

Walter Mora F
Alex Borbón A

Edición de Textos Científicos de

Alta Calidad

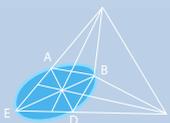
LaTeX

LaTeX2html y Presentaciones Beamer

$$\iint_S \frac{1}{a^2+z^2} dS = 2 \iint_D \frac{1}{a^2+z^2} \frac{a}{\sqrt{a^2-y^2}} dy dz$$

$$= 2 \int_{-a}^a \frac{a}{\sqrt{a^2-y^2}} dy \int_0^h \frac{1}{a^2+z^2} dz$$

$$= \lim_{\epsilon \rightarrow 0^+} 2a \arcsen \left(\frac{y}{a} \right) \Big|_{-a+\epsilon}^{a-\epsilon} \cdot \frac{1}{a} \arctan \left(\frac{z}{a} \right) \Big|_0^h = 2\pi \arctan \left(\frac{h}{a} \right)$$



Textos Universitarios

Revista Digital Matemática Educación e Internet (www.cisde.itcr.ac.cr)

EDICIÓN DE TEXTOS CIENTÍFICOS

LATEX

LaTeXtoHTML y Presentaciones Beamer

Walter Mora F.,
Alex Borbón A.
Escuela de Matemática
Instituto Tecnológico de Costa Rica.

Contenido

Prefacio	2
1 L^AT_EX	1
1.1 T _E X L ^A T _E Xy MikT _E X	1
1.2 Acciones en una sesión con L ^A T _E X	4
2 Creando un documento	8
2.1 Preámbulo y cuerpo	8
2.1.1 Plantilla básica	8
2.1.2 Otros comandos para el preámbulo	9
2.1.3 Cuerpo del documento	9
2.1.4 Otros ajustes	9
2.1.5 Idioma	9
2.2 Tipos y tamaños de letras.	12
2.2.1 Caracteres especiales.	12
2.2.2 Algunos tipos de fuentes (fonts).	12
2.2.3 Tamaños de letras.	13
2.3 Párrafos y efectos especiales.	17
2.3.1 Centrar	17
2.3.2 Cajas	18
2.3.3 Doble columna	18

2.3.4	El comando multicol	18
2.3.5	El ambiente minipage	18
2.3.6	El comando parbox	20
2.3.7	Otros efectos de texto	23
2.3.8	Texto como en la pantalla	23
2.3.9	Espacio horizontal y vertical	26
2.4	Enumerado automático.	26
3	Texto en modo matemático	30
3.1	Potencias, subíndices y superíndices	31
3.2	Tamaño natural	31
3.3	Raíces	31
3.4	Fracciones y “fracciones”	32
3.5	Elipsis (puntos)	34
3.6	Delimitadores	34
3.7	LLaves y barras horizontales	34
3.8	Acentos y “sombros” en modo matemático	35
3.9	Negritas en modo matemático	35
3.10	Espacio en modo matemático	36
3.11	Centrado	36
3.12	Contadores automáticos	36
3.13	Matrices	37
3.14	Alineamiento	39
3.15	Tablas de símbolos matemáticos frecuentes	40
3.15.1	Letras griegas	40
3.15.2	Operadores binarios	41
3.15.3	Relaciones	41
3.15.4	Negación de relaciones	41
3.15.5	Flechas	42
3.15.6	Operadores grandes	42
3.15.7	Otros símbolos	43
3.15.8	Especiales	43
3.15.9	Símbolos del paquete amssymb	43
4	Objetos y gráficos	44
4.1	Tablas	44
4.2	Inserción de gráficos.	50
4.3	Otros formatos gráficos	52
4.4	Los ambientes figure y table	54
4.5	Los ambientes wrapfigure y floatflt	56
4.5.1	wrapfigure	56

iv	CONTENIDO	
	4.5.2 floatflt	57
5	Formato del documento	59
5.1	Título, índice, secciones,...,bibliografía	59
5.2	Modulación	61
5.3	Abreviando comandos. Comando con opciones.	62
5.4	Numeración automática de definiciones, teoremas, ejemplos...	65
5.5	El paquete theorem	65
5.6	Paquete algorithm2e	67
6	Traducción \LaTeX a HTML: \LaTeX2HTML	71
6.1	Generalidades	71
6.2	Descarga e instalación de los programas	72
6.2.1	Instalación de PERL	72
6.2.2	GhostScript	73
6.2.3	Instalación de Mi \TeX	73
6.2.4	Instalación de \LaTeX 2HTML	74
7	Cómo hacer Transparencias con la clase Beamer	78
7.1	Introducción	78
7.2	Instalar Beamer	79
7.3	Un documento Beamer	79
7.4	Marcos	82
7.5	Velos (overlays)	82
7.5.1	Opciones <code><i-></code> y <code>\uncover<i-></code>	83
7.5.2	Opción <code><i- alert@ i></code>	84
7.6	Comando pause.	85
7.7	Entornos para teoremas, definición, etc.	86
7.8	Blocks.	86
7.9	Opción <code>fragile</code>	87
7.10	Entorno <code>semiverbatim</code>	88
7.11	Beamer y el paquete <code>algorithm2e</code>	89
7.12	Gráficos	90
7.13	Ligas y botones.	91
7.14	Efectos de Transición. Color	92
7.15	Ligas a Documentos Externos	94
7.16	Animaciones	95
	Appendix A: Ambiente <code>picture</code> de \LaTeX	96
A.1	\LaTeX Draw	97
A.2	Pi \TeX	99

	1
A.2.1 Comandos de PiCTeX	99
A.2.2 Escala y caja del dibujo.	99
A.2.3 Texto en un dibujo.	100
Bibliografía	101
Index	103

Prefacio

Este texto cubre la edición \LaTeX básica y también cubre tópicos relativamente avanzados, usualmente 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 y programación según nuestra experiencia. Sin embargo, dado la cantidad enorme de paquetes, los detalles más finos usualmente se deben consultar usando las máquinas de búsqueda en Internet (por ejemplo Google).

Este libro se ha usado en algunos cursos en el Instituto Tecnológico de Costa Rica y lo usan frecuentemente los asistentes, en la revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr).

W. MORA, A. BORBÓN.

Cartago, Costa Rica
Enero, 2009

Capítulo 1



1.1 $\text{T}_{\text{E}}\text{X}$ $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ Y $\text{M}\text{I}\text{K}\text{T}_{\text{E}}\text{X}$

“ $\text{T}_{\text{E}}\text{X}$ is intended for the creation of beautiful books - and especially for books that contain a lot of mathematic”.

Donald Knuth



Donald Knuth

$\text{T}_{\text{E}}\text{X}$ (implementado por Donald Knuth) es un sofisticado programa para preparar documentos *científicos* tales como artículos, reportes, libros, etc. $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ es un conjunto adecuado de comandos $\text{T}_{\text{E}}\text{X}$ preparado por Leslie Lamport. $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ no es un procesador de textos, es un programa que nos permite preparar automáticamente un documento de apariencia estándar y de alta calidad.

En general, solo necesitamos editar texto y algunos comandos y $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ se encarga de componer automáticamente la formulería del documento. A diferencia de un procesador de textos, con $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ tenemos un control más fino sobre cualquier aspecto tipográfico del documento.

■ EJEMPLO 1.1

LaTeX. Walter Mora F., Alex Borbón A.

Derechos Reservados © 2009 Revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr)

2 L^AT_EX

L^AT_EX formatea las páginas de acuerdo a la *clase* de documento especificado por `\documentclass{}`, por ejemplo, `\documentclass{book}`.

Un documento L^AT_EX puede tener texto ordinario junto con texto en *modo matemático*. Los comandos vienen precedidos por el símbolo “\”. Un ejemplo de código L^AT_EX es el siguiente:

```
\documentclass{book}
\usepackage{latexsym}

\begin{document}
$0^0$ es una expresi\on indefinida.
Si $a>0,$ $a^0=1$ pero $0^a=0.$
Sin embargo, convenir en que $0^0=1$ es adecuado para que
algunas f\ormulas se puedan expresar de manera sencilla,
sin recurrir a casos especiales, por ejemplo
Sin embargo, convenir en que $0^0=1$ es adecuado para que
algunas f\ormulas se puedan expresar de manera sencilla,
sin recurrir a casos especiales, por ejemplo

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

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

\end{document}
```

Este código, una vez compilado, produce una página con el texto:

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

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

MiK_TE_X

MiK_TE_X es una implementación de T_EX para Windows de distribución gratuita. Una de sus mejores facetas es su habilidad de actualizarse por sí mismo descargando nuevas versiones de componentes (vía Internet) y paquetes instalados previamente, y su fácil proceso de instalación.

Hay otras implementaciones de T_EX, por ejemplo TeX Live (multiplataforma) y MacTeX (Mac OS X).

LaTeX2HTML

LaTeX2HTML traduce documentos LaTeX para el Web a través de MiKTeX.

¿Cómo obtener MiKTeX?

Para obtener MiKTeX se puede ir directamente al sitio web de [MiKTeX](http://miktex.org/) (<http://miktex.org/>).

Se puede optar por el sistema básico ('Installing a basic MiKTeX system') o por el sistema completo ('Installing the complete MiKTeX system: MiKTeX 2.x Net Installer').

El sistema básico se puede configurarse para que se conecte a Internet (a un repositorio) cada vez que necesita un paquete.



Con el sistema completo, podemos indicar a MiKTeX (en el proceso de instalación, por ejemplo) la dirección en disco del conjunto de paquetes. Ambos sistemas vienen con un archivo "setup.exe" que se encarga de la instalación.

Si tiene una carpeta con todos los paquetes, puede indicarle a MiKTeX esta ubicación con:
INICIO - MiKTeX2.x - Browse Packages - Repository - Change Package Repository.

En este libro asumimos que el lector tiene el sistema completo a mano.

Adicionalmente, se debe descargar:

- Aladdin Ghostscript: <http://www.cs.wisc.edu/~ghost/index.html>
Ghostscript es un intérprete para el lenguaje PostScript. YAP (el visualizador de archivos L^AT_EX) usa Ghostscript para mostrar los gráficos EPS
Descargue Ghostscript 8.14 en la dirección:
<ftp://mirror.cs.wisc.edu/pub/mirrors/ghost/AFPL/gs814/gs814w32.exe>
- Descargue GSView en la dirección: <http://www.cs.wisc.edu/~ghost/index.html>
- Adobe Reader: <http://www.adobe.com/>

Si desea intentar instalar LaTeX2HTML, deberá descargar también

4 \LaTeX

- ActivePerl <http://www.activestate.com/Products/ActivePerl/>
ActivePerl es una implementación del lenguaje PERL para windows. PERL es usado por LaTeX2HTML (en realidad LaTeX2HTML es un conjunto de scripts escritos en Perl). MiKTeX usa algunas utilidades que son scripts en PERL
- LaTeX2HTML <http://saftack.fs.uni-bayreuth.de/~latex2ht/current/>
Se debe descargar la versión `Latex2html-2002-2.tar.gz` (se selecciona y se descarga con el botón derecho del mouse y se descomprime con 'Winzip' o 'Winrar')

Un Editor

MiKTeX no viene con un editor ('shell') de texto, sin embargo, se puede utilizar alguno de los editores siguientes:

- WinShell: <http://www.winshell.org/> (gratis, recomendado)
- LEd: <http://www.latexeditor.org/> (gratis, recomendado)
- WinEdit: <http://www.winedt.com> (shareware; 30 días de prueba; no vence, pero despliega una molesta ventana)
- FelTeX <http://www.itcr.ac.cr./revistamate/> (en v1, número 1, 2000. Gratis)

WinShell 330 y WinEdt

WinShell y WinEdt se instalan después de MiKTeX. Ambos buscan la máquina LaTeX revisando el registro. La configuración se hará de manera automática.

1.2 ACCIONES EN UNA SESIÓN CON \LaTeX

En una sesión \LaTeX ejecutamos varias acciones:

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

```
\begin{document}... texto ... \end{document}
```

- **Editamos**: escribimos texto corriente y texto en *modo matemático* (posiblemente combinando ambos). El texto en modo matemático va entre $\$ \$$ o $\backslash [\]$ si se quiere centrado, otra posibilidad para este último es $\$ \$ \$ \$$. Esto le indica al programa que interprete el texto y lo convierta en símbolos matemáticos.

- **Compilamos:** esto nos permite detectar, por ejemplo, errores en los comandos.

Compilar con WinShell

La compilación se hace presionando el botón \LaTeX en WinShell (ver la figura 1.1). Esto generará una información de salida. Si la compilación encuentra algún error se indicará con un mensaje corto en rojo, por ejemplo

```
archivo.tex(89): Error: ! Missing $ inserted.
```

Esto nos indica que falta el símbolo \$ en la línea 89.

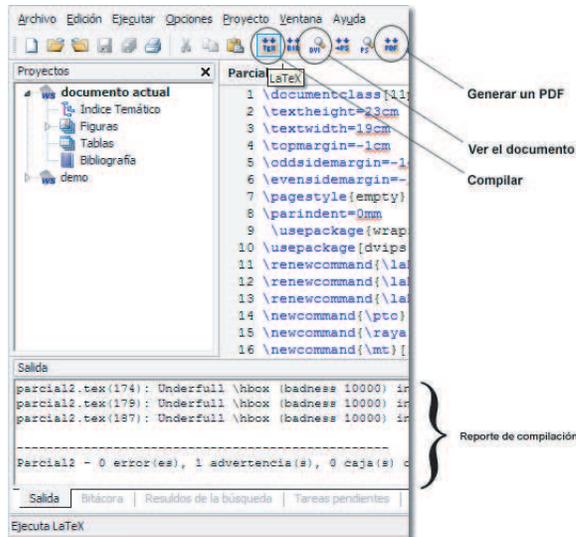


Figura 1.1 Ventana de edición de WinShell

Compilar con WinEdt

La compilación con WinEdt se hace también el botón \LaTeX pero en este caso se abre una ventana MS-DOS.

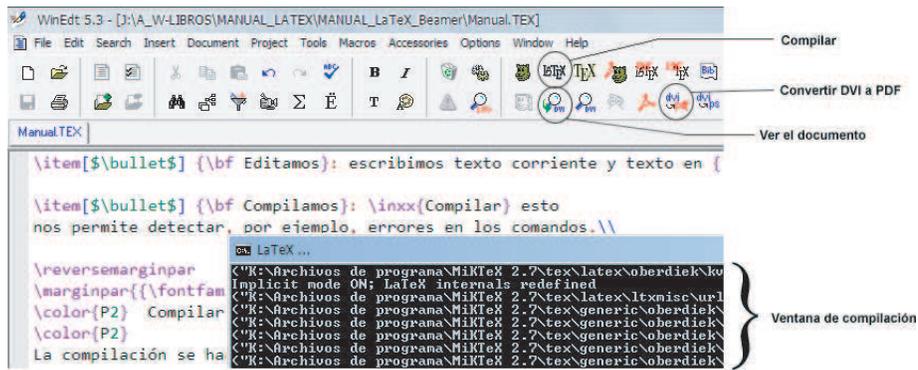


Figura 1.2 Ventana de edición de WinEdt

Al detectarse algún error en el documento durante la compilación, aparecerá un mensaje e inmediatamente después el símbolo ?. Hay que digitar algo. Se puede digitar (dando enter después) lo siguiente:

- (a) ? x : vuelve al documento sin generar un nuevo archivo ‘.dvi’ (necesario para ver el documento e imprimir).
- (b) ? r : sigue adelante la compilación sin detenerse en los errores detectados. Todos los mensajes se guardan en el archivo respectivo ‘.log’.
- (c) ? e : se devuelve al editor y se posiciona en el número de línea del error
- (d) ? h : se obtiene una explicación adicional del error. Para proseguir se da enter o alguna de las opciones anteriores.

A veces se despliega un mensaje de error que no corresponde exactamente al error cometido y también, número de línea del error puede ser que no corresponda a la al número de línea exacto. Los errores más comunes son olvidar una } o un \$ o un \ o intercambiar] con }. A veces dejar renglones en blanco en algunos “ambientes” produce un error.

Después de compilar se producen varios archivos: *.tex, *.dvi, *.aux, *.log, *.toc. El archivo de edición tiene extensión *.tex. Para imprimir un documento L^AT_EX solo necesitamos el archivo *.dvi y los archivos de los gráficos incluidos en el documento (si hubiera).

- **Ver el archivo DVI:** Una vez que hemos compilado, presionamos el botón DVI en WinShell.
- **Imprimir el archivo DVI**

- **Convertir de DVI a PostScript:** Esto se logra con el botón DVI→PS en WinShell.
- **Convertir de TEX a PDF:** Esto se hace con el botón PDF T_EX de WinShell.

Capítulo 2

CREANDO UN DOCUMENTO

2.1 PREÁMBULO Y CUERPO

Un documento básico en $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ se compone de dos partes: el *preámbulo* del documento y el *cuerpo* del mismo. Al inicio del documento se debe especificar la clase de documento y lo relativo al ajuste de las páginas, este es el preámbulo, 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) del documento en sí, es decir, el cuerpo es la parte que aparecerá impresa como producto final.

