



Curso Introductorio de \LaTeX

Instituto de Tecnología Nuclear Dan Beninson

30 de marzo de 2010

Índice

1. Introducción	6
2. Editores WYSIWIG vs. Latex	7
2.1. Esquema de trabajo con Latex: Compilador + Editor	8
2.2. Software para trabajar con Latex	9
2.2.1. Editores de Latex	9
2.2.2. Compiladores de Latex	10
2.2.3. Visores de Documentos DVI y PDF	10
2.3. Instalación del software	11
2.3.1. Instalación en Linux	11
2.3.2. Instalación en Windows	11
3. Estructura básica de un documento de Latex	12
3.1. Programación en Latex	12
3.2. Caracteres reservados de Latex	12
3.3. Comenzando con Latex	13
3.4. Estructura básica	13
3.5. Encabezado o preámbulo de un documento	15
3.5.1. El comando documentclass	15
3.5.2. Paquetes y utilidades del encabezado	16
3.6. Cuerpo del documento	18
3.7. En resumen: Encabezado + Cuerpo del documento	18
3.8. Tipos de comandos y su funcionamiento	18
3.8.1. Combinación de comandos	20
3.9. Primer ejemplo sencillo	20
3.9.1. Marcación básica de texto	21
3.9.2. Primeros formatos al texto	22
3.10. Segundo ejemplo sencillo	22
4. Documentos pre-formateados	23
4.1. Tercer ejemplo sencillo: Un libro con capítulos	24
5. Algunos comandos útiles	27
5.1. Comandos para el encabezado	27
5.2. Entornos dentro del documento	28
5.2.1. Listas: numeradas, con viñetas y de descripción simple	28
5.2.2. Centrado de texto, tablas e imágenes	29
5.3. Formato de la tipografía	29
5.4. Comillas y acentos	30
5.5. Comandos útiles para la edición del texto	31

6. Escritura y simbología matemática	31
6.1. Definición de entorno matemático	31
6.1.1. Fórmulas inmersas en el texto	32
6.1.2. Espacio matemático destacado	32
6.2. Expresiones más comunes	33
6.2.1. Agrupar elementos dentro del espacio matemático	33
6.2.2. Alfabeto griego	33
6.2.3. Potencias y raíces	34
6.2.4. Fracciones	34
6.2.5. Subíndices	35
6.2.6. Sumatorias, productorias e integrales	35
6.2.7. Paréntesis, corchetes y llaves	36
6.2.8. Matrices	37
6.2.9. Algunas expresiones matemáticas	39
6.2.10. Comentarios y edición dentro del espacio matemático	40
6.2.11. Definición de entornos dedicados: uso del comando <code>\newtheorem{...}{...}[...]</code>	40
7. Tablas	42
7.1. El entorno <code>\begin{tabbing} ... \end{tabbing}</code>	42
7.2. El entorno <code>\begin{tabular} ... \end{tabular}</code>	43
7.3. El entorno <code>\begin{table} ... \end{table}</code>	44
8. Inclusión de imágenes	46
8.0.1. Tipos de imagen y tipos de compilación	47
8.0.2. Inclusión de archivos PDF	47
8.0.3. Propiedades de una imagen en Latex	47
9. Referencias, notas al pie, citas y bibliografía	48
9.1. El entorno figure	48
9.2. Utilización de los comandos <code>\label{}</code> y <code>\ref{}</code>	49
9.3. Notas al pie de página con el comando <code>\footnote{}</code>	50
9.4. Bibliografía y citas	50
9.4.1. Bibliografía	50
9.4.2. Citas bibliográficas	52
9.4.3. Normas bibliográficas	52
10. Índices	52
10.1. Índice de contenidos	53
10.2. Índice de tablas	53
10.3. Índice de Figuras	53
11. Anexo: Símbolos	54
12. Anexo I: Ejemplos	54

Índice de Gráficos

1.	Editores wysiwig vs. Latex	7
2.	Esquema Básico	8
3.	Esquema del documento	14
4.	Ejemplo básico 1	20
5.	Ejemplo básico 2	21
6.	Ejemplo básico 3	22
7.	Ejemplo básico 4	23
8.	Tabla del Ejemplo 1	45
9.	Símbolos matemáticos varios	54
10.	Símbolos varios	55
11.	Flechas horizontales	55
12.	Flechas verticales	55
13.	Simbología de orden (símbolos matemáticos)	56
14.	Barras verticales y oblicuas	56
15.	Paréntesis, corchetes, llaves y otros	57
16.	Tipografía matemática	57
17.	Operadores p/conjuntos (símbolos matemáticos)	57
18.	Vectores (símbolos matemáticos)	58
19.	Espacios entre caracteres	58
20.	Alfabeto Griego	58

Índice de Tablas

1.	Propiedades y comandos de tablas	44
2.	Tabla del Ejemplo 2	46

Prólogo

El trabajo de edición de documentos con \LaTeX es esencialmente una tarea de programación. Para aquellas personas que alguna vez hayan programado, el aprendizaje de \LaTeX consistirá en la mera adquisición de un lenguaje simple —casi intuitivo— y de su estructura. Para no programadores el uso de \LaTeX será novedoso debido a que el modo de edición difiere de cualquier editor de textos convencional.

Este curso es inicial y está especialmente dirigido a personas sin experiencia en lenguajes de programación. El contenido ha sido diseñado siguiendo las formas estéticas más utilizadas, tratando de cubrir el mayor espectro de casos particulares con la menor cantidad de herramientas. No se ha hecho hincapié en modificaciones de diseño estructural de un documento, sino que se ha centrado la programación en la distribución del contenido dejando que \LaTeX resuelva toda la parte estética. Esto mostrará su enorme potencial, rápidamente.

Como el lenguaje es muy extenso y no es en absoluto práctico memorizar comandos, es común trabajar con un material como ayuda memoria. Este apunte pretende ser —en una primera instancia— dicho material de consulta. En la medida en que el usuario/programador vaya conociendo e interesándose en el uso de \LaTeX , necesitará material más completo y específico.

1. Introducción

\LaTeX ¹ es un conjunto de sentencias escritas en un lenguaje llamado \TeX . Nombre que se adopta también para denominar el intérprete de dicho lenguaje o compilador. Pero \TeX , el verdadero motor creado por Donald Ervin Knuth entre los años 1977 y 1978, no es un lenguaje de programación usual, sino uno orientado a la escritura de textos de excelente calidad. No se trata naturalmente de un editor de textos al estilo Microsoft Word o similares.

\TeX no es un editor de la familia WYSIWIG ("what you see is what you get"), acrónimo empleado para denominar a los editores que sólo trabajan sobre la pantalla dando un formato visual al texto, y en los cuales *lo que ves es lo que tenés*. Muy al contrario, en \TeX usted escribe el texto acompañado de órdenes que el compilador, posteriormente, interpreta y ejecuta para proporcionar un texto perfectamente compuesto. Un solo ejemplo bastará: en los editores WYSIWIG si desea escribir el símbolo ∇ tendrá que buscarlo en un menú que se puede denominar *Insertar | Símbolos* o algo similar; en \TeX debe escribir `∇` .

Leslie Lamport creó \LaTeX en 1982, con la intención de simplificar la tarea de aquéllos que desean utilizar \TeX . El prólogo de \LaTeX —A Document Preparation System, la obra en la que Lamport presenta su producto, comienza así:

\LaTeX (sistema para la preparación de documentos) es una versión especial de \TeX , programa creado por Donald Knuth. \TeX es un programa sofisticado diseñado para producir documentos de alta calidad, especialmente documentos matemáticos. \LaTeX añade a \TeX una colección de comandos que simplifican el mecanografiado, permitiendo al usuario concentrarse en la escritura del texto, en vez de en los comandos para dar formato. Al transformar \TeX en \LaTeX , he tratado de convertir un coche de carreras muy bien afinado en un comfortable sedán familiar. El sedán familiar no está pensado para ir tan rápido o ser excitante de conducir como el coche de carreras, pero es comfortable y lo transportará sin problemas. Sin embargo, \LaTeX tiene toda la potencia de \TeX escondida bajo el capó, y el conductor más aventurero puede hacer con él todo lo que puede hacer con \TeX .

Posteriormente \LaTeX ha ido actualizándose gracias al trabajo de un grupo de personas, reunidas en el denominado \LaTeX Team. Hoy en día \LaTeX es un programa de unas 8000 líneas de código \TeX , junto a un buen número de otros archivos con contenido relacionado.

\LaTeX es *free software*, esto implica que puede conseguirse su código fuente, modificarlo según las propias necesidades y utilizarlo libremente ya que se distribuye bajo licencia LPPL (LaTeX Project Public License) que permite su uso y distribución legal e ilimitado.

¹ Esta introducción es un extracto de la introducción de [Cascales Salinas et al., 2003, pag. V y VI]

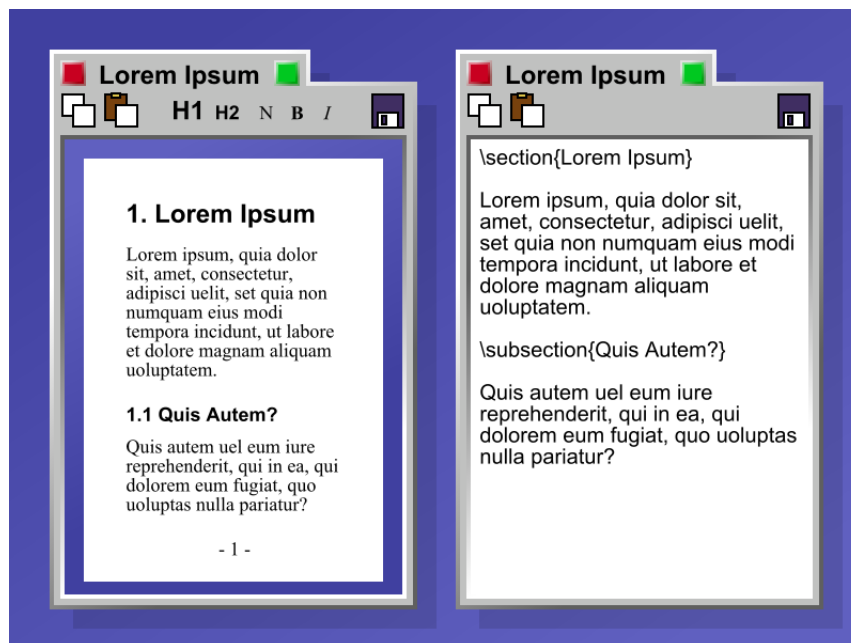


Figura 1: Editores wysiwig vs. Latex

2. Editores WYSIWIG vs. Latex

Como ya se ha dicho WYSIWYG es el acrónimo de *What You See Is What You Get* (en inglés, “lo que ves es lo que tenés”). Se aplica a los editores de *texto con formato* como el Word del Microsoft Office, el Writer del Open Office, editores como el Dreamweaver, editores de texto on line en internet, etc... que permiten *escribir un documento viendo directamente el resultado final*, frecuentemente el resultado impreso.

Se los llama así en contraposición a otros editores en los que se escribe sobre una vista que no muestra el formato sino hasta la impresión del documento. Usualmente en estos editores el texto no puede *enriquecerse*, es decir, no pueden incorporarse formatos de tamaño, tipo de carácter, alineación, etc.

Recalcamos entonces la diferencia fundamental entre los *editores de texto con formato* (Ej. el Word) y los *editores de texto plano* (Ej. el Bloc de Notas). El presente curso abordará la edición de documentos con Latex en editores de texto plano, es decir, *editores de texto sin formato* de modo que *el usuario no verá en el editor la imagen del documento final*, sino que tendrá que *transformarlo* mediante una operación muy sencilla, en el documento con formato –archivo DVI o PDF–, listo para imprimir. Dicho de otro modo, la disposición del texto dentro del editor no se condice con la disposición de dicho texto en el archivo final.

Esquema básico de trabajo de edición de documentos con LaTeX

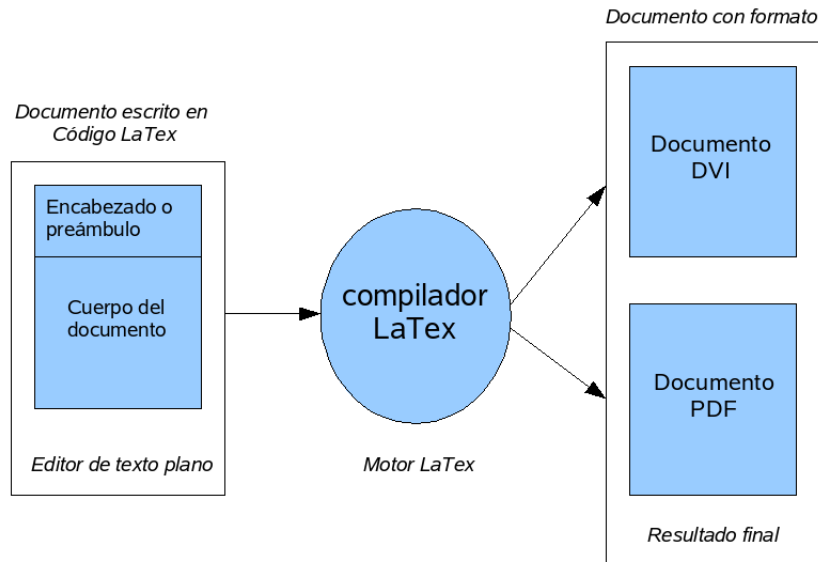


Figura 2: Esquema básico

2.1. Esquema de trabajo con Latex: Compilador + Editor

Latex es un *lenguaje* y un *compilador* que combinados conforman un completo sistema de edición.

Este *lenguaje* consiste básicamente en una serie de comandos, cada uno de ellos con un fin determinado. Podemos utilizarlos y combinarlos de muchas maneras para poder crear nuestro documento.

