

Tarea 1 - Reporte de L^AT_EX

Kathy Brenes Guerrero, Barnum Castillo Barquero

Maestría en Ciencias de la Computación, Introducción a la Investigación, ITCR

Abstract—One of the biggest issues that an operating system can experience is privilege escalation. Privilege escalation is the act of exploiting a bug, design flaw, or configuration oversight in an operating system or software application to gain elevated access to resources that are normally protected from an application or user. Understanding the weaknesses and flaws of a security level issue for the operating system can help implement better approaches and techniques to improve the software itself. Just because you have updated your computer to the latest update or patch, doesn't mean that it has been secured. Windows, for example, has a series of vulnerabilities that can affect the operating system and can't be solved by Microsoft because the updates can create incompatibilities with an older system or with some security protocols. The Privilege Escalation technique takes advantage of these vulnerabilities to gain privileges (access) within a remote system, in order to run applications and make commands on it. The focus of this paper is to list the vulnerabilities that have been demonstrated by third party systems in different operating system, and provide a technical point of view on what can be done to avoid these breaches (vulnerabilities or impacts). An Operation System breach can enable attackers to increase their level of control over target systems, such that they are free to access any data or make any configuration changes. This study reveals the importance of the way in which current systems should be defended from this mechanism.

Index Terms—Operating System, Penetration Testing, Cyber-security, Internet of Things.

I. INTRODUCCIÓN

El siguiente reporte detalla los conceptos básicos y principales del lenguaje de programación L^AT_EX, además de su historia. Cada una de las siguientes secciones incorpora un ejemplo aplicado de los términos comentados.

II. HISTORIA DE L^AT_EX

TeX es el programa original de composición matemática desarrollado alrededor de 1980 por Donald Knuth para la composición tipográfica digital de alta calidad.[3] TEX es un lenguaje de bajo nivel con el que las computadoras pueden trabajar, pero a la mayoría de las personas les resultaría difícil usarlo; entonces L^AT_EX ha sido desarrollado para hacerlo más fácil.[2]. Se pronuncia como "tecnología" como en alta tecnología. La X es la letra griega chi, que hace que el sonido "ch" aparezca al final de "tech". En la documentación original para TeX, hay mucha discusión acerca de "pegar" objetos juntos y de "estirar" el espacio entre los objetos que componen una página. [3]

El látex es un producto natural pegajoso que forma la base del caucho, y creo que ese es el motivo de la palabra L^AT_EX. L^AT_EX está construido sobre TeX. Fue escrito a principios de

la década de 1980 por Leslie Lamport. Tiene más funciones de alto nivel integradas que TeX, por lo que tiende a ser más fácil de usar. L^AT_EX no es un intento de sonar en francés; probablemente no sea La TeX, o "The TeX". La razón de la divertida alternancia de letras mayúsculas y minúsculas es que TeX y L^AT_EX, cuando se escriben correctamente de esta manera:, son todas letras mayúsculas pero en diferentes tamaños de puntos y diferentes alturas por encima y por debajo de la línea de base. Alternando mayúsculas y minúsculas imita esto. [3]

L^AT_EX se creó para facilitar la producción de libros y artículos de uso general dentro de TeX. Debido a que L^AT_EX es una extensión del sistema de composición tipográfica TeX, tiene la capacidad de TeX para compilar documentos técnicos que contienen ecuaciones matemáticas complejas. Esta característica hizo que L^AT_EX fuera popular entre científicos e ingenieros. [1]

La producción de un documento L^AT_EX comienza con un archivo de texto que contiene contenido etiquetado con códigos especiales L^AT_EX utilizados para indicar cómo se diseñará el texto. Cuando el archivo se ejecuta a través de un procesador L^AT_EX, se producen páginas de composición tipográfica. Debido a que la composición tipográfica L^AT_EX requiere envolver el texto en códigos informáticos complicados, tiene una curva de aprendizaje bastante empinada. Aunque ahora hay programas de software que ayudan a automatizar la creación de documentos L^AT_EX, un conocimiento práctico de L^AT_EX sigue siendo deseable para este tipo de composición tipográfica.

L^AT_EX fue uno de los primeros programas de composición tipográfica capaz de producir ecuaciones matemáticas complejas. Con los años se ha utilizado para componer muchas revistas científicas, matemáticas y de ingeniería. La American Mathematical Society (AMS) incluso tiene su propio conjunto de extensiones, llamado AMS-L^AT_EX, que sus contribuyentes usan para su revista. Pero los programas de autoedición como Quark Inc. de Quark Inc. y FrameMaker de Adobe Systems Incorporated se volvieron más capaces de producir expresiones matemáticas complejas, L^AT_EX se hizo menos popular.[1] La versión actual de L^AT_EX es L^AT_EX2_ε. [2]

III. USOS ACADÉMICOS, EXTENSIÓN, IMPORTANCIA

L^AT_EX es un sistema de preparación de documentos para producir documentos de aspecto profesional, no es un procesador de textos. Es particularmente adecuado para producir documentos largos y estructurados, y es muy bueno para escribir ecuaciones. Está disponible como software libre para

la mayoría de los sistemas operativos.

