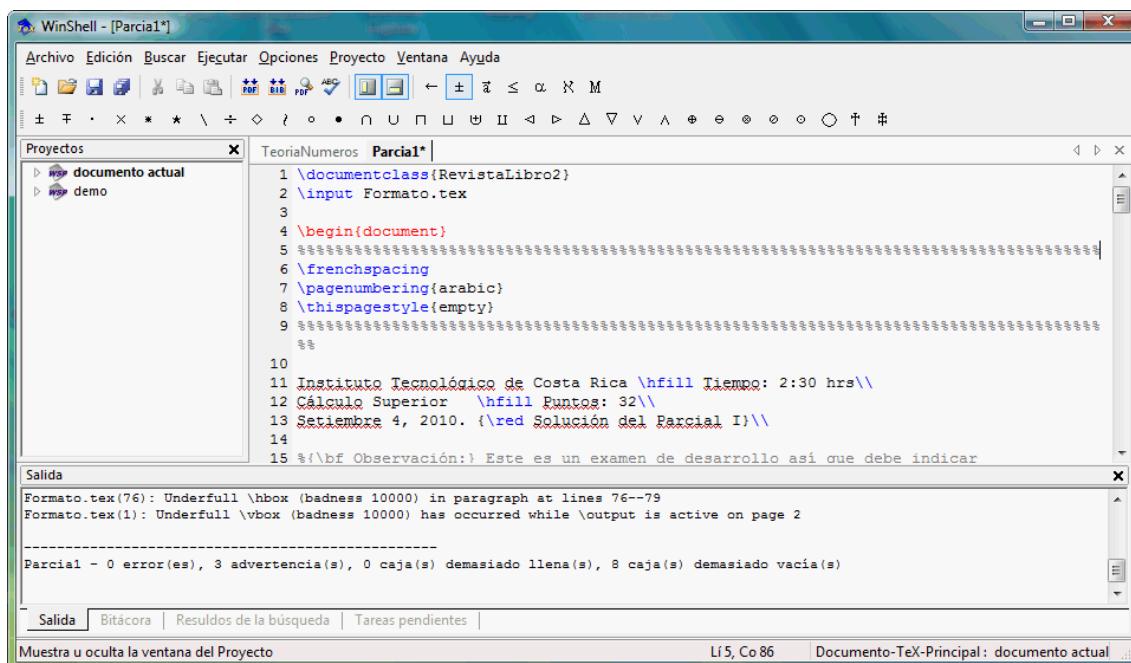
**Windows:** Hay varios editores: [TeXMaker](#), [Texstudio](#), [WinShell](#), etc.

**TeXMaker:** El instalador se descarga en <http://www.xmlmath.net/texmaker/>. La configuración adicional es la misma que la que se menciono para [Ubuntu](#) excepto el visor DVI pues en Windows se usa [Yap](#).

Para hacer búsqueda inversa ([código↔visor Yap](#)) desde el DVI (en [Windows](#)) se requiere configurar el **YAP**. Abrimos el **YAP** desde [TeXMaker](#) con la tecla **F3**, luego vamos a **View-Opciones-Inverse DVI Search** y pulsamos el botón **New**. En **Name:** digitamos **texmaker** (el nombre del editor para referencia), en la cejilla que sigue usamos el botón de navegación para indicar el ejecutable (**C:\ProgramFiles\texmaker\texmaker.exe**) y en la tercera cejilla digitamos **"%f" "%l"**. Pulsamos el botón **Aplicar** y luego **Aceptar**. Para hacer búsqueda inversa usando [PDFLaTeX](#) se debe usar [SumatraPDF-TeX](#) (ver más abajo).

**Texstudio:** Es una derivación de [TeXMaker](#) con varias facetas adicionales muy útiles. Se descarga en <http://texmakerx.sourceforge.net/>.

**WinShell:** El instalador se descarga en <http://www.winshell.org/>.



**Figura A.2** Editor WinShell para Windows

#### Configuración adicional

Aunque la instalación procede de manera automática, hay un par de cosas que podemos configurar.