El *compilador* es la aplicación que lee el lenguaje escrito por nosotros y lo transforma en un documento con formato. Dicho de otro modo el compilador es el motor de nuestro sistema de edición.

Lo que hacemos al trabajar con Latex es escribir nuestro documento utilizando los comandos del lenguaje y luego procesarlo mediante el compilador o mejor dicho *compilarlo*.

Es decir, el usuario trabaja sobre el documento –o mejor dicho sobre el *código* del documento– en un editor de texto plano utilizando el lenguaje Latex. Una vez terminado se invoca al *compilador* y éste crea el documento final en formato DVI² o PDF. La figura 2 muestra gráficamente el esquema general de trabajo.

²DVI es el formato nativo de LaTeX y resulta muy parecido a los documentos PDF, mucho más difundidos.

Este punto es de vital importancia y reviste la mayor dificultad en el uso de edición de textos por ser este un modo de trabajo muy diferente de los editores convencionales.

2.2. Software para trabajar con Latex

Para trabajar con Latex es necesario contar con tres herramientas básicas:

- Un editor de texto plano donde escribir el código del documento
- El motor o compilador Latex propiamente dicho
- Un programa visor de documentos DVI o PDF

2.2.1. Editores de Latex

Aunque con un simple editor tipo Bloc de Notas nos bastaría para poder escribir libros con Latex, existen muchos editores muy avanzados que facilitan enormemente la tarea de edición.

Como lo que necesitamos es conocer y escribir en lenguaje Latex, estos editores nos proporcionan comandos que escriben el lenguaje por nosotros, lo cual nos evita no sólo tener que recordar comandos sino tener que escribirlos. El uso frecuente de este sistema hace que poco a poco vayamos recordando comandos básicos y de esa manera nos vayamos habituando a su uso y lectura.

Aquí proporcionamos una breve lista de editores de Latex para windows:

- Texmaker
- LED (gratis)
- Lyx (gratis)
- Winedt (pago)

Lista de editores Latex para Linux:

- Kile (gratis)
- Lyx (gratis)
- Emacs (gratis)
- TexMaker (gratis)

En general todos los editores son parecidos en aspecto y prestaciones. Su elección depende del gusto y costumbre del usuario.

Para este curso se adoptará el uso del editor Texmaker por ser de prestaciones estándar y por ser free software.

2.2.2. Compiladores de Latex

En el sitio www.latex-project.org se encuentra la documentación del compilador original para Unix/Linux. Si trabajamos en este sistema operativo no será necesario descargarlo ya que lo proveen todas (o casi todas) las distribuciones.

Para los usuarios de Windows existe el compilador *MikTeX* que no es otra cosa que el mismo compilador Latex escrito para este sistema. Está disponible su descarga gratuita en www.miktex.org.

2.2.3. Visores de Documentos DVI y PDF

Se ha dicho ya que el resultado final de la compilación del código Latex es un archivo tipo DVI o PDF. La primera ventaja de estos formatos es la calidad. Los DVI o PDF son archivos de imágenes tipo vectoriales, eso significa que conservan la misma definición a cualquier escala. Esta cualidad garantiza que su impresión, ya sea en impresoras o en imprenta, será de óptima calidad. Esta propiedad es importante ya que Latex se utiliza para editar artículos, reportes, tesis, libros, etc... todos publicables.

Este tipo de documentos tienen además una segunda la propiedad importante y es la de estar “cerrados”. Es decir, no es posible –en la mayoría de los casos– abrir DVI’s ni PDF’s para modificar su contenido³, con lo cual obtenemos como resultado un archivo seguro.

Visualización en Linux

La visualización de archivos DVI y PDF está completamente integrada a los sistemas gráficos de Linux (Gnome y KDE son los más conocidos) con lo que no necesitaremos instalar ningún visor específico.

Visualización en Windows

Para visualizar PDF’s contamos con el famoso *Acrobat Reader* que es freeware⁴.

Para visualizar DVI en windows contamos con el programa *YAP* que es una aplicación que viene incluida en el paquete *MikTeX* –el compilador de Latex–, con lo cual no nos hace falta instalarlo.

En resumen si trabajamos en Linux o Windows casi no necesitamos instalar ninguna aplicación para visualizar los DVI o PDF, salvo el Acrobat Reader en windows, que en la mayoría de los casos se encuentra instalado.

Más adelante veremos cuáles son las ventajas y desventajas de compilar en DVI y/o en PDF.

³Este punto puede ser discutible ya que existen en la actualidad editores de archivos PDF en los cuales pueden modificarse algunas características y tipos. Ello depende de cómo fue generado dicho PDF.

⁴Puede descargarse gratuitamente desde su sitio oficial www.adobe.com/es/

2.3. Instalación del software

Se ha dicho ya que todos los programas que se utilizarán para el dictado de este curso son “free software”. Este punto es importante ya que todas las herramientas que utilizaremos para trabajar tienen licencia para su uso legal e ilimitado.

Es necesario pues instalar:

- Un editor
- El compilador de Latex
- Un visor PDF si hiciera falta

2.3.1. Instalación en Linux

Por lo general casi todas las distribuciones Linux incluyen el compilador Latex con lo cual sólo habría que verificarlo en nuestra distribución.

Además los sistemas gráficos Gnome y KDE –incluidos en casi todas las distribuciones Linux– incluyen editores con funcionalidades para Latex y visores de DVI y PDF, con lo cual, en la mayoría de los casos, no hacen falta instalar.

Es decir, los sistemas Linux por lo general incorporan por defecto todas las aplicaciones necesarias para la edición con Latex.

2.3.2. Instalación en Windows

A diferencia de Linux, Windows no trae aplicaciones para Latex con lo cual deberemos instalar:

- El motor MikTeX (que incluirá el visor YAP para documentos DVI)
- El editor Texmaker⁵
- El Acrobat Reader para documentos PDF si hiciera falta

⁵Puede descargarse directamente desde <http://www.xm1math.net/texmaker/download.html>

3. Estructura básica de un documento de Latex

3.1. Programación en Latex

Latex es un lenguaje que posee una estructura determinada y términos propios. Si no respetáramos su estructura o bien escribiéramos mal sus términos, el compilador Latex no podría transformar nuestro código en el documento final ya que *el lenguaje estaría mal escrito y el compilador no lo comprendería*. Como consecuencia de esto el sistema envía un error. En virtud de poder entender qué sucede cuando ocurren errores, es necesario pues entender someramente cómo trabaja el motor de Latex.

Se ha dicho que aquello que nosotros escribimos en nuestro editor será interpretado por el compilador. Esta aplicación leerá el documento en forma ascendente de izquierda a derecha, línea por línea. Es decir, comenzará por el primer carácter de la primera línea, luego por el segundo, hasta llegar al final. Seguirá pues con la segunda línea de la misma manera, carácter por carácter, y así hasta llegar al final de nuestro documento. Nunca retrocederá.

Si durante este *camino* el compilador encuentra algún error –por lo general algún comando mal escrito o algún carácter especial mal utilizado– tratará pues de seguir adelante y generar el DVI o PDF aunque se muestre con errores. Si no puede, se frenará indicando en qué línea se encontró dicho error y no generará el archivo con formato. El compilador informará siempre si hubo algún error y en qué línea del programa se ha producido para poder buscarlo fácilmente y corregirlo.

3.2. Caracteres reservados de Latex

En el lenguaje Latex hay *caracteres reservados*, es decir, caracteres que no pueden utilizarse directamente porque se han definido como comandos. Luego para evitar errores es necesario saber cuáles son y cuál es su función:

#: Se utiliza para nombrar los argumentos de un comando.

\$: Crea un espacio matemático

?: Comentarios en documento

&: Usado para separar columnas en tablas o matrices.

_: sub-índice

{ }: Abre y cierra entornos

~: Se utiliza para evitar la separación de palabras en un cambio de renglón.
Carácter para la ñ.

^: Super-índice

\: Marca el inicio de un comando

Para poder incluir estos caracteres dentro del documento, como parte del texto debe anteponérsele un *backslash* (una barra \) o se debe utilizar el comando `\verb' '`, por ejemplo:

- Para obtener # , escribe \#
- Para obtener \$, escribe \\$
- Para obtener % , escribe \%
- Para obtener & , escribe \&
- Para obtener _ , escribe _
- Para obtener { ó } , escribe \{ ó \}
- Para obtener ~ , escribe \verb'~'
- Para obtener ^ , escribe \verb'^'
- Para obtener \ , escribe \verb'\'

3.3. Comenzando con Latex

Ahora que tenemos un editor de texto (el Texmaker), el motor Latex instalado y un visor DVI o PDF y además sabemos que la escritura de nuestro documento debe ajustarse a determinadas reglas del lenguaje Latex, podemos empezar a trabajar.

Nuestro entorno de trabajo será siempre el editor Texmaker. Desde aquí escribiremos el documento, lo compilaremos y veremos su resultado.

Notación

Para que este apunte resulte más claro estableceremos una notación:

- El texto en formato máquina de escribir será utilizado para representar el código de nuestro documento, es decir lo que se escribe en el editor Texmaker.
- A menudo se utilizará el texto normal, quizás en itálica, para denotar el resultado final, es decir el formato en PDF o DVI.

3.4. Estructura básica

Todos los documentos Latex se dividen en dos grandes partes:

- **Encabezado o preámbulo del documento:**

Es el lugar en donde se define todo el formato del documento. A saber:

- Tipo de documento: si es un libro, artículo, reporte, tesis, etc...⁶

⁶Este punto es importante y se explicará en detalle más adelante

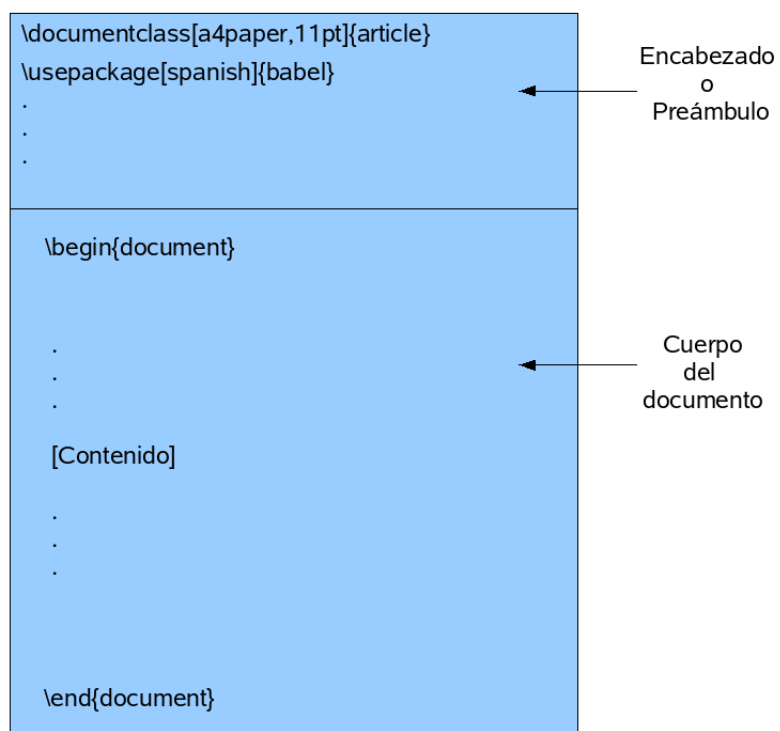


Figura 3: Esquema del documento

- Tamaño de la hoja, tamaño y tipo de letra
- Idioma: El idioma por defecto es el inglés, pero puede configurarse para muchos otros entre los que se encuentra por supuesto el español.
- Encabezado y pie de página
- Autor, título, fecha
- Tipos de imágenes que se van a incorporar al documento
- etc ...

El comienzo del encabezado está determinado por el comando

```
\documentclass[]{}{}
```

■ **El documento propiamente dicho:**

Aquí es donde se escribe el contenido. El comienzo y fin del documento se define por medio del comando

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

3.5. Encabezado o preámbulo de un documento

El encabezado de un documento Latex consiste en una serie de comandos que se incorporan por única vez y que usualmente no se modifican. Por lo general el formato se define una sola vez al principio y no sufre luego grandes modificaciones. Tendremos aquí una breve explicación de cuáles son las herramientas que necesitamos para construir un encabezado sencillo.

La mecánica de trabajo es la de establecer un formato general y aquellos formatos que queremos que sean diferentes deberemos modificarlos particularmente. Así este procedimiento establece un criterio estético general para todo el documento.

3.5.1. El comando `documentclass`

```
\documentclass[]{}{}
```

Este comando es lo primero que en un documento Latex debe aparecer. En él se definen entre otras cosas el tipo de documento, el tamaño de la hoja y tamaño de la letra de la siguiente manera:

```
\documentclass[a4paper,11pt]{article}
```

Podemos editar sus propiedades y modificar el tamaño de la letra simplemente cambiando *11* por *12* o por el tamaño que plazca, o cambiando el tipo de papel *a4paper* por *letter* o modificando el tipo de documento *article* por *book*. Este cambio afectará a todo el documento.

```
\documentclass[letter,12pt]{book}
```

3.5.2. Paquetes y utilidades del encabezado

Latex posee una gran cantidad de paquetes que pueden incorporarse en el encabezado y que poseen funcionalidades diversas. Por ejemplo:

Se ha dicho ya que el idioma por defecto es el inglés. Si empezáramos a trabajar en español notaríamos que el resultado final no sería el apropiado. Luego para trabajar en nuestro idioma hay que indicarle al compilador qué paquete de idioma debe utilizar.

Esto se define en el encabezado de la siguiente manera:

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

Así el encabezado estará lleno de comandos tipo `\usepackage[...]{...}` que determinarán los paquetes que se utilizarán según las necesidades de edición del texto.