2.1.1 Plantilla básica

```
Preámbulo { \documentclass{article}
            { \textheight=21cm
              \textwidth=17cm
              \topmargin=-1cm
              \oddsidemargin=0cm
              \parindent=0mm
              \usepackage{graphicx} %paquete para incluir gr'\aficos
              \begin{document}
Cuerpo { ... texto + texto en modo matemático + comandos ...
        { \end{document}
```

- `\documentclass{article}`: Es la clase de documento, `article` se refiere al archivo `article.cls` y se utiliza para hacer artículos. En vez de 'article' se puede utilizar 'report' o 'book' para un reporte o un libro.
- `\textheight=21cm`: Establece el largo del texto en cada página. El default es 19 cm.
- `\textwidth=17cm`: Establece el ancho del texto en cada página (en este caso, de 17 cm). El default es 14 cm.
- `\topmargin=-1cm`: Establece el margen superior. El default es de 3 cm, en este caso la instrucción sube el margen 1 cm hacia arriba.
- `\oddsidemargin=0cm`: Establece el margen izquierdo de la hoja. 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.

2.1.2 Otros comandos para el preámbulo

- `\renewcommand{\baselinestretch}{1.5}` genera un texto a espacio y medio. Si se pone 2, lo hace a doble espacio.
- `\pagestyle{empty}` elimina la numeración de las páginas.
- `\parskip=Xmm` genera un espacio de X mm entre los párrafos.
- `\parindent=0mm` elimina la sangría.
- `\pagestyle{myheadings}` coloca la numeración de página en la parte superior.

2.1.3 Cuerpo del documento

El cuerpo del documento es el que se pone entre los delimitadores `\begin{document}` y `\end{document}`.

En esta parte se coloca el texto del documento junto con el texto matemático.

2.1.4 Otros ajustes

- `\markright{'texto'}` coloca 'texto' en la parte superior de la página. Se pueden poner varios `\markright` en el texto (en cada sección).

Ejemplo: `\markright{\LaTeX \hrulefill W. Mora, A. Borbn $}; \;$}`

- `\newpage` le indica a \LaTeX que siga imprimiendo en la página siguiente.

2.1.5 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.

Por ejemplo, L^AT_EX normalmente no acepta tildes, ni la letra 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
<code>\’a</code>	á	<code>?‘</code>	¿
<code>\’e</code>	é	<code>!‘</code>	¡
<code>\’{\i}</code>	í	<code>‘ ‘ . ’ ’</code>	“ . ”
<code>\’o</code>	ó	<code>‘ . ’</code>	‘ . ’
<code>\’u</code>	ú	<code>\~n</code>	ñ

Tabla 2.1 Acentos en modo texto y otros símbolos

Sin embargo, si en el preámbulo se coloca la instrucción

```
\usepackage[latin1]{inputenc} % Caracteres con acentos.
```

se tendrá un soporte completo para el español, ahora sólo las comillas se tendrán que seguir poniendo mediante la instrucción dada en la tabla.

Otro problema que tiene L^AT_EX con el idioma es que los títulos de las secciones 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 poner en el *preámbulo* del documento la instrucción

```
\usepackage[spanish]{babel} % Caracteres con acentos.
que carga la opción en español de la librería babel; esta librería también tiene soporte para otros idiomas como alemán, francés, italiano, etc.
```

Tópico
adicional

A veces el paquete `babel` tiene conflictos con algún paquete que deseamos usar. Por eso tenemos que tener en cuenta una segunda opción: Poner en el *preámbulo*

```
\renewcommand{\contentsname}{Contenido}
\renewcommand{\partname}{Parte}
\renewcommand{\indexname}{Lista Alfab\’etica}
\renewcommand{\appendixname}{Ap\’endice}
\renewcommand{\figurename}{Figura}
\renewcommand{\listfigurename}{Lista de Figuras}
\renewcommand{\tablename}{Tabla}
\renewcommand{\listtablename}{Lista de Tablas}
\renewcommand{\abstractname}{Resumen}
\renewcommand{\chaptername}{Cap\’{\i}tulo}
\renewcommand{\refname}{Bibliograf\’{\i}a}
```

Nota: Si el tipo de documento es un libro, para la bibliografía se debe poner

```
\renewcommand{\bibname}{Bibliograf\'}{\i}a}
```

Esto también funciona si queremos cambiar algún título en español de los que pone la librería babel.

Una tercera opción es hacer este cambio permanente: ir a la carpeta de instalación de MiKTeX (usualmente `C:\Archivos de programa\MiKTeX 2.7\tex\latex\base`), buscar y abrir el archivo de texto `article.cls` (o `report.cls` o `book.cls`)

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 Alfab\'}{etica}
...
```

Luego, simplemente se salva (Guardar) el archivo.

2.1.5.1 *División de palabras* Con la inclusión del paquete

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

en el preámbulo se logra que L^AT_EX divida 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. Por ejemplo:

```
de\-ci\-si\'on.
```

Este sistema tiene el inconveniente de que L^AT_EX sólo divide bien la palabra en ese punto del documento y si dicha palabra aparece otra vez habrá que volver a decirle como se divide, y tiene la ventaja de que funciona con palabras que tienen acento.

2.2 TIPOS Y TAMAÑOS DE LETRAS.

2.2.1 Caracteres especiales.

Algunos caracteres están **reservados** para que cumplan alguna función, por eso no se pueden obtener digitándolos 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 para imprimirlos.

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

Carácter	Comando
<code>\</code>	<code>\$\$\backslash\$</code>
<code>{, }</code>	<code>\$\$\{ \$, \$ \} \$</code>
<code>\$</code>	<code>\\$</code>
<code>&</code>	<code>\&</code>
<code>_, ⊕</code>	<code>_ , \^{}</code>
<code>#</code>	<code>\#</code>
<code>Λ</code>	<code>\~{}{ }</code>
<code>%</code>	<code>\%</code>

El comando `\verb` permite imprimir los caracteres tal y como aparecen en pantalla. Por ejemplo, si se quiere digitar `\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.

2.2.2 Algunos tipos de fuentes (fonts).

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

Comando	Produce
<code>{\rm Roman }</code>	Roman
<code>{\em Enftico }</code>	<i>Enfático</i>
<code>{\bf Negrita }</code>	Negrita
<code>{\it Itlica }</code>	<i>Itálica</i>
<code>{\sl Slanted }</code>	<i>Slanted</i>
<code>{\sf Sans Serif }</code>	Sans Serif
<code>{\sc Small Caps }</code>	SMALL CAPS
<code>{\tt Typewriter }</code>	Typewriter
<code>\underline{ Subrayado }</code>	<u>Subrayado</u>

2.2.3 Tamaños de letras.

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

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

■ EJEMPLO 2.1

`{\large \bf 'Ejemplo'}` produce **Ejemplo**.

Otros tamaños de fuente.

El tamaño de la fuente se puede controlar usando el paquete `anyfontsize`. Debemos poner el *preámbulo* `\usepackage{anyfontsize}`. Luego podemos usar el comando `\fontsize{x}{1}` para establecer el tamaño de la fuente (ver ejemplo 2.3).

■ EJEMPLO 2.2

El código:

```
\documentclass{article}
\usepackage{anyfontsize}
\usepackage{t1enc}
%\input t1cmr.fd
%\DeclareFontFamily{T1}{cmr}{}
%\DeclareFontShape{T1}{cmr}{m}{n}{<10>ecrm1000}{}
\begin{document}
\fontsize{80}{28}\selectfont \LaTeX
\end{document}
```

Produce: una página con L^AT_EX

Tópico
adicional

Fuentes y tamaño de la fuente.

Muchas fuentes son instaladas por default en la instalación de MiKTeX, otras fuentes especiales se pueden agregar de manera automática: Al usar un paquete para fuentes, MiKTeX lo carga de manera automática en la primera compilación (en el caso de que esté disponible).

Si una fuente está disponible, puede aplicar esta fuente a un texto o de manera global.

Cualquier fuente en LaTeX tiene cinco atributos:

`\fontencoding{}`: El orden en el que aparecen los caracteres en la fuente. Lo más común es

```
OT1  TEX text
T1   TEX extended text
OML  TEX math italic
OMS  TEX math symbols
OMX  TEX math large symbols
```

`\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 (‘‘oblique’’)
 sc Caps and small caps

`\fontsize{x}{1}`: Tamaño de la fuente.

Usualmente usamos instrucciones tales como

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

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

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

Para ver un conjunto completo de fuentes disponibles y los paquetes y el código que se deben usar para usar cada fuente, se puede ir a [“The LaTeX Font Catalogue”](http://www.tug.dk/FontCatalogue/) en

<http://www.tug.dk/FontCatalogue/>

■ EJEMPLO 2.3

Para usar la fuente Calligra, debemos poner en el *preábul*

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

Luego podemos cambiar la fuente de un poco de texto con la instrucción:

```
{\fontfamily{calligra}\fontsize{15}{1}\selectfont{Hab\’ia una vez ...}}
```

Esta instrucción produce:

Había una vez ...

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

{\fontfamily{pzc}\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\xxx`. Aquí, `xxx` se refiere al nombre de la familia, por ejemplo:

```
\normalfont\calligra ... texto ...\normalfont\normalfont
```

Cambio global de fuente.

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

```

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

```

■ EJEMPLO 2.4

concrete

cmbright

Sea $P(x) = a_n x^n + \dots + a_0 \in \mathbb{Z}[x]$ un polinomio *primitivo* (el máximo común divisor de sus coeficientes es 1). Sea $x = y/a_n$, entonces $Q(y) = a_n^{n-1} P(y/a_n)$ es mónico y tiene coeficientes enteros. Además,

$$P(x) = \frac{1}{a_n^{n-1}} Q(a_n x).$$

Sea $P(x) = a_n x^n + \dots + a_0 \in \mathbb{Z}[x]$ un polinomio *primitivo* (el máximo común divisor de sus coeficientes es 1). Sea $x = y/a_n$, entonces $Q(y) = a_n^{n-1} P(y/a_n)$ es mónico y tiene coeficientes enteros. Además,

$$P(x) = \frac{1}{a_n^{n-1}} Q(a_n x).$$

2.3 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 indentar 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.

■ EJEMPLO 2.5

El texto:

```
{\bf Introducci'on.} \\  
Se parte de un conjunto  $\Omega$   
de  $n$  patrones, objetos  
o ‘individuos’,  
descritos por un vector  
de  $p$  atributos.
```

Produce:

Introducción.

Se parte de un conjunto Ω de n patrones, objetos o ‘individuos’, descritos por un vector de p atributos.

2.3.1 Centrar

Para centrar un texto se pone éste entre los comandos `\begin{center}` `\end{center}`

■ EJEMPLO 2.6

El texto:

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

Produce:

Manual de
 \LaTeX

2.3.2 Cajas

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

■ EJEMPLO 2.7

El texto:

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

Produce:

Manual de


2.3.3 Doble columna

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

2.3.4 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}{#}`; así, la instrucción

```
\begin{multicols}{2}
Hace que...
\end{multicols}
```

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.

Una limitación de este ambiente es que las columnas se distribuyen con el mismo ancho y no se puede tener control sobre la cantidad

de texto que aparece en cada una de ellas; sin embargo, esto es una gran ventaja cuando no se quiere tener control sobre estos detalles.

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.

2.3.5 El ambiente minipage

La segunda manera de insertar dentro del documento texto a doble columna, es utilizar el formato `minipage` (también se puede utilizar `twocolumn`), la sintaxis sería de la siguiente manera.

```
\begin{minipage}[t]{Xcm}
PRIMERA COLUMNA \
*****\
```

LaTeX. Walter Mora F., Alex Borbón A.

Derechos Reservados © 2009 Revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr)

```

*****\\
*****\\

\end{minipage}
\ \ \hfill \begin{minipage}[t]{Xcm}

SEGUNDA COLUMNA \\
*****\\
*****\\
*****\\
\end{minipage}

```

Donde ‘X cm’ especifica el ancho de cada columna.

En vez de ‘X cm’ 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” (botton) se usa para alinear las dos columnas en el “fondo” del ambiente minipage. También se puede usar la opción “t” (top). La elección depende del contenido de cada ambiente “minipage”.

■ EJEMPLO 2.8

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 satlites son un icosaedro, un dodecaedro y un tetraedro.
Las figuras fueron generadas con {\sc Mathematica} y {\sc Corel Photo House}.
\end{minipage}
\ \ \hfill \begin{minipage}[b]{0.6\textwidth}
\begin{center}
\includegraphics{images/ML_fig3.eps}
\caption{{\small 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 ADOBE ILLUSTRATOR.

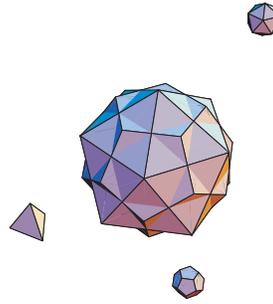


Figura 2.1 Poliedros

2.3.6 El comando parbox

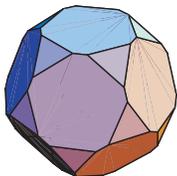
Un comando similar a minipage es `\parbox{xcm}{texto}`. Parbox también se puede usar en combinación con `\framebox` para hacerle un recuadro a la imagen: `\framebox{\parbox{xcm}{texto}}`. Este comando no está pensado para grandes bloques de texto. Veamos un ejemplo:

■ EJEMPLO 2.9

El texto:

```
\parbox{3cm}{\includegraphics{images/ML_fig4.eps}}
\parbox{10cm}{En {\sc Mathematica}, podemos eliminar una o varias
caras de un dodecahedro, seleccionar el color y el grosor de las
aristas y poner color a las caras. Para esto debemos utilizar los
comandos {\tt FaceForm[{}], EdgeForm[{}],
Drop[Dodecahedron[{}],{}] Lighting->False}.
Podemos también controlar la posición y
el tamaño indicando coordenadas en el espacio y
un factor de escalamiento.
Tanto FaceForm[...] como EdgeForm[...] pueden recibir primitivas
relacionadas con color y grosor. Poner el comando {\tt EdgeForm[]} tiene
como efecto eliminar las aristas.}
```

produce:



En MATHEMATICA, podemos eliminar una o varias caras de un dodecahedro, seleccionar el color y el grosor de las aristas y poner color a las caras. Para esto debemos utilizar los comandos `FaceForm[]`, `EdgeForm[]`, `Drop[Dodecahedron[],]`, `Lighting->False`. Podemos también controlar la posición y el tamaño indicando coordenadas en el espacio y un factor de escalamiento. Algunas opciones solo se pueden habilitar en el contexto `Graphics3D[]` mientras que otras son propias de los contextos `Polyhedron[]` y `Stellate`. Tanto `FaceForm[...]` como `EdgeForm[...]` pueden recibir primitivas relacionadas con color y grosor. Poner el comando `EdgeForm[]` tiene como efecto eliminar las aristas.

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 cubrir todo el ancho de la página sino \LaTeX seguirá escribiendo en el espacio sobrante.

Color y cajas.

Tópico
Adicional

- Para usar colores se debe llamar al paquete `color:\usepackage{color}`. 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:

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

La definición de color se pone en el *preámbulo*.

■ EJEMPLO 2.10

Usando el selector de colores de Microsoft Word, podemos definir un anaranjado personalizado.



(a) Selector de Color

El anaranjado personalizado tiene parámetros 234 112 0. Para poderlas usar, deben ser números entre cero y uno, y el selector devuelve valores entre 0 y 255, así que los dividimos por 255. Ponemos en el *preámbulo*

$$\backslash\text{definecolor}\{\text{miorange}\}\{\text{rgb}\}\{0.91, 0.43, 0.0\}$$

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

$$\backslash\text{fcolorbox}\{\text{color fondo}\}\{\text{color borde}\}\{\text{texto}\}$$

■ EJEMPLO 2.11

El código `\fcolorbox{miorange}{miorange}{ \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 2.12

```
{\fboxsep 12pt%
\fcolorbox{miorange}{white}{%
\begin{minipage}[t]{10cm}
$0^0$ es una expresi'on indefinida.
Si $a>0$, $a^0=1$ pero $0^a=0.$
Sin embargo, convenir en que $0^0=1$ es adecuado para que
algunas f'ormulas se puedan expresar de manera sencilla,
sin recurrir a casos especiales, por ejemplo

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

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

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

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

2.3.7 Otros efectos de texto

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

■ EJEMPLO 2.13

```
{ Instituto Tecnol\`ogico de Costa Rica \hfill Tiempo: 2:45 hrs}
```

producirá:

Instituto Tecnológico de Costa Rica Tiempo: 2:45 hrs

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

■ EJEMPLO 2.14

```
{Instituto Tecnol\`ogico de Costa Rica \hrulefill Tiempo: 2:45 hrs}
```

produce:

Instituto Tecnológico de Costa Rica _____Tiempo: 2:45 hrs

■ EJEMPLO 2.15

```
{Instituto Tecnol\`ogico de Costa Rica \dotfill Tiempo: 2:45 hrs}
```

produce:

Instituto Tecnológico de Costa RicaTiempo: 2:45 hrs

- `\footnote{ 'texto1' }`: produce una nota al pie de la página con el texto 'texto1'. El comando se escribe exactamente donde se quiere que quede la etiqueta que hará referencia al pie de página.

2.3.8 Texto como en la pantalla

Para reproducir exactamente lo que esta en la pantalla (incluyendo espacios) se pone el texto entre los comandos `\begin{verbatim} ... \end{verbatim}`.

