A Short Introduction to LaTeX

Swalpa Kumar Roy, Ph.D
Personal Website: `https://swalpa.github.io`

CSE Department
Jalpaiguri Government Engineering College

January 13, 2022

NEED

- We produce a lot of documents for diverse academic purposes; e.g.,
  - Report (Dissertation, thesis and similar other materials)
  - Presentation
  - Book
  - Article (The so called 'Research paper')
- The word-processor program that we use offers a limited amount of typesetting options leading to
  - Symbol based non-structured Mathematics.
  - Very little reference, citation and indexing capabilities.
  - Poor and non-professionally structured document.

Word vs. LaTeX
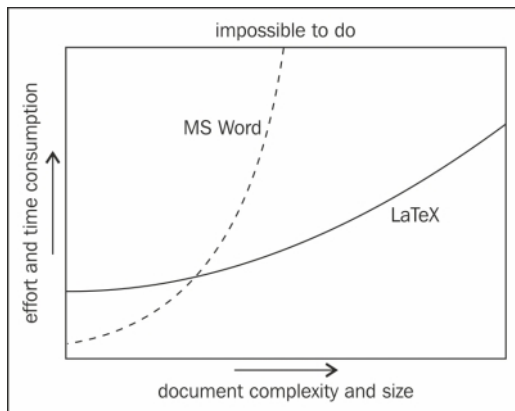
Word is nice for making a quick little document, but when it comes to large complex ones with strict structuring rules LaTeX provides benefits and features that are hard to come across.

What is LaTeX ?

- TeX is a typesetting program
- It is created (released in 1982) by famous computer scientist Donald E Knuth.
  - ▸ For the printing industry who were using digital technology but poor old typesetting standards.
  - ▸ To see his own papers/books in a properly printed form
- LaTeX (Developed by Leslie Lamport in 1989) is an easier form that uses TeX for typesetting.

Steps to create a document in general

- Type the text using Notepad, word etc.
- The s/w helps formatting the texts into paragraphs, lines and pages.
- The written content is displayed (in a what you see what you get style).
- The document (a page or more) is printed.

Steps to create a document in LaTeX

LATEX is concerned more on text formatting; so

- Type the text as well as the text formatting commands in any text editor.
- Compile (latex filename.tex) the document to get type-set page(s).
- Display the type-set page(s) using a special previewer (say, xdvi filename.dvi)
- Convert the document to PDF form (latex2pdf filename) to take a print.

Is LaTeX worth doing?

Yes; it is. If

- you are creating a
  - complex scientific or engineering document
  - with a lot of mathematics, references and cross refernence, TOC, Index etc.
- you are a purist
  - with strong aesthetic and
  - a passion to excel not only in the contents (though most important) and
  - like to establish the underlying structure that conforms to the heighest professional typesetting standrd.

So;

It is worth doing and you will be convinced if you

- stay focussed up to the end of this session and
- you care to produce (typeset) a document in latex.

Are you A JGEC STUDENT

Guess Why?

So;

It is worth doing and you will be convinced if you

- stay focussed up to the end of this session and
- you care to produce (typeset) a document in latex.

Are you A JGEC STUDENT

Guess Why?

What you have is only "the structure" – and "nothing in-between"

Beginning

- The input file (filename.tex) is a text file.
- So, it is highly portable and the input file can be prepared using any text editor.
- Text (the contents) and commands (start with \\) are mingled. .
- The following characters have special meaning and cannot be produced directly; they are:

(i) # (ii) $ (ii) % (iv) ^ (v) & (vi) _ (vii) { (viii) } and (ix) \\

- They must be preceded by a \\

The escape sequence \\{ would produce { only. and for the symbol \\ you need to type \\textbackslash.