Si está acostumbrado a producir documentos con Microsoft Word, encontrará que \LaTeX es un estilo de trabajo muy diferente. Microsoft Word es 'Lo que ves es lo que obtienes' (WYSIWYG), esto significa que puedes ver cómo se verá el documento final mientras escribes. Cuando trabaje de esta manera, probablemente realice cambios en la apariencia del documento (como espacios entre líneas, encabezados, saltos de página) mientras escribe. Con \LaTeX no verá cómo se verá el documento final mientras lo está escribiendo; esto le permite concentrarse en el contenido más allá de la apariencia. Para producir esto en la mayoría de los sistemas de tipografía o procesamiento de textos, el autor debería decidir qué diseño usar, por lo que seleccionaría (digamos) 18pt Times Roman para el título, 12pt Times Italic para el nombre, y así sucesivamente. Esto tiene dos resultados: los autores pierden su tiempo con los diseños; y muchos documentos mal diseñados. [2] \LaTeX se basa en la idea de que es mejor dejar el diseño del documento a los diseñadores de documentos y permitir que los autores continúen con la escritura de documentos. [2]

Un documento \LaTeX es un archivo de texto sin formato con una extensión de archivo .tex. Se puede escribir en un editor de texto simple como el Bloc de notas, pero la mayoría de las personas encuentran que es más fácil usar un editor de \LaTeX dedicado. Mientras escribe, marque la estructura del documento (título, capítulos, subtítulos, listas, etc.) con etiquetas. Cuando finaliza el documento, compílelo; esto significa convertirlo a otro formato. [2]

Existen varios formatos de salida diferentes, pero probablemente el más útil sea Portable Document Format (PDF), que aparece tal como se imprimirá y se puede transferir fácilmente entre computadoras. [2]

Implementaciones de \LaTeX

- 1) Composición de artículos de revistas, informes técnicos, libros y presentaciones de diapositivas.
- 2) Control sobre documentos grandes que contienen secciones, referencias cruzadas, tablas y figuras.
- 3) Composición tipográfica de fórmulas matemáticas complejas.
- 4) Composición tipográfica avanzada de las matemáticas con AMS- \LaTeX .
- 5) Generación automática de bibliografías e índices.
- 6) Composición tipográfica multilingüe.
- 7) Inclusión de obras de arte, y color de proceso o mancha.
- 8) Utilizando fuentes PostScript o Metafont.

A. Tipo de archivo

Los archivos \LaTeX no son tan self-contained como, por ejemplo, los documentos de Word. Sin embargo, sus tamaños de archivo son generalmente más pequeños. Aquí hay una descripción muy básica.

Un archivo fuente \LaTeX debe recibir la extensión .tex. Es un archivo de texto, uno que puede editar con Notepad o SimpleText, a diferencia de los documentos de Word, que solo puede editar con Word. El archivo fuente \LaTeX contiene el texto de sus documentos más los comandos que indican cómo deben formatearse el texto y las ecuaciones. De esta manera,

es un poco como código de computadora o HTML sin formato. Los archivos fuente \LaTeX deben ser procesados por un programa que sepa cómo interpretar los diversos comandos integrados en ellos. En una PC, puede descargar varias implementaciones de \LaTeX , como MiKTeX. En Macintosh, puede usar OzTeX. En una máquina UNIX, debería ser capaz de instalar \LaTeX . Me referiré genéricamente a estos programas como \LaTeX . Tenga en cuenta que el programa texify puede ser el programa real que se utiliza para procesar un documento. Cuando latex procesa un archivo fuente \LaTeX llamado example.tex, crea un archivo llamado example.dvi. dvi significa Device Independent. Es un formato anterior a .pdf, y pretendía ser lo que dice, un formato que pueden manejar muchos dispositivos diferentes (plataformas de computadora). Cada plataforma de computadora tiene sus propios programas que pueden ver (algunos dicen vista previa) archivos .dvi. En una PC, MikTeX viene con YAP (Yet Another Previewer), en Macintosh, una vista previa dvi está integrada en OzTeX, y en una máquina UNIX, generalmente puede encontrar un programa llamado xdvi. latex también produce un archivo .aux y .log, del cual no tenemos que preocuparnos ahora.

La progresión habitual al componer un documento en \LaTeX es escribir el archivo .tex usando un editor de texto, luego procesarlo usando latex, luego ver el resultado de .dvi usando un reproductor de vista previa. Comparado con Word, esto parece bastante torpe, lo sé. Se debe principalmente al hecho de que \LaTeX se desarrolló en una plataforma UNIX, y así es como los programas UNIX tienden a estructurarse. Porque es más modular y más fácil tener a muchas personas involucradas en el desarrollo, en lugar de una sola empresa monolítica de desarrollo de software. En balance, eso es algo bueno.

No puede enviar un archivo .dvi a una impresora. La mayoría de las impresoras aceptan entradas PostScript (.ps). En una PC, puede abrir el archivo .dvi con YAP para imprimirlo. Con OzTeX, puede seleccionar Archivo, Imprimir. En UNIX, primero utiliza un programa como dvips para convertir el archivo .dvi a un archivo .ps, luego envía el archivo .ps a la impresora usando un comando como lp file.ps.

Finalmente, una vez que crea un documento \LaTeX , puede compartir el resultado electrónicamente con otros, ya sea por correo electrónico como un archivo adjunto o por publicarlo en una página web. Las personas que no usan TeX o \LaTeX generalmente no tienen instaladas las previsualizaciones .dvi y .ps, pero a menudo tienen Adobe Acrobat Reader, que puede leer y visualizar archivos .pdf (Portable Document Format). En una PC, probablemente use pdflatex para procesar el código fuente de \LaTeX y producir salida .pdf automáticamente. O puede usar un convertidor de DVI a PDF como dvipdfm, que viene con MiKTeX. En un Macintosh, puede usar Adobe Distiller (use Sherlock para buscarlo, luego ejecútelo) para convertir un archivo .ps en un archivo .pdf, luego use Acrobat Reader para verlo. En UNIX, puede usar dvipdf o ps2pdf. La principal desventaja de producir directamente salida .pdf de \LaTeX al editar su archivo fuente .tex es que una vez que Adobe Reader tiene un archivo abierto, latex no puede modificar el archivo en el disco. Las previsualizaciones dvi como YAP no tienen este problema.