■ EJEMPLO 2.16

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 este ejemplo, para controlar el tamaño del texto se puso

```
{\footnotesize
\begin{verbatim}
...
\end{verbatim}
}
```

Tópico
adicional

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 dentro del ambiente `verbatim`.

Estas cosas se pueden hacer con el paquete `fancyvrb`: Ponemos en el *preámbulo* `\usepackage{fancyvrb}`.

- `verbatim` en notas al pie de página: se usa `footnote` de la manera usual pero, en este caso, precedido por `\VerbatimFootnotes`.

■ EJEMPLO 2.17

El texto:

```
El peor ejemplo de programación recursiva es el del factorial
\VerbatimFootnotes\footnote{Se refiere al código}
\begin{verbatim}
int factorial(int n){
    if (n == 0) return 1;
    return n * factorial(n-1);}
\end{verbatim}} aunque es un ejemplo muy claro.
```

Produce:

El peor ejemplo de programación recursiva es el del factorial ¹ aunque es un ejemplo muy claro.

¹Se refiere al código

```
int factorial(int n)
{
    if (n == 0) return 1;
    return n * factorial(n-1);
}
```

A veces es adecuado introducir texto en modo matemático en un ambiente `verbatim`. Para hacer esto, usamos el comando `Verbatim`. En el ejemplo que sigue, se usan dos varias opciones: `frame=lines` para poner segmentos de línea al inicio y al final, `xleftmargin` y `xrightmargin` para ajustar estos segmentos y `commandchars` para especificar los caracteres de código matemático que se van a permitir en el ambiente `Verbatim`.

■ EJEMPLO 2.18

El código:

```
\begin{Verbatim}[
    xleftmargin=3.1cm,
    xrightmargin=4.5cm,
    resetmargins=true,
    frame=lines,
    fontfamily=ptm,
    commandchars=\\{\},
    codes={\catcode'$_=3\catcode'^=7\catcode'_{=8}]
$x$           $y=x^2$
-2            4
0             0
2             4
0.000005      2.5$\times 10^{-11}$
\end{Verbatim}
```

Produce:

x	$y = x^2$
-2	4
0	0

$$\frac{0.000005^2}{2.5 \times 10^{-11}}^4$$

2.3.9 Espacio horizontal y vertical

Para dejar espacio horizontal se usa el comando `\hspace{Xcm}`. El efecto es abrir espacio o correr horizontalmente texto o tablas o gráficos. X cm o X mm o X in es el corrimiento a la derecha o a la izquierda según sea X positivo o negativo.

■ EJEMPLO 2.19

`\subset \hspace{-3.5mm} / \hspace{-1mm} / $ 30` produce: $\# 30$
 Para abrir espacio verticalmente se usa el comando `\vspace{Xcm}` que funciona de manera análoga a `\hspace` excepto que para que haga efecto debe dejarse antes, un renglón en blanco.

■ EJEMPLO 2.20

<p>El texto:</p> <p><code>\LaTeX...</code></p> <p><code>\vspace{-0.8cm}</code> <code>\hspace{1cm}\LaTeX</code></p> <p><code>\hspace{2cm}\TeX...</code></p> <p><code>\vspace{-0.2cm}</code> <code>\hspace{1cm}\TeX</code></p>	<p>Produce:</p>
--	-----------------

2.4 ENUMERADO AUTOMÁTICO.

Uno de los comandos más usados para hacer listas es `enumerate`. Cada nuevo item 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 2.21

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}

LaTeX.. Walter Mora F., Alex Borbón A.
Derechos Reservados © 2009 Revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr)
```

```

\item {\bf [3 Puntos]} Sea  $A=\{1,b,c,d,7\}$  y  $B=\{1,2,c,d\}$ .
Calcule  $\mathcal{P}(A \Delta B)$ .

\item {\bf [5 Puntos]} Muestre que  $A-(B \cap C)=(A-B) \cup (A-C)$ 

\item {\bf [5 Puntos]} Mostrar que  $[A \cup C \subseteq B \cup C \wedge A \cap C = \emptyset] \implies A \subseteq B$ 

\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 relaci' on 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}

```

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 relaci' on de equivalencia.
 - (b) **[2 Puntos]** Determine las clases de equivalencia $\overline{1}$ y $\overline{-1}$.
 - (c) **[1 Punto]** Determine \mathbb{R}^*/\Re (el conjunto cociente).

■ EJEMPLO 2.22

Ejemplo con 4 niveles, usamos itemize en vez de 'enumerate'

```

\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}
\end{enumerate}
\end{enumerate}
\end{enumerate}

```

```

        \end{itemize}
        \item Paso c.3
    \end{enumerate}
\item Paso d.
\end{enumerate}
\item[\fbox{3.}] {\bf fin}
\end{enumerate} % fin del primero

```

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 con las instrucciones

```

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

```

las cuales, en este caso, indican que en el primer nivel se utilizarán números romanos en mayúscula (para ponerlo en minúscula se pone `\roman{enumi}`), para el segundo nivel números arábigos, en el tercer nivel se usarán letras en minúscula (si se quieren en mayúscula se utiliza `\Alph{enumiii}`) y en el cuarto nivel se utilizarán símbolos.

Cada nivel se puede definir de acuerdo a lo que se necesite, observe que también se pueden agregar paréntesis o puntos para darle el formato. De hecho, para enumerar también se pueden poner símbolos, por ejemplo `\renewcommand{\labelenumii}{\bullet}` pondrá un círculo pequeño relleno en el segundo nivel.

■ EJEMPLO 2.23

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}

```

produce:

```

\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}

```

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

Capítulo 3

TEXTO EN MODO MATEMÁTICO

Recordemos que el texto matemático va entre $\$$ $\$$. Para tener acceso a varios paquetes de fuentes (que no están en el formato estándar) y para usar abreviaciones para algunos símbolos especiales, lo mejor es usar la siguiente

Plantilla básica

```
\documentclass{article}%o report o book

%Paquetes con fuentes adicionales
\usepackage{amsmath,amssymb,amsfonts,latexsym,cancel}
%Comandos especiales
\newcommand{\sen}{\mathop{\rm sen}\nolimits} %seno
\newcommand{\arcsen}{\mathop{\rm arcsen}\nolimits}
\newcommand{\arcsec}{\mathop{\rm arcsec}\nolimits}
\def\max{\mathop{\mbox{\rm m`ax}}}\qquad %mx
\def\min{\mathop{\mbox{\rm m`{i}n}}}\qquad %mn

\begin{document}
...
\end{document}
```

Aquí suponemos que se tiene una versión completa de MiKTeX o, en su defecto, una conexión a Internet para MiKTeX automáticamente agrgue los pquetes que le hagan falta.

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>
$\sin^2(x)$	<code>\sen^2(x)</code>	$x^{\sin(x)+\cos(x)}$	<code>x^{\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.

■ EJEMPLO 3.1

El texto:

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

Produce:

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

3.3 RAÍCES

■ EJEMPLO 3.2

$$\sqrt{x+1} \quad \backslash\text{sqrt}\{x+1\}$$

$$\sqrt[n]{x+\sqrt{x}} \quad \backslash\text{displaystyle}\{ \backslash\text{sqrt}[n]\{x+\backslash\text{sqrt}\{x\}\} \}$$

$$\sqrt[n]{x+\sqrt{x}} \quad \backslash\text{sqrt}[n]\{x+\backslash\text{sqrt}\{x\}\}$$

3.4 FRACCIONES Y “FRACCIONES”

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

■ EJEMPLO 3.3

$$\frac{x+1}{x-1} \quad \{x+1 \backslash\text{over } x-1\}$$

$$\frac{x+1}{x-1} \quad \backslash\text{displaystyle } \backslash\text{frac}\{x+1\}\{x-1\}$$

$$\frac{\frac{x+1}{3}}{x-1} \quad \{\{x+1 \backslash\text{over } 3\} \backslash\text{over } x-1\}$$

$$\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}} \quad \backslash\text{displaystyle}\{\backslash\text{left}(1+ \{1 \backslash\text{over } x\} \backslash\text{right})^{\{n+1 \backslash\text{over } n\}}\}$$

$$\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}} \quad \backslash\text{displaystyle } \backslash\text{left}(1+ \backslash\text{frac}\{1\}\{x\} \backslash\text{right})^{\backslash\text{frac}\{n+1\}\{n\}}$$

$$\left(1 + \frac{1}{x}\right)^{\frac{n+1}{n}} \quad \backslash\text{displaystyle}\{\backslash\text{left}(1+ \{1 \backslash\text{over } x\} \backslash\text{right})\}^{\backslash\text{displaystyle}\{n+1 \backslash\text{over } n\}}$$

$$\frac{x+1}{x-1} \quad \{x+1 \backslash\text{atop } x-1\}$$

$$\frac{x+1}{x-1} \quad \{\text{x+1 \above 2pt x-1}\} \quad (2\text{pt es el grosor})$$

$$\{x+1\}_{x-1} \quad \{\text{x+1 \brace x-1}\}$$

$$[x+1]_{x-1} \quad \{\text{x+1 \brack x-1}\}$$

$$a \xrightarrow{f} b \quad \{\text{displaystyle}\{a \stackrel{f}{\rightarrow} b\}\}$$

$$\lim_{x \rightarrow 0} f(x) \quad \{\text{displaystyle}\{\lim_{x \rightarrow 0} f(x)\}\}$$

$$\binom{a}{b} \quad \{\text{displaystyle}\{a \choose b\}\}$$

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

$$\int_C \mathbf{F} \cdot d\mathbf{r} \quad \{\text{displaystyle}\{\int_C \boldsymbol{F} \cdot d\mathbf{r}\}\}$$

$$\oint_C \mathbf{F} \cdot d\mathbf{r} \quad \{\text{displaystyle}\{\oint_C \boldsymbol{F} \cdot d\mathbf{r}\}\}$$

$$\iint_D f(x,y) dA \quad \{\text{displaystyle}\{\iint_D f(x,y) dA\}\}$$

$$\iiint_Q f(x,y,z) dA \quad \{\text{displaystyle}\{\iiint_Q f(x,y,z) dA\}\}$$

Los subíndices y los superíndices se pueden centrar con el símbolo si usamos la opción `intlimits` en el paquete `amsmath`, es decir, si ponemos en el *preámbulo* `\usepackage[intlimits]{amsmath}`.

$$\iiint_Q f(x,y,z) dA$$

`intlimits` no es opción de otros paquetes, así que en el *preámbulo* que estamos usando haríamos el cambio

```
\usepackage{amssymb,amsfonts,latexsym,cancel}
\usepackage[intlimits]{amsmath}
```

3.5 ELIPSIS (PUNTOS)

Una elipsis es un grupo de tres puntos que indican 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`, por ejemplo

■ EJEMPLO 3.4

```

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

```

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

```

$$\left\{ n \in \mathbb{N} \text{ at } r \neq 1 \right\}$$

```

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

3.7 LLAVES Y BARRAS HORIZONTALES

Barras horizontales.

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

■ EJEMPLO 3.5

El texto

Leyes de DeMorgan:

```

$$\left\{ \overline{A \cup B} = \overline{A} \cap \overline{B} \right.$$

```

```

$$\left. \overline{A \cap B} = \overline{A} \cup \overline{B} \right\}$$

```

produce:

Leyes de DeMorgan: $\left\{ \begin{array}{l} \overline{A \cup B} = \overline{A} \cap \overline{B} \\ \overline{A \cap B} = \overline{A} \cup \overline{B} \end{array} \right.$

LLaves y llaves horizontales

Las llaves se ponen con `\{+ y \verb+\}`.

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`, `\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 3.6

El texto

```
\[ \overbrace{(x_i-1)}^{K_i}f(x)+\underbrace{(x_i-1)}_{K_i}g(x)=K_i(f(x)+g(x)) \]
```

produce:

$$\overbrace{(x_i-1)}^{K_i}f(x) + \underbrace{(x_i-1)}_{K_i}g(x) = K_i(f(x) + g(x))$$

3.8 ACENTOS Y “SOMBREROS” EN MODO MATEMÁTICO

\hat{i} `\hat{\imath}` \acute{a} `\acute{a}`

\bar{p} `\bar{p}` \vec{p} `\vec{p}`

3.9 NEGRITAS EN MODO MATEMÁTICO

En una fórmula matemática, el comando `\boldmath` solo pone en negrita 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 3.7

$$\cos(x+2\pi) = \cos x \quad \text{\code{\pmb{\cos(x+2\pi)=\cos x}}}$$

$$\cos(x+2\pi) = \cos x \quad \text{\code{\cos(x+\pmb{2\pi})=\cos x}}$$

3.10 ESPACIO EN MODO MATEMÁTICO

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

■ EJEMPLO 3.8

$$n \in \mathbb{N}, x \in \mathbb{R} \quad \text{\code{\n \in \N, x \in \R}} \quad \text{Mejor: } n \in \mathbb{N}, x \in \mathbb{R} \quad \text{\code{n \in \N, \; \; ; x \in \R}}$$

$$\int f(x) dx \quad \text{\code{\int f(x) dx}} \quad \text{Mejor: } \int f(x) dx \quad \text{\code{\int f(x)\, dx}}$$

3.11 CENTRADO

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

■ EJEMPLO 3.9

El texto

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

produce:

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

3.12 CONTADORES AUTOMÁTICOS

L^AT_EX puede llevar un conteo automático de capítulos, secciones, etc. Podemos llevar también un conteo automático de teoremas, ecuaciones, etc.

Marcamos cada objeto que queremos contar. Por ejemplo, con `\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 3.10

```
%ecuaci'on 3.1 (cap 3)
\begin{equation}
\log_{2}(xy)=\log_{2}x + \log_{2}y
\end{equation}
```

$$\log_2(xy) = \log_2 x + \log_2 y \quad (3.1)$$

$$\log_2(a^b) = b \log_2 a \quad (3.4)$$

```
%ecuaci'on 4
\setcounter{equation}{3}
\begin{equation}
\log_{2}(a^b)=b\log_{2}a
\end{equation}
```

Una vez establecido el contador, se puede usar el ambiente `subequations` para lograr una “subnumeración”:

■ EJEMPLO 3.11

```
\begin{subequations}
\begin{equation}
\log_{2}(xy)=\log_{2}x + \log_{2}y
\end{equation}
\begin{equation}
\log_{2}(a^b)=b\log_{2}a
\end{equation}
\end{subequations}
```

$$\log_2(xy) = \log_2 x + \log_2 y \quad (3.5a)$$

$$\log_2(a^b) = b \log_2 a \quad (3.5b)$$

3.13 MATRICES

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: `& & \\\`
- ‘\\’ indica el cambio de fila
- Final del “array”: `\end{array}`

■ EJEMPLO 3.12

LaTeX. Walter Mora F., Alex Borbón A.

Derechos Reservados © 2009 Revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr)

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:

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

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

■ EJEMPLO 3.13

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:

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

■ EJEMPLO 3.14

El texto

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

produce:

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

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

■ EJEMPLO 3.15

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

El código

```

$$
\left\{
\begin{array}{lclcl}
\sen x &=& -1 & \Longrightarrow & x=(4k+3)\frac{\pi}{2}+2k\pi, \; k \in \mathbb{Z} \\
&&& & \\
\cos(2x) &=& \frac{1}{2} & \Longrightarrow & \left\{ \begin{array}{l} x = \frac{\pi}{6} + k\pi, \; z \in \mathbb{Z} \\ x = -\frac{\pi}{6} + k\pi, \; z \in \mathbb{Z} \end{array} \right. \\
&&& & \\
\end{array}
\right.
$$

```

produce

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

3.14 ALINEAMIENTO

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 (antes de `\`).