- **Diccionario en español:** El diccionario se debe descargar en <http://wiki.services.openoffice.org/wiki/Dictionaries>. El contenido el comprimido se debe pegar en **C:/Archivosdeprograma/WinShell/Dictionaries** (puede ser Spanish Mex), luego vamos a [Opciones-Idioma...](#) en la cejilla **Corrector ortográfico:** cambiamos a **es\_MX**.
- **Búsqueda inversa:** Esta opción es muy útil para pasar del PDF, generado con [LATEX](#) (DVI→PDF) o [PDFLaTeX](#), al código [TeX](#) viceversa de tal manera que nos ubiquemos en el lugar actual del documento. En Windows esto se puede hacer con el visor [SumatraPDF-TeX](#) que se descarga en <http://william.famille-blum.org/software/sumatra/index.html> (no funciona con Adobe Reader). No se instala, solo se pega en **C:/Archivos de programa**. Ahora hay que configurar ambos programas,

- En [WinShell](#) vamos a [Opciones-Lamadas a programas....](#) En la cejilla Programa: seleccionamos PDFView y en la cejilla Archivo-exe: usamos el botón Examinar para poner la dirección del archivo SumatraPDF-TeX.exe (C:/Archivos de programa/SumatraPDF-TeX.exe). Eso es todo, luego reiniciamos [WinShell](#).
- Vamos a C:/Archivos de programa y ejecutamos SumatraPDF-TeX.exe, luego vamos a [Configuración-Opciones](#) y en la cejilla que está al final seleccionamos "C:\ProgramFiles\winShell\WinShell.exe" ... y pulsamos el botón Aceptar y cerramos.

Para hacer búsqueda inversa desde el DVI se requiere configurar el [YAP](#). Abrimos el [YAP](#) desde [WinShell](#) con la tecla [F7](#), luego vamos a [View-Opciones-Inverse DVI Search](#) y pulsamos el botón New. En Name: digitamos winshell (el nombre del editor para referencia), en la cejilla que sigue usamos el botón de navegación para indicar el ejecutable (C:/Archivos de programa\winShell\WinShell.exe) y en la tercera cejilla digitamos -c "%f" -l %. Pulsamos el botón Aplicar y luego Aceptar.

## A.1 Software adicional

---



### Software adicional (Windows)<sup>2</sup>

Además de la edición de texto, es usual trabajar con gráficos, imágenes y también con diseño editorial (en los archivos PDF generados con [LATEX](#)). Para hacer esto vamos a usar [Inkscape](#), [Gimp](#) y otros programas. [Inkscape](#) requiere la instalación previa de [Ghostscript](#), [Gsview](#) y [Pstoedit](#). Adicionalmente podemos descargar la extensión [TeXtext](#) de [Inkscape](#) para editar texto [LATEX](#) en esta aplicación.

También tres programas muy útiles son [PdfSam](#) (para extraer, unir o dividir páginas en un archivo pdf), [FastStone V5.3](#) (última versión libre, se usa para recortar y editar capturas de pantalla) y [Gimp](#) para edición de fotos e imágenes generales. Descargar e instalar *en este orden*:



- ① [GPL Ghostscript](#): Descargar gs900w32.exe o gs900w64.exe (Windows 64 bits) desde <http://sourceforge.net/projects/ghostscript/files/GPL%20Ghostscript/9.00/>
- ② [Gsview](#): Descargar desde <http://pages.cs.wisc.edu/~ghost/gsview/get49.htm>
- ③ [Pdfsam](#): Descargar desde [http://www.pdfsam.org/?page\\_id=32](http://www.pdfsam.org/?page_id=32)
- ④ [FastStone V5.3](#): Descargar desde <http://www.portablefreeware.com/?id=775>
- ⑤ [Gimp](#): Descargar desde <http://www.gimp.org/>
- ⑥ [Inkscape](#): Descargar desde <http://sourceforge.net/projects/inkscape/files/inkscape/0.48/>.