IV. ESTILOS DE DOCUMENTO

```
\documentclass[options]{article}
Preamble (for LaTeX commands only)
\begin{document}
Document text (text with embedded LaTeX
commands)
\end{document}
```

Mediante el comando **document class** se determina el diseño y la estructura general del documento. Además la opción **article** permite establecer específicamente los usos y las propiedades que se van a desarrollar, otras clases comúnmente utilizadas son:

A. Article:

como su nombre indica, destinada a escribir artículos. Esto significa documentos relativamente cortos que no contienen capítulos o partes, solo secciones, subsecciones, etc. Como una de las clases base, el formato es bastante básico. Sin embargo, como la clase de artículo proporciona la función básica que la mayoría de la gente espera de LaTeX, a menudo se usa con modificaciones para documentos más largos.

```
\documentclass[opciones]{article}
Preamble (declaraciones : paquetes,
comandos; ttulo, autor, fecha)
\begin{document} Documento
\maketitle
\begin{abstract} ...
\end{abstract}
\section{... }
\subsection{... }
\subsubsection{...}
\end{document}
```

Comandos importantes

- 1) `\maketitle` Hace que se produzcan las líneas para el título, autor y fecha. Debe ubicarse después de `\begin{document}`, si se omite, no se generan dichos campos.
- 2) `\date` Se imprime la fecha vigente del computador, o el valor que se ingrese al campo obligatorio, si se desea que no aparezca de debe escribir `\date`.
- 3) `\thanks{...}` Se puede utilizar en `\title`, `\author`, `\date`, produce notas al pie de página con la información del autor.
- 4) `\begin{abstract}...\end{abstract}` En este entorno se coloca el resumen del artículo y debe ubicarse después de `\maketitle`.
- 5) `\section`, `\subsection` Son numeradas automáticamente.

B. Report:

La clase de informe está destinada a documentos más largos que tendrán capítulos, mientras que el libro está destinado a documentos muy grandes. La configuración estándar para informe y libro es ligeramente diferente de artículo. Por ejemplo, el valor predeterminado para el artículo es poner la información de `\maketitle` en la parte superior de la

primera página, mientras que el informe y el libro usan páginas de título separadas. libro incluye accesos directos predefinidos para `\frontmatter` (capítulos sin numerar con números de página romanos), `\mainmatter` (capítulos numerados y números de página arábiga) y `\backmatter`. Tiene la misma estructura de book. Imprime por una sola cara: `\oneside`. El entorno abstract esta disponible, se crea en una página independiente.

C. Thesis:

Para escribir una tesis de RPI.

D. Book:

Para la estructura, formato y composición de libros.

```
\documentclass{book}
Preamble (declaraciones : paquetes, comandos;
título, autor, fecha)
\begin{document}
\maketitle
\frontmatter
\mainmatter
\chapter{...}
\section{...}
\subsection{...}
\appendix
\backmatter
\end{document}
```

Comandos importantes

- 1) `\frontmatter`: Apertura del libro, se presenta todo aquel contenido que no tenga que ver con el tema central tratado en el libro: prólogo, agradecimientos, tabla de contenido, derechos de autor, índice de figuras, índice de tablas etc. La numeración se realiza utilizando numeración romana.
- 2) `\mainmatter` Contiene la parte central del documento, se desarrolla el tema tratado en el libro, también se ubican los apéndices mediante el comando `\appendix` los cuales se numeran con las letras mayúsculas A, B, C,...
- 3) `\backmatter` Es el cierre del documento, contiene el índice alfabético, bibliografía, conclusiones, reconocimientos, información editorial, etc. Los capítulos no son numerados

E. Letter:

Para redactar el formato de una carta.

```
\documentclass{letter}
\usepackage[spanish]{babel}
\usepackage[latin1]{inputenc}
\begin{document}
\address{
Revista Colombiana de Estadística \\\
Universidad Nacional de Colombia}
\signature{
Pepito Prez \\\
Editor}
\date{30 septiembre de 2006}
\begin{letter}{
Dr. Donald Knuth \\\
```

```
Departamento de Ciencias de la Computación \
Universidad de Stanford \
EEUU
}
```

F. Slides:

Para el diseño y confección de las diapositivas elaboradas con Beamer. El entorno slides da lugar a una transparencia, la numeración es consecutiva en la parte inferior derecha. Slides no se utilizan `\chapter`, `\section`, `\pagestyle`, `\thispagestyle`, ni los entornos `table` y `figure`. Los demás comandos se pueden utilizar libremente.

```
\documentclass{slides}
Preamble (declaraciones : paquetes, comandos;
         ttulo, autor, fecha)
\begin{document}
\begin{slides}
Contenido de la transparencia
\end{slides}
\end{document}
```

Entre otras cosas, las clases proporcionan comandos de encabezado, como `\ part`, `\ chapter`, `\ section`

```
\documentclass[options]{article}
Preamble (for LaTeX commands only)
\begin{document}
Document text (text with embedded LaTeX
              commands)
\end{document}
```

V. USO DE ELEMENTOS

A. Párrafos y Saltos de Línea

Para comenzar un nuevo párrafo en LaTeX se debe dejar una línea en blanco en el medio. Hay otra manera de comenzar un nuevo párrafo, mira el siguiente fragmento de código.

```
This is the text in first paragraph. This is
the text in first
paragraph. This is the text in first
paragraph. \par
This is the text in second paragraph. This is
the text in second
paragraph. This is the text in second
paragraph.
```

This is the text in first paragraph. This is the text in first paragraph. This is the text in first paragraph.