A Starting Document

\documentclass[OPTIONS]{TYPES}
\begin{document}

\end{document}

The OPTIONS are numerous; like
- 12pt
- onecolumn
- a4wide; and so many

The TYPES are primarily
- Article
- Book
- Slides
- Report and
- Letter

Unconstrained Input

You are supposed to type the text and commands freely. The form of the input file will be totally changed (typesetted) by the LATEXcompiler. Note

- More than one spaces are treated as a single space; and
- a blank line indicates the beginning of a new paragraph.
- Several blank lines are also treated as a single blank

```
It does not matter whether you
enter one or several                    spaces
after a word in your text.


An empty line starts a new
paragraph. Multiple empty lines are ignored.
```

Sample Page: After Typesetting

It does not matter whether you enter one or several spaces after a word in your text.
An empty line starts a new paragraph. Multiple empty lines are ignored.

If you do not want any typesetting or the special characters use

```
\begin{verbatim}
Anything in any form here including the special charcaters.
\end{verbatim}
```

The effect is shown next.

No Typesetting

```
ABC    def   # $ / y3t  md nmcd \\

\\\\\ ........
sfdjsdkj 57801'2kjdv
+  gfdfd &  \begin{document}
-> <-
```

Font Size Control

The font size for the normal text as a whole can be set by:

$\backslash$**documentstyle[12pt]**$\{$**report**$\}$

Note: That appropriate text sizes are automatically selected for headings, sections etc. to the default settings as we shall see later.

Other sizes can be controlled locally by:

$\{$ $\backslash$large Text$\}$, Text

$\{$ $\backslash$tiny Text$\}$, Text

$\{$ $\backslash$Huge Text$\}$, Text

Other Common text forms

Creating Bold-face text, emphasized (italics) text and underlined texts;

1. **Bold** {\bf Bold} 2. *Emphasized* {\em Emphasized} 3. <u>Underline</u> \underline{Underline}

As well as centered text

\centerline{Text is in the centre}

Text is in the centre

\begin{center} Line 1\\ Line 2 \end{center}

Line 1
Line 2

Common Structural Requirements

```
Leftmost portion        \hfill        Rightmost portion
```

Leftmost portion   Rightmost portion

```
Big Gap        \hspace{3cm}        Rightmost portion
```

Big Gap   Rightmost portion

```
Big Gap\\
 \vspace{1cm}
below
```

Big Gap

below

Emphasising Structures ... contd.

To get the following effect
- First
  - First.1
    - First.1.1
  - First.2
- Second

Emphasising Structures ... contd.

Another very common scenario is structure with Numbers and alphabets
to show the main and sub-structures.

```
\begin{enumerate}
  \item First
  \begin{enumerate}
     \item First.1
       \begin{itemize}
          \item First.1.1
       \end{itemize}
   \item First.2
 \end{enumerate}
\item Second
\end{enumerate}
```

Ephasizing Structures ... contd.

The result is

```
1.   First
      1.1   First.1
            1.1.1 First.1.1
      1. 2   First.2
2. Second
```

Mathematical Formula

```
1. The formula $x_i=\sqrt{y_i^2 + z_i^2}$ : in-line Math form.
2. The  same  \[ x_i=\sqrt{y_i^2 + z_i^2 \] in
displayed-Math form.
```

1. The formula $x_i = \sqrt{y_i^2 + z_i^2}$: in-line Math form.
2. The same

$$x_i = \sqrt{y_i^2 + z_i^2}$$

in displayed-Math format.

Mathematics: Contd

Fractions are written by the command: **frac**{**numerator**}{**denominator**}

So the following in displayed Math form

\\[f(x) = \sum_{i=1}^4 \frac{x^i}{i} \\]
would produce

$$f(x) = \sum_{i=1}^{4} \frac{x^i}{i}$$

A Simple Test

What it would produce?

```
\begin{equation}
\lim_{n \to \infty}
\sum_{k=1}^n \frac{1}{k^2}
= \frac{\pi^2}{6} \nonumber
\end{equation}
```