Nota. [Inkscape](#)<sup>3</sup> requiere Ghostscript, GsView y además,

- [Pstoedit](#): Descargar desde <http://www.pstoedit.net/pstoedit>
- [TeXtext](#): Se usa para introducir texto [LATEX](#) en la edición de gráficos e imágenes. Descargar en <http://pav.iki.fi/software/textext/>.
- [Replace Font](#): Se usa para reemplazar una fuente por otra fuente. Se descarga el comprimido replace\_font0.x.zip en <http://code.google.com/p/inkscape-replace-font/downloads/list> y se descomprime en C:/Archivodeprograma/Inkscape/share/inkscape/extensions

<sup>3</sup>En Windows esta extensión a veces no funciona bien con la versión 0.48, en todo caso en el repositorio que se menciona está la versión [0.46](#) de Inkscape que es también funcional para nuestros propósitos. A veces es necesario instalar la versión 0.46 después de la versión 0.46 para todo funcione.

Fuentes  $\text{\LaTeX}$  adicionales: **Inkscape** usa las fuentes del sistema. Se pueden instalar fuentes .ttf (Truetype) y .otf (Opentype). Si vamos a abrir PDF generados con  $\text{\LaTeX}$  es conveniente instalar al menos las fuentes .ttf básicas (las fuentes AMS en formato .ttf, generados con **FontForge**). Estas fuentes las puede obtener en [http://ifile.it/h4beg9c/ttf\\_LaTeXFonts\\_Minimo.zip](http://ifile.it/h4beg9c/ttf_LaTeXFonts_Minimo.zip), luego las descomprimimos y desde la terminal las movemos a la carpeta de fuentes de Windows (en el panel de control)

Un manual básico de **Inkscape** se encuentra en <http://inkscape.org/doc/basic/tutorial-basic.es.html>.

## **Software adicional (Ubuntu)**

Además de la edición de texto, es usual trabajar con gráficos, imágenes y también con diseño editorial (en los archivos PDF generados con  $\text{\LaTeX}$ ). Para hacer esto vamos a usar **Inkscape**, **Gimp** y otros programas. Si tenemos una conexión a Internet, los programas se pueden instalar con el **Centro de Software de Ubuntu** o con el **Gestor de paquetes Synaptic**, por ejemplo.



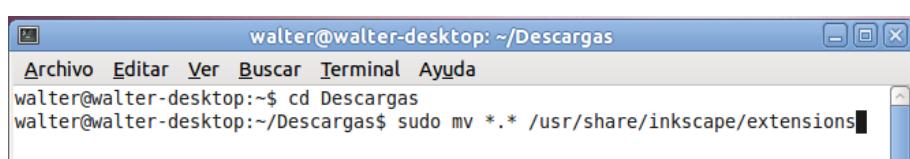
- ➊ Fuentes de Microsoft: Buscar 'ttf-mscorefonts' en 'Centro de Software de Ubuntu' e instalar. Fuentes adicionales se instalan desde **Centro de Software de Ubuntu-Tipografías**
- ➋ **Shutter** (para recortar y editar capturas de pantalla): Se instala con Synaptic
- ➌ **PdfSam** (para unir, separar o combinar pdfs): Se instalan con Synaptic
- ➍ **Adobe Reader**: Se instalan con Synaptic
- ➎ **Okular** (visualizador adicional): Se instala con Synaptic
- ➏ **Gimp**: Se instalan con Synaptic
- ➐ **Inkscape**. En Synaptic marcar **Inkscape**, **pstoedit** y **pdf2svg** y luego aplicar.

También vamos a usar las extensiones,

- ➑ **TeXtext**: Se usa para introducir texto  $\text{\LaTeX}$  en la edición de gráficos e imágenes. Se descarga el comprimido `textext-0.4.4.tar.gz` desde <http://pav.iki.fi/software/textext/> y se *descomprime*, digamos en la carpeta 'Descargas'.