■ EJEMPLO 3.16

El texto

```

% Sin numeraci'on >>
\begin{eqnarray*}
y=\sqrt[n]{x} & \Longrightarrow & y^n=x \\
\end{eqnarray*}

```

```

& \Longrightarrow & n\log \,y=\log \,x, \; \mbox{si}\; x>0,\; y>0\\
& \Longrightarrow & \log \sqrt[n]{x}={1 \over n}\log \,x
\end{eqnarray*}

% Con numeraci'on >>
\begin{eqnarray}
y=\sqrt[n]{x} & \Longrightarrow & y^n=x \\
& \Longrightarrow & n\log \,y=\log \,x, \; \mbox{si}\; x>0,\; y>0\\
& \Longrightarrow & \log \sqrt[n]{x}={1 \over n}\log \,x
\end{eqnarray}

% Numeraci'on selectiva >>
\begin{eqnarray}
y=\sqrt[n]{x} & \Longrightarrow & y^n = x \nonumber \\
& \Longrightarrow & n\log \,y= \log \,x, \; \mbox{si}\; x>0,\; y>0 \\
& \Longrightarrow & \log \sqrt[n]{x}={1 \over n}\log \,x
\end{eqnarray}

```

produce

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

$$y = \sqrt[n]{x} \implies y^n = x \tag{3.6}$$

$$\implies n \log y = \log x, \text{ si } x > 0, y > 0 \tag{3.7}$$

$$\implies \log \sqrt[n]{x} = \frac{1}{n} \log x \tag{3.8}$$

$$\begin{aligned}
 y = \sqrt[n]{x} &\implies y^n = x \\
 &\implies n \log y = \log x, \text{ si } x > 0, y > 0
 \end{aligned}
 \tag{3.9}$$

$$\implies \log \sqrt[n]{x} = \frac{1}{n} \log x \tag{3.10}$$

3.15 TABLAS DE SÍMBOLOS MATEMÁTICOS FRECUENTES

3.15.1 Letras griegas

α	<code>\alpha</code>	ε	<code>\varepsilon</code>	ι	<code>\iota</code>	ξ	<code>\xi</code>
β	<code>\beta</code>	ζ	<code>\zeta</code>	κ	<code>\kappa</code>	o	<code>o</code>
γ	<code>\gamma</code>	η	<code>\eta</code>	λ	<code>\lambda</code>	π	<code>\pi</code>
δ	<code>\delta</code>	θ	<code>\theta</code>	μ	<code>\mu</code>	ϖ	<code>\varpi</code>
ε	<code>\varepsilon</code>	ϑ	<code>\vartheta</code>	ν	<code>\nu</code>	ρ	<code>\rho</code>

ρ	<code>\varrho</code>	ϕ	<code>\phi</code>	Δ	<code>\Delta</code>	Υ	<code>\Upsilon</code>
σ	<code>\sigma</code>	φ	<code>\varphi</code>	Θ	<code>\Theta</code>	Φ	<code>\Phi</code>
ς	<code>\varsigma</code>	χ	<code>\chi</code>	Λ	<code>\Lambda</code>	Ψ	<code>\Psi</code>
τ	<code>\tau</code>	ψ	<code>\psi</code>	Ξ	<code>\Xi</code>	Ω	<code>\Omega</code>
υ	<code>\upsilon</code>	ω	<code>\omega</code>	Π	<code>\Pi</code>		
		Γ	<code>\Gamma</code>	Σ	<code>\Sigma</code>		

3.15.2 Operadores binarios

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

3.15.3 Relaciones

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

3.15.4 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.

$\not<$	<code>\not<</code>	$\not\geq$	<code>\not\geq</code>
$\not\leq$	<code>\not\leq</code>	$\not\succ$	<code>\not\succ</code>
$\not\prec$	<code>\not\prec</code>	$\not\succeq$	<code>\not\succeq</code>
$\not\preceq$	<code>\not\preceq</code>	$\not\supset$	<code>\not\supset</code>
$\not\subset$	<code>\not\subset</code>	$\not\supseteq$	<code>\not\supseteq</code>
$\not\subseteq$	<code>\not\subseteq</code>	$\not\sqsupseteq$	<code>\not\sqsupseteq</code>
$\not\sqsubseteq$	<code>\not\sqsubseteq</code>	\neq	<code>\neq</code>
$\not\supseteq$	<code>\not\supseteq</code>	$\not\equiv$	<code>\not\equiv</code>
$\not\sqsupseteq$	<code>\not\sqsupseteq</code>		
$\not=$	<code>\not=</code>		
$\not>$	<code>\not></code>		

$\not\sim$	<code>\not\sim</code>	$\not\approx$	<code>\not\approx</code>
$\not\cong$	<code>\not\cong</code>	$\not\equiv$	<code>\not\equiv</code>

También puede usar el comando `\cancel` del paquete `cancel` (definido en el preámbulo propuesto al inicio del capítulo)

■ EJEMPLO 3.17

El código:

```
\displaystyle{ \frac{\cancel{ab}c}{\cancel{ab}d}=\frac{c}{d} }
```

produce:

$$\frac{\cancel{ab}c}{\cancel{ab}d} = \frac{c}{d}$$

3.15.5 Flechas

\leftarrow	<code>\leftarrow</code>	\longleftarrow	<code>\longleftarrow</code>
\Lleftarrow	<code>\Lleftarrow</code>	\Longleftarrow	<code>\Longleftarrow</code>
\rightarrow	<code>\rightarrow</code>	\longrightarrow	<code>\longrightarrow</code>
\Rightarrow	<code>\Rightarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>
\hookleftarrow	<code>\hookleftarrow</code>	\rightharpoonup	<code>\rightharpoonup</code>
\leftharpoonup	<code>\leftharpoonup</code>	\rightharpoondown	<code>\rightharpoondown</code>
\leftharpoondown	<code>\leftharpoondown</code>	\Uparrow	<code>\Uparrow</code>
\rightleftharpoons	<code>\rightleftharpoons</code>	\Uparrow	<code>\Uparrow</code>
\Rightarrow	<code>\Rightarrow</code>	\Downarrow	<code>\Downarrow</code>
\longleftrightarrow	<code>\longleftrightarrow</code>	\Downarrow	<code>\Downarrow</code>
\Llongleftrightarrow	<code>\Llongleftrightarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\mapsto	<code>\mapsto</code>	\nearrow	<code>\nearrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\searrow	<code>\searrow</code>
\Lleftrightarrow	<code>\Lleftrightarrow</code>	\swarrow	<code>\swarrow</code>
\mapsto	<code>\mapsto</code>	\nwarrow	<code>\nwarrow</code>

3.15.6 Operadores grandes

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

■ EJEMPLO 3.18

El código:

```
$$L_{\{n,k\}}(x)
```

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

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 $\overset{i=0}{i \neq k}$ para producir: $i \neq k$

3.15.7 Otros símbolos

\aleph	<code>\aleph</code>	∂	<code>\partial</code>	\parallel	<code>\parallel</code>	\natural	<code>\natural</code>
\hbar	<code>\hbar</code>	∞	<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>
ℓ	<code>\ell</code>	∇	<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>	\perp	<code>\perp</code>	\flat	<code>\flat</code>		

3.15.8 Especiales

\tilde{A}	<code>\widetilde{A}</code>	\vec{v}	<code>\overrightarrow{v}</code>
\hat{A}	<code>\widehat{A}</code>		

3.15.9 Símbolos del paquete amssymb

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

\mathbb{R}	<code>\mathbb{R}</code>	\triangleright	<code>\gtrdot</code>	\lesseqgtr	<code>\lesseqgtr</code>	\supseteq	<code>\Supset</code>
\mathbb{Q}	<code>\mathbb{Q}</code>	\triangleleft	<code>\gtrless</code>	\doteqdot	<code>\doteqdot</code>	\sqsubset	<code>\sqsubset</code>
\mathbb{Z}	<code>\mathbb{Z}</code>	\trianglelefteq	<code>\eqslantless</code>	\risingdotseq	<code>\risingdotseq</code>	\sqsupset	<code>\sqsupset</code>
\mathbb{I}	<code>\mathbb{I}</code>	\lesssim	<code>\lesssim</code>	\fallingdotseq	<code>\fallingdotseq</code>	\succcurlyeq	<code>\succcurlyeq</code>
\therefore	<code>\therefore</code>	\lessapprox	<code>\lessapprox</code>	\circeq	<code>\circeq</code>	\preccurlyeq	<code>\preccurlyeq</code>
\because	<code>\because</code>	\approx	<code>\approx</code>	\triangleq	<code>\triangleq</code>	\curlyeqprec	<code>\curlyeqprec</code>
\leqq	<code>\leqq</code>	\lessdot	<code>\lessdot</code>	\thicksim	<code>\thicksim</code>	\curlyeqsucc	<code>\curlyeqsucc</code>
\geqq	<code>\geqq</code>	\gtreqless	<code>\gtreqless</code>	\thickapprox	<code>\thickapprox</code>	\precsim	<code>\precsim</code>
\leqslant	<code>\leqslant</code>	\gtreqqless	<code>\gtreqqless</code>	\backsim	<code>\backsim</code>	\succsim	<code>\succsim</code>
\geqslant	<code>\geqslant</code>	\lll	<code>\lll</code>	\backsimeq	<code>\backsimeq</code>	\precapprox	<code>\precapprox</code>
\leqslantgtr	<code>\leqslantgtr</code>	\ggg	<code>\ggg</code>	\subseteq	<code>\subseteq</code>	\succapprox	<code>\succapprox</code>
\gtrsim	<code>\gtrsim</code>	\lessgtr	<code>\lessgtr</code>	\supseteq	<code>\supseteq</code>	\vartriangleright	<code>\vartriangleright</code>
\gtrapprox	<code>\gtrapprox</code>	\lesseqgtr	<code>\lesseqgtr</code>	\Subset	<code>\Subset</code>	\trianglerighteq	<code>\trianglerighteq</code>

Capítulo 4

OBJETOS Y GRÁFICOS

4.1 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 poner líneas verticales se ponen marcas como `|` o `||` en la parte que corresponde al alineamiento de columnas.
- Para poner líneas horizontales, al final de cada fila se especifica

i. `\hline`: línea tan larga como la tabla

ii. `\cline{i-j}`: línea de columna i a columna j

■ EJEMPLO 4.1

El texto

produce:

```

\begin{tabular}{|c|c|c|} \hline


$p$ & $q$ & $p \rightarrow q$

 \\ \hline
0 & 0 & 1 \\
0 & 1 & 1 \\
1 & 0 & 0 \\
1 & 1 & 1 \\
\end{tabular}

```

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

Fuente en tabular.

Tópico
Avanzado

En la tabla del ejemplo (4.2) los números están en modo texto. En vez de ponerlos en modo matemático uno por uno, podríamos solamente cambiar la fuente a la familia `cmm`.

■ EJEMPLO 4.2

El texto

produce:

```

{\fontencoding{OML}\fontfamily{cmm}
 \selectfont{
 \begin{tabular}{|c|c|}
 $x$ & $x^2+3$ \\ \hline
 2 & 7 \\
 4 & 19 \\
 \end{tabular}
 }%fontfamily

```

x	$x^2 + 3$
2	7
4	19

Color en tablas.

Para poner color en las filas o las columnas de una tabla debemos usar el paquete `xcolor`: ponemos `\usepackage[table]{xcolor}` en el *preámbulo*.

Usamos el comando `\rowcolors {nfile}{color fila-impar}{color fila-par}`

`nfile` es el número de fila de la primera fila en ser coloreada. Los colores de fila par e impar se pueden dejar en blanco (no se pondrá color en esa fila).

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

Nota: En el capítulo sobre transparencias con Beamer, se requiere poner `\documentclass[xcolor=table...]{beamer}` sino se tiene un choque (“clash”) de paquetes.

■ EJEMPLO 4.3

En este ejemplo se colorean con un 20% gris (gray!30) las filas impares y se dejan con fondo blanco las pares.

El código:

```
\begin{table}[h]
\centering
{\fontfamily{ptm}\selectfont{
\rowcolors{1}{gray!30}{}}
\begin{tabular}{ll}
$x_{n+1}$ & $|x_{n+1}-x_n|$\ \hline
1.20499955540054 & 0.295000445\ \hline
1.17678931926590 & 0.028210236\ \hline
1.17650196994274 & 0.000287349\ \hline
1.17650193990183 & 3.004$\times 10^{-8}$\ \hline
1.17650193990183 & 4.440$\times 10^{-16}$\ \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
1.17650196994274	0.000287349
1.17650193990183	3.004×10^{-8}
1.17650193990183	4.440×10^{-16}

Tabla 4.1 Iteración de Newton para $x^2 - \cos(x) - 1 = 0$ con $x_0 = 1.5$.

■ EJEMPLO 4.4

En este ejemplo se colorean con un 20% gris una sola celda: Simplemente agregamos `\cellcolor[gray]{0.80}` en la celda que queremos.

El código:

```
\begin{table}[h]
\centering
{\fontfamily{ptm}\selectfont{
\begin{tabular}{ll}
$x_{n+1}$ & $|x_{n+1}-x_n|$\ \hline
\end{tabular}
}}
```

```
\cellcolor[gray]{0.80} 1.17 & 3.$\times 10^{-8}$\\
                        1.17 & 4.$\times 10^{-16}$\\ \hline
\end{tabular}
}}%font
\end{table}
```

Produce:

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

Rotación de texto en celdas.

A veces es conveniente rotar una tabla completa o simplemente el texto en las celdas. Esto se hace con el ambiente `\begin{sideways} . . . \end{sideways}` aplicado directamente a la tabla o a la(s) celda(s). Necesita el paquete `\usepackage{rotating}`

■ EJEMPLO 4.5

El código:

```
\begin{table}[h]
\centering
\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 & & 4.$\times 10^{-16}$ \\ \hline
\end{tabular}
\end{sideways}
\end{table}
```

Produce:

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

■ EJEMPLO 4.6

El código:

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

Produce:

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

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}{Estimaci'on del error} \\
& \multicolumn{2}{l}{absoluto y relativo} \\
$x_n$ & $x_{n+1}$ & $|x_{n+1}-x_n|/|x_{n+1}|$ \\ \hline
-3.090721649 & 2.990721649 & 1.6717 \\
-2.026511552 & 1.064210097 & 0.525143859 \\ \hline
\end{tabular}
\end{table}
```

```
-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 4.2

Espaciado en celdas.

A veces el texto matemático queda muy pegado a las paredes de las celdas: Necesitamos hacer un poco de espacio hacia arriba, hacia abajo o variar el ancho de la celda. Esto lo hacemos con el comando `\vrule`. La altura (`height`), la profundidad (`depth`) y el ancho (`width`) lo manejamos en unidades `pt` (1 pulgada=72.27 pt)

■ EJEMPLO 4.8

En este ejemplo tenemos una tabla problemática: el texto matemático está muy ajustado. La solución es crear espacio en la celda con la instrucción

```
@{\vrule height xpt depth ypt width zpt}
```

aplicado a una columna.

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

$$\begin{array}{c} \hline x \quad \sqrt{x} \quad x^{2^n} \\ \hline \end{array}$$

Como se ve, necesitamos hacer espacio arriba y en la parte de abajo de la celda.

```
\begin{tabular}{l l l@{\vrule height 11pt depth 5pt width 0pt}}\hline
$x$ & $\sqrt{x}$ & $x^{2^n}$\\ \hline
\end{tabular}
```

$$\begin{array}{c} \hline x \quad \sqrt{x} \quad x^{2^n} \\ \hline \end{array}$$

4.2 INSERCIÓN DE GRÁFICOS.

Para incluir objetos gráficos en su documento LaTeX, se debe poner en el *preámbulo*

```
\usepackage[dvips]{graphicx}
\DeclareGraphicsRule{.emf}{bmp}{}{}
\DeclareGraphicsExtensions{.pdf, .png, .jpg}
```

Un gráfico nombre.ext, que está en una subcarpeta “imagenes”, se incluye en el documento escribiendo

```
\includegraphics{imagenes/nombre.ext}
```

Usualmente es conveniente poner `\includegraphics` en un ambiente `figure`

```
\begin{figure}[h]
\includegraphics{imagenes/nombre.ext}
\caption{comentario al pie de la figura}
\label{texto para referencia}
\end{figure}
```

Nota: Si usamos el paquete `float` (ver sección 4.4, más adelante), podemos poner `\begin{figure}[H]` para que el gráfico se despliegue exactamente donde es llamado.

Detalles.

Un gráfico es un objeto flotante. \LaTeX soporta varios formatos de gráficos: *.wmf (Metafile), *.bmp (bimap), *.ps (PostScript), *.eps (PostScript encapsulado), etc.

En general tratamos de evitar el formato *.bmp pues es un formato no muy fino (y pesado) Metafile es un formato adecuado y “eps” deseable.

Si queremos generar un archivo pdf (compilando con PDFLaTeX), podemos usar los formatos *.gif, *.jpeg o *.png.

Si compilamos con LaTeX, lo deseable es tener los gráficos en formato “eps”. Hay muchas maneras de convertir cualquier tipo de gráficos “eps”. Lo mejor sería un convertidor general. Pero tenemos otras opciones, usando software conocido.

Los formatos *.tiff (de DeRiVe, por ejemplo) o *.gif, pueden ser leídos y convertidos a Metafile en PowerPoint por ejemplo. Adobe Photoshop (*.psd) hace copias en formato *.tiff

Para obtener gráficos en formato “eps”, puede usar *Mathematica* (www.wolfram.com). Los puede editar de manera adicional con, por ejemplo, Adobe illustrator.

Para insertar un gráfico, éste debe estar en el directorio del documento que se está editando, en otro caso se debe especificar la dirección donde se encuentra.

Lo primero que se debe hacer es poner en el *preámbulo* las instrucciones que habilitan el uso de los comandos de inserción y manejo de gráficos. Una de sus variantes es

```
\usepackage[dvips]{graphicx}
```

Si se quieren incluir gráficos en formato “emf” y “wmf” hay que poner además:

```
\DeclareGraphicsRule{.emf}{bmp}{}{}
```

Si se quieren incluir gráficos en formato “pdf”, “png” o “gif” hay que poner además:

```
\DeclareGraphicsExtensions{.pdf,.png,.jpg}
```

Para incluir gráficos en formato “bmp” o “eps” no hay que poner nada extra.

Debemos usar `\DeclareGraphicsExtensions{.pdf,.png,.jpg}` si vamos a usar estos formatos en el documento siempre y cuando compilemos con PDFLaTeX para generar un “pdf”.

Para incluir un gráfico usamos el comando `\includegraphics[scale=x]{dir/nombre-graf}`
Por ejemplo,

Si compila con LaTeX

```
\includegraphics[scale=0.5]{images/ML_fig30.eps}
```

Si compila con PDFLaTeX