This is the text in second paragraph. This is the text in second paragraph. This is the text in second paragraph.

Como puede ver, el comando `\ par` también comienza un nuevo párrafo.

De forma predeterminada, los párrafos tienen una sangría de 1.5 veces el tamaño del punto de la fuente actual. Además, no hay espacios en blanco adicionales insertados entre los párrafos.

Otra forma de crear párrafos es através de los entornos. Los entornos en LaTeX son zonas del fichero fuente delimitadas

por los comandos `\begin` y `\end` que sirven para dar instrucciones al compilador LaTeX sobre su comportamiento en el interior de éstos. Los entornos tienen la siguiente apariencia: [7]

```
\begin{Entorno}
Texto
\end{Entorno}
```

donde Entorno es el nombre del entorno que queremos utilizar y Texto es el texto al que se le aplican las instrucciones indicadas por el entorno.

Los entornos que se utilizan para la alineación de párrafos son los siguientes:

Entorno `center` para centrar el texto. Entorno `flushleft` para alinear el texto a la izquierda. Entorno `flushright` para alinear el texto a la derecha.

Un ejemplo claro de estos tres tipos es el siguiente:

```
\documentclass{article}
\usepackage[latin1]{inputenc}
\usepackage[spanish]{babel}
```

```
\begin{document}
\begin{flushleft}
Este párrafo está alineado a la izquierda. Los
siguientes párrafos serán una copia de este
cambiando el tipo de alineación.
\end{flushleft}
```

```
\begin{center}
Este párrafo está centrado. Los siguientes
párrafos serán una copia de este cambiando
el tipo de alineación.
\end{center}
```

```
\begin{flushright}
Este párrafo está alineado a la derecha. Los
siguientes párrafos serán una copia de este
cambiando el tipo de alineación.
\end{flushright}
```

Por último tenemos un párrafo cuya alineación es la del entorno por defecto en LaTeX.

```
Esto es todo por hoy.\\Hasta pronto.
\end{document}
```

Que visualmente se puede apreciar de la siguiente manera:

Este párrafo está alineado a la izquierda. Los siguientes párrafos serán una copia de éste cambiando el tipo de alineación.

Este párrafo está centrado. Los siguientes párrafos serán una copia de éste cambiando el tipo de alineación.

Este párrafo está alineado a la derecha. Los siguientes párrafos serán una copia de éste cambiando el tipo de alineación.

Por último tenemos un párrafo cuya alineación es la del entorno por defecto en LaTeX.

Esto es todo por hoy.
Hasta pronto.

Con este ejemplo podemos observar dos formas distintas de realizar un salto de línea:

- 1) Dejando una línea en blanco en el fichero fuente. Así se ha hecho en el ejemplo para que aparezca en otra línea el texto Esto es todo por hoy.
- 2) Añadiendo `\\` al final de la línea. En el ejemplo esto se ha utilizado para añadir un salto de línea entre Esto es todo por hoy y Hasta pronto. En este caso el párrafo siguiente no aparece con sangría en la primera línea.
- 3) Utilizando el comando `\newline`.

Otra forma de alinear un párrafo es por medio de Sangría de párrafo. Por defecto, LaTeX no deja sangría en el primer párrafo de una sección. El tamaño de las sangrías del párrafo posterior está determinado por el parámetro. `\parindent`

```
\setlength{\parindent}{10ex}
This is the text in first paragraph. This is
the text in first
paragraph. This is the text in first
paragraph. \par
\noindent %The next paragraph is not indented
This is the text in second paragraph. This is
the text in second
paragraph. This is the text in second
paragraph.
```

This is the text in first paragraph. This is the text in first paragraph. This is the text in first paragraph. This is the text in second paragraph. This is the text in second paragraph. This is the text in second paragraph.

La longitud predeterminada de este parámetro está establecida por la clase de documento utilizada. Es posible cambiar el tamaño de sangría del párrafo usando el comando `\setlength`. En el ejemplo, los párrafos siguientes `\setlength{\parindent}{10ex}` tendrán una sangría de 10ex (un "ex" equivale a la longitud de la "x" en la fuente actual). Si desea crear un párrafo sin sangría, como el segundo en el ejemplo, puede usar el comando `\noindent` al principio del párrafo.

Si desea sangrificar en un párrafo que no la tiene, puede usar `\indent` encima de él. Se debe tener en cuenta que este comando solo tendrá un efecto cuando `\parindent` no se establezca en cero.

B. Efectos de letra

En este artículo se explicarán tres herramientas básicas de formato de texto: cursiva, negrita y subrayado. Comencemos con un ejemplo:

```
Some of the \textbf{greatest}
discoveries in \underline{science}
were made by \textbf{\textit{accident}}.
```

Some of the **greatest** discoveries in science were made by *accident*.

Como puede ver, hay tres comandos básicos y se pueden anidar para obtener efectos combinados.

Cursiva: para que un texto en cursiva sea sencillo, utilice el comando `\textit`.

Negrita: para hacer un texto en negrita, use el comando `\textbf`.

`\textbf`.

Subrayado: Resaltar texto: también es muy simple, use el comando `\underline`.

Énfasis en el texto: el texto se puede enfatizar usando el comando `\emph`. A veces, el comando `\emph` se comporta igual que `\textit`, pero no es exactamente el mismo.

C. Tildes

Se recomienda utilizar comandos como `\`a` para producir la letra acentuada "á". E incluso se menciona al paquete `spanish` de `babel` que te permite simplificar esto un poco y escribir simplemente ``a`.