A continuación se muestra un encabezado típico (bastante completo) con un breve comentario que indica cuál es su utilidad:

```
%TIPO DE DOCUMENTO
\documentclass[a4paper,11pt]{article}

%PAQUETE PARA CORRECCION DE TIPOGRAFIA PARA ESPAÑOL
\usepackage[utf8]{inputenc}

%PAQUETE DICCIONARIO ESPAÑOL
\usepackage[spanish]{babel}

%PAQUETES COMPLEMENTARIOS PARA TIPOGRAFIA MATEMATICA
\usepackage{amsmath,amssymb,amsfonts,textcomp}

%PAQUETE PARA LA INCLUSION DE IMAGENES EPS
\usepackage{graphicx}

%PAQUETE PARA LA INCLUSION DE IMAGENES jpg, png Y pdf
\usepackage[pdftex]{graphicx}

%PAQUETE PARA ESTILO DE PAGINA
\usepackage{fancyhdr}

%DEFINICION DEL ESTILO DE LA PAGINA
\pagestyle{fancy}

%ENCABEZADO A LA IZQUIERDA SEGUN ESTILO fancy
\fancyhead[L]{\slshape \leftmark}

%ENCABEZADO A LA DERECHA SEGUN ESTILO fancy
\fancyhead[R]{\slshape \thepage}
```



```
%ENCABEZADO AL CENTRO SEGUN ESTILO fancy
\fancyfoot[C]{\slshape \author}

%ESPACIO DE INTERLINEADO EN MILIMETROS
\renewcommand{\baselinestretch}{1.2}

%AUTOR DEL DOCUMENTO
\author{Instituto de Tecnología Nuclear Dan Beninson}

%FECHA DE EDICION DEL DOCUMENTO
\date{\today}

%TITULO DEL DOCUMENTO
\title{Curso Introductorio de Latex}

%Fin del preambulo y comienzo del documento
```

Sin los comentarios⁷ el preámbulo se reduce a unas pocas líneas:

```
\documentclass[a4paper,11pt]{article}
\usepackage[utf8]{inputenc}
\usepackage[spanish]{babel}
\usepackage{amsmath,amssymb,amsfonts,textcomp}
\usepackage{graphicx}
\usepackage[pdftex]{graphicx}
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhead[L]{\slshape \leftmark}
\fancyhead[R]{\slshape \thepage}
\fancyfoot[C]{\slshape \author}
\renewcommand{\baselinestretch}{1.2}
\author{Instituto de Tecnología Nuclear Dan Beninson}
\date{\today}
\title{Curso Introductorio de Latex}
%Fin del preambulo y comienzo del documento
```

En la práctica lo que habitualmente se hace es configurar el preámbulo una sola vez y utilizarlo siempre, con pequeñas modificaciones. Para nuestro documento utilizaremos un preámbulo más reducido:

```
\documentclass[a4paper,11pt]{article}
\usepackage[utf8]{inputenc}
```

⁷Nótese que los comentarios están siempre precedidos del carácter%. Éste carácter es especial y tiene la cualidad de que todo texto que esté detrás de% será ignorado por el compilador. Así podemos incorporar texto dentro de la programación que no queremos que sea incluido en el documento final y que no queremos que sea tomado en cuenta por la compilación del documento.

```
\usepackage[spanish]{babel}
\usepackage{amssymb}
\usepackage{graphicx}\usepackage[pdftex]{graphicx}
\pagestyle{headings}
\author{Instituto de Tecnología Nuclear Dan Beninson}
\date{\today}
\title{Curso Introductorio de Latex}
%Fin del preambulo y comienzo del documento
```

3.6. Cuerpo del documento

Finalmente podremos ahora empezar a trabajar en nuestro contenido. El comando `\begin{document} ... \end{document}` es el que define el *entorno* donde incorporar el contenido propiamente dicho de nuestro documento.

3.7. En resumen: Encabezado + Cuerpo del documento

Nuestro documento se compone entonces de la siguiente manera:

```
%Comienzo del encabezado

\documentclass[a4paper,11pt]{article}
\usepackage[utf8]{inputenc}
\usepackage[spanish]{babel}
\usepackage{amssymb}
\usepackage{graphicx}\usepackage[pdftex]{graphicx}
\pagestyle{headings}
\author{Instituto de Tecnología Nuclear Dan Beninson}
\date{\today}
\title{Curso Introductorio de Latex}

%Fin del encabezado y comienzo del documento

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

3.8. Tipos de comandos y su funcionamiento

Aunque ya se han mostrado algunos comandos daremos aquí una forma general. Latex admite tres tipos de comandos:

- `\comando`:

Comandos simples que no necesitan parámetros. Por ejemplo:

`\maketitle`: confecciona un título con los datos del preámbulo

`\bigskip`: deja un espacio grande entre dos renglones

`\item`: comando utilizado para entornos de numeración o descripción.

- `\comando{ }:`

Estos comandos sí necesitan al menos un parámetro, por ejemplo:

`\section{Nombre de la sección}`: Crea una sección dentro de nuestro documento

`\usepackage{paquete}`: incorpora un paquete a nuestro encabezado

`\includegraphics[escala]{ruta de la imagen}`: incorpora una imagen a nuestro documento

Nótese que hay comandos a los que se les puede pasar parámetros adicionales. En el caso de `\includegraphics` puede agregarse un factor de escala a la imagen para ajustar su tamaño.

- `\begin{ } ... \end{ }:`

Por lo general todo comando de esta forma crea un *entorno* dentro del documento. Si hay escrito un `\begin{ }`, DEBE estar su correspondiente `\end{ }`, si no el compilador arrojará un error. *No puede quedar un entorno abierto* en nuestro documento.

Ejemplos:

```
\begin{center}
```

```
Texto centrado
```

```
\end{center}
```

```
\begin{itemize}
```

```
\item Espacio con viñetas
```

```
\end{itemize}
```

- Entornos matemáticos `$... $` ó `$$... $$`:

Para la edición de fórmulas o contenido en lenguaje matemático es necesario definir un espacio específico para ello. Deben utilizarse los siguientes caracteres:

- `$ Fórmula $`: para incorporar una fórmula al texto
- `$$ Fórmula $$`: para que la fórmula aparezca destacada.

```

\documentclass[a4paper,11pt]{report}%TIPO DE DOCUMENTO
\usepackage[utf8]{inputenc}%PAQUETE PARA CORRECCION DE TIPOGRAFIA PARA ESPAÑOL
\usepackage[spanish]{babel}%PAQUETE DICCIONARIO ESPAÑOL
%-----
%Fin del preambulo y comienzo del documento
%-----

\begin{document}

\chapter{Viajes}

    Cuando los famas salen de viaje, sus costumbres al pernoctar en una ciudad son las siguientes: Un fama va al hotel y
    averigua cautelosamente los precios, la calidad de las sábanas y el color de las alfombras. El segundo se traslada a la
    comisaría y labra un acta declarando los muebles e inmuebles de los tres, así como el inventario del contenido de sus
    valijas. El tercer fama va al hospital y copia las listas de los médicos de guardia y sus especialidades.
    Terminadas estas diligencias, los viajeros se reúnen en la plaza mayor de la ciudad, se comunican sus observaciones, y
    entran en el café a beber un aperitivo. Pero antes se toman de las manos y danzan en ronda. Esta danza recibe el nombre de
    "Alegria de los famas".
    Cuando los cronopios van de viaje, encuentran los hoteles llenos, los trenes ya se han marchado, llueve a gritos, y los taxis
    no quieren llevarlos o les cobran precios altísimos. Los cronopios no se desaniman porque creen firmemente que estas cosas les
    ocurren a todos, y a la hora de dormir se dicen unos a otros: "La hermosa ciudad, la hermosísima ciudad". Y sueñan toda la
    noche que en la ciudad hay grandes fiestas y que ellos están invitados. Al otro día se levantan contentísimos, y así es como
    viajan los cronopios.
    Las esperanzas, sedentarias, se dejan viajar por las cosas y los hombres, y son como las estatuas que hay que ir a verlas
    porque ellas ni se molestan.

\end{document}

```

Figura 4: Ejemplo básico 1

3.8.1. Combinación de comandos

Una herramienta importante es la posibilidad de combinar comandos uno dentro de otro.

Por ejemplo:

- Para que una palabra aparezca en negrita deberemos escribir:

```
\textbf{palabras en negrita}
```

- Para aparezca en negrita e itálica:

```
\textit{\textbf{palabras en negrita e itálica}}
```

- Para que aparezca en negrita, itálica y centrado:

```

\begin{center}
\textit{\textbf{palabras en negrita, itálica y centradas}}
\end{center}

```

De esta manera pueden anidarse comandos indiscriminadamente, siempre y cuando se respeten las reglas de cada uno de ellos según se mostró anteriormente.

3.9. Primer ejemplo sencillo

La figura 4 muestra un código sencillo cuyo resultado compilado está representado por la figura 5.

Si nos fijamos en su correspondiente encabezado, entenderemos que se trata de un archivo de tamaño *a4* con un tamaño de letra *11* y un documento

Capítulo 1

Viajes

Cuando los famas salen de viaje, sus costumbres al pernoctar en una ciudad son las siguientes: Un fama va al hotel y averigua cautelosamente los precios, la calidad de las sábanas y el color de las alfombras. El segundo se traslada a la comisaría y labra un acta declarando los muebles e inmuebles de los tres, así como el inventario del contenido de sus valijas. El tercer fama va al hospital y copia las listas de los médicos de guardia y sus especialidades. Terminadas estas diligencias, los viajeros se reúnen en la plaza mayor de la ciudad, se comunican sus observaciones, y entran en el café a beber un aperitivo. Pero antes se toman de las manos y danzan en ronda. Esta danza recibe el nombre de ⁴“alegría de los famas”. Cuando los cronopios van de viaje, encuentran los hoteles llenos, los trenes ya se han marchado, llueve a gritos, y los taxis no quieren llevarlos o les cobran precios altísimos. Los cronopios no se desaniman porque creen firmemente que estas cosas les ocurren a todos, y a la hora de dormir se dicen unos a otros: “La hermosa ciudad, la hermosísima ciudad”. Y sueñan toda la noche que en la ciudad hay grandes fiestas y que ellos están invitados. Al otro día se levantan contentísimos, y así es como viajan los cronopios. Las esperanzas, sedentarias, se dejan viajar por las cosas y los hombres, y son como las estatuas que hay que ir a verlas porque ellas ni se molestan.

Figura 5: Resultado de la compilación del código de la figura 4

tipo *report*. Notemos también que está configurado el idioma español y se ha agregado el paquete para el manejo de caracteres de este idioma⁸.

3.9.1. Marcación básica de texto

Espacios entre palabras

Si en la escritura dejamos más de un espacio entre palabras (golpes de barra espaciadora), el compilador eliminará todos los espacios dejando sólo uno. Si quisiéramos agregar espacios entre palabras debemos utilizar la barra \ entre golpes de barra espaciadora.

Ejemplo:

código en Texmaker:

```
Un fama va al hotel y averigua    cautelosamente
```

Resultado en PDF/DVI:

Un fama va al hotel y averigua cautelosamente

⁸Por defecto el lenguaje de Latex es el inglés, para poder escribir nuestro documento en español es necesario indicarle al compilador que utilizaremos en nuestro editor Texmaker caracteres que no existen en ese idioma como las vocales acentuadas, la ñ y la diéresis.

```

\documentclass[a4paper,11pt]{report}%TIPO DE DOCUMENTO
\usepackage[utf8]{inputenc}%PAQUETE PARA CORRECCION DE TIPOGRAFIA PARA ESPAÑOL
\usepackage[spanish]{babel}%PAQUETE DICCIONARIO ESPAÑOL
%-----
%Fin del preambulo y comienzo del documento
%-----

\begin{document}

\chapter{Viajes}

Cuando los \textbf{famas} salen de viaje, sus costumbres al pernoctar en una ciudad son las siguientes: Un fama va al hotel y
averigua cautelosamente los precios, la calidad de las sábanas y el color de las alfombras. El segundo se traslada a la comisaría
y labra un acta declarando los muebles e inmuebles de los tres, así como el inventario del contenido de sus valijas. El tercer
fama va al hospital y copia las listas de los médicos de guardia y sus especialidades.

\textit{Terminadas estas diligencias, los viajeros se reúnen en la plaza mayor de la ciudad, se comunican sus observaciones, y
entran en el café a beber un aperitivo. Pero antes se toman de las manos y danzan en ronda. Esta danza recibe el nombre de
"Alegría de los famas".}

Cuando los cronopios van de viaje, encuentran los hoteles llenos, los trenes ya se han marchado, llueve a gritos, y los taxis no
quieren llevarlos o les cobran precios altísimos. Los cronopios no se desaniman porque creen firmemente que estas cosas les
ocurren a todos, y a la hora de dormir se dicen unos a otros: "\underline{La hermosa ciudad, la hermosísima ciudad}". Y sueñan
toda la noche que en la ciudad hay grandes fiestas y que ellos están invitados. Al otro día se levantan contentísimos, y así es
como viajan los cronopios.