```
\includegraphics[scale=0.5]{images/logo.png}
```

El comando `\includegraphics` admite atributos opcionales para escalar el gráfico, hacer que el ancho del gráfico sea igual al ancho del texto o para rotarlo.

En el ejemplo que sigue vamos a incluir un gráfico en el ambiente `minipage`

■ EJEMPLO 4.9

El código:

```
\begin{minipage}[t]{8cm}
\includegraphics{images/ML_fig8.eps}
\end{minipage}
\ \ \hfill
\begin{minipage}[t]{8cm}
\includegraphics[angle=45]{images/ML_fig8.eps}
```

```
\end{minipage}
```

rota el gráfico 45 grados en contra de las manecillas del reloj



Figura 4.1 Rotación de 45 grados

■ EJEMPLO 4.10

El código:

```
\centering
\includegraphics{images/ML_fig8.eps}

\includegraphics[scale=0.5]{images/ML_fig8.eps}
```

escala el gráfico el ancho usado por el texto en el primer caso y a la mitad de sus dimensiones en el segundo.



Figura 4.2 Escalamiento

Nota: Esta rotación y escalamiento funciona bien en gráficos con formato postscript (.ps) y postscript encapsulado (.eps) pero puede presentar problemas con los demás formatos.

4.3 OTROS FORMATOS GRÁFICOS

En el documento, cuando se quiera incluir un gráfico '*.bmp' hay que poner:

```
\includegraphics[0cm,0cm][xcm,ycm]{ejemplo1.bmp}
```

Si se desea incluir un gráfico '*.wmf' se pone

```
\includegraphics[0cm,0cm][xcm,ycm]{ejemplo2.wmf}
```

En donde el primer argumento opcional (los dos primeros valores) indica el espacio que se deja a la izquierda de la imagen y arriba, el segundo argumento opcional indica el ancho y el alto de la imagen (en postscript no hacía falta poner estos argumentos porque el gráfico internamente tiene estos datos).

Nota 1: es recomendable poner el gráfico en algún ambiente flotante tal como `figure`, `floatflt` o `wrapfigure`.

Nota 2: la impresión de un gráfico es más fina si está en formato PostScript o PostScript encapsulado (como los gráficos de MATHEMATICA), estos son los formatos recomendados para publicaciones profesionales. Estos gráficos no se ven en la presentación preliminar del archivo DVI, excepto si se tiene instalado el programa 'GSview'.

Nota 3: un gráfico se puede escanear y/o **recortar** digamos en PAINT, IMAGE COMPOSER o FIREWORKS u otra aplicación que permita estas operaciones. Se recorta no solo para seleccionar la figura, también para controlar el tamaño del gráfico. En PAINT el gráfico se recorta y se guarda usando Edit-Copiar A.

Con IMAGE COMPOSER o con MACROMEDIA FIREWORKS se puede recortar una figura con formas caprichosas además se puede agregar efectos. La figura se guarda con Save Selection As. En MATHEMATICA, los gráficos se seleccionan (se hace click sobre ellos) y se guardan con File o Edit - Save Selection As - EPS.

Los gráficos de DeRive se salvan con Crtl-F9. Estos quedan en el directorio actual de DeRive, sus nombres son, si se usa el default, derive.tiff, ...,derive3.tiff,... etc. Se pueden *insertar* en una presentación en blanco de PowerPoint, y se guarda con Save-As en formato WMF.

Si compila usando PDFLaTeX no habrá problema con los gráficos en formato *.gif, *.jpeg, *.png: Agregamos (como vimos antes) en el *preámbulo*

```
\DeclareGraphicsExtensions{.pdf,.png,.jpg}
```

■ EJEMPLO 4.11

```
\begin{figure}[h]
\begin{minipage}[b]{0.5\linewidth}
{\bf 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  $f(x)=6-(x-2)^3+x$ ,
y si  $a=2$  y  $b=4$  ;  $\Rightarrow \xi = \frac{2}{3}(3+\sqrt{3})$  .
\end{minipage}
\ \ \ \ \ \hfill \begin{minipage}[b]{0.45\linewidth}
\vspace{-3cm}
```

```

\includegraphics[scale=0.7]{images/ML_fig10.eps}
\caption{\small Teorema del valor medio}
\end{minipage}
\end{figure}

```

produce:

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

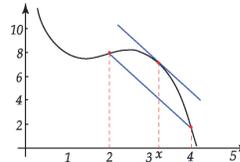


Figura 4.3 Teorema del valor medio

4.4 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. \LaTeX resuelve (y a veces complica) este problema manipulando las figuras como objetos flotantes en el documento.

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

```

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

```

- `[h]` le indica a \LaTeX que queremos la figura o la tabla, exactamente en ese lugar (`h=here`, esto no es tan exacto, ya que \LaTeX en realidad lo acomoda lo más cerca posible de ese lugar). Otras opciones son `[t]=top`, `[b]=bottom`, `[htb]=here o top o bottom,...`. Si no se pone algo, el default es `[htbp]`
- `\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 4.12

El texto

```
\begin{table}[h]
\bc
\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{
{\small Tabla de verdad para $p \rightarrow q$}
}
\ec
\end{table}
```

produce

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

Tabla 4.3 Tabla de verdad para $p \rightarrow q$

Nota: los ambientes figure y table se usan fuera de los ambientes minipage.

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: Ponemos en el *preámbulo* `\usepackage{float}`.

Ahora, en vez de digitar `\begin{table}[h]` o `\begin{table}[h]`, digitamos `\begin{table}[H]` o `\begin{table}[H]` (con H): El gráfico o la tabla quedará donde está.



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 4.13

El código:

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

produce:

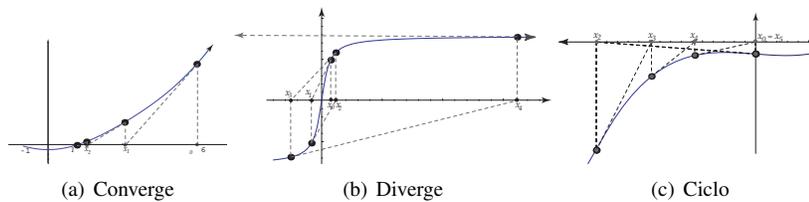


Figura 4.4 Iteración de Newton

4.5 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} %Inclusin de grficos al lado de texto
\usepackage[rflt]{floatflt} %Para meter figuras flotantes entre el texto
```

4.5.1 wrapfigure

El ambiente `wrapfigure` permite incluir gráficos o texto en un recuadro al lado del documento, \LaTeX 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.

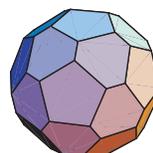
■ EJEMPLO 4.14

El código

```
\begin{wrapfigure}{r}{2.5cm}
\includegraphics{images/ML_fig11.eps}
\end{wrapfigure}
Al incluir...
```

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.

4.5.2 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}
```

LaTeX. Walter Mora F., Alex Borbón A.

Derechos Reservados © 2009 Revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr)

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.

■ EJEMPLO 4.15

El código:

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

Este ambiente...

Genera:

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 \LaTeX 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 `floatflt` 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.

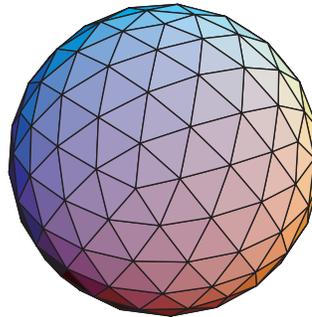


Figura 4.5 Un poliedro

Capítulo 5

FORMATO DEL DOCUMENTO

5.1 TÍTULO, ÍNDICE, SECCIONES,...,BIBLIOGRAFÍA

La primera versión de este documento era de clase report.

PLANTILLA PARA UN DOCUMENTO DE CLASE report

```
\documentclass{report}
\textheight=20cm
\textwidth=18cm
\topmargin=-2cm
\oddsidemargin=-1cm
\parindent=0mm
\usepackage[dvips]{graphicx}
\DeclareGraphicsRule{.wmf}{bmp}{}{}
\usepackage{latexsym,amsmath,amssymb,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{ejem}{\it Ejemplo}[chapter]
\newtheorem{defi}{Definición}[chapter]
```

```

\begin{document}
\title{\LaTeX\
      {\small \sc Instituto Tecnol\`ogico de Costa Rica}\
      {\small \sc Escuela de Matem\`atica}\
      {\small \sc Ense\~anza de la Matem\`atica Asistida por
              Computadora}
      }

\author{ {\sl Preparado por Prof. Walter Mora F. y Alexander Borb\`on A. }}
\date{2005}

\maketitle %despliega el t\`{\i}tulo

\tableofcontents

\chapter{\LaTeX}

\section{?'Qu\`e es \LaTeX ?}
...
\subsection{Pre\`ambulo}
...
\subsubsection{T\`{\i}tulo}
...

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

```

Bibliografía

Como se ve en la plantilla anterior, `\begin{thebibliography}{99}... \end{thebibliography}` es el ambiente para la bibliografía. Es un ambiente parecido al de `enumerate`.

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

```

\clearpage
\addcontentsline{toc}{chapter}{Bibliogr\`{\i}a}
\begin{thebibliography}{99}
...

```

Referencias a la Bibliografía

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 eveyone’’.
    Prentice Hall, New Jersey, 1993.
\end{thebibliography}
```

Entonces podemos hacer referencia a este libro en el texto, por ejemplo:

En `\cite{Hahn}` se pueden ver los aspectos relativos al ‘‘Picture Environment’’.

Esto produce:

En [1] se pueden ver los aspectos relativos al ‘‘Picture Environment’’.

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 definici\’on (`\ref{definicion1}`)...

Lo mismo podemos hacer en los ambientes `table`, `figure`, `eqnarray`, `equation`, etc.

5.2 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.

5.3 ABREVIANDO COMANDOS. COMANDO CON OPCIONES.

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

- `\newcommand{\nombre}{\comando original}`
- `\newcommand{\nombre}[n]{\f{#1}... \h{#n}}`. *n* es el número de parámetros.

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}`. El archivo se invoca, en el preámbulo, por ejemplo como `\input miscom.tex`. Este archivo puede estar en el directorio de trabajo preferiblemente.

Vamos a ver algunos ejemplos de abreviaciones

■ EJEMPLO 5.1

```

\newcommand{\bc}{\begin{center}}
\newcommand{\ec}{\end{center}}
\newcommand{\ds}[a]{\displaystyle{#1}}
\newcommand{\sii}{\Leftrightarrow}
\newcommand{\imp}{\Rightarrow}
\newcommand{\suma}{\ds{\sum_{k=1}^N u_k}} %usamos \ds{}

```

Ahora podemos escribir, por ejemplo:

Si $S_n = \sum_{k=1}^n u_k \Rightarrow S_{N+1} = S_N + u_{N+1}$

para producir

$$S_n = \sum_{k=1}^n u_k \Rightarrow S_{N+1} = S_N + u_{N+1}$$

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

■ EJEMPLO 5.2

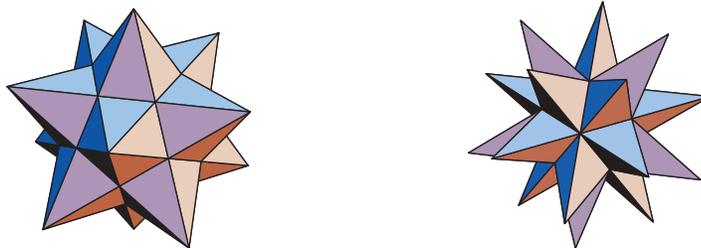
```
\newcommand{\mpage}[2]{\begin{minipage}[t]{0.5 \textwidth}
#3
\end{minipage}
\ \ \hfill \begin{minipage}[t]{0.5 \textwidth}
#4
\end{minipage}}
```

```
\newcommand{\limite}[2]{\lim_{#1} \rightarrow #2} }
```

Así, el texto:

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

produce:



El texto:

```
$$\limite{n}{\infty} \arctan (n) $$
```

produce:

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

Otros ejemplos son

```
\def\max{\mathop{\mbox{\rm m}'ax}}           %\max con acento
\def\min{\mathop{\mbox{\rm m}'{i}n}}         %\min con acento
\def\ngrr#1{\hbox{\boldmath$#1$\unboldmath}} %\ngrr{} negrita
```

Tópico
Avanzado

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 abreviar 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 5.3

El comando

```
\newcommand*\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. El parámetro #2 permite cambiar elegir entre x , u , etc.

El código:

produce:

```

                                 $x_1, \dots, x_n$ 
                                 $y_0, \dots, y_n$ 
\coord{x}$ \ \ %--->(x1, . . . , xn)
\coord[0]{y}$ \ \ %--->(y0, . . . , ym)
\coord{z}[m]$ \ \ %--->(z1, . . . , zm)
\coord[0]{t}[m]$ \ \ %--->(t0, . . . , tm)
```

■ EJEMPLO 5.4

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

```
\newcommand*\mpage[4][1=0.45, 2=0.45]{
\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{...}{...}`

5.4 NUMERACIÓN AUTOMÁTICA DE DEFINICIONES, TEOREMAS, EJEMPLOS...

Para que \LaTeX numere automáticamente definiciones, teoremas, axiomas, ejemplos, etc.; se pone en el preámbulo (por comodidad) el comando

```
\newtheorem{abreviacion}{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 Definicin}[chapter]
```

■ EJEMPLO 5.5

El Texto:

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

produce:

Definición 5.1 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

```
{\em
\begin{defi}
. . .
\end{defi}
}
```

5.5 EL PAQUETE THEOREM

Este paquete es una extensión del enumerado automático de teoremas que ofrece \LaTeX 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{theorem}
```

En el preámbulo se pueden escribir los comandos

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

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

el primero 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:

- plain: Este es idéntico al estilo por defecto de \LaTeX .
- break: El título es seguido por un paso de línea.
- margin: El número del teorema se coloca antes del título.
- marginbreak: Igual a 'margin' pero el título va seguido por un paso de línea.

Para cambiar el tipo del 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.

Para este folleto se utilizaron los siguientes comandos:

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

■ EJEMPLO 5.6

El texto:

```
\begin{defi}[Polinomio caracter'\i{stico}]

Si  $A$  es una matriz cuadrada de orden  $n$ , al polinomio
 $p$  definido por  $p(x)=|A-\lambda \cdot I_n|$ , se le
llama {\bf polinomio caracter'\i{stico}} de  $A$  y la ecuaci'on
 $p(x)=0$  se llama ecuaci'on caracter'\i{stica} de  $A$ .
\end{defi}
```

Produce:

Definición 5.2 (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 .

Tópico
adicional

5.6 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{ condici'on }{ c'odigo }
\eIf{ condici'on }{ c'odigo}{ else ... c'odigo }
\For{ condici'on }{ c'odigo }
\While{ condici'on }{ c'odigo }
\Repeat{ condici'on ("Until") }{ c'odigo }
```

»; se usa para el cambio de línea.

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

■ EJEMPLO 5.7

El código:

```
\begin{algorithm}[h]
\caption{M'aximo com'un divisor}\label{CER}
\SetLine
```

```

\KwData{\mt{a,\, b \es \N.}}
\KwResult{MCD$(a,b)$}
\linesnumbered
\SetVline
\mt{c = |a|,\; d = |b|}\;
\While{\mt{d\neq 0}}{
    $r = $ rem$(c,d)$\;
    $c = d$\;
    $d = r$\;
}
\Return MCD$(a,b)=|c|$\;
\end{algorithm}

```

produce:

Algoritmo 5.1: Máximo común divisor

Entrada: $a, b \in \mathbb{N}$.

Resultado: $\text{MCD}(a, b)$

```

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

```

El código:

```

\begin{algorithm}[h]
\caption{Inverso Multiplicativo mod $m$.}\label{CER}
\SetLine
\KwData{$a\es \Z_m$}
\KwResult{$a^{-1}\mbox{mod}\;m, $ si existe.}
\linesnumbered
\SetVline
Calcular  $x, t$  tal que  $xa+tm=\mbox{\rm MCD}(a,m)$ \;
\eIf{ $\mbox{\rm MCD}(a,m)>1$ }{ $a^{-1}\; \mbox{mod}\; m$  no existe}{
\Return  $\mbox{rem}\;(x,m).$ 
}
\end{algorithm}

```

produce:

Algoritmo 5.2: Inverso Multiplicativo mod m .