A Simple Test

What it would produce?

```
\begin{equation}
\lim_{n \to \infty}
\sum_{k=1}^n \frac{1}{k^2}
= \frac{\pi^2}{6} \nonumber
\end{equation}
```

$$\lim_{n\to\infty} \sum_{k=1}^{n} \frac{1}{k^2} = \frac{\pi^2}{6}$$

More Tests

1.

$$w(x, y) * f(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x + s, y + t)$$

More Tests

1.

$$w(x, y) * f(x, y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x + s, y + t)$$

`\[w(x,y)*f(x,y)=\sum_{s=-a}^a\sum_{t=-b}^b w(s,t)f(x+s,y+t)\]`

More Tests

1.

$$w(x,y) * f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

\[w(x,y)*f(x,y)=\sum_{s=-a}^a\sum_{t=-b}^b w(s,t)f(x+s,y+t)\]

2. $f(x,y) = \int_0^\pi f_\theta(x,y) d\theta$

More Tests

1.
$$w(x,y) * f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)f(x+s,y+t)$$

`\[w(x,y)*f(x,y)=\sum_{s=-a}^a\sum_{t=-b}^b w(s,t)f(x+s,y+t)\]`

2. $f(x,y) = \int_0^\pi f_\theta(x,y)d\theta$

`$f(x,y)=\int_0^\pi f_\theta (x,y)d\theta$`

More Tests

1.
$$w(x,y) * f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)$$

\[w(x,y)*f(x,y)=\sum_{s=-a}^a\sum_{t=-b}^b w(s,t)f(x+s,y+t)\]

2. $f(x,y) = \int_0^{\pi} f_{\theta}(x,y) d\theta$

$f(x,y)=\int_0^\pi f_\theta (x,y)d\theta$

3. $f(x,y) = \int_0^{2\pi} \int_0^{\infty} F(\omega cos\theta, \omega sin\theta) e^{j2\pi\omega(xcos\theta + ysin\theta)} \omega d\omega d\theta$

More Tests

1.

$$w(x,y) * f(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)f(x+s,y+t)$$

`\[w(x,y)*f(x,y)=\sum_{s=-a}^a\sum_{t=-b}^b w(s,t)f(x+s,y+t)\]`

2. $f(x,y) = \int_0^\pi f_\theta(x,y)d\theta$

`$f(x,y)=\int_0^\pi f_\theta (x,y)d\theta$`

3. $f(x,y) = \int_0^{2\pi} \int_0^\infty F(\omega cos\theta, \omega sin\theta)e^{j2\pi\omega(xcos\theta+ysin\theta)}\omega d\omega d\theta$

```
$f(x,y)=\int_0^{2\pi} \int_0^\infty F(\omega cos \theta,
 \omega sin \theta)
 e^{j2\pi \omega (x cos\theta + y sin \theta)}
 \omega d \omega d \theta $
```

More Examples

```
$\underbrace{\overbrace{a+b+c}^6\cdot \overbrace{d+e+f}^9}
_{\rm  meaning\  of\  life} = 42$
```

More Examples

```
$\underbrace{\overbrace{a+b+c}^6\cdot \overbrace{d+e+f}^9
_{\rm  meaning\  of\  life} = 42$
```

$$\underbrace{\overbrace{a+b+c}^6 \cdot \overbrace{d+e+f}^9}_{\text{meaning of life}} = 42$$

```
\[ f(x) = \sum_{i=1}^4 \frac{x^i}{i} \]
```

More Examples

```
$\underbrace{\overbrace{a+b+c}^6\cdot \overbrace{d+e+f}^9
_{\rm  meaning\  of\  life} = 42$
```

$$\underbrace{\overbrace{a+b+c}^{6}\cdot\overbrace{d+e+f}^{9}}_{\rm meaning\ of\ life} = 42$$

```
\[ f(x) = \sum_{i=1}^4 \frac{x^i}{i} \]
```

$$f(x) = \prod_{i=1}^{4} \frac{x^i}{i}$$

Equation

$$s = 1 + \frac{1}{1 + \frac{1}{1+x}} \qquad (1)$$

```
\begin{eqnarray}
 s & = & 1 + \frac{1}{1 + {\frac{1}{1+x}}}
\end{eqnarray}

$\tau\epsilon\chi$\\
100~m$^{3}$ of water\\[15pt]
This comes from my $\diamondsuit$
```