D. Títulos y Subtítulos

La mayoría de los títulos y subtítulos se establecen por medio de los Chapters o Sections que se desglosan a su vez en sub capítulos y sub secciones que forman a ser parte de los subtítulos como lo podemos visualizar en el siguiente ejemplo.

```
\section{Titulo}
\subsection{Subtitulo}
```

E. Referencias

Para las referencias se puede utilizar el entorno `thebibliography` que se emplea de la siguiente manera:

```
\begin{thebibliography}{1}
\bibitem{[1]}The Editors of Encyclopaedia
Britannica (2013) \emph{LaTeX COMPUTER
PROGRAMMING LANGUAGE} [Blog post].
Consultado desde ...
\bibitem{[2]} \emph{Introduction to LaTeX.}
(2018). Consultado desde ...
\bibitem{[3]} \emph{Basic description of file
types and how LaTeX works.} (2018).
Consultado desde ...
\bibitem{a} \emph{Latex-Tutorial.} (2018).
Consultado desde ...
\bibitem{b} \emph{ShareLatex.} (2018).
Consultado desde ...
\bibitem{c} \emph{Sascha-Frank.} (2018).
Consultado desde ...
\bibitem{[7]} \emph{Alineacin de prrafos.}
(2018). Consultado desde ...
\end{thebibliography}
```

En el ejemplo anterior se puede observar que los caracteres que se encuentran entre `{}` corresponden al nombre del elemento que se desea citar, para utilizarlo se emplea el uso de `\cite{ nombre de la referencia}`.

F. Marcas de agua

Se utiliza el paquete `\usepackage{draftwatermark}` para insertar la marca de agua en cada página. Por defecto, la marca de agua se agrega a todas las páginas. Para insertar solo en la primera página, se utiliza `\usepackage[firstpage]{draftwatermark}`. El

texto predeterminado de la marca de agua es DRAFT. Para insertar un texto de marca de agua diferente se utiliza `\SetWatermarkText {Text Reporte 1 :) }`

Para cambiar la claridad u oscuridad se utiliza el comando `\SetWatermarkLightness {0.5}` El valor de luminosidad oscila entre 1.0 para blanco y 0.0 para negro. Por defecto está configurado a 0.8.

G. Headers y Footers

LaTeX tiene algunos estilos predeterminados que cambian la forma en que se muestran el encabezado y el pie de página. El pie de página y el encabezado también se pueden personalizar para adaptarse a cualquier diseño en particular. La información que se muestra en el pie de página y el encabezado de un documento depende del estilo de página actualmente activo, estos estilos de página son más notorios en la clase de documento del libro.

```
\documentclass[a4paper,12pt,twoside]{book}
\usepackage[english]{babel}
\usepackage[utf8]{inputenc}

\pagestyle{headings}

\begin{document}
\chapter{Sample Chapter}
\section{New section}

Hello, here is some text without a meaning.
  This text should
show what aprinted text will look like at
  this place. If you
read this text, you will get noinformation.
  Really? Is there
no information? Is there a dience betweenthis
  text and some
nonsense like Huardest gefburn? Kjift { not
  at all!...
```

El comando `\pagestyle {encabezados}` establece el estilo de página llamado encabezados en el documento actual.

H. Columnas de la página

Los documentos de dos columnas se pueden crear fácilmente pasando el parámetro `\twocolumn` a la declaración de la clase del documento. Si necesita más flexibilidad en el diseño de la columna, o para crear un documento con varias columnas, el paquete `multicol` proporciona un conjunto de comandos para eso. Una herramienta flexible para manejar documentos de múltiples columnas en LATEX es `multicol`, por ejemplo:

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}

\usepackage{multicol}

\begin{document}
\begin{multicols}{3}
```

```
[
\section{First Section}
All human things are subject to decay. And
  when fate summons, Monarchs must obey.
]
Hello, here is some text without a meaning.
  This text should show what
a printed text will look like at this place.
If you read this text, you will get no
  information. Really? Is there
no information? Is there...
\end{multicols}

\end{document}
```

Para importar el paquete, la línea `\usepackage {multicol}` se agrega al preámbulo. Una vez que se importa el paquete, se pueden usar las multicolumnas del entorno. El entorno toma dos parámetros.

Número de columnas, este parámetro debe pasarse dentro de llaves. Su valor es 3 en el ejemplo. Cualquier comando LATEX se puede usar aquí, excepto elementos flotantes como figuras y tablas. En el ejemplo, el título de la sección y un pequeño párrafo se establecen aquí. El texto incluido dentro de las etiquetas `\begin {multicols}` y `\end {multicols}` se imprime en formato de varias columnas.

Separación de columnas: la separación de columnas está determinada por `\columnsep`.

```
\documentclass{article}
\usepackage[utf8]{inputenc}
\usepackage[english]{babel}

\usepackage{multicol}
\setlength{\columnsep}{1cm}

\begin{document}
\begin{multicols}{2}
[
\section{First Section}
All human things are subject to decay. And
  when fate summons, Monarchs must obey.
]
Hello, here is some text without a meaning.
  This text should show what
a printed text will look like at this place.
If you read this text, you will get no
  information. Really? Is there
no information? Is there...
\end{multicols}

\end{document}
```

Aquí, el comando `\setlength {\columnsep} {1cm}` establece la separación de la columna en 1 cm.

VI. MANEJO DE TABLAS

Durante esta sección se estarán discutiendo el atributo de Latex de creación de tablas, partiendo del ejemplo de “Tabla básica” incluyendo su código en Latex hasta las diferentes características para personalizarla según las necesidades.

A. Tabla Básica

La siguiente tabla refleja la implementación más básica, por razones de claridad se le ha agregado las líneas negras.

Columna 1	Columna 2	Columna 3
Fila 11	Fila 12	Fila 13
Fila 21	Fila 22	Fila 23

El código de la tabla es el siguiente:

```
\begin{table}[H]\centering
\begin{tabular}{|l c|r|}
\hline
Columna 1 & Columna 2 & Columna 3 \\ \hline
Fila 11 & Fila 12 & Fila 13 \\ \hline
Fila 21 & Fila 22 & Fila 23 \\ \hline
\end{tabular}
\end{table}
```

El comienzo de la tabla se representa con `\begin{tabular}`, dentro de esta sección se describe el contenido de la tabla (Columna 1, Columna 2, etc), la separación de la columna se da por `&` y el de cada fila por `\\`.

La alineación del texto se da por `{|l|c|r|}` siendo l = left, c = center, r = right y los símbolos pipe (“|”) representa el delineado de las columnas, para delinear las líneas horizontales se hace uso del `\hline` al final de cada fila.

B. Agregar columnas y filas

Tomando el ejemplo de tabla básica y agregándole una nueva columna y una nueva fila da como resultado el siguiente código.

```
\begin{table}[H]\centering
\begin{tabular}{|l|l|l|l|}
\hline
Columna 1 & Columna 2 & Columna 3 & C.Nueva \\ \hline
Fila 11 & Fila 12 & Fila 13 & \\ \hline
Fila 21 & Fila 22 & Fila 23 & \\ \hline
Nueva fila & & & \\ \hline
\end{tabular}
\end{table}
```

Las diferencias notorias como resultado de agregar la nueva fila y columna son “Nueva fila & & & \\hline” al final del código que representa a la fila y en todas las columnas se ha agregado un `&` dando la siguiente tabla.

Columna 1	Columna 2	Columna 3	C.Nueva
Fila 11	Fila 12	Fila 13	
Fila 21	Fila 22	Fila 23	
Nueva fila			

C. Múltiples columnas

Para el proceso de combinar columnas en una misma fila es necesario el comando:

```
\multicolumn{ancho}{alineamiento}{contenido}
```

Teniendo en cuenta la línea anterior y si lo agregamos al código de tabla básica resulta en:

Columna 1	Columna 2	Columna 3
Fila 11 + fila 12		Fila 13
Fila 21	Fila 22	Fila 23

Conociendo la función del comando `\multicolumn`, podemos ver el código agregado al de tabla básica y explicar su funcionamiento.

```
\begin{table}[H]\centering
\begin{tabular}{|l|l|l|}
\hline
Columna 1 & Columna 2 & Columna 3 \\ \hline
\multicolumn{2}{|l|}{Fila 11 + fila 12} & Fila 13 \\ \hline
Fila 21 & Fila 22 & Fila 23 \\ \hline
\end{tabular}
\end{table}
```

`{2}` es la cantidad de columnas que se expandió la celda, `{|l|}` es la delineación vertical de la celda más la alineación del texto y el último parámetro es el contenido “Fila 11 + fila 12”.

D. Múltiples filas

La funcionalidad de combinar filas es dada por el paquete.

```
\usepackage{multirow}
```

La siguiente tabla ejemplifica su uso.

Columna 1	Columna 2	Columna 3
Fila 11 + fila 21	Fila 12	Fila 13
	Fila 22	Fila 23

```
\begin{table}[H]\centering
\begin{tabular}{|l|l|l|}
\hline
Columna 1 & Columna 2 & Columna 3 \\ \hline
\multirow{2}{*}{Fila 11 + fila 21} & Fila 12 & Fila 13 \\ \cline{2-3}
& Fila 22 & Fila 23 \\ \hline
\end{tabular}
\end{table}
```

Como se puede apreciar `\multirow` tiene los mismos parámetros que `\multicolumn` con la diferencia en que `{2}` indica la cantidad de filas a ser combinadas.

E. Colores celdas

Para agregar color a la tabla basta el uso de dos comandos `\rowcolor` y `\cellcolor`, en el siguiente código se muestra la ubicación correspondiente de cada uno.

```
\begin{table}[H]\centering
\begin{tabular}{|l|l|l|}
\hline
\rowcolor[HTML]{FFCE93}
\end{tabular}
```



```

Columna 1 & Columna 2 & Columna
3 \\ \hline
\cellcolor[HTML]{3166FF}Fila 11 & Fila 12 &
Fila 13 \\ \hline
Fila 21 & Fila 22 & Fila 23
\\ \hline
\end{tabular}
\end{table}

```

Columna 1	Columna 2	Columna 3
Fila 11	Fila 12	Fila 13
Fila 21	Fila 22	Fila 23

Se puede mostrar que ambos comandos hacen uso del código del color, pero se diferencian en que `\cellcolor` sólo es aplicado a una celda y `\rowcolor` a toda la fila.

VII. MANEJO DE FIGURAS Y GRÁFICOS

En esta sección del documento se explica las funcionalidades de LaTeX con respecto al manejo de gráficos.

A. Directorio de imágenes

Cuando se agrega una referencia a una imagen en un documento .tex, esta cuando sea compilada por el LaTeX buscará a la imagen en el directorio en el que se encuentra el .tex. A continuación, se presentan las formas en las que se puede modificar dicho directorio.

```

%Ruta relativa al archivo .tex
\graphicspath{ {./images/} }

```

```

%Ruta absoluta
\graphicspath{ {c:/user/images/} }

```

Ambas formas tienen que incorporarse al principio del .tex.

B. Modificar tamaño y rotación

La imagen por insertar en el documento tiene que existir en directorio de imágenes especificado en el LaTeX. El código para insertarlo es el siguiente.