Entrada: $a \in \mathbb{Z}_m$

Resultado: $a^{-1} \bmod m$, 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   | retorna  $\text{rem}(x, m)$ .

```

El código:

```

\begin{algorithm}[h]
\caption{Criba de Erat'ostenes}\label{CER}
\SetLine
\KwData{\$n \es \N\$}
\KwResult{Primos entre \$2\$ y \$n\$}
\linesnumbered
max$=\left[;(n-3)/2;\right]$;
boolean esPrimo[$i],\;;\; i=1,2,\dots,$max\;

\SetVline
\For{\$j=1,2,\dots,$max }{ esPrimo[$j]=\$True\;}
$i=0\$;
\While{\$(2i+3)(2i+3) \leq n\$}{
    $k=i+1\$;
    \While{\$(2k+1)(2i+3) \leq n\$}
    {
        esPrimo[\$((2k+1)(2i+3)-3)/2]=\$False\;
        $k=k+1\$;
    }
    $i=i+1\$;
}
\Imprimir\;
\For{\$j=1,2,\dots,$max }{
\If{esPrimo[$j]=\$True}{\Imprima $j$ }
}
\end{algorithm}

```

produce:

Algoritmo 5.3: Criba de Eratóstenes

Entrada: $n \in \mathbb{N}$ **Resultado:** Primos entre 2 y n

```
1 max =  $\lfloor (n-3)/2 \rfloor$ ;  
2 boolean esPrimo[i], i = 1, 2, ..., max;  
3 for j = 1, 2, ..., max do  
4   esPrimo[j] = True;  
5 i = 0;  
6 while  $(2i+3)(2i+3) \leq n$  do  
7   k = i + 1;  
8   while  $(2k+1)(2i+3) \leq n$  do  
9     esPrimo[ $((2k+1)(2i+3)-3)/2$ ] = False;  
10    k = k + 1;  
11   i = i + 1;  
12 Imprimir;  
13 for j = 1, 2, ..., max do  
14   if esPrimo[j] = True then  
15     Imprima j
```

Capítulo 6

TRADUCCIÓN \LaTeX A HTML: \LaTeX 2HTML

6.1 GENERALIDADES

LaTeX2HTML Translator es un conjunto de scripts en PERL. LaTeX2HTML convierte un documento \LaTeX (un archivo *.tex) en un documento adecuado para la internet. LaTeX2HTML fue creado por Nikos Drakos y Ross Moore. Para correr LaTeX2HTML Translator bajo Windows 98/Millennium/2000/XP, se necesitan algunos programas: Perl, MiKTeX, GhostScript y Netpbm (biblioteca de conversión de imágenes).

La mejor recomendación es: si quiere traducir LaTeX a html, use Linux. Entre los programas asociados a LaTeX está LaTeX2HTML.

Si quiere probar en XP, pues deberá armarse de paciencia.

Nota 1: Algunas versiones de XP no permiten la instalación de LaTeX2HTML.

Nota 2: La instalación que se describe se ha probado en Windows 98/Millennium/2000/ y algunas versiones de XP.

Un archivo que compile bien no tiene garantizada la traducción con LaTeX2HTML. El traductor es sensible a expresiones LaTeX correctas pero “extrañas”. Por ejemplo, el código

```
\newcommand \sen {\sin}
```

compila bien, pero hace que la traducción sea un desastre. Todo se arregla cambiando este código por

```
\newcommand{\sen}{\sin}
```

Sobre Windows, LaTeX2HTML se corre en una ventana MS-DOS. LaTeX2HTML genera una carpeta con el nombre del archivo *.tex actual. En esta carpeta están los nodos (páginas web) junto con los archivos *.gif que componen la mayoría del texto matemático de la página. En general, LaTeX2HTML le asigna al texto “tex” la marca correspondiente en HTML, en el caso de encontrar texto “fuertemente formateado” (como una fórmula compleja) lo convierte en un archivo gif.

Nota: Este manual se generó utilizando programas un poco desactualizados de Perl, Ghostscript, GSView y MiKTeX. En general, el manual debe funcionar con las versiones más actuales con algunos cambios menores, por lo tanto, se debe utilizar tan sólo como una guía. Aquí seguimos la descripción del proceso de instalación de ([6]) el cual ya probamos en varias máquinas con XP.

Las direcciones donde se pueden encontrar las versiones más recientes de los programas se dan al inicio de este manual, sin embargo, en esta parte se pondrán las direcciones tal y cómo se utilizaron en ese momento.

Otras Opciones

Existen otras opciones de poner texto matemático en el web (suponiendo que no tenemos ya un archivo LaTeX que queramos aprovechar). Por ejemplo se puede editar en Matemática Media 4.0 (o versiones anteriores) y guardar el archivo en el formato HTML. También Matemática guarda en el formato MathML (un formato con marcas para el manejo del texto matemático <http://www.w3.org/TR/REC-MathML/>). Hay varios navegadores y que pueden interpretar este formato. También hay programas comerciales y libres para la edición de texto matemático en este formato.

6.2 DESCARGA E INSTALACIÓN DE LOS PROGRAMAS

6.2.1 Instalación de PERL

Debemos descargar los archivos de instalación de Perl, al momento de esta publicación está disponible en www.activestate.com/ActivePerl/download.htm. Se deben descargar dos archivos: el instalador de Windows 98 (en este caso InstMsi.exe) y Api522e.exe. Los usuarios de Windows NT deben descargar el instalador correspondiente. Los de Windows 2000/XP no necesitan el instalador, solamente `verb+Api522e.exe+`

Bien, ahora hacemos la instalación de PERL

1. Se deben cerrar todas las aplicaciones abiertas, luego corremos `InstMsi.exe`. Se debe reiniciar el sistema.
2. Corremos el programa instalador `Api522e.exe`. En las ventanas de diálogo, presionamos siempre Next. Dejemos que Perl quede grabado en `C:/perl`

6.2.2 GhostScript

Como referencia y para mantener el orden en el disco duro, creamos una carpeta nueva, digamos `c:\texutils`

Ghostscript es un interpretador para el lenguaje PostScript (TM). GSview es una interface gráfica para MS-Windows o OS/2. Para los documentos que siguen ciertas convenciones (Adobe PostScript Document Structuring Conventions), GSview permite seleccionar páginas para ver o imprimir.

La version actual de LaTeX2HTML funciona bien con GhostScript 5.50. Esta versión se puede conseguir en

<http://www.ccp14.ac.uk/ccp/ccp14/ftp-mirror/ghostscript/ghost/aladdin/g550/>

Descargar el archivo `gsv26550.exe` y ejecútelo. Instale GS5.50 en `c:\texutils` de modo que quede

```
c:\texutils\GS5.50
```

Nota: Para usar GhostScript 6.01, debe editar `config.pl`, después de la línea

```
close(GS);
```

y antes de

```
#Configure things determined by the Ghostscript version
```

añadir

```
@gs_devs=qw(pnmraw ppmraw);
$gs_version=6.01;
```

6.2.3 Instalación de MiKTeX

La instalación de LaTeX2html solo la hemos probado con MiKTeX2.5. Después de la instalación de MiKTeX, se debe tener (y/o editar en caso necesario) el `autoexec.bat` (o las variables de entorno), con las líneas

```
C:\PERL\BIN;C:\TEXMF\MiK\TeX $%, $\BIN;C:\TEXUTILS\GS5.50;
C:\TEXUTILS\GSVIEW;C:\TEXUTILS\NETPBM\BIN;
```

6.2.4 Instalación de L^AT_EX₂HTML

Descargue LaTeX2HTML del sitio:

<http://saftsack.fs.uni-bayreuth.de/~latex2ht/current/>

El archivo que descargamos, 'Latex2html-...', Lo ponemos en la carpeta `c:\mktemp\`

1. Descomprimos el archivo `netpbm.zip` en `c:\texutils\netpbm`
2. En el archivo `c:\mktemp\latex2html-...` editamos el archivo `prefs.pm` (editamos con NotePad o WordPad). Una configuración básica se obtiene buscando y editando las líneas siguientes:

```
$prefs{'EXTRAPATH'}='C:\\TEXUTILS\\GS5.50;C:\\TEXUTILS\\NETPBM\\BIN';
```

...

```
$prefs{'PREFIX'} = 'C:\\TEXUTILS\\L2H';
```

Con esto LaTeX2HTML se instalará en `C:\TEXUTILS\L2H`

3. En `c:\mktemp\latex2html-...` ejecutamos `config.bat` (se abre una ventana MS-Dos). Esta es la parte más delicada de la instalación. Aquí debe se debe reconocer todo lo que hemos instalado. Posiblemente la corrida se detenga en la línea

```
Checking for dvips version...
```

pero basta dar ENTER un par de veces para que siga adelante. En todo caso, la corrida debe quedar con los siguientes elementos:

```
config.pl, Release ...
...
...
checking for old config file
(cfgcache.pm)... not found (ok)
checking for platform... MSWin32
(Windows 32 bit)
checking for C:\Perl\bin\perl.exe...
C:\Perl\bin\perl.exe
checking perl version... 5.00503
checking if
perl supports some dbm... yes
checking if perl globbing works...
yes
```

```

checking for tex... C:\TEXMF\MiK\TeX $,\$BIN\tex.exe
checking
for latex... C:\TEXMF\MiK\TeX $,\$BIN\latex.exe
checking for
initex... C:\TEXMF\MiK\TeX $,\$BIN\initex.exe
checking for
kpsewhich... no
checking for TeX include path... NONE Warning:
Will not automatically install LaTeX2HTML style files.
checking for dvips... C:\TEXMF\MiK\TeX $,\$BIN\dvips.exe
checking dvips version... 5.86
checking if dvips supports the combination of -E
and -i -S 1... yes
checking for html4-check... no
checking for
gswin32c... \TEXUTILS\GS5.50\gswin32c.exe
checking for ghostscript
version... 5.50
checking for ghostscript portable bitmap device...
pnmraw
checking for full color device for anti-aliasing... ppmraw
checking for ghostscript library and font paths... built-in paths
are correct
checking for pnmcrop...
\TEXUTILS\NETPBM\BIN\pnmcrop.exe
checking if pnmcrop can crop from
one direction... yes
checking for pnmflip...
\TEXUTILS\NETPBM\BIN\pnmflip.exe
checking for ppmquant...
\TEXUTILS\NETPBM\BIN\ppmquant.exe
checking for pnmfile...
\TEXUTILS\NETPBM\BIN\pnmfile.exe
checking for pnmcat...
\TEXUTILS\NETPBM\BIN\pnmcat.exe
checking for pbmmake...
\TEXUTILS\NETPBM\BIN\pbmmake.exe
checking for ppmtogif...
\TEXUTILS\NETPBM\BIN\ppmtogif.exe
checking if ppmtogif can make
transparent GIFs... yes
checking if ppmtogif can make interlaced
GIFs... yes
checking for pnmtopng...
\TEXUTILS\NETPBM\BIN\pnmtopng.exe
checking if multiple pipes
work... no Unfortunately multiple pipes are not reliable on this
OS.
checking for temporary disk space... C:\WINDOWS\TEMP

```

```

creating
cfgcache.pm
creating test.bat creating install.bat Note: Will
install...
... executables to   : C:\TEXUTILS\L2H\bin
... library items to : C:\TEXUTILS\L2H

```

Esta salida se graba en el archivo `config.log` en este mismo directorio

4. Ejecutamos `install.bat`.

Este programa copia los archivos a `C:\TEXUTILS\L2H`.
Después de esto se puede borrar el archivo temporal `c:\mktemp`

5. Ahora configuramos LaTeX2HTML. En `C:\TEXUTILS\L2H` abrimos `l2hconfig.pm` (digamos con NotePad) buscamos y hacemos los siguientes cambios.

```

@IMAGE_TYPES=qw(gif png);
$LOCAL_ICONS=1;
$HTML_VERSION='3.2, math';
$UNSEGMENT = 1;
$EXTRA_IMAGE_SCALE = 2;
$MATH_SCALE_FACTOR = 1.5;

```

`$MATH_SCALE_FACTOR` define el tamaño del texto matemático, 1.2 puede ser muy pequeño y 1.6 muy grande.

Aquí podemos cambiar el idioma, buscamos por ejemplo "Contents" y lo cambiamos por "Contenido", etc.

Ya todo está listo. Ahora podemos, desde una ventana MS-DOS, traducir `miarchivo.tex` (con la sintaxis apropiada) escribiendo

```
latex2html  miarchivo.tex  (ENTER)
```

Y esperar. Esto genera una carpeta "miarchivo" con las páginas web y los archivos adicionales indispensables (en `l2hconf.pm` hay una variable de instalación llamada `$TEXINPUTS` la cual le dice a LaTeX2HTML dónde buscar los archivos `*.tex` para procesar).

Si esto no funciona, se puede llevar el archivo `*.tex` a de `C:\TEXUTILS\L2H\BIN` y desde una ventana MS-Dos hacer la misma corrida. La carpeta quedará en este directorio.

Algunas opciones se pueden poner en esta línea de comandos. Por ejemplo

```
latex2html -no_math -html_version 3.2 -no_navigation miarchivoal.tex
```

o

```
latex2html -split, -no_navigation -show_section_numbers miarchivo.tex
```

Un Manual completo de LaTeX2HTML (en Latex) se puede encontrar y descargar en

<http://cbl.leeds.ac.uk/nikos/tex2html/doc/manual/manual.html>

Capítulo 7

CÓMO HACER TRANSPARENCIAS CON LA CLASE BEAMER

7.1 INTRODUCCIÓN

“Beamer” es una clase LaTeX (`\documentclass{beamer}`) para generar transparencias para presentaciones (à la Power Point).

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 “Warsaw”) se ve en la figura 7.1.

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 ε 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 7.1 Transparencia Beamer.

- ¿Qué significa "beamer"? Formalmente "der Beamer (-s)" significa un proyector digital o LCD ([?]). Beamer fue desarrollado por Till Tantau.

7.2 INSTALAR BEAMER

Beamer viene incluido en MikTeX 2.5, 2.6 y 2.7. Si tiene una instalación completa (la cual puede obtener en <http://miktex.org>, se descarga 'Download MiKTeX 2.x Net Installer') puede usar beamer sin problemas.

Si no tiene una instalación completa, haga lo siguiente

1. Vaya a INICIO-Programas-MikTex2.x-BrowsePackages
2. Seleccione beamer e instale el paquete desde internet o desde un disco (usted puede cambiar esta opción, antes de seleccionar beamer, en el menú Repository-Change Package Repository)
3. Vaya a INICIO-Programas-MikTex2.x-Settings y luego haga clic en 'Refresh FNDB' y luego en 'Update Formats'

7.3 UN DOCUMENTO BEAMER

La estructura general de un posible documento Beamer es,

```

\documentclass{beamer}
\usefonttheme{professionalfonts}% font de LaTeX
\usetheme{Warsaw} % Tema escogido en este ejemplo
\setBeamercovered{transparent}

%% packages y comandos personales %%%
\usepackage[latin1]{inputenc}
\usepackage{latexsym} % S'imbolos
\usepackage{amsmath}
\usepackage{amssymb}
\newtheorem{Teorema}{Teorema}
\newtheorem{Ejemplo}{Ejemplo}
\newtheorem{Definicion}{Definici'on}
\newtheorem{Corolario}{Corolario}
\newtheorem{Prueba}{Prueba}

\begin{document}
\title{Factorizaci'on en  $Z_p[x]$  y en  $Z[x].\$\\}$ 
\subtitle{Parte I}
\author{{\large Walter Mora F.}\}
{\small Escuela de Matem'atica, ITCR}\{\small www.itcr.ac.cr/revistamate/}\}
\vspace*{0.5cm}
\date{Junio 2007}
\frame{\titlepage}

\section{...}
\begin{frame}{T\titulo de este marco}
%texto
\end{frame}

\begin{frame}{T\titulo de este marco}
%texto
\end{frame}
\end{document}

```

Compilar

- Un documento Beamer se puede compilar directamente con **PDFLaTeX** y verlo con Acrobat Reader. Pero...
 - si hay gráficos, pdfLaTeX funciona bien si los gráficos son PDF
 - si hay gráficos EPS, podemos cambiar el formato de todas las imágenes a PDF con Acrobat Distiller o Adobe Illustrator, por ejemplo.

La primera transparencia del código anterior es

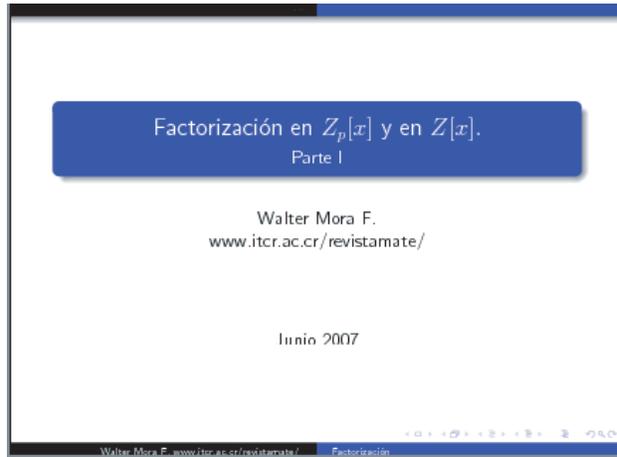


Figura 7.2 Transparencia Beamer. Tema Warsaw

Notas.

- Beamer carga (por default) “graphicx” para el soporte de gráficos.
- La lista de temas que viene con Beamer es

```

\usetheme{Bergen}
\usetheme{Boadilla}
\usetheme{Copenhagen}
\usetheme{Dresden}
\usetheme{Hannover}
\usetheme{Luebeck}
\usetheme{AnnArbor}
\usetheme{Berkeley}
\usetheme{Darmstadt}
\usetheme{Frankfurt}
\usetheme{Ilmenau}
\usetheme{Madrid}%azulito-li;la
\usetheme{Warsaw}%int
\usetheme{Antibes}
\usetheme{Berlin}
\usetheme{CambridgeUS}%rojo-gris
\usetheme{Malmoe}
\usetheme{PaloAlto}

```

- Una vista de los temas se puede encontrar en [7] o también en

<http://mike.polycat.net/gallery/beamer-themes>

- En internet hay otros tantos temas Beamer disponibles, usualmente temas de particulares o instituciones pero de distribución libre. Los temas se pueden personalizar (ver [7]).

7.4 MARCOS

Marcos con una lista de items.

El uso más común de un marco es poner una lista de items

```
\begin{frame}
\frametitle{Campo Galois  $GF(p^r)$ }
\framesubtitle{Resumen}
\begin{enumerate}
\item Todo dominio integral {\em finito} es un campo\\

\item Si  $F$  es un campo con  $q$  elementos, y  $a \in F$ 
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}
```

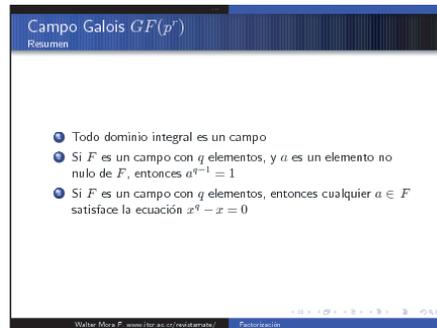


Figura 7.3 Marco Beamer. Tema Warsaw

7.5 VELOS (OVERLAYS)

En una presentación puede ser deseable que los items vayan apareciendo uno a la vez, mientras los otros permanecen con un *velo*. Esto se puede lograr agregando la opción [$\langle + - \rangle$] los entornos `enumerate` o `itemize`. La salida de este código se muestra en la figura 7.4.