$\tau\epsilon\chi$
100 m^3 of water

This comes from my $\diamondsuit$

Tabular Form

Tables are often needed and are represented in a tabular form as below:

```
\begin{tabular}{|l|c|r|} \hline
left & Centered & Right \\ \hline
l L  & c C & r R \\ \hline
\end{tabular}
```

Tabular Form

Tables are often needed and are represented in a tabular form as below:

```
\begin{tabular}{|l|c|r|} \hline
left & Centered & Right \\ \hline
l L  & c C & r R \\ \hline
\end{tabular}
```

left	Centered	Right

Tabular Form

Tables are often needed and are represented in a tabular form as below:

```
\begin{tabular}{|l|c|r|} \hline
left & Centered & Right \\ \hline
l L  & c C & r R \\ \hline
\end{tabular}
```

left	Centered	Right
l L	c C	r R

Table with caption and reference mechanism

```
\begin{table}[h]
\centering
\begin{tabular}{|l|c|r|} \hline
left & Centered & Right \\ \hline
l L  & c C & r R \\ \hline
\end{tabular}
\caption{My first Table}
\label{table1}
\end{table}
```

left	Centered	Right
l L	c C	r R

Table 1: My first Table

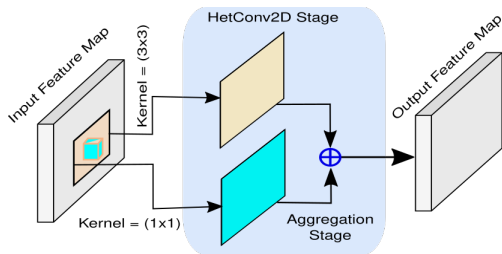To refer write: Did I refer Table \ref{table1}? to get Did I refer Table 1?

Figures

Figure may be included in a similar manner by using the package in the beginning

\usepackage{graphicx}

folowed by the following commands in the appropriate location of the document

\includegraphics[scale=factor]{figfilename}

Figures with captions etc.

```
\begin{figure}[h]
\centering
\includegraphics[scale=factor]{figfilename}
\caption{My First Figure}
\label{figLabel}
\end{figure}
```
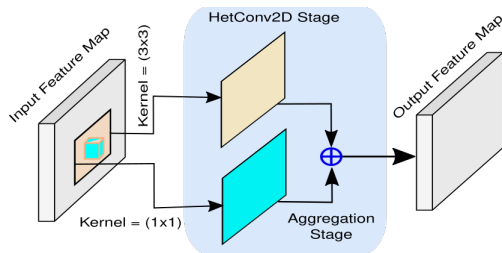

Figure 1: My First Figure

To refer write: Did I refer Fig. \ref{figLable}? to get Did I refer Figure 4?

PDF as Figures

```
\begin{figure}[h]
\centering
\includegraphics[clip=true, trim=10 495 350 10]{figfilename}
\caption{My First Figure}
\label{figLabel}
\end{figure}
```


Figure 2: My First Figure

trim = left bottom right top

PDF as Figures

```
\begin{figure}[h]
\centering
\includegraphics[clip=true, trim=10 495 350 10]{figfilename}
\caption{My First Figure}
\label{figLabel}
\end{figure}
```
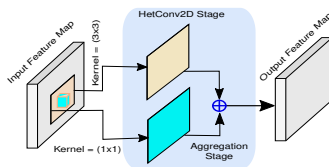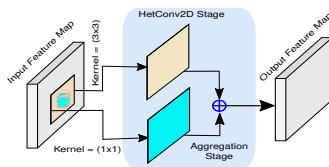