```

\includegraphics[scale=1]{latex-logo}

```

Se recalca que la imagen latex-logo no tiene extensión de archivo. La imagen a continuación es el resultado del código anterior y será utilizada como punto de referencia para las demás modificaciones.



En el código el tamaño de la imagen puede ser modificado por `[scale=1]` con un valor mayor o menor, o puede ser cambiado por `[width=x cm, height= x cm]`.

```

\includegraphics[width=3cm,
height=4cm]{latex-logo}

```



El cambio se observa como:

```

\includegraphics[width=\textwidth]{latex-logo}

```

Como adición a lo anterior, se puede tomar el ancho del texto como referencia para el ancho de la imagen con `\textwidth`.

La modificación de la orientación de la imagen se ve cambiado por el parámetro `angle` en términos de 360 grados.

```

\includegraphics[scale=0.5,
angle=45]{latex-logo}

```



C. Posicionamiento

```

\begin{figure}[posicionamiento]
\includegraphics[width=8cm]{latex-logo}
\end{figure}

```

Texto

Parámetro	Posicionamiento
n	Coloque el flotador aquí, es decir, aproximadamente en el mismo punto en el que aparece el texto fuente (sin embargo, no exactamente en el punto).
t	Posición en la parte superior de la página.
b	Posición en la parte inferior de la página.
p	Se pone en una página especial para flotadores solamente.
!	Anula los parámetros internos que LaTeX esté utilizando.
	Coloca el flotador exactamente en la ubicación del código LaTeX. Requiere el paquete flotante. Esto es algo equivalente a h!.
H	

D. Referenciar figuras

Texto

```
\begin{figure}[h]
\centering
\includegraphics[width=0.25\textwidth]{grafico}
\label{fig:refejeml}
\end{figure}
```

Como puede ver en la figura
`\ref{fig:refejeml}`, en la P.#
`\pageref{fig:refejeml}`.

Las referencias una vez compiladas se ven como:

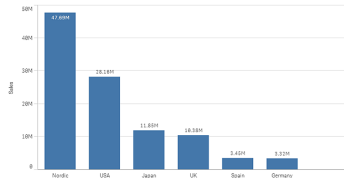


Fig. 1: refejeml

En la figura 1 en la P.# 9.

VIII. MANEJO DE FIGURAS AL LADO DE TABLAS (MINIPAGE)

El minipage es una funcionalidad de Latex utilizada para poner lado a lado estructuras que de otra forma sería difícil, su armazón en código se ve como:

```
\begin{minipage}[ajuste]{ancho del minipage}
  Texto ... \ \
  Imagenes ... \ \
  Tablas ... \ \
\end{minipage}
```

Partiendo de esta implementación se pueden crear diferentes configuraciones.

Ejemplo de tabla + tabla.

```
\begin{minipage}{0.2\textwidth}
\begin{tabular}{|c|c|c|}
\hline
A & B & C \\
\hline
1 & 2 & 3 \\
\hline
4 & 5 & 6 \\
\hline
\end{tabular}
\end{minipage}
\begin{minipage}{0.2\textwidth}
\begin{tabular}{|c|c|c|}
\hline
A & B & C \\
\hline
1 & 2 & 3 \\
\hline
4 & 5 & 6 \\
\hline
\end{tabular}
\end{minipage}
```

A	B	C
1	2	3
4	5	6

A	B	C
1	2	3
4	5	6

Ejemplo de imagen + texto.

```
\begin{minipage}[t]{0.2\textwidth}
\includegraphics[scale=0.4]{tec-logo}
\end{minipage}
\begin{minipage}{0.2\textwidth}
Tecnologico de CR\\
Tecnologico de CR\\
Tecnologico de CR\\
Tecnologico de CR\\
Tecnologico de CR\\
\end{minipage}
```



Tecnológico de Costa Rica

Tecnológico de CR
Tecnológico de CR
Tecnológico de CR
Tecnológico de CR
Tecnológico de CR

Ejemplos de imagen + imagen + imagen.

```
\begin{minipage}[b]{0.15\textwidth}
\includegraphics[width=\textwidth]{tec-logo}
\end{minipage}
\begin{minipage}[t]{0.15\textwidth}
\includegraphics[width=\textwidth]{tec-logo}
\end{minipage}
\begin{minipage}[t]{0.15\textwidth}
\includegraphics[width=\textwidth]{tec-logo}
\end{minipage}
```



Tecnológico de Costa Rica



Tecnológico de Costa Rica



Tecnológico de Costa Rica

IX. ECUACIONES MATEMÁTICAS

En este segmento se presentan las características básicas para generar fórmulas matemáticas en Latex.

A. Modo matemáticas

Latex permite dos tipos de fórmulas: inline y display, la primera permite escribir la fórmula en línea con el texto y la segunda lo opuesto, por lo que cuando se escriba ocupará una línea completa por si sola.

Las fórmulas “inline” se caracterizan por estar delimitadas por $\$formula\$$ y las “display” por $\\$formula\\$$ o $\\begin{equation} formula \\end{equation}$. Cabe recalcar que en el modo inline las expresiones son compresas para seguir con la línea del texto.

Ejemplo de inline:

La formula $E=MC^2$ fue formulada por Einstein anos despues de presentar su teoria de la relatividad.

La fórmula $E = MC^2$ fue formulada por Einstein años después de presentar su teoría de la relatividad.

Ejemplo de display:

La formula $E=MC^2$ fue formulada por Einstein anos despues de presentar su teoria de la relatividad.En unidades naturales ($c = 1$), representa la identidad