```

\begin{frame}
\frametitle{Campo Galois  $GF(p^r)$ }
\framesubtitle{Resumen}
\begin{enumerate}[<+>]% <- NUEVA OPCION
\item Sea  $F$  un campo con  $q$  elementos y  $a$  un elemento no nulo de  $F$ .
Si  $n$  es el orden de  $a$ , entonces  $n|(q-1)$ .\\

\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}

```

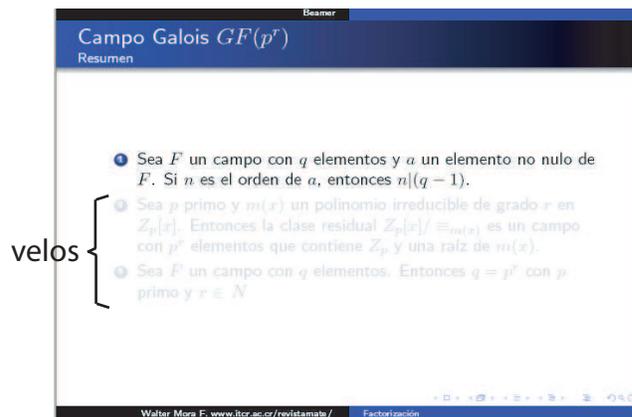


Figura 7.4 Marco Beamer con dos “velos”.

7.5.1 Opciones $\langle i \rangle$ y $\langle i \rangle$

En vez de usar la opción $[<+>]$ en el entorno `enumerate` (o `itemize`), se puede agregar un comportamiento un poco más dinámico usando las opciones $\langle i \rangle$ y $\langle i \rangle$.

Con estas opciones podemos controlar la secuencia en la que se presentan distintas líneas en una transparencia.

- $\langle i \rangle$ 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.
- $\langle i \rangle$ 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 items se marcan con <1->. Después se muestra el procedimiento, que corresponde a los items <2->, <3->, <4->

```
\begin{frame}{Ejemplo}
  \begin{enumerate}
    \item<1->  $x^4-x=0$  % <-
    \item<2->  $x(x^3-1)=0$ 
    \item<3->  $x=0$  \; o \;  $x^3-1=0$ 
    \item<4->  $x=0$  \; o \;  $x=\sqrt[3]{1}$ 
    \item<1->  $\Longrightarrow x=0, x=1$  % <-
  \end{enumerate}
\end{frame}
```

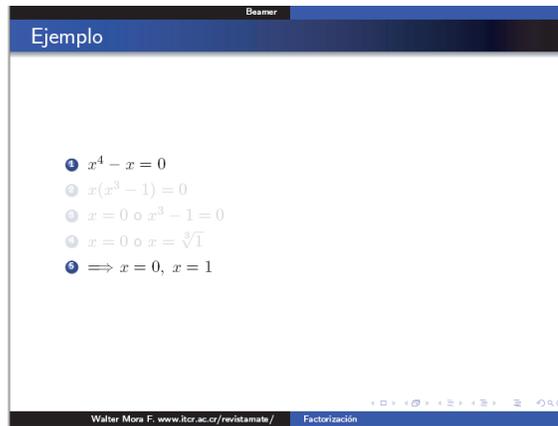


Figura 7.5 Marco Beamer con overlays.

- Un ejemplo del comando `\uncover` se presenta más adelante, en la sección 7.10.

7.5.2 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 item.

```
\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>  $x=0$  \; o \;  $x^3-1=0$ 
    \item<4-|alert@4>  $x=0$  \; o \;  $x=\sqrt[3]{1}$ 
    \item<1-|alert@1>  $\Longrightarrow x=0, x=1$  % <-
  \end{enumerate}
\end{frame}
```

```
\end{frame}
```

Nota: si solo queremos el efecto de “alerta” en cada item, podemos poner

```
\begin{frame}{Ejemplo}
  \begin{enumerate}[<+ -| alert@+>] % <- opcin
    \item  $x^4-x=0$ 
    \item  $x(x^3-1)=0$ 
    \item  $x=0$  \; o \;  $x^3-1=0$ 
    \item  $x=0$  \; o \;  $x=\sqrt[3]{1}$ 
    \item  $\Rightarrow x=0, x=1$ 
  \end{enumerate}
\end{frame}
```

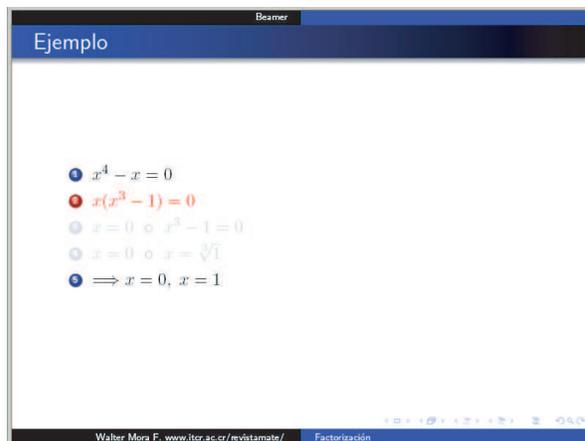


Figura 7.6 Marco Beamer con opciones `<i-|alert@ i>`

7.6 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`.

`\pause` se puede usar varias veces en el el marco si queremos fragmentar los velos.

Los ejemplos de cómo se podría usar `pause` están en la sección que sigue.

7.7 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

```
\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 % <-- PAUSE
\begin{example}
  Sea  $P(x) = x^3 - 2 \in \mathbb{Q}[x]$ .  $P(x)$  es irreducible. Aunque tiene una raíz en  $\mathbb{R}$ ,
  a saber  $2^{1/3}$ ,  $\mathbb{R}$  no es un campo de escisión para  $P$ .
\end{example}

\end{frame}
```

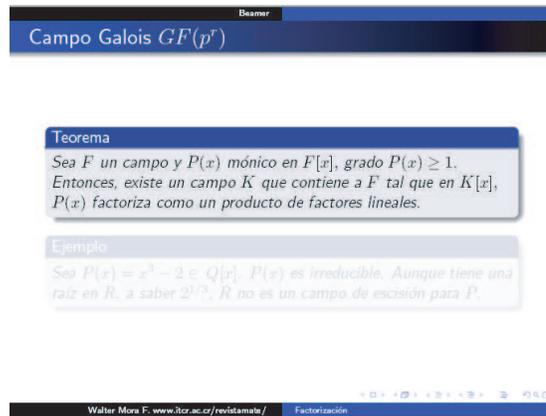


Figura 7.7 Entornos Teorema y Ejemplo.

Nota. Beamer tiene su propio entorno para ejemplos, teoremas, definiciones, etc. Los entornos se declaran `begin{example}...end{example}`, etc. El idioma se puede cambiar en el archivo

`C:\Archivos de programa\MiKTeX 2.5\tex\latex\beamer\base\beamerbasetheorems.sty`,

7.8 BLOCKS.

Los blocks son parecidos al entorno Teorema. Veamos un ejemplo

```

\begin{frame}{Búsqueda de factores lineales}

  \begin{block}{Teorema de Descartes (1637)}
    Si  $P(x)=a_nx^n+\dots+a_0 \in \mathbb{Z}[x]$ 
    y tiene a  $x=r/s$  como raíz, con  $r,s$ 
    enteros primos relativos, entonces  $s|a_n$  y  $r|a_0$ .
  \end{block}
  %\onslide
\end{frame}

```

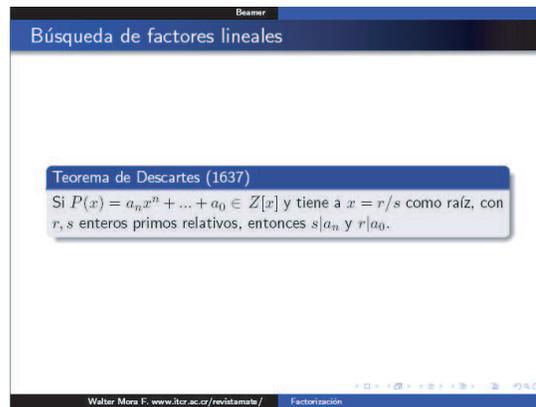


Figura 7.8 Block.

7.9 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 7.9),

```

\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}

```

LaTeX. Walter Mora F., Alex Borbón A.

Derechos Reservados © 2009 Revista digital Matemática, Educación e Internet (www.cidse.itcr.ac.cr)

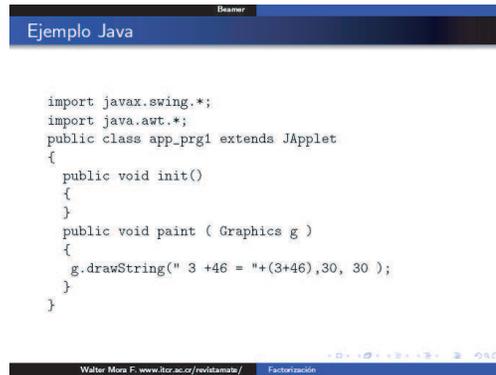


Figura 7.9 Opción fragile.

7.10 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 semi-verbatim. Veamos un ejemplo de código en C.

En este ejemplo se usa

- `\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.

```
\begin{frame}[fragile]
\frametitle{Un algoritmo para buscar n\'umeros 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->{N\’otese el uso de \alert{\texttt{std::}}.}
\end{frame}

```

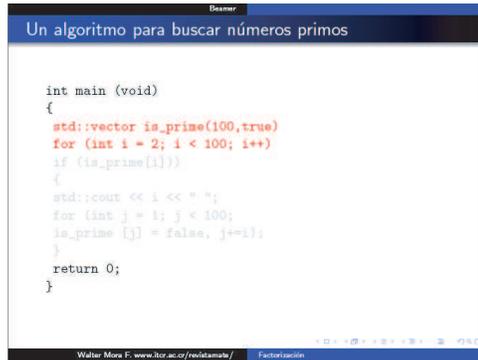


Figura 7.10 semiverbatim

7.11 BEAMER Y EL PAQUETE ALGORITHM2E

En esta sección vamos a ver un ejemplo en el que se usa el paquete `algorithm2e` en un entorno `frame`. Para este ejemplo, se puso en el preámbulo

```

%prebulo
\usepackage[ruled, ,vlined,lined,linesnumbered,algorithsection,portugues]{algorithm2e}

```

Observe que necesitamos la opción `fragile` para `frame` y la opción `[H]` para `algorithm`.

```

\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
\For{$j=1$ \KwTo $m$}
{
  $s=y_j-\alpha_0$;
  \For{$k=1$ \KwTo $j-1$}
  {$s=s-\alpha_k \cdot f$;

```

```

    $f=(x_j-x_k)\cdot f$ \;
  }
  \Return $\alpha_j=s/f$ \;
}
\end{algorithm}
\end{frame}

```

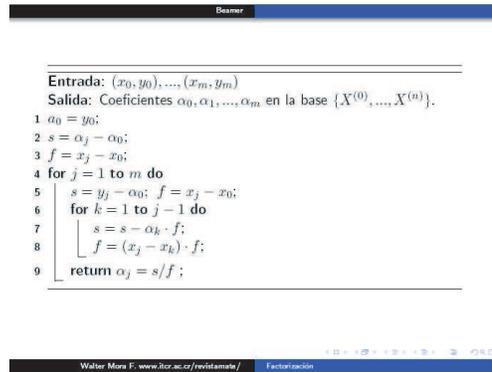


Figura 7.11 Beamer con algorithm2e

7.12 GRÁFICOS

- Beamer maneja gráficos de tamaño máximo 128 mm por 96 mm.
- pdfLaTeX soporta los formatos JPEG, PNG, PDF, y MetaPost.
 - compilar con pdfLaTeX funciona bien si los gráficos son PDF.
 - si hay gráficos EPS, podemos cambiar el formato de todas las imágenes a PDF con Acrobat Distiller o Adobe Illustrator, por ejemplo.
- la manera de incluir un gráfico es la usual

```
\includegraphics{img1.pdf}
```

Se puede poner la imagen en un entorno `\begin{figure}... \end{figure}`

- En el siguiente ejemplo, vamos a poner una superficie S en una transparencia. Inicialmente la imagen tenía formato EPS. Fue convertida a PDF con Adobe Distiller.

```

\begin{frame}{Superficie $$$ y proyecci\'on}
\begin{figure}[h]
\centering
\includegraphics{images/B_fig12.pdf}
\caption{Superficie $$$}\label{Beamer:fig12}
\end{figure}
\end{frame}

```

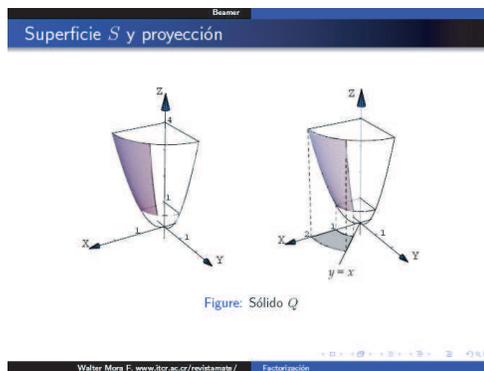


Figura 7.12 Incluir un gráfico

7.13 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 dos botones Beamer, `\beamergotobutton` y `\beamerreturnbutton`.

```

\begin{frame}{MARCO 1}
\hyperlink{Liga1}{\beamergotobutton{Ir a Marco 2}}
\hypertarget<2>{Liga2}{}
\end{frame}

```

```

\begin{frame}{MARCO 2}
\hyperlink{Liga2}{\beamerreturnbutton{Regresar a Marco 1}}
\hypertarget<2>{Liga1}{}
\end{frame}

```

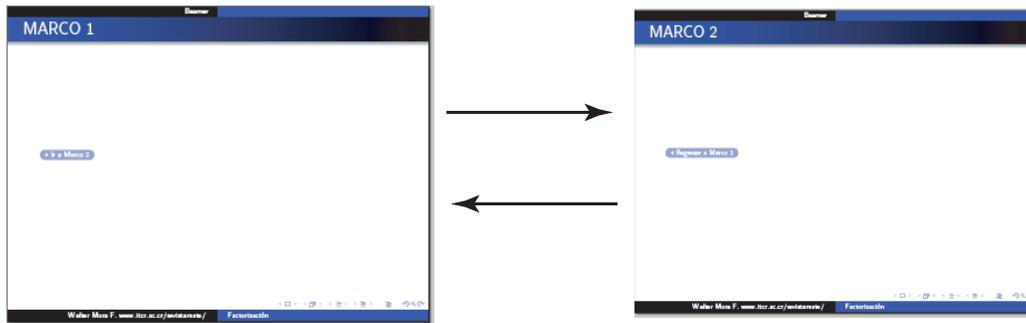


Figura 7.13 Ligas

- También se puede incorporar ligas sin botones,

```
\begin{frame}{MARCO 3}
\hyperlink{Liga2}{Ir a Marco 4}
\hypertarget<2>{RegresarLiga2}{}
\end{frame}
```

```
\begin{frame}{MARCO 4}
\hyperlink{RegresarLiga2}{Regresar a Marco 3}
\hypertarget<2>{Liga2}{}
\end{frame}
```

7.14 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 son

1. `\transblindshorizontal`
2. `\transblindsvertical<2,3>`
3. `\transboxin`
4. `\transboxout`
5. `\transglitter<2-3>[direction=90]`
6. etc.

- Se pueden incluir películas, animaciones, etc. con `\usepackage{multimedia}`.

Ejemplo.

En este ejemplo, además de poner un efecto de transición vamos a crear un 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 el paquete xcolor (que Beamer carga automáticamente) 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.