\bigskip

\begin{center}
\textit{Las esperanzas, sedentarias, se dejan viajar por las cosas y los hombres, y son como las estatuas que hay que ir a
verlas porque ellas ni se molestan.}
\end{center}

\end{document}

```

Figura 6: Segundo ejemplo básico

Espacios entre párrafos

La separación entre párrafos está determinada por un renglón vacío, es decir, no basta con dar *enter* y enviar el texto al renglón inferior, sino que debe existir al menos una línea vacía.

Análogamente al uso de los espacios entre palabras, si entre dos párrafos existieran muchas líneas vacías (renglones), el compilador asumirá que sólo hay una. Si quisiéramos dejar más espacio entre párrafos deberemos utilizar el comando `\bigskip`.

3.9.2. Primeros formatos al texto

A continuación se muestran los comandos básicos para dar formato al texto:

- **Negrita:** `\textbf{Negrita}`
- Subrayado: `\underline{Subrayado}`
- *Italica:* `\textit{Italica}`
- Maquina de escribir: `\texttt{Maquina de escribir}`

3.10. Segundo ejemplo sencillo

La figura 6 muestra el esquema general de un documento típico modificado con algunos elementos de formato.

El resultado de la compilación de este código corresponde a la figura 7.

Capítulo 1

Viajes

Cuando los **famas** salen de viaje, sus costumbres al pernoctar en una ciudad son las siguientes: Un fama va al hotel y averigua cautelosamente los precios, la calidad de las sábanas y el color de las alfombras. El segundo se traslada a la comisaría y labra un acta declarando los muebles e inmuebles de los tres, así como el inventario del contenido de sus valijas. El tercer fama va al hospital y copia las listas de los médicos de guardia y sus especialidades.

Terminadas estas diligencias, los viajeros se reúnen en la plaza mayor de la ciudad, se comunican sus observaciones, y entran en el café a beber un aperitivo. Pero antes se toman de las manos y danzan en ronda. Esta danza recibe el nombre de .Alegría de los famas”.

Cuando los cronopios van de viaje, encuentran los hoteles llenos, los trenes ya se han marchado, llueve a gritos, y los taxis no quieren llevarlos o les cobran precios altísimos. Los cronopios no se desaniman porque creen firmemente que estas cosas les ocurren a todos, y a la hora de dormir se dicen unos a otros: “La hermosa ciudad, la hermosísima ciudad”. Y sueñan toda la noche que en la ciudad hay grandes fiestas y que ellos están invitados. Al otro día se levantan contentísimos, y así es como viajan los cronopios.

Las esperanzas, sedentarias, se dejan viajar por las cosas y los hombres, y son como las estatuas que hay que ir a verlas porque ellas ni se molestan.

Figura 7: Resultado de la compilación del código de la figura 6

4. Documentos pre-formateados

En Latex existen ya algunos formatos típicos de documentos. Los que veremos aquí son:

- book y report
- article

Estos formatos son los que se definen en la primera línea y poseen importantes características de edición. Aquí es donde empezamos a ver las verdaderas ventajas del uso de Latex ya que al indicar en el `\documentclass[] {book}`, `\documentclass[] {report}` o `\documentclass[] {article}` le estamos indicando que configure no sólo cuestiones de formato sino de estructura del contenido. Por ejemplo los libros (book) poseen en general estructuras jerárquicas de administración del contenido, a saber: tomos, partes, capítulos, secciones, etc..., pues eso es lo que precisamos cuando indicamos book en el comando `\documentclass[] {book}`. El formato report es muy parecido al formato book.

A continuación se muestran las posibles divisiones de contenidos para cada tipo de documento:

- book y report: `\part{}`, `\chapter{}`, `\section{}`, `\subsection{}`,

`\subsubsection{}`, `\paragraph{}`, `\subparagraph{}`.

- Article: `\section{}`, `\subsection{}`, `\subsubsection{}`, `\paragraph{}`, `\subparagraph{}`.

4.1. Tercer ejemplo sencillo: Un libro con capítulos

En los ejemplos anteriores puede verse que ya habíamos incorporado el comando `\chapter{Viajes}` sin haberlo mencionado, y cuya compilación arrojó el texto **Capítulo 1**. Podemos agregar todos los capítulos que necesitemos y el compilador realizará el resto.

```
\documentclass[a4paper,11pt]{report}%TIPO DE DOCUMENTO
\usepackage[utf8]{inputenc}%PAQUETE PARA CORRECCION DE TIPOGRAFIA PARA ESPAÑOL
\usepackage[spanish]{babel}%PAQUETE DICCIONARIO ESPAÑOL
%-----
%Fin del preambulo y comienzo del documento
%-----

\begin{document}

\chapter{Viajes}

    Cuando los famas salen de viaje, sus costumbres al pernoctar en una ciudad son las siguientes: Un fama va al hotel y averigua
    cuidadosamente los precios, la calidad de las sábanas y el color de las alfombras. El segundo se traslada a la comisaría y
    labra un acta declarando los muebles e inmuebles de los tres, así como el inventario del contenido de sus valijas. El tercer
    fama va al hospital y copia las listas de los médicos de guardia y sus especialidades...

\chapter{Inconvenientes en los servicios públicos}

    Vea lo que pasa cuando se confía en los cronopios. Apenas lo habían nombrado Director General de Radiodifusión, este cronopio
    llamó a unos traductores de la calle San Martín y les hizo traducir todos los textos, avisos y canciones al rumano, lengua no
    muy popular en la Argentina. ...

\chapter{La foto salió movida}

    Un cronopio va a abrir la puerta de calle, y al meter la mano en el bolsillo para sacar la llave lo que saca es una caja de
    fósforos, entonces este cronopio se aflige mucho y empieza a pensar que si en vez de la llave encuentra los fósforos, sería
    horrible que el mundo se hubiera desplazado ...

\end{document}
```

el resultado de la compilación de este código es el siguiente:

<p>Capítulo 1</p> <p>Viajes</p> <p>Cuando los famas salen de viaje, sus costumbres al permanecer en una ciudad son las siguientes: Un fama va al hotel y averigua exhaustivamente los precios, la calidad de las sábanas y el color de las alfombras. El segundo se traslada a la comisaría y habla su acta declarando los nombres e iniciales de los tres, así como el inventario del contenido de sus valijas. El tercer fama va al hospital y copia las listas de los médicos de guardia y sus especialidades...</p> <p>1</p>	<p>Capítulo 2</p> <p>Inconvenientes en los servicios públicos</p> <p>Via lo que pasa cuando se confía en los cronopios. Apenas lo habían nombrado Director General de Radioffusión, este cronopio llamó a unos traductores de la calle San Martín y los hizo traducir todos los textos, avisos y canciones al rumano, lengua no muy popular en la Argentina. ...</p> <p>2</p>
<p>Capítulo 3</p> <p>...</p>	

Un ejemplo de un documento tipo `article` podría ordenarse de la siguiente manera:

```
%documentclass[a4paper,11pt]{article}%TIPO DE DOCUMENTO
\usepackage[utf8]{inputenc}%PAQUETE PARA CORRECCION DE TIPOGRAFIA PARA ESPAÑOL
\usepackage[spanish]{babel}%PAQUETE DICCIONARIO ESPAÑOL
%-----
%Fin del preambulo y comienzo del documento
%-----

\begin{document}

\section{Sección Primera}
La sección primera se clasifica en dos subsecciones:

.      \subsection{Subsección primera}
.      .      \subsubsection{sub-subsección}
.      .      .      \paragraph{Párrafo}
.      .      .      \subparagraph{Sub Párrafo}

.      \subsection{Subsección segunda}
.      .      \subsubsection{sub-subsección}
.      .      .      \paragraph{Párrafo}
.      .      .      \subparagraph{Sub Párrafo}

\section{Sección Segunda}
La sección segunda se clasifica en dos subsecciones:

.      \subsection{Subsección}
.      .      \subsubsection{sub-subsección}
.      .      .      \paragraph{Párrafo}
.      .      .      \subparagraph{Sub Párrafo}

.      \subsection{Subsección}
.      .      \subsubsection{sub-subsección}
.      .      .      \paragraph{Párrafo}
.      .      .      \subparagraph{Sub Párrafo}

\end{document}
```

el resultado de la compilación de este código es el siguiente:

1. Sección Primera

La sección primera se clasifica en dos subsecciones:

1.1. Subsección primera

1.1.1. sub-subsección

Párrafo

Sub Párrafo

1.2. Subsección segunda

1.2.1. sub-subsección

Párrafo

Sub Párrafo

2. Sección Segunda

La sección segunda se clasifica en dos subsecciones:

2.1. Subsección

2.1.1. sub-subsección

Párrafo

Sub Párrafo

2.2. Subsección

2.2.1. sub-subsección

Párrafo

Sub Párrafo

Nótese cómo el compilador realiza automáticamente la numeración de las secciones y sub secciones.

Si se desea quitar la numeración puede utilizarse el * antes de las llaves de la siguiente manera:

```
\section*{Viajes} o \subsection*{Viajes}, etc ...
```

5. Algunos comandos útiles

En esta etapa inicial lo más recomendable es modificar lo menos posible las cuestiones de estilo y formato para que el compilador se encargue de todo. Por lo general las configuraciones por defecto del sistema resolverán muy eficientemente todo el documento.

A continuación mostraremos una serie variada de comandos de uso cotidiano para edición del texto:

5.1. Comandos para el encabezado

- `\author{}`: Setea el autor del documento.
- `\title{}`: Setea el título del documento.
- `\date{}`: Setea la fecha de edición del documento.
- `\pagenumbering{<estilo>}`: establece la numeración de páginas convencional. Los estilos pueden ser:
 - `arabic`: 1, 2, 3, ...
 - `roman`: i, ii, iii, iv, ...
 - `Roman`: I, II, III, IV, ...
 - `alph`: a, b, c, ...
 - `Alph`: A, B, C, ...
- `\pagestyle{<estilo>}`: Establece un formato integral. Algunos estilos pueden ser:
 - `plain`: Indica que el número de página va en el final y que el encabezado es vacío, ésta es la opción por omisión en el caso de article y report.
 - `empty`: Indica que tanto el encabezado como el pie de página van vacíos.
 - `headings`: El número de página y cualquier otra información, que ha sido determinada en el documentclass va en el encabezado, y el pie de página va vacío.
 - `fancy`: Es posible configurar información adicional. Requiere la carga previa del paquete `\usepackage{fancy}`.

5.2. Entornos dentro del documento

5.2.1. Listas: numeradas, con viñetas y de descripción simple

Contamos con los siguientes espacios para realizar listas:

Para viñetas:

```
\begin{itemize}
\item Dato 1
\item Dato 2
\item Dato 3
\end{itemize}
```

Para espacios con numeración:

```
\begin{enumerate}
\item Dato 1
\item Dato 2
\item Dato 3
\end{enumerate}
```

Para descripciones simples:

```
\begin{description}
\item Dato 1
\item Dato 2
\item Dato 3
\end{description}
```

Ejemplo de uso para un espacio con viñetas:

Código en Texmaker:

```
\begin{itemize}
\item viñeta 1
\item viñeta 2
\item viñeta 3
\end{itemize}
```

resultado compilado:

- viñeta 1
- viñeta 2
- viñeta 3

Los espacios pueden anidarse uno dentro de otro. El sistema respetará para todos los casos el orden tabular y las numeraciones correspondientes.

5.2.2. Centrado de texto, tablas e imágenes

Por defecto el sistema justifica el texto de manera inteligente cortando las palabras mediante el uso del diccionario, con lo cual –por lo general– no hace falta alinearlo. Sin embargo para casos en los que se quiera centrar texto, imágenes o tablas, puede utilizarse el siguiente entorno:

```
\begin{center}
<texto, imágenes o tablas>
\end{center}
```

5.3. Formato de la tipografía

Ya se ha mostrado en la sección 3.9.2 como transformar texto en negrita, itálica, subrayado y tipo máquina de escribir. Ahora veremos cómo modificar el estilo y tamaño:

- **Estilos:** Pueden aplicarse los siguientes estilos a cualquier porción de texto dentro de un espacio `\begin{}`... `\end{}`:

<code>\rm</code>	roman
<code>\em</code>	italic
<code>\bf</code>	boldface
<code>\sl</code>	slanted
<code>\sf</code>	sans serif
<code>\sc</code>	small caps
<code>\tt</code>	typewriter

```
\begin{\sf}
Texto con letra sans serif
\end{\sf}
```

- **Tamaño:** El tamaño del texto se configura en el encabezado. Eso no implica que todo el texto esté en ese tamaño ya que por defecto los títulos de los capítulos, secciones, subsecciones, etc... tienen un tamaño ya definido y que está acorde al número que se ingresa en el encabezado. Es decir, cuando ingresamos `\documentclass[a4paper,10pt]{article}` esto define en realidad un rango de tamaños de letras.

Si se desea realizar cambios particulares, Latex tiene preconfigurados diferentes tamaños que pueden utilizarse en cualquier porción del texto. La forma de utilizarla es mediante un espacio tipo `\begin{}`... `\end{}`. Los tamaños están representados por los siguientes comandos:

```
\tiny
```

```

\scriptsize
\footnotesize
\small
\normalsize
\large
\Large
\LARGE

\begin{\Large}
Texto con tamaño \Large
\end{\Large}

```

5.4. Comillas y acentos

Las comillas (") y apóstrofes (') en Latex no son caracteres especiales, por tanto no es un problema incluirlos en el texto. Sin embargo para poner acentos y ñ es necesario poner un apóstrofo (') antes de la vocal a acentuar o de la n, por ejemplo:

```
\'o \'A \'n \'N
```

Da como resultado

ó Á ñ Ñ

Otro ejemplo puede ser el siguiente:

```
H\^otel, na\"ive, \'el\'eve,\\
sm\o rrebr\o d, !\'Se\'norita!,\\
Sch\"onbrunner Schlo\ss{} Stra\ss e
```

que genera la siguiente salida:

Hôtel, naïve, élève,
smørrebrød, !Señorita!,
Schönbrunner Schloß Straße

Reconocemos que la escritura de acentos como se presenta aquí es realmente incómoda ya que estamos acostumbrados a incorporar desde el teclado los acentos correspondientes. Esto lo podemos solucionar fácilmente si incorporamos a nuestro enabezado el paquete `\usepackage[latin1]{inputenc}`⁹.

⁹Si se trabaja en Linux es frecuente la opción *latin1* —que corresponde a la codificación de archivos trabajados en windows— no funcione, en ese caso puede utilizarse la codificación *utf8*: `\usepackage[utf8]{inputenc}`

5.5. Comandos útiles para la edición del texto

- `\maketitle`: configura un título para nuestro documento obteniendo los datos de los comandos `\author{}`, `\title{}` y `\date{}`.
- `\newpage`: fuerza un cambio de página, y los comandos `\nolinebreak` y `\nopagebreak` previenen un cambio de línea o página donde no se lo desea.
- `\clearpage`: su uso práctico es parecido al de `\newpage`, sólo que limpia el resto de la página.
- `\noindent`: Si no queremos que aparezca la sangría de cada párrafo podemos utilizar este comando al inicio del mismo.
- `\:`: esto indica al compilador que el texto continúa en la próxima línea, al contrario se escribe un tilde (~) entre dos palabras, esto significa que el compilador no debe cortar la línea entre estas dos palabras.
- Espacios verticales: LaTeX provee de tres sencillos comandos para insertar espacios predefinidos entre líneas, estos son
 - `\smallskip`
 - `\medskip`
 - `\bigskip`

Puede utilizarse también el comando `\vspace*{<tamaño>}` donde el tamaño puede ser un número seguido de una unidad de medida. Por ejemplo:

- 1in = 1 pulgada.
 - 2.34cm = 2.34 centímetros.
 - -0.7m = -0.7 milímetros. El signo negativo significa que el salto será hacia arriba en vez de hacia el final de la página.
 - 2ex = el largo igual a dos veces una letra x.
 - 30pt = es el tamaño del punto utilizado para definir el tamaño de las letras.
- Espacios horizontales: puede utilizarse el comando `\hspace*{<tamaño>}` con el mismo criterio que el anterior.

6. Escritura y simbología matemática

6.1. Definición de entorno matemático

LaTeX tiene la posibilidad de incorporar orgánicamente simbología matemática sin la necesidad de un editor especial como en el caso del word.

Es necesario para esto definir un *espacio* determinado por caracteres \$ según las necesidades.

Hay dos posibilidades de incorporar fórmulas:

- Fórmulas inmersas en el texto
- Espacio matemático destacado

6.1.1. Fórmulas inmersas en el texto

\$ *fórmula* \$: Este espacio hace que una fórmula aparezca correctamente en el medio de un párrafo. Por ejemplo:

Código:

Dada una función $f(x)$ de una variable real x y un intervalo $[a, b]$ de la recta real, la integral $\int_a^b f(x) dx$ es igual al área de la región del plano xy limitada entre la gráfica de f , el eje x , y las líneas verticales $x = a$ y $x = b$, donde son negativas las áreas por debajo del eje x .

Resultado:

Dada una función $f(x)$ de una variable real x y un intervalo $[a, b]$ de la recta real, la integral $\int_a^b f(x) dx$ es igual al área de la región del plano xy limitada entre la gráfica de f , el eje x , y las líneas verticales $x = a$ y $x = b$, donde son negativas las áreas por debajo del eje x .

6.1.2. Espacio matemático destacado

\$\$ *fórmula* \$\$ ó \[*fórmula* \]: De esta manera aparecerá la fórmula resaltada y centrada dentro de la página. Por ejemplo:

Código:

Dada una función $f(x)$ de una variable real x y un intervalo $[a, b]$ de la recta real, la integral $\int_a^b f(x) dx$ es igual al área de la región del plano xy limitada entre la gráfica de f , el eje x , y las líneas verticales $x = a$ y $x = b$, donde son negativas las áreas por debajo del eje x .

Resultado:

Dada una función $f(x)$ de una variable real x y un intervalo $[a, b]$ de la recta real, la integral

$$\int_a^b f(x) dx$$

es igual al área de la región del plano xy limitada entre la gráfica de f , el eje x , y las líneas verticales $x = a$ y $x = b$, donde son negativas las áreas por debajo del eje x .

6.2. Expresiones más comunes

De ahora en adelante se asume que toda simbología matemática se escribe dentro de un espacio $\$ \dots \$$ o $\$ \$ \dots \$ \$$.

En el texto anterior se ha mostrado brevemente cómo se incorporan algunos símbolos matemáticos. Veremos ahora los caracteres y expresiones más comunes y cómo vincularlos.

6.2.1. Agrupar elementos dentro del espacio matemático

Resulta fundamental saber que las llaves simples $\{\}$ agrupan elementos sin que se éstas impriman en el documento.

Por ejemplo:

Si deseo obtener como resultado $a^{x_1^2+x_2^2+x_3^2}$ es necesario escribir $\$a^{\{x_1^2 + x_2^2 + x_3^2\}}\$$. Nótese cómo las llaves agruparon la expresión pero sin embargo no aparecieron en el resultado final. Esta será una situación muy común en la escritura de fórmulas.

El agrupamiento de elementos es fundamental para entender la mecánica del lenguaje. Debido a que un elemento puede ser parte de otro, escribiremos las posibles relaciones o formas de escritura refiriéndonos a elementos encerrados entre llaves.

Si quisiéramos que aparezcan las llaves deberemos anteponer la barra invertida, ejemplo:

$\$ \$A=\{x: x \in \mathbb{N}\} \$$ arroja

$$A = \{x : x \in \mathbb{N}\}$$

6.2.2. Alfabeto griego

LaTeX incluye todo el alfabeto Griego:

¹⁰El comando $\backslash\mathrm{mathbb}\{\dots\}$ transforma el contenido de las llaves a tipografía matemática. Para poder utilizarlo es necesario incorporar en el preámbulo el paquete $\backslash\mathrm{usepackage}\{\mathrm{amssymb}\}$

código	símbolo	código	símbolo	código	símbolo	código	símbolo
\Alpha	A	\Beta	B	\Gamma	Γ	\Delta	Δ
\Epsilon	E	\Zeta	Z	\Eta	H	\Theta	Θ
\Iota	I	\Kappa	K	\Lambda	Λ	\Mu	M
\Nu	N	\Xi	Ξ	\Pi	Π	\Rho	P
\Sigma	Σ	\Upsilon	Υ	\Phi	Φ	\Chi	X
\Psi	Ψ	\Omega	Ω				
\alpha	α	\beta	β	\gamma	γ	\delta	δ
\epsilon	ϵ	\zeta	ζ	\eta	η	\theta	θ
\iota	ι	\kappa	κ	\lambda	λ	\mu	μ
\nu	ν	\xi	ξ	\pi	π	\rho	ρ
\sigma	σ	\tau	τ	\upsilon	υ	\phi	ϕ
\chi	χ	\psi	ψ	\omega	ω		

6.2.3. Potencias y raíces

Aunque en el ejemplo anterior nos ayamos adelantado, damos la forma general de escribir potencias y raíces.

Para escribir potencias debe utilizarse el caracter \wedge de la siguiente manera:

$$\{base\}^{\{potencia\}}$$

Ejemplos:

- El código $\$(a+b+c)^2\ $$ arroja $(a + b + c)^2$
- El código $\$(a+b+c)^{\{(c+d)\}}\ $$ arroja $(a + b + c)^{(c+d)}$
- El código $\$c+d^{\{(a+b+c)^2\}}\ $$ arroja $c + d^{(a+b+c)^2}$

la forma para escribir raíces es de la siguiente manera: $\sqrt[\text{raiz}]{\text{base}}$

Ejemplos:

- $\sqrt[2]{a+b}$ arroja $\sqrt[2]{a+b}$
- $\sqrt[(a+b+c)]{Z}$ arroja $\sqrt[(a+b+c)]{Z}$
- $\sqrt[(a+b+c)]{(a+b+c)^{\{(c+d)\}}}$ arroja $\sqrt[(a+b+c)]{(a+b+c)^{(c+d)}}$

6.2.4. Fracciones

Para escribir fracciones debemos utilizar el comando

$$\frac{\text{numerador}}{\text{denominador}}$$

Ejemplos:

- $\frac{a}{b}$ arroja $\frac{a}{b}$
- $\frac{\sqrt[2]{a+b}}{b}$ arroja $\frac{\sqrt[2]{a+b}}{b}$
- $\frac{\sqrt[2]{a+b}}{(a+b+c)^2}$ arroja $\frac{\sqrt[2]{a+b}}{(a+b+c)^2}$
- $\frac{\sqrt[2]{a+b}}{(a+b+c)^2} + \frac{\sqrt[2]{a+b}}{b}$ arroja $\frac{\sqrt[2]{a+b}}{(a+b+c)^2} + \frac{\sqrt[2]{a+b}}{b}$

6.2.5. Subíndices

Para indicar un subíndice debe utilizarse el caracter `_` del siguiente modo: `{objeto}_{subíndice}`. Ejemplos:

- x_1, x_2, \dots, x_n arroja x_1, x_2, \dots, x_n
- α_β arroja α_β
- A_{B_C} arroja A_{B_C}
- $\sum_{i=1}^n$ arroja $\sum_{i=1}^n$

6.2.6. Sumatorias, productorias e integrales

- Para indicar una **sumatoria** desde $i = 0$ hasta infinito puede utilizarse el comando `\sum` de la siguiente manera:

$\sum_{i=0}^{\infty}$ cuyo resultado es

$$\sum_{i=0}^{\infty}$$

`\sum` utiliza los comandos `_` y `^` de manera especial, incorporando la expresión $i = 0$ por debajo de la Sigma y al ∞ por encima.

- Para indicar una **productoria** desde $i = 0$ hasta infinito puede utilizarse el comando `\prod` de la siguiente manera:

$\prod_{i=0}^{\infty}$ cuyo resultado es

$$\prod_{i=0}^{\infty}$$

- Para indicar una **integral** desde a hasta b , puede utilizarse el comando `\int` de la siguiente manera:

\int_a^b cuyo resultado es

$$\int_a^b$$

- Una variante de este comando es `\oint`:

`\oint_{a}^{b}` cuyo resultado es¹¹

$$\oint_a^b$$

Recordamos que para todos los casos pueden incorporarse dentro de las llaves cualquier expresión algebraica.

6.2.7. Paréntesis, corchetes y llaves

En expresiones algebraicas extensas suelen utilizarse paréntesis, corchetes y llaves globales, es decir, que abarcan a una o varias operaciones. Hay pues una manera de que LaTeX adapte estos elementos a la expresión.

Ejemplo:

Supongamos que de la expresión $\frac{a+b}{c} + \sqrt{a+c}$ queremos elevar el primer término al cuadrado. Es necesario indicarlo con un paréntesis que abarque a toda la fracción.

Utilizamos pues la expresión

`\left(\frac{a+b}{c} \right)^2 + \sqrt{a+c}` para obtener:

$$\left(\frac{a+b}{c} \right)^2 + \sqrt{a+c}$$

Para que los paréntesis, corchetes, llaves y signos se adapten a la expresión hay que utilizar los comandos `\left` y `\right` de la siguiente manera:

- `\left(... \right)` arroja (...)
- `\left[... \right]` arroja [...]

Nótese que dentro del entorno matemático no pueden utilizarse las llaves directamente ya que son caracteres especiales, el lugar de ello debe utilizarse el comando `\lbrace ... \rbrace` o bien las llaves precedidas de la barra: `\{ ... \}`.

- `\left\lbrace ... \right\rbrace` arroja {...}

Ejemplos:

- `\left(\frac{a}{b} \right)` arroja $\left(\frac{a}{b} \right)$
- `\left[\frac{a}{b} \right]` arroja $\left[\frac{a}{b} \right]$
- `\left[\frac{a}{b} \right]` arroja $\left\{ \frac{a}{b} \right\}$
- `\left\langle \frac{a}{b} \right\rangle` arroja $\left\langle \frac{a}{b} \right\rangle$

Si deseamos que sólo aparezca un solo paréntesis, corchete, llave o signo, puede simplemente omitirse uno de los dos símbolos al comienzo o al final. Por ejemplo:

¹¹Esta expresión podría no tener sentido matemático, sin embargo la exponemos para mostrar las posibilidades del comando `\oint_{...}^{...}`.

6.2.8. Matrices

La escritura de matrices puede ser muy variada. Daremos aquí una forma general de escribir cualquiera de ellas. Aquí vemos un ejemplo de matriz de 3 columnas por 2 filas:

```
\left[
\begin{array}{lcr}
Col1 & Col2 & Col3 & \\\%fila 1
Col1 & Col2 & Col3 & \%fila 2
\end{array}
\right]
```

- `\left[... \right]`: Indica que los datos se encerrarán entre corchetes. Podríamos utilizar paréntesis, llaves o signos, según hemos visto en la sección 6.2.7.
- `\begin{array} ... \end{array}`: Este es el comando que crea la matriz propiamente dicha.
- `\begin{array}{ parámetros }`: El comando `\begin{array}` necesita parámetros que indiquen la cantidad de columnas. Esos parámetros pueden ser `{lcr}` donde
 - l: indica la primera columna alineada a la izquierda (left)
 - c: indica la segunda columna alineada al centro (center)
 - r: indica la tercera columna alineada a la derecha (right)
- `Col1 & Col2 & Col3 \\\`: el caracter `&` define columnas y la doble barra `\\` indica que se pasa a una nueva fila. La última fila de la matriz no precisa las `\\`.

Ejemplos:

Código:

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

Resultado:

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

Código:

```

$$
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)

```

Resultado:

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

Código:

```

$$
\left. \begin{array}{l}
a \\
b \\
\vdots \\
z
\end{array} \right\} = B

```

Resultado:

$$\left. \begin{array}{l} a \\ b \\ \vdots \\ z \end{array} \right\} = B$$

Código:

```

$$
B = \langle a \ b \ \cdots \ z \rangle

```

Resultado:

$$B = \langle a \ b \ \cdots \ z \rangle$$

Código:

```


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


```

Resultado:

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

6.2.9. Algunas expresiones matemáticas

código	resultado
$0 = a_{11} + a_{12}$	$0 = a_{11} + a_{12}$
$x^{a+b} = x^a x^b$	$x^{a+b} = x^a x^b$
$x_i = \sqrt[n]{\frac{a_i}{b_i}}$	$x_i = \sqrt[n]{\frac{a_i}{b_i}}$
$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$	$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta \end{pmatrix}$
$\int_{ x-x_0 < X_0} \Phi(x)$	$\int_{ x-x_0 < X_0} \Phi(x)$
$\lim_{ x-x_0 < X_0} \Phi(x)$	$\lim_{ x-x_0 < X_0} \Phi(x)$
$\oint F(x) dx$	$\oint F(x) dx$
$\iint \Phi(x, y) dx dy$	$\iint \Phi(x, y) dx dy$
$\sum_{0 \leq i \leq m, 0 < j < n} P(i, j)$	$\sum_{0 \leq i \leq m, 0 < j < n} P(i, j)$
$\lim_{n \rightarrow \infty} \frac{n!}{2^n} = \pi$	$\lim_{n \rightarrow \infty} \frac{n!}{2^n} = \pi$
$\binom{n}{r} = \frac{n!}{r!(n-r)!}$	$\binom{n}{r} = \frac{n!}{r!(n-r)!}$
$x' + x'' = \dot{x} + \ddot{x}$	$x' + x'' = \dot{x} + \ddot{x}$
$\vec{v} = a\hat{x} + b\hat{y}$	$\vec{v} = a\hat{x} + b\hat{y}$

6.2.10. Comentarios y edición dentro del espacio matemático

En ocasiones es necesario incluir texto dentro de una fórmula o definición. El problema es que no se puede escribir texto directamente en el espacio matemático, para ello debe utilizarse el comando `\mbox{ ... }`. Por ejemplo:

Código:

```
\begin{small}
$$
h_{\epsilon}^s(A)=\mbox{inf} \left\{ \sum_{i=0}^{\infty} \text{diam}^s(U_i) \mid \{U_1, U_2, U_3, \dots\} \right.
\begin{array}{l}
\text{cubr. por abiertos de } A \text{ con } \text{diam}(U_i) < \epsilon
\end{array}
\left. \right\}
\end{small}
```

Resultado:

$$h_{\epsilon}^s(A) = \inf \left\{ \sum_{i=0}^{\infty} \text{diam}^s(U_i) \mid \{U_1, U_2, U_3, \dots\} \text{ cubr. por abiertos de } A \text{ con } \text{diam}(U_i) < \epsilon \right\}$$

6.2.11. Definición de entornos dedicados:

uso del comando `\newtheorem{...}{...}[...]`

En aquellos documentos donde se definen términos técnicos, se demuestran teoremas, se enuncian conjeturas, etc... es habitual el uso de referencias cruzadas. La escritura y modificación constante de contenido hace muy incómodo el seguimiento de las referencias numeradas. El problema surge cuando incorporamos en el medio un entorno al que hay que hacer referencia (el mismo problema que sucedía con los capítulos, secciones, subsecciones, etc...). Para definir estos entornos podremos utilizar el comando `\newtheorem{ ... }{ ... }` en el preámbulo:

```
\newtheorem{EntornoTeorema}{Teorema}
```

Este comando crea dentro de nuestro documento un nuevo entorno cuyo nombre es el que se ha definido como primer parámetro. El segundo parámetro es el que se mostrará en documento. Así, a partir de esta definición, podemos utilizar en cualquier parte del documento el entorno

```
\begin{EntornoTeorema}
...
\end{EntornoTeorema}
```


Ejemplo:

Definimos tres tipos de entornos diferentes en el preámbulo de nuestro documento de la siguiente manera:

```
\newtheorem{EntornoDefinicion}{Definición}
\newtheorem{EntornoTeorema}{Teorema}
```

Luego utilicemos los espacios definidos:

```
\begin{EntornoTeorema}
Dado un triángulo rectángulo, el cuadrado de la hipotenusa es
igual a la suma de los cuadrados de sus catetos.
\end{EntornoTeorema}

\begin{EntornoDefinicion}
Dados dos conjuntos  $X, Y \subset \mathbb{R}$  se define función o transformaci
ón  $f$  de  $X$  hacia  $Y$  } como una relaci
ón o f
órmula que asocia a un punto  $f(x)$  de  $Y$  por cada punto  $x$  de  $X$ . Se denota como  $f: X \rightarrow Y$ . El conjunto  $X$  es llamado dominio de la funci
ón  $f$ , e  $Y$  co-dominio de  $f$ .
\end{EntornoDefinicion}

\begin{EntornoDefinicion}
Un conjunto infinito  $A$  es numerable si es coordinable con  $\mathbb{N}$ , es decir, si sus elementos pueden ser listados  $x_1, x_2, \dots$  con cada elemento de  $A$  indicado con un n
úmero natural; de otra manera se dice que  $A$  es infinito no-numerable.
\end{EntornoDefinicion}
```

Cuyo resultado final es el siguiente:

Teorema 1 *Dado un triángulo rectángulo, el cuadrado de la hipotenusa es igual a la suma de los cuadrados de sus catetos.*

Definición 1 *Dados dos conjuntos $X, Y \subset \mathbb{R}$ se define función o transformación f de X hacia Y como una relación o fórmula que asocia a un punto $f(x)$ de Y por cada punto x de X . Se denota como $f: X \rightarrow Y$. El conjunto X es llamado dominio de la función f , e Y co-dominio de f .*

Definición 2 *Un conjunto infinito A es numerable si es coordinable con \mathbb{N} , es decir, si sus elementos pueden ser listados x_1, x_2, \dots con cada elemento de A indicado con un número natural; de otra manera se dice que A es infinito no-numerable.*

Nótese cómo los entornos se enumeran. Todos los que se definen mediante *newtheorem* admiten el uso del comando `\label{}`¹² para referencias cruzadas.

Los entornos *newtheorem* tienen ya una tipografía definida por defecto, es por eso que sin haberlo indicado, el compilador puso el texto en itálica y el título en negrita.

7. Tablas

Para el diseño de tablas existen muchas variantes y herramientas que no se verán en este curso por ser inicial. Se recomienda especialmente la lectura de [Mata Botana, 2003] donde se encuentra un fabuloso compendio de comandos y formatos para el diseño de tablas. Puede consultarse también [Cascales Salinas et al., 2003] en el capítulo 13.

En la sección 6.2.8 se ha visto una forma de disponer datos en forma de tablas (de matrices). Aunque aquella es una forma típica utilizada específicamente para matemática, existen otras maneras de disponer datos tabulados o directamente en tablas.

7.1. El entorno `\begin{tabbing} \dots \end{tabbing}`

Este entorno de trabajo no genera exactamente tablas, pero permite presentar a texto encolumnado, de manera similar a como lo haría un tabulador. Del entorno de trabajo se entra y se sale mediante los comandos `\begin{tabbing}` y `\end{tabbing}` respectivamente. Cuenta con los siguientes comandos:

- `\=` Fija la posición de los tabuladores
- `\>` Salta al siguiente tabulador
- `\\` Fin de línea
- `\kill` Si una línea finaliza con este comando no se imprime
- `\+` Hace que la línea sig. empiece en el tabulador sig. al previsto
- `\-` Hace que la línea sig. empiece en el tabulador anterior al previsto
- `\<` Salta al tabulador anterior

Ejemplo 1:

Si ponemos:

```
\begin{tabbing}
Nombre \= Apellido \= Padrón\\
Esteban \> Quito \> 80000\\
```

¹²Que se verá en la sección 9.2

```
Elena \> Nito \> 80001\\
Olga \> Sana \> 80002\\
\end{tabbing}
```

Lo que obtendremos es:

Nombre	Apellido	Padrón
Esteban	Quito	80000
Elena	Nito	80001
Olga	Sana	80002

Ejemplo 2:

También podemos fijar el ancho de las columnas:

```
\begin{tabbing}
\hspace*{2cm} \= \hspace*{5cm} \= \hspace*{3cm} \kill
Nombre \> Apellido \> Padrón\\
Esteban \> Quito \> 80000\\
Elena \> Nito \> 80001\\
Olga \> Sana \> 80002\\
\end{tabbing}
```

Lo que obtendremos es:

Nombre	Apellido	Padrón
Esteban	Quito	80000
Elena	Nito	80001
Olga	Sana	80002

7.2. El entorno `\begin{tabular} ... \end{tabular}`

Este entorno de trabajo nos permite trabajar con muchas más posibilidades que el entorno `tabbing`. Tiene dos formatos posibles que son los siguientes:

```
\begin{tabular}[posición]{columnas}
columna 1 & columna 2 & columna n \\
...
...
...
\end{tabular}
```

ó

```
\begin{tabular*}{ancho}[posición]{columnas}
columna 1 & columna 2 & columna n \\
...
...
```

```
...
...
\end{tabular*}
```

<i>columnas</i>	Determina la cantidad de columnas y su posición o {1}: alinea a izquierda {c}: centra {r}: alinea a derecha { }: colocado entre dos columnas genera una línea vertical p{ancho}: Establece el ancho de una columna @{texto}: Inserta el texto entre columnas @{\hspace{ancho}}: Inserta espacio entre columnas *{num}{cols}: num establece la cantidad de columnas y cols el formato de las columnas (que puede ser una expresión)
<i>posición</i>	Determina la posición de la tabla con respecto al texto b: pone la parte inferior de la tabla en la base del texto (opción por defecto) c: pone la mitad de la tabla en la base del texto t: pone la parte superior de la tabla en la base del texto
&	Salta a la siguiente columna
\\	Salta a la siguiente fila
\hline	Genera una línea horizontal
\cline(n-m)	Genera una línea horizontal desde el inicio de la columna n al fin de la m
\vline	Genera una línea vertical a lo largo de una fila

Cuadro 1: Propiedades y comandos de tablas

7.3. El entorno `\begin{table} ... \end{table}`

Lo utilizaremos aquí para poder crear un índice de tablas creadas con el comando *tabular*. Su forma general es la siguiente:

```
\begin{table}
\begin{tabular}[]{}
...
\end{tabular}
\caption[Contenido para el índice de cuadros]
{Leyenda al pie de la tabla}
\end{table}
```

Veremos algunos ejemplos sencillos de *tabular*, *table* y *figure* combinados:

Ejemplo 1:

Si deseamos una tabla:

- de tres columnas con la primera columna de ancho 3cm, la segunda de 5cm y la tercera de 1cm
- y queremos trazar las líneas verticales y horizontales
- y queremos que la tabla aparezca centrada en la hoja.
- y queremos que sea una *figura* para poder incluirla en el índice de figuras.

deberemos escribir:

```
\begin{figure}
\begin{center}
\begin{tabular}{|p{3cm}|p{5cm}|p{1cm}|}
\hline
dato11 & dato12 & dato13 \\
\hline
dato21 & dato22 & dato23 \\
\hline
dato31 & dato32 & dato33 \\
\hline
\end{tabular}
\end{center}
\caption[Tabla del Ejemplo 1]
{Esta es una tabla de prueba\label{tabla_de_prueba}}
\end{figure}
```

lo que dará como resultado la siguiente tabla

dato11	dato12	dato13
dato21	dato22	dato23
dato31	dato32	dato33

Figura 8: Esta es una tabla de prueba

Ejemplo 2:

Si deseamos una tabla:

- de cuatro columnas con la primera columna de ancho 2cm, la segunda de 2cm, la tercera de 2cm y la cuarta 6cm
- y queremos trazar sólo líneas verticales y dos horizontales que enmarquen la tabla.

dato11	dato12	dato13
dato21	dato22	dato23
dato31	dato32	dato33

Cuadro 2: Tabla de ejemplo

- y queremos que la tabla aparezca centrada en la hoja.
- y queremos incluirla en el índice de cuadros.

deberemos escribir:

```
\begin{table}
\begin{center}
\begin{tabular}{|p{2cm}|p{2cm}|p{2cm}|}
\hline
dato11 & dato12 & dato13 \\
dato21 & dato22 & dato23 \\
dato31 & dato32 & dato33 \\
\hline
\end{tabular}
\end{center}
\caption[Tabla del Ejemplo 2]{Tabla de ejemplo}
\end{table}
```

lo que dará como resultado la siguiente tabla

8. Inclusión de imágenes

LaTeX tiene la capacidad de incluir imágenes de un modo práctico sin que nuestro documento adquiera un tamaño sideral. El hecho de trabajar en un entorno de texto –recordemos que el Texmaker y la mayoría de los editores de LaTeX son de texto plano– hace imposible la incorporación de imágenes directamente. Esta es otra diferencia fundamental con el uso de editores WYSIWYG. La manera de hacerlo es utilizando el comando `\includegraphics{dirección de la imagen}` de la siguiente manera:

- En Linux:
`\includegraphics {/home/usuario/CarpetaImagenes/imagen.jpg}`
- En Windows: `\includegraphics {c:/Mis Documentos/imagen.jpg}`

Al momento de la compilación de nuestro archivo **.tex** el compilador buscará la imagen en la dirección que hemos indicado y la incorporará en el archivo PDF o DVI.

8.0.1. Tipos de imagen y tipos de compilación

Encontramos aquí un buen motivo para decidir en qué formato compilar o trabajar: si en DVI o PDF.

Se ha dicho ya que el formato DVI –que es el nativo de Latex– es vectorial y presenta ventajas de manejo frente a los PDF ya que éstos por lo general son de tamaño mayor y los lectores de PDF's suelen ser programas más pesados, especialmente en Windows.

El formato DVI sólo admite imágenes de extensión **.eps**, formato muy conveniente ya que son imágenes de tipo vectoriales de alta calidad y bajo peso. El problema que presenta es que el formato no está muy difundido.

Los formatos de imágenes más populares –sobre todo en internet– son **.jpg**, **.gif** y **.png**. Existen aplicaciones gratuitas que transforman estos formatos a **.eps**.

Si quisiéramos trabajar directamente con imágenes **.jpg** y **.png** –el formato **.gif** no está soportado por Latex– tendremos pues que compilar o trabajar en PDF que sí los admite.

8.0.2. Inclusión de archivos PDF

El comando `\includegraphics { ... }` para la compilación en PDF admite también archivos **.pdf**. Es decir les da el mismo tratamiento que a una imagen **.jpg** o **.png**, por ejemplo:

```
\includegraphics{/home/manuel/imagenes/POINT_PDF1.pdf}
```

8.0.3. Propiedades de una imagen en Latex

El comando `\includegraphics { ... }` admite algunas propiedades. El formato completo del comando es `\includegraphics[] { ... }`:

- `{ ... }`: aquí se indica el *path* o dirección de la imagen.
- `[]`: aquí se indican parámetros que el sistema aplicará a la imagen en el momento de compilar el tex. :
 - **scale**: puede aplicarse a la imagen un factor de escala. Ejemplo:
`\includegraphics[scale=0.75]{myfig.eps}`
 - **width**, **height**: aplica un factor al ancho y de alto a la imagen, respectivamente. Ejemplo:
`\includegraphics[height=45mm,width=52mm]{myfig.png}`
 - **angle**: aplica un factor de giro a la imagen. Ejemplo:
`\includegraphics[angle=-60]{myfig.png}`

Ejemplo:

La imagen que sigue ha sido incorporada y modificada por medio del siguiente comando, compilado en PDF dentro de un entorno `\begin{center} ... \end{center}` para que aparezca centrada en el cuerpo del documento:

```
\begin{center}
\includegraphics[scale=0.15, angle=-214]{/home/APUNTES/CursoLatex/
imagenes/wyisiwigLatex.png}
\end{center}
```



9. Referencias, notas al pié, citas y bibliografía

Latex tiene automatizado todo un sistema de citas y referencias. También nos da la posibilidad de incorporar citas bibliográficas¹³.

9.1. El entorno figure

Este entorno será de gran ayuda a la hora de referenciar diferentes objetos con párrafos e imágenes —las tablas se referenciarán de otra manera—.

Su estructura es la siguiente:

```
\begin{figure}
Objeto
\caption[TextoLeyendaIndice]
{Texto Leyenda\label{Referencia DelObjeto}}
\end{figure}
```

El *Objeto* puede ser cualquier cosa: uno o varios párrafos de texto, texto escalado, un dibujo lineal, un gráfico externo incluido con `\includegraphics{}` o cualquier combinación de éstos y otros elementos.

Se ha incluido la etiqueta `\label{}` —que no es obligatoria— dentro de `\caption{}` para poder de esta manera hacer referencia a cualquier objeto de nuestro documento.

¹³El sistema ideal para realizar citas bibliográficas es BibTeX que permite establecer una verdadera base de datos bibliográfica e incorporarla a nuestro trabajo. BibTeX no será abordado en este curso por su extensión, para más información puede ingresar en www.bibtex.org

9.2. Utilización de los comandos `\label{}` y `\ref{}`

La construcción de textos científicos puede volverse un tanto compleja si frecuentemente tenemos que hacer referencias a fórmulas, tablas, secciones, teoremas, etc... de nuestro trabajo. Supongamos que numeramos todas las referencias manualmente y nos diéramos cuenta de que nos ha faltado una fórmula al inicio del documento, esto provocaría reenumerar todo el documento. Latex resuelve este problema con los comandos `\label{}` y `\ref{}` muy sencillamente:

- `\label{nombre_de_referencia}` se utiliza para poner un *nombre* a un objeto (capítulos, secciones, tablas, imágenes, etc ...)
- `\ref{nombre_de_referencia}` se utiliza para concretar la referencia.
- Ejemplo:

Si queremos hacer referencia a la sección *Documentos pre-formateados* deberemos incorporarle un `\label{}` y elegir un nombre –en lo posible que sea representativo de aquello que estamos nombrando– en este caso será simplemente *Documentos.preformateados*. Quedaría así expresado:

```
\section{Documentos pre-formateados}
\label{Documentos_preformateados}
```

Para referenciarlo finalmente utilizamos `\ref{}` con el nombre de referencia, es decir: `\ref{Documentos_preformateados}`

El resultado sería el siguiente:

el código:

```
... se dijo en la sección \ref{Documentos_preformateados}
devolverá:
... se dijo en la sección 4
```

- Pueden así referenciarse todo tipo de objetos que contengan un label. Por ejemplo el código

```
En la sección \ref{Estructura_basica} se ha incorporado
la imagen \ref{ejemplo_basico_2} que pertenece a
la subsección \ref{primer_ejemplo_sencillo}.
```

devuelve

En la sección 3 se ha incorporado la imagen 5 que pertenece a la subsección 3.9

9.3. Notas al pie de página con el comando `\footnote{}`

Para realizar notas al pie contamos con el comando `\footnote{ ... }`. El criterio general es incluirlas directamente dentro del texto. Cuando el compilador encuentre este comando, colocará su contenido al pie de la página, si la nota al pie es muy extensa, continuará al pie de la página siguiente. Incorporará en el texto un número de referencia.

Dentro del `\footnote{ ... }` pueden incorporarse cualquier tipo de datos, imágenes, tablas, etc. Ejemplo:

El código

```
La dimensión de Hausdorff\footnote{Felix Hausdorff
(8 de noviembre de 1868, 26 de enero de 1942) fue
un matemático alemán que está considerado como uno
de los fundadores de la Topología moderna} es una
generalización métrica del concepto de
dimensión de un espacio topológico. Su formula es
$d=\frac{\log N(r)}{\log \frac{1}{r}}$
```

devuelve el siguiente texto y la nota al pie de esta misma página:

La dimensión de Hausdorff¹⁴ es una generalización métrica del concepto de dimensión de un espacio topológico. Su formula es $d = \frac{\log N(r)}{\log \frac{1}{r}}$.

9.4. Bibliografía y citas

Latex posee herramientas potentes para automatizar el uso de bibliografía y para poder realizar citas. Es necesario pues conocer la estructura y comandos que manipulan los datos bibliográficos para poder hacer luego referencia a ellos.

9.4.1. Bibliografía

Para incorporar citas bibliográficas podemos utilizar el entorno *thebibliography* cuya estructura es la siguiente:

```
\begin{thebibliography}{LongitudMaxima}
...
\bibitem[Leyenda]{EtiquetaDeReferencia} Texto
...
\end{thebibliography}
```

A continuación se presenta el significado de los distintos elementos de esta sintaxis.

¹⁴Felix Hausdorff (8 de noviembre de 1868, 26 de enero de 1942) fue un matemático alemán que está considerado como uno de los fundadores de la Topología moderna.

- *LongitudMaxima*: Es una cadena de caracteres de anchura mayor o igual que la máxima que va a impresa en la numeración de la bibliografía. Por ejemplo, si se sabe que no vamos a tener más de 200 referencias bibliográficas, la orden `\begin{thebibliography}{999}` instruye a Latex para que reserve un espacio igual a longitud de la cadena 999 para 1a numeración que aparece a la izquierda de nuestras referencias en la lista de bibliografía.
- *EtiquetaDeReferencia*: Se utiliza para identificar la referencia bibliográfica y es el argumento para el uso posterior en las citas mediante el comando `\cite{EtiquetaDeReferencia}`.
- *Texto*: Es el contenido de la cita bibliográfica que normalmente contiene: autor o autores, título del documento, editor o revista, año de publicación, y número de páginas si es relevante.
- *Leyenda*: Se utiliza para modificar la identificación que en la lista de referencias se imprimirá a la izquierda de las mismas. Como ya se ha comentado, por defecto Latex numera las referencias. Si en una entrada `\bibitem [Leyenda] {EtiquetaDeReferencia} Texto` llenamos el campo opcional *Leyenda*, esta *Leyenda* será la que se imprima en lugar del número de orden correspondiente. Cuando vayamos a utilizar esta opción debemos modificar la longitud *LongitudMaxima* hasta alcanzar la longitud máxima de todos los argumentos *Leyenda*.

Ejemplo: el siguiente código

```
\begin{thebibliography}{99}
\bibitem[Dow]{DandJ} Dow, W. \& Jones,
E.A., {\it Wall Street Journal}, March 29, 1929.
\end{thebibliography}
```

produce

Referencias

[Dow, 1929] Dow, W. & Jones, E.A., *Wall Street Journal*, March 29, 1929.

Por defecto este entorno titula la lista de referencias según el diccionario que se ha configurado en el preámbulo. En nuestro caso `\usepackage [spanish]{babel}` escribe **Referencias**. Para modificar dicho título puede utilizarse el comando

```
\renewcommand{\refname}{NuevoTitulo}15
```

¹⁵En general el comando `\renewcommand{}` permite modificar parámetros por defecto. Un ejemplo es el parámetro `\renewcommand{\baselinestretch{1.3}}` mencionado en la sección 3.5.2 que modifica el interlineado `\baselinestretch{1}` que Latex ha definido por defecto.

por ejemplo el código siguiente

```
\renewcommand{\refname}{Bibliografía}
\begin{thebibliography}{99}
\bibitem[Dow]{DandJ} Dow, W. \& Jones,
E.A., {\it Wall Street Journal}, March 29, 1929.
\end{thebibliography}
```

produce

Bibliografía

[Dow, 1929] Dow, W. & Jones, E.A., *Wall Street Journal*, March 29, 1929.

9.4.2. Citas bibliográficas

Se ha mencionado brevemente el comando `\cite{NombreDeReferencia}`. Su inclusión dentro de un texto provocará, al compilarlo, que Latex invoque la lista del entorno *thebibliography* y por medio del parámetro de referencia *NombreDeReferencia* incorpore la *Leyenda*¹⁶ como resultado. Por ejemplo:

el código

Así se cita a `\cite{DandJ}`, único libro de la lista
produce

Así se cita a [Dow, 1929], único libro de la lista

9.4.3. Normas bibliográficas

Por lo general si nuestro documento es de carácter científico seguro estará sujeto a alguna norma de referencias bibliográficas que determinará los formatos tanto para la lista de bibliografía y referencias, como para las citas dentro del texto. Es importante tener en cuenta este aspecto ya que la bibliografía suele ser parte fundamental en cualquier documento científico.

10. Índices

LaTeX posee potentes herramientas que confeccionan automáticamente índices de acuerdo a la estructura de nuestro documento, incluyendo la numeración:

- Índices de contenido
- Índices de figuras (gráficos u objetos)

¹⁶Ver en 9.4.1 la estructura del entorno *thebibliography*.

- Índices de cuadros o tablas
- Índices de palabras
- Glosario

En este curso se comentarán sólo los tres primeros.

10.1. Índice de contenidos

Una de las grandes ventajas de LaTeX es la generación automática de índices que se generará a partir de la estructura jerárquica de capítulos, secciones, subsecciones, subsubsecciones, etc... que numerará automáticamente.

La incorporación del índice de contenidos es extremadamente sencilla, sólo hay que incluir el comando

```
\tableofcontents
```

donde se quiera que aparezca. Requiere ser escrito dentro del documento, entre el comando `\begin{document}` y `\end{document}`. Es necesario compilar dos veces el documento, pues genera un archivo auxiliar de extensión `.toc` en la primera compilación y que es utilizado en la segunda.

10.2. Índice de tablas

Si hemos incluido tablas dentro de nuestro documento mediante el uso del entorno `\begin{table} ... \end{table}` y le hemos incluido el comando `\caption[]{}{}` con todos sus parámetros, podemos entonces crear fácilmente un índice de tablas o *cuadros* mediante la inclusión del comando

```
\listoftables
```

El uso del diccionario español babel –configurado en el preámbulo– coloca por defecto el título **Índice de Cuadros**. Si deseamos modificarlo por, por ejemplo, **Índice de tablas**, deberemos anteponer al comando `\listoftables` la redefinición de dicho título de la siguiente manera:

```
\renewcommand{\listtablename}{Índice de Tablas}
```

10.3. Índice de Figuras

Si hemos incluido figuras¹⁷ dentro de nuestro documento y le hemos incluido el comando `\caption[]{}{}` con todos sus parámetros, podemos entonces crear fácilmente un índice de figuras mediante la inclusión del comando

```
\listoffigures
```

Para cambiar el título del índice por, por ejemplo, **Lista de Gráficos** deberemos anteponer al comando `\listoffigures`:

```
\renewcommand{\listfigurename}{Índice de Gráficos}
```

¹⁷Ver sección [9.1](#)

11. Anexo: Símbolos

\LaTeX posee una enorme cantidad de símbolos. Se dan una serie de figuras con símbolos y su comando correspondiente que puede utilizarse como ayuda memoria.

$*$	$*$	\odot	<code>\circledcirc</code>	\ominus	<code>\ominus</code>
$+$	$+$	\oslash	<code>\circledddash</code>	\oplus	<code>\oplus</code>
$-$	$-$	\cup	<code>\cup</code>	\oslash	<code>\oslash</code>
\amalg	<code>\amalg</code>	\cup	<code>\Cup</code>	\otimes	<code>\otimes</code>
$*$	<code>\ast</code>	\curlyvee	<code>\curlyvee</code>	\pm	<code>\pm</code>
\barwedge	<code>\barwedge</code>	\curlywedge	<code>\curlywedge</code>	\rightthreetimes	<code>\rightthreetimes</code>
\bigcirc	<code>\bigcirc</code>	\dagger	<code>\dagger</code>	\rtimes	<code>\rtimes</code>
\bigtriangledown	<code>\bigtriangledown</code>	\ddagger	<code>\ddagger</code>	\setminus	<code>\setminus</code>
\bigtriangleup	<code>\bigtriangleup</code>	\diamond	<code>\diamond</code>	\smallsetminus	<code>\smallsetminus</code>
\boxdot	<code>\boxdot</code>	\div	<code>\div</code>	\sqcap	<code>\sqcap</code>
\boxminus	<code>\boxminus</code>	\divideontimes	<code>\divideontimes</code>	\sqcup	<code>\sqcup</code>
\boxplus	<code>\boxplus</code>	\dotplus	<code>\dotplus</code>	\star	<code>\star</code>
\boxtimes	<code>\boxtimes</code>	\doublebarwedge	<code>\doublebarwedge</code>	\times	<code>\times</code>
\bullet	<code>\bullet</code>	\gtrdot	<code>\gtrdot</code>	\triangleleft	<code>\triangleleft</code>
\cap	<code>\cap</code>	\intercal	<code>\intercal</code>	\triangleright	<code>\triangleright</code>
\Cap	<code>\Cap</code>	\leftthreetimes	<code>\leftthreetimes</code>	\uplus	<code>\uplus</code>
\cdot	<code>\cdot</code>	\lesssim	<code>\lesssim</code>	\vee	<code>\vee</code>
\centerdot	<code>\centerdot</code>	\ltimes	<code>\ltimes</code>	\veebar	<code>\veebar</code>
\circ	<code>\circ</code>	\mp	<code>\mp</code>	\wedge	<code>\wedge</code>
\circledast	<code>\circledast</code>	\odot	<code>\odot</code>	\wr	<code>\wr</code>

Figura 9: Símbolos matemáticos varios

12. Anexo I: Ejemplos

En este anexo se detallan documentos de diferentes tipos, breves pero completos. Se describen primero el resultado compilado y seguido el código correspondiente. Se detallan los modelos de menor hacia mayor complejidad.

#	\#	/	\diagup	¬	\neg
&	\&	◇	\diamondsuit	∄	\nexists
∠	\angle	∅	\emptyset	/	\prime
\	\backprime	∃	\exists	♯	\sharp
★	\bigstar	♭	\flat	♠	\spadesuit
◆	\blacklozenge	∀	\forall	◁	\sphericalangle
■	\blacksquare	♥	\heartsuit	□	\square
▲	\blacktriangle	∞	\infty	√	\surd
▼	\blacktriangledown	◇	\lozenge	⊤	\top
⊥	\bot	∠	\measuredangle	△	\triangle
♣	\clubsuit	∇	\nabla	▽	\triangledown
\	\diagdown	ℵ	\natural	∅	\varnothing

Figura 10: Símbolos varios

↻	\circlearrowleft	⇐	\Lleftarrow	⇒	\rightarrow
↻	\circlearrowright	←	\longleftarrow	↘	\nwarrow
↶	\curvearrowleft	⇐	\Longleftarrow	→	\rightarrow
↷	\curvearrowright	↔	\longleftrightarrow	⇒	\Rightarrow
⇓	\downdownarrows	⇔	\Longleftrightarrow	↗	\rightarrowtail
↶	\downharpoonleft	↦	\longmapsto	↘	\rightharpoondown
↷	\downharpoonright	→	\longrightarrow	↗	\rightharpoonup
↶	\hookleftarrow	⇒	\Longrightarrow	↔	\rightleftarrows
↷	\hookrightarrow	⇒	\Longrightarrow	⇐	\rightleftharpoons
←	\leftarrow	↶	\looparrowleft	⇒	\rightrightarrows
⇐	\Lleftarrow	↷	\looparrowright	↗	\rightsquigarrow
↵	\leftarrowtail	↶	\Lsh	⇒	\Rightarrow
↶	\leftharpoondown	↦	\mapsto	↶	\Rsh
↷	\leftharpoonup	↶	\multimap	↘	\searrow
⇐	\leftleftarrows	⇐	\nleftarrow	↗	\swarrow
↔	\leftrightarrow	⇐	\nLeftarrow	⇐	\twoheadleftarrow
⇐	\Leftrightarrow	⇒	\nRightarrow	⇒	\twoheadrightarrow
⇐	\leftrightharpoons	↗	\nearrow	↶	\upharpoonleft
⇐	\leftrightsquigarrow	↶	\nleftarrow	↷	\upharpoonright
		↷	\nletrightarrow	↗	\upuparrows

Figura 11: Flechas horizontales

↑	\uparrow	↑	\Uparrow	↓	\downarrow
↓	\Downarrow	↑	\updownarrow	↑	\Updownarrow

Figura 12: Flechas verticales

\angle	<code><</code>	\gtrsim	<code>\gnsim</code>	\nlessapprox	<code>\nleqslant</code>
$=$	<code>=</code>	\gtrapprox	<code>\gtrapprox</code>	\nless	<code>\nless</code>
$>$	<code>></code>	\gtreqless	<code>\gtreqless</code>	\nprec	<code>\nprec</code>
\approx	<code>\approx</code>	\gtreqqless	<code>\gtreqqless</code>	\npreceq	<code>\npreceq</code>
\approxeq	<code>\approxeq</code>	\gtrless	<code>\gtrless</code>	\nsim	<code>\nsim</code>
\asymp	<code>\asymp</code>	\gtrsim	<code>\gtrsim</code>	\nsucc	<code>\nsucc</code>
\backsim	<code>\backsim</code>	\gvertneqq	<code>\gvertneqq</code>	\nsucceq	<code>\nsucceq</code>
\backsimeq	<code>\backsimeq</code>	\leq	<code>\leq</code>	\prec	<code>\prec</code>
\bumpeq	<code>\bumpeq</code>	\leqq	<code>\leqq</code>	\precapprox	<code>\precapprox</code>
\Bumpeq	<code>\Bumpeq</code>	\leqslant	<code>\leqslant</code>	\preccurlyeq	<code>\preccurlyeq</code>
\circeq	<code>\circeq</code>	\lessapprox	<code>\lessapprox</code>	\preceq	<code>\preceq</code>
\cong	<code>\cong</code>	\lesseqgtr	<code>\lesseqgtr</code>	\precnapprox	<code>\precnapprox</code>
\curlyeqprec	<code>\curlyeqprec</code>	\lesseqqgtr	<code>\lesseqqgtr</code>	\precneqq	<code>\precneqq</code>
\curlyeqsucc	<code>\curlyeqsucc</code>	\lessgtr	<code>\lessgtr</code>	\precnsim	<code>\precnsim</code>
\doteq	<code>\doteq</code>	\lessssim	<code>\lessssim</code>	\precsim	<code>\precsim</code>
\doteqdot	<code>\doteqdot</code>	\ll	<code>\ll</code>	\risingdotseq	<code>\risingdotseq</code>
\eqcirc	<code>\eqcirc</code>	\lll	<code>\lll</code>	\sim	<code>\sim</code>
\eqsim	<code>\eqsim</code>	\lnapprox	<code>\lnapprox</code>	\simeq	<code>\simeq</code>
\eqslantgtr	<code>\eqslantgtr</code>	\lneq	<code>\lneq</code>	\succ	<code>\succ</code>
\eqslantless	<code>\eqslantless</code>	\lneqq	<code>\lneqq</code>	\succapprox	<code>\succapprox</code>
\equiv	<code>\equiv</code>	\lnsim	<code>\lnsim</code>	\succcurlyeq	<code>\succcurlyeq</code>
\fallingdotseq	<code>\fallingdotseq</code>	\lvertneqq	<code>\lvertneqq</code>	\succeq	<code>\succeq</code>
\geq	<code>\geq</code>	\ncong	<code>\ncong</code>	\succnapprox	<code>\succnapprox</code>
\geqq	<code>\geqq</code>	\neq	<code>\neq</code>	\succneqq	<code>\succneqq</code>
\geqslant	<code>\geqslant</code>	\ngeq	<code>\ngeq</code>	\succnsim	<code>\succnsim</code>
\gg	<code>\gg</code>	\ngeqq	<code>\ngeqq</code>	\succsim	<code>\succsim</code>
\ggg	<code>\ggg</code>	\ngeqslant	<code>\ngeqslant</code>	\thickapprox	<code>\thickapprox</code>
\gnapprox	<code>\gnapprox</code>	\ngtr	<code>\ngtr</code>	\thicksim	<code>\thicksim</code>
\gneq	<code>\gneq</code>	\nleq	<code>\nleq</code>	\triangleq	<code>\triangleq</code>
\gneqq	<code>\gneqq</code>	\nleqq	<code>\nleqq</code>		

Figura 13: Simbología de orden (símbolos matemáticos)

$\text{\textbackslash vert}$	$\text{\textbackslash Vert}$	$\text{\textbackslash slash}$	$\text{\textbackslash slash}$	$\text{\textbackslash backslash}$
$\text{\textbackslash arrowvert}$	$\text{\textbackslash Arrowvert}$	$\text{\textbackslash bracevert}$		

Figura 14: Barras verticales y oblicuas

(())	<	\langle	>	\rangle
[[]]	\lceil	\lceil	\rceil	\rceil
{	\lbrace	}	\rbrace	\lfloor	\lfloor	\rfloor	\rfloor
	\lvert		\rvert	\lg	\lg	\rg	\rg
	\lVert		\rVert	\lgroup	\lgroup	\rgroup	\rgroup
				\lmoustache	\lmoustache	\rmoustache	\rmoustache

Figura 15: Paréntesis, corchetes, llaves y otros

\mathnormal	X	x	0	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathbb	\mathbb	\mathbb	\mathbb	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathbf	\mathbf	\mathbf	\mathbf	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathcal	\mathcal	\mathcal	\mathcal	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathfrak	\mathfrak	\mathfrak	\mathfrak	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathit	\mathit	\mathit	\mathit	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathrm	\mathrm	\mathrm	\mathrm	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathsf	\mathsf	\mathsf	\mathsf	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re
\mathtt	\mathtt	\mathtt	\mathtt	()	+	-	=	\xi	\infty	\aleph	\Sigma	\Pi	\Re

Figura 16: Tipografía matemática

\backslashbackepsilon	\ntrianglerightright	\subsetneq
\because	\ntrianglerightrighteq	\subsetneqq
\between	\nvdash	\supset
\blacktriangleleft	\nVdash	\Supset
\blacktriangleright	\nvDash	\supseteq
\bowtie	\nVDash	\supseteqq
\dashv	\parallel	\supsetneq
\frown	\perp	\supsetneqq
\in	\pitchfork	\therefore
\mid	\propto	\trianglelefteq
\models	\shortmid	\trianglerighteq
\ni	\shortparallel	\varpropto
\nmid	\smallfrown	\varsubsetneq
\notin	\smallsmile	\varsubsetneqq
\nparallel	\smile	\varsupsetneq
\nshortmid	\sqsubset	\varsupsetneqq
\nshortparallel	\sqsubseteq	\vartriangle
\subseteq	\sqsupset	\vartriangleleft
\subseteqq	\sqsupseteq	\vartriangleright
\supseteq	\subset	\vdash
\supseteqq	\Subset	\Vdash
\triangleleft	\subseteq	\vDash
\trianglelefteq	\subseteqq	\Vdash

Figura 17: Operadores p/conjuntos (símbolos matemáticos)

\acute{x}	<code>\acute{x}</code>	\tilde{x}	<code>\tilde{x}</code>	\dot{x}	<code>\dot{x}</code>	\check{x}	<code>\check{x}</code>
\grave{x}	<code>\grave{x}</code>	\widetilde{xxx}	<code>\widetilde{xxx}</code>	\ddot{x}	<code>\ddot{x}</code>	\mathring{x}	<code>\mathring{x}</code>
\bar{x}	<code>\bar{x}</code>	\hat{xxx}	<code>\hat{xxx}</code>	\dddot{x}	<code>\dddot{x}</code>	\vec{x}	<code>\vec{x}</code>
\breve{x}	<code>\breve{x}</code>	\widehat{xxx}	<code>\widehat{xxx}</code>	\ddddot{x}	<code>\ddddot{x}</code>		

Figura 18: Vectores (símbolos matemáticos)

Comando	Sinónimo	Espacio	Comando	Sinónimo	Espacio
<code>\thinspace</code>	<code>\,</code>	$\! \!$	<code>\negthinspace</code>	<code>\!</code>	$\! \!$
<code>\medspace</code>	<code>\:</code>	$\! \!$	<code>\negmedspace</code>		$\! \!$
<code>\thickspace</code>	<code>\;</code>	$\! \!$	<code>\negthickspace</code>		$\! \!$

Figura 19: Espacios entre caracteres

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

Figura 20: Alfabeto Griego

Bibliografía y Referencias

- [Cascales Salinas et al., 2003] Cascales Salinas, B., Saorín, P., Mira Ros, J., Pallarés Ruiz, A., and Sánchez-Pedreño, S. (2003). *El Libro de \LaTeX* . Pearson Prentice Hall, Madrid.
- [Foundation, 2008] Foundation, F. S. (2008). *Sitio oficial de la Free Software Foundation*. Sitio web disponible en: <http://www.fsf.org/> (consultado el 21-07-2008).
- [Ivorra, 2008a] Ivorra, C. (2008a). *Algunas cosas útiles sobre el Tex*. Apunte obtenido del sitio web personal de Carlos Ivorra. Disponible en web: <http://www.uv.es/=ivorra/Latex/latex.htm> (consultado el 20-08-2008), y descarga directa disponible en web: <http://www.uv.es/=ivorra/Latex/TeX.pdf> (consultado el 20-08-2008).
- [Ivorra, 2008b] Ivorra, C. (2008b). *Preparación de textos con Latex*. Apunte obtenido del sitio web personal de Carlos Ivorra. Disponible en web: <http://www.uv.es/=ivorra/Latex/latex.htm> (consultado el 20-08-2008), y descarga directa disponible en web: <http://www.uv.es/=ivorra/Latex/LaTeX.pdf> (consultado el 20-08-2008).
- [Latex, 2008] Latex (2008). *Sistema de documentación Latex*. Sitio web disponible en: <http://www.latex-project.org/>. Disponible su descarga en <http://www.latex-project.org/ftp.html> (consultado el 21-07-2008).
- [Mata Botana, 2003] Mata Botana, R. (2003). *Tablas en \LaTeX , Versión 1.00*. Apunte obtenido del sitio <http://www.lug.fi.uba.ar/> perteneciente a la Facultad de Ingeniería (UBA). Disponible en web <http://www.lug.fi.uba.ar/documentos/tablas/tablas.pdf> (consultado el 15-07-2009).
- [Project, 2008] Project, M. (2008). *Sitio oficial del proyecto MikTeX*. Sitio web disponible en: <http://miktex.org/> (consultado el 17-07-2009).