```
\begin{equation}
E = m
\end{equation}
```

La fórmula

$$E = MC^2$$

fue formulada por Einstein años después de presentar su teoría de la relatividad.En unidades naturales ($c = 1$), representa la identidad

$$E = m \quad (1)$$

B. Símbolos especiales

La utilización de símbolos especiales se da por el modo matemática explicado en el punto anterior, solamente basta escribir $\backslash simbolo$ dentro de los $\$ \$$.

$\delta \alpha$

$$\delta \alpha$$

C. Fracciones

Las fracciones pueden ser utilizadas en conjunto con el texto $\frac{1}{2}$ o por si solas con los modos inline y display.

$$\frac{1}{2}$$

Las fracciones son muy versátiles, pudiendo ser anidadas para expresiones complejas.

Ejemplo sencillo:

$\frac{1}{2}$

Ejemplo anidado:

$\frac{1 + \frac{a}{b}}{1 + \frac{1}{1 + \frac{1}{a}}}$

Ejemplo sencillo:

$$\frac{1}{2}$$

Ejemplo anidado:

$$\frac{1 + \frac{a}{b}}{1 + \frac{1}{1 + \frac{1}{a}}}$$

D. Operadores

El manejo de operadores es muy variado por lo que a continuación se explican los más utilizados.

$$\int_a^b x^2 dx$$

Las integrales como la anterior son definidas en Latex como:

$\int_{\text{inferior}}^{\text{superior}}$

$\int_a^b x^2 dx$

Integrales anidadas pueden resultar ms complejas, pero se pueden obtener modificando el inicio de la expresin (int) como en los siguientes ejemplos:

$\iint_V \mu(u, v) \, du \, dv$
 $\iiint_V \mu(u, v, w) \, du \, dv \, dw$
 $\int \dots \int_V \mu(u_1, \dots, u_k) \, du_1 \dots du_k$

$$\iint_V \mu(u, v) \, du \, dv$$

$$\iiint_V \mu(u, v, w) \, du \, dv \, dw$$

$$\int \dots \int_V \mu(u_1, \dots, u_k) \, du_1 \dots du_k$$

Al igual que las integrales, las sumatorias están dadas por:

$\sum_{\text{inferior}}^{\text{superior}}$

Ejemplo:

$\sum_{n=1}^{\infty} 2^{-n} = 1$ inside text

$$\sum_{n=1}^{\infty} 2^{-n} = 1$$

En el caso de los límites se utiliza el comando

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

Ejemplo:

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

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

X. MANEJO DE COLORES

El manejo de colores en Latex está dado por los paquetes

```
\usepackage{color}
\usepackage{xcolor}
```

Ambos paquetes permiten la manipulación de colores, pero a la vez tiene la flexibilidad de modificar los colores y secciones a gusto. El siguiente ejemplo muestra los colores predefinidos.

```
\begin{itemize}
\color{blue}
\item Primer item
\item Segundo item
\end{itemize}
```

```
\noindent
{\color{red} \rule{\linewidth}{0.5mm} }
```

- Primer item
 - Segundo item
-

Los colores red y blue está predefinidos en el paquete color, algunos otros ejemplos básicos son.

```
El color del texto puede ser cambiado a
\textcolor{red}{rojo}. Tambien puede
cambiar el background del
\colorbox{BurntOrange}{texto}.
```

El color del texto puede ser cambiado a **rojo**. Tambien puede cambiar el background del **texto**.

Para una mayor variedad de colores se pueden crear variables de color dándole una etiqueta a un valor RGB.

```
\definecolor{rosado}{rgb}{0.858, 0.188, 0.478}
\definecolor{gris}{gray}{0.6}
```

- 1) **Rosado**
- 2) **Gris**

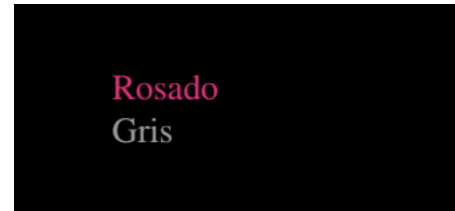
El definecolor utiliza una de las siguientes 4 opciones para el primer parámetro.

- 1) **rgb**: rojo, verde, azul. Tres valores separados por comas entre 0 y 1 definen los componentes del color.
- 2) **RGB**: lo mismo que rgb, pero los números son enteros entre 0 y 255.
- 3) **cmyk**: cian, magenta, amarillo y black. Lista de cuatro números separados por comas entre 0 y 1 que determina el color de acuerdo con el modelo aditivo utilizado en la mayoría de las impresoras.

- 4) **gray**: escala de grises. Un solo número entre 0 y 1.

Además de modificar el color del texto de puede cambiar el color de la página con

```
\pagecolor{black}
```



REFERENCES

- [1] The Editors of Encyclopaedia Britannica (2013) *LaTeX COMPUTER PROGRAMMING LANGUAGE* [Blog post]. Consultado desde <https://www.britannica.com/technology/LaTeX-computer-programming-language>
- [2] *Introduction to LaTeX*. (2018). Consultado desde <https://www.latex-project.org/about/>
- [3] *Basic description of file types and how LaTeX works*. (2018). Consultado desde http://personal.bgsu.edu/~zirbel/5920/latex/latex_basics.htm
- [4] *Latex-Tutorial*. (2018). Consultado desde <https://www.latex-tutorial.com>
- [5] *ShareLatex*. (2018). Consultado desde <https://www.sharelatex.com/learn>
- [6] *Sascha-Frank*. (2018). Consultado desde <http://www.sascha-frank.com>
- [7] *Alineacin de prrafos*. (2018). Consultado desde <https://laxtips.wordpress.com/2009/01/27/alineacion-de-parrafos/>