```
%Agregar al pre'ambulo
%\documentclass[xcolor=pdftex,table]{Beamer}
%\definecolor{ZurichBlue}{rgb}{.255,.41,.884}
\begin{frame}{Marco B}
  \transdissolve % <- Efecto de transici'on
  \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}
```

Marco B		
Enteros	long	de -2^{63} a $2^{63} - 1$
	int	de -2^{31} a $2^{31} - 1$
	short	de -2^{16} a $2^{16} - 1$
	byte	de -2^7 a $2^7 - 1$
Coma flotante	float	de 3.4×10^{-38} a 3.4×10^{38}
	double	de 1.7×10^{-308} a 1.7×10^{308}
Caracteres	char	
	boolean	true o false

Walter Mora F. www.itcr.ac.cr/revistamate/ Factorización

Figura 7.14 Transición y Color

7.15 LIGAS A DOCUMENTOS EXTERNOS

Para hacer ligas a documentos externos podemos usar el comando `\href{http://...}{ texto}`. Por ejemplo

1. Liga a un documento en el disco duro

El código del programa está aquí

El código del programa está `\a`
`\href{file://C:/MiJava/programa1.java}{\underline{aquí}\{i}}`

2. Otras ligas

Prof. Walter Mora F.
`{\href{wmora2@yahoo.com.mx}`
`(wmora2@yahoo.com.mx)}\}`

`{\color{ZurichBlue}`
 En la revista digital Matemática, Educación e Internet encontrar `\a`
`\href{http://www.cidse.itcr.ac.cr/revistamate/}`
`{\underline{el Manual de LaTeX},}` en la liga correspondiente a
 ‘‘Tecnologías de Internet para la Enseñanza de la Matemática’’.
`}\}%fin color`

7.16 ANIMACIONES

Se puede preparar una animación simplemente desplegando una secuencia de gráficos, por ejemplo

```
\begin{frame}
\frametitle{Mi animacin}
\begin{figure}[t]
\centering
\includegraphics<1>[scale=0.2]{images/picture_1.eps}
\includegraphics<2>[scale=0.2]{images/picture_2.eps}
\includegraphics<3>[scale=0.2]{images/picture_3.eps}
\includegraphics<4->[scale=0.2]{images/picture_4.eps}
\end{figure} \end{frame}
```

Lecturas recomendadas

Se recomienda las lecturas [10] y [7].

Appendix A

Ambiente `picture` de LaTeX

El ambiente `picture` de LaTeX es un ambiente espacial para insertar figuras simples implementados con comandos simples como `\plot`, `\put`, etc.

Por supuesto, esto se puede hacer de manera sencilla con paquetes como Adobe Illustrator, Macromedia Fireworks, *Mathematica*, etc. Pero la idea es incorporar figuras LaTeX simples y pequeñas.

Aún así, usando el paquete `PsTricks` o usando `PixTex`, se habilitan un gran cantidad de comandos para hacer figuras LaTeX.

Por supuesto, existen varios editores que permiten hacer figuras y generan el código LaTeX, listo para introducirlo en nuestro documento.

En este apéndice solo vamos a considerar brevemente

1. `LaTeXDraw`. Este es un editor gratuito multiplataforma (implementado en Java) basado en `PsTricks`. `LaTeXDraw` genera el código LaTeX de las figuras.
2. `PiCTeX`: un paquete de macros TeX para dibujar figuras.

*

A.1 LATEXDRAW

Este editor se descarga desde

<http://latexdraw.sourceforge.net/download.html>

Obtenemos un comprimido “LaTeXDraw2.0.2.zip”. Se ejecuta el archivo `installer.jar`. Debe tener Java (<http://www.java.com/es/download/>) en su máquina, lo cual de por sí, es muy adecuado.

En el directorio de instalación que Ud. eligió en el proceso de instalación (el default es `C:\Program Files\latexdraw`) está el ejecutable "LaTeXDraw.jar" (doble clic y listo).

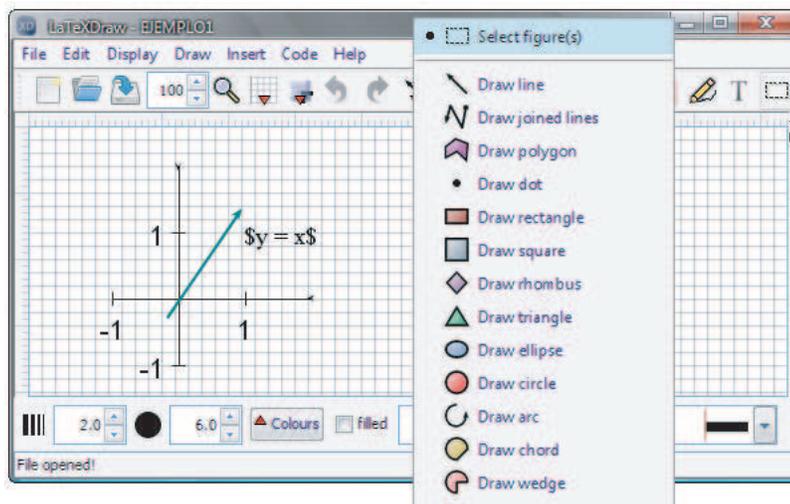


Figura A.1 Editor LaTeXDraw

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.

Para usar TeXDraw debemos poner en el *preámbulo*

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

■ EJEMPLO A.1

La figura que se ve en el editor LaTeXDraw, en el ejemplo (A.1), 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}
}

```

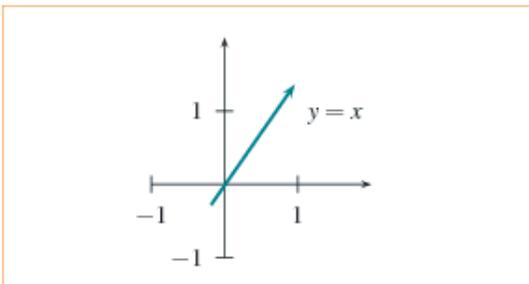
Ahora este código lo pegamos en nuestro documento LaTeX, por ejemplo

```

{\fboxsep 12pt \fboxrule 0.2pt%
\fcolorbox{orange}{white}{%
  \begin{minipage}[t]{0.5 \textwidth}
  \bc% Nuestro \begin{center} abreviado
  \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}
  }
  \ec
  \end{minipage}
}}%

```

Y obtenemos



A.2 PICTEX

PiCTeX es una colección de macros T_EX. Con PiCTeX se tiene control del dibujo, es decir, se pueden poner las cosas exactamente, donde uno quiere. Solo se necesita conocer las coordenadas exactas. Por esto último es que es conveniente, en dibujos que llevan muchos detalles, diseñarlos previamente en papel milimétrico para tener acceso rápido a las coordenadas de cada punto. Algunas coordenadas se deben calcular con las ecuaciones de los objetos geométricos. Con PiCTeX se obtienen dibujos de gran precisión, usualmente indispensables en libros, artículos o documentos que requieran un acabado fino.

Necesitamos algunos paquetes, así que necesitamos un documento con un preámbulo adecuado:

```
\documentclass{article}
\usepackage{amsmath,amssymb,amsfonts,latexsym,cancel}
\usepackage{rawfonts}
\usepackage{pictexwd}
\begin{document}
...
\end{document}
```

Los dibujos PiCTeX contienen instrucciones sobre el tamaño de la caja (invisible) donde estará el dibujo, escala, nombre del dibujo, etc. La caja que le definimos al dibujo es exactamente el espacio que ocupará en el texto.

Todo dibujo estará entre `\beginpicture` y `\endpicture`:

```
\beginpicture
  Comandos PiCTeX
\endpicture
```

A.2.1 Comandos de PiCTeX

Los comandos de PiCTeX deben ir seguidos de al menos un espacio en blanco y los `'/'` no deben quedar pegados a los números (a veces esto no es problema pero otras veces si).

A.2.2 Escala y caja del dibujo.

Escala:	<code>\setcoordinatesystemunits <xunits,yunits> point at xcoord ycoordl</code>
Caja:	<code>\setplotareax from x1 to x2, y from y1 to y2l</code>

Los objetos son posicionados en el dibujo especificando sus coordenadas (x,y) . Se debe empezar especificando la escala que se va a usar. Esto se hace con el comando `\setcoordinatesystem xunits` se refiere a la escala en el eje X y `yunits` a la escala en el eje Y . `xcoord` y `ycoord` se refiere al punto de referencia del sistema, si omitimos esto último, la referencia será $(0,0)$. Las unidades son `mm`, `cm`, `in`, `pt` (`1in=72.27pt`), `pc`, (`1pc=12pt`) etc.

Si no se define un área para el dibujo, entonces puede pasar que éste caiga encima del texto del documento original. La caja del dibujo se define especificando el rango de x (ancho), y rango de y (alto). Estos dos rangos se dan respecto al origen de coordenadas.

El resultado de estos dos comandos juntos, `escala` y `caja`, es abrir el espacio exacto en el documento para la caja definida

En el siguiente ejemplo, se declara una caja $4\text{cm} \times 4\text{cm}$ y se dibuja la caja punteada (eso es lo que hace las instrucciones `\setdots <2pt>`: separa los puntos dos pt's y `\plot` que dibuja líneas.). Las esquinas serían $(-2, -2)$ y $(2, 2)$. El código y lo que resulta es

```
\[
\beginpicture
\setcoordinatesystem units <1cm,1cm>
\setplotarea x from -2 to 2, y from -2 to 2
  \setdots <3pt>
  \setlinear
  \plot -2 -2 -2 2 2 2 2 -2 -2 -2 /
\axis bottom label { Caja de Dibujo } invisible /
\endpicture \]
```



Caja de Dibujo

En este ejemplo se usaron los comandos `\plot`, `\setdot` y `\setlinear` para dibujar las líneas punteadas que marcan el contorno de la caja. El default de las unidades es 1 pt . Una manera de agrandar o empequeñecer el dibujo es cambiar la escala. Al variar la escala, es posible que algún texto deba ser reubicado pues el tamaño de las letras no varía.

En el mismo archivo se pueden usar varias escalas. Cada instrucción actúa de acuerdo a la escala declarada que le antecede. Esto se usa cuando uno quiere que en un mismo dibujo queden figuras con distintas escalas.

A.2.3 Texto en un dibujo.

```
\put { TEXTO } [o_x,o_y]<xshift,yshift> at xcoord ycoord
```

El comando `\put` coloca un bloque de texto (o un dibujo) en un punto específico de la caja de dibujo. Cada bloque de texto está encerrado en un rectángulo invisible, respecto a esta caja es que se orienta la posición del texto en el dibujo. En el comando, `TEXTO` se refiere a cualquier bloque de texto $\text{T}_{\text{E}}\text{X}$. o_x, o_y se usa para orientar la caja de tal manera que $(x\text{coord}, y\text{coord})$ quede en alguna de sus esquinas. Es decir, pone una esquina de la caja en las coordenadas $(x\text{coord}, y\text{coord})$. Si o_x y o_y se omiten entonces se produce un centrado horizontal y vertical, respectivamente, de la caja.

$$o_x \in \{l, r\} = \{\text{esquina izquierda, esquina derecha}\}$$

$$o_y \in \{t, B, b\} = \{\text{esquina superior, línea base del texto, esquina inferior}\}$$

Una vez posicionada la caja, $\langle x\text{shift}, y\text{shift} \rangle$ se usa para correrla x unidades a la derecha (+) o izquierda (-) y y unidades hacia arriba (+) o abajo (-) del punto $(x\text{coord}, y\text{coord})$.

Por ejemplo, vamos a dibujar un sistema de coordenadas para ver el efecto de las instrucciones (ej `ej2.tex` es el archivo donde está el código de los ejes que aquí aparecen)

```

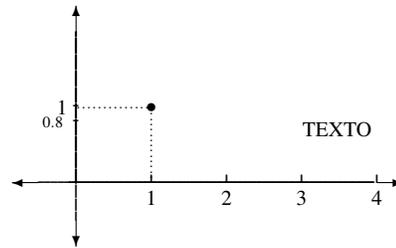
\l
\beginpicture
  \setcoordinatesystem units <1cm,1cm>
  \setplotarea x from -1 to 5, y from -1 to 2

  \setlinear
    \plot -.5 0 4 0 0 0 0 2 0 0 0 -.5 /
    \put{\vector(1,0){10}} [B1] at 4 0
    \put{\vector(-1,0){10}} [B1] at -.5 0
    \put{\vector(0,1){10}} [B1] at 0 2
    \put{\vector(0,-1){10}} [B1] at 0 -.5
    \put {\footnotesize{$_{0.8}$}} at -.3 .8
    \put {\footnotesize{$1$}} at -.15 1
    \put {\footnotesize{$1$}} at 1 -.2
    \put {\footnotesize{$2$}} at 2 -.2
    \put {\footnotesize{$3$}} at 3 -.2
    \put {\footnotesize{$4$}} at 4 -.2
    \put {$\bullet$} at 1 1
    \put {-} at 0 1
    \put {-} at 0 0.8
    \put { TEXTO }
      [rt] <3cm,-.2cm> at 1 1

    \plot 1 0 1 0.1 /
    \plot 2 0 2 0.1 /
    \plot 3 0 3 0.1 /
    \plot 4 0 4 0.1 /

  \setdots <2pt>
  \setlinear
  \plot 1 0 1 1 0 1 /
  \axis bottom label { Texto en un dibujo} invisible /
\endpicture
\l

```



Texto en un dibujo

Lo que hace la instrucción [rt] <3cm,-.2cm> es poner la esquina superior (t) derecha (r) de la caja invisible de 'TEXTO' a 3cm a la derecha de (1,1) y luego baja la caja 0.2cm . Observe que en la segunda instrucción, por default, (1,1) es el centro del punto (\$\bullet\$).

Un texto un poco más detallado lo puede encontrar en "Dibujar figuras LaTeX con PiCTeX" (<http://www.cidse.itcr.ac.cr/revistamate/HERRAmInternet/>).

Bibliografía

- [1] Hahn, J. *LT_EX for everyone*. Prentice Hall, New Jersey, 1993.
- [2] M. Goossens; F. Mittelbach; A. Samarin. *The LT_EX Companion*. Addison-Wesley. 1993.
- [3] G. Grätzer *The New Standard LT_EX*. Personal T_EX Inc. California. 1998.
- [4] L. Lamport. *LT_EX*. Addison-Wesley. 1996.
- [5] N. Drakos, R. Moore. *The LaTeX2HTML Translator*.
En <http://cbl.leeds.ac.uk/>
- [6] L. Seidel. "LaTeXtoHTML".
En <http://apollo.us.es/CervanTeX/>
- [7] Till Tantau *User Guide to the Beamer Class, Version 3.07*
En <http://latex-beamer.sourceforge.net>, 2007.
- [8] Andrew Mertz y William Slough. *Beamer by Example*.
En <http://www.tug.org/pracjourn/2005-4/mertz/mertz.pdf>

- [9] Gilles Bertrand. "Preparing a presentation (Beamer)".
En <http://www.rennes.enst-bretagne.fr/~gbertran/>
- [10] KiJoo Kim. "Beamer v3.0 Guide".
En <http://faq.ktug.or.kr/wiki/uploads/>
- [11] Wichura M. "*The Pictex Manual*". Universidad de Chicago, 1987.

Indice

A

Algoritmos
algorithm2e, 67

B

babel, 9
baselinestretch, 8
beamer
gráficos, 91
ligas, 92
marcos, 83
pdfLaTeX, 81
tablas, 95
temas, 82
velos, 83
alert, 85
algorithm2e, 90
blocks, 87
boldmath (negrita), 34
boldmath
boldsymbol, 34
pbm, 34
unboldmath, 34

C

calligra
usepackage, 14
center, 16

centrar en modo matemático, 34
choose, 31
Comandos, 61
newcommand, 62
opciones por default, 63
Compilar, 5
Contenido, 9

D

definecolor, 20
displaystyle, 30
documentclass, 2
DVI
archivo, 6
imprimir, 6

E

enumerate, 26
eqnarray, 38
equation, 35

F

fbox, 17
fboxsep, 20
fcolorbox, 20
figure
subfigure, 54
fontenc
usepackage, 13

fontencoding, 13
fontfamily, 13
fontseries, 13
fontsize, 13
footnote, 22
Fuentes, 11

G

graphicx, 49
 bmp, 49
 DeclareGraphicsExtensions, 49
 DeclareGraphicsRule, 49
 dvips, 49
 eps, 49
 gif, 49
 includegraphics, 50
 pdf, 49
 rotación, 50
 scale, 50

H

hfill, 21
hspace, 25

I

inputec
 usepackage, 9
int, 41
 iiint, 41
 iint, 41
 oint, 41
intllimits, 32
item, 26

K

Knuth
 D., 1

L

labelenumi, 27
Lamport
 L., 1
LaTeX2HTML, 3
 url, 77

M

markright, 8
Matrices
 array, 36
mbox, 37
MiKTeX, 2
minipage, 18

 con figure, 18
multicols, 17
myheadings, 8

N

Negrita
 en modo matemático, 34
newpage, 8
nonumber, 38

O

overset, 41

P

pagestyle, 8
parindent, 8
parskip, 8
PDF
 TeX a Pdf, 6
Perl, 3
Preámbulo, 4

R

renewcommand, 8

S

selectedfont, 13
setcounter, 35
stackrel, 31
subequations, 35
substack, 31

T

tabular, 44
 cellcolor, 46
 multicolumn, 48
 rotación, 47
 rowcolors, 45
 vrule, 49

V

Verbatim
 paquete fancyvrb, 24
VerbatimFootnote, 24
vspace, 25

W

WinShell, 4