Desde la terminal nos vamos a la carpeta 'Descargas' (con el comando 'change-directory': `cd`) y luego digitamos

```
sudo mv *.* /usr/share/inkscape/extensions [Enter]
```



- ➒ **Replace Font**: Se usa para reemplazar una fuente por otra fuente. Se descarga el comprimido `replace_font0.x.zip` desde <http://code.google.com/p/inkscape-replace->

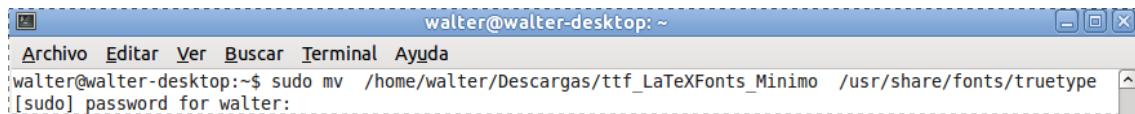
`font/downloads/list` y se *descomprime*, digamos en la carpeta 'Descargas'.

Desde la terminal nos vamos a la carpeta 'Descargas' (con el comando 'change-directory': `cd`) y luego digitamos,

```
sudo mv *.* /usr/share/inkscape/extensions [Enter]
```

Fuentes  $\text{\LaTeX}$  adicionales: Inkscape usa las fuentes del sistema. Se pueden instalar fuentes .ttf (Truetype) y .otf (Opentype). Si vamos a abrir PDF generados con  $\text{\LaTeX}$  es conveniente instalar al menos las fuentes .ttf básicas (las fuentes AMS en formato .ttf, generados con [FontForge](#)). Estas fuentes las puede obtener en [http://ifile.it/h4beq9c/ttf\\_LaTeXFonts\\_Minimo.zip](http://ifile.it/h4beq9c/ttf_LaTeXFonts_Minimo.zip), luego las descomprimimos y desde la terminal las movemos a la carpeta de fuentes,

```
sudo mv /home/walter/Descargas/ttf_LaTeXFonts_Minimo /usr/share/fonts/truetype [Enter]
```



En el apéndice B se recomienda software adicional.

## Apéndice B

### Ubuntu

---



**Ubuntu** es una distribución Linux basada en Debian GNU/Linux. **Ubuntu** es un sistema operativo para el usuario promedio, con un fuerte enfoque en la facilidad de uso y de instalación del sistema. Al igual que otras distribuciones se compone de múltiples paquetes de software normalmente distribuidos bajo una licencia libre o de código abierto.

#### B.1 Instalar **Ubuntu** desde Windows

---

Ubuntu se puede instalar y desinstalar desde Windows. Una vez instalado, al inicio de la sesión aparece la opción de entrar con Windows o Ubuntu

---

Microsoft Windows  
Ubuntu

---

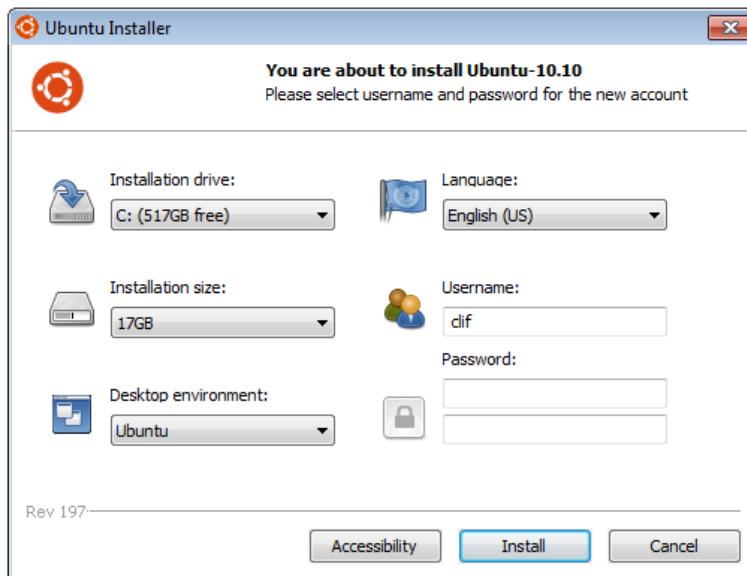
El instalador se es "Wubi.exe". También se puede instalar Ubuntu de manera independiente, dándonos la oportunidad de escoger las particiones.



\*



- ➊ Descargar en [Ubuntu-Descargas](#).  
Hay tres opciones: "Ubuntu Desktop Edition" en 32 bits o 64 bits y "Ubuntu Netbook Edition".
- ➋ Descomprimimos el archivo<sup>1</sup> Ubuntu-10.10-desktop... y ejecutamos el programa Wubi



- ➌ Al reiniciar la computadora, [Ubuntu](#) se termina de instalar. La distribución viene con algunos programas: [OpenOffice.org](#), [Firefox](#), etc. Si tenemos una conexión a internet<sup>2</sup>,
- ➍ Después de instalar [Ubuntu](#) es conveniente actualizar el sistema desde [Administración-Gestor de Actualizaciones](#).
- ➎ Es conveniente abrir el Centro de Software de [Ubuntu](#) para tener una idea de qué software hay disponible (siempre hay cosas nuevas)
- ➏ Antes de instalar, habilitamos más repositorios: Ir a [Administración-Gestor de Actualizaciones-Configurar-Software de Ubuntu](#) y habilitar las primeras 4 casillas
- ➐ Recomendación para instalación:
  - (a) [Texlive-full](#): Se instala con Synaptic
  - (b) [Lyx](#), [Kile](#), [TexMaker](#). (editores): Se instalan con Synaptic
  - (c) Fuentes de Microsoft: Buscar ttf-mscorefonts en '[Centro de Software de Ubuntu](#)' e instalar. Fuentes adicionales se instalan desde [Centro de Software de Ubuntu-Tipografías](#)
  - (d) [Shutter](#) (para recortar y editar capturas de pantalla): Se instala con Synaptic
  - (e) [PdfSam](#) (para unir, separar o combinar pdfs): Se instalan con Synaptic
  - (f) [QtOctave](#), [wxMaxima](#), [Scilab](#) (equivalente a MatLab): Se instalan con Synaptic
  - (g) [Google Chrome](#) (otro navegador): Se instala con Synaptic
  - (h) [Evolution](#) o [Thunderbird](#) (Gestión de correo)
  - (i) [Google Earth](#)
  - (j) [Skype](#)
  - (k) [Unrar](#) (para descomprimir): Se instalan con Synaptic

<sup>2</sup>En general, la conexión a internet es automática. Sino, seleccionar la red "eth0" y hacer clic en "editar". En la pestaña de "Ajustes de IPv4" seleccionar "manual" y rellenar datos de IP y DNS

(1) [Adobe Reader](#): Se instalan con Synaptic

8 [TurboPrint](#) es un programa (requiere licencia) para instalar drivers de impresoras posiblemente algo antiguas. TurboPrint agrega los drivers a la lista de la distribución. La descarga y las instrucciones están en <http://www.turboprint.info/download.html> Antes de comprar, hay que tomar en cuenta que en los foros de Internet hay mucha información sobre cómo conseguir e instalar drivers de impresoras en [Ubuntu](#).

9 [Inkscape](#).

- (a) [Inkscape](#) se instala con Synaptic. Además es adecuado instalar [pstoeedit](#) y [pdf2svg](#) (requiere Texlive-full) para trabajar con archivos pdf generados con LaTeX.
- (b) Extensión TeXtext para [Inkscape](#): Se descarga el comprimido [textext-0.4.4.tar.gz](#) y se descomprime.
- (c) Desde la terminal nos vamos a la carpeta que nos quedó (posiblemente "Descargas") y luego movemos el archivo a `inkscape/extensions`. En nuestra máquina sería,

```
walter@walter-desktop:~$ cd /home/walter/Descargas  
walter@walter-desktop:~$ sudo mv *.* /usr/share/inkscape/extensions [Enter]
```