Figure 3: My First Figure

trim = left bottom right top

Scaling Figures

```
\begin{figure}[h]
\centering
\includegraphics[clip=true, trim=10 495 350 10,
                width=0.7\textwidth]{figfilename}
\caption{My First Figure}
\label{figLabel}
\end{figure}
```
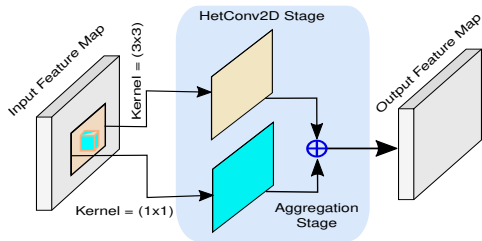

Figure 4: My First Figure

Figures in row(s)/col(s)

```
\begin{figure}[h]
\centering
\begin{tabular}{c c c }
\includegraphics[clip=true,trim=10 10 10 10]{figfilename} &
\includegraphics[clip=true,trim=10 10 10 10]{figfilename} &
\includegraphics[clip=true,trim=10 10 10 10]{figfilename}
\end{tabular}
\caption{My First Figure}
\label{figLabel}
\end{figure}
```
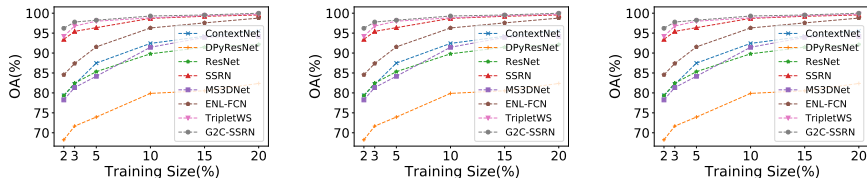

Figure 5: My Figures in a row

Pseudo Code Environment in LaTeX

The pseudo code environment is invoked as follows:

```
\begin{pseudocode}{<Name>}{<Parameters>}
        pseudo code constructs
\end{pseudocode}
```

The argument ⟨*Name*⟩ is the name of the algorithm, and ⟨*Parameters*⟩ is a list of parameters for the algorithm. For example, the commands

Pseudo Code Environment in LaTeX

The pseudo code environment is invoked as follows:

```
\begin{pseudocode}{<Name>}{<Parameters>}
        pseudo code constructs
\end{pseudocode}
```

The argument ⟨*Name*⟩ is the name of the algorithm, and ⟨*Parameters*⟩ is a list of parameters for the algorithm. For example, the commands

```
\begin{pseudocode}{CelsiusToFahrenheit}{c}
f \GETS {9c/5} + 32\\
\RETURN{f}
\end{pseudocode}
```

produce the following output when included in a document:

Pseudo Code Environment in LaTeX

The pseudo code environment is invoked as follows:

```
\begin{pseudocode}{<Name>}{<Parameters>}
        pseudo code constructs
\end{pseudocode}
```

The argument ⟨*Name*⟩ is the name of the algorithm, and ⟨*Parameters*⟩ is a list of parameters for the algorithm. For example, the commands

```
\begin{pseudocode}{CelsiusToFahrenheit}{c}
f \GETS {9c/5} + 32\\
\RETURN{f}
\end{pseudocode}
```

produce the following output when included in a document:

Algorithm 0.3: CELSIUSTOFAHRENHEIT(c)

$f \leftarrow 9c/5 + 32$
return (f)

The begin-end Construct in LaTeX

To form compound statements from simple statements, the begin-end
construct is used as follows:

```
\BEGIN
    some statement\\
    another statement\\
    yet another statement
\END
```

This generates the following output in in document:

The begin-end Construct in LaTeX

To form compound statements from simple statements, the begin-end construct is used as follows:

```
\BEGIN
    some statement\\
    another statement\\
    yet another statement
\END
```

This generates the following output in in document:

$\begin{cases} \text{some statement} \\ \text{another statement} \\ \text{yet another statement} \end{cases}$

The if-then-else Construct in LaTeX

The if-then-else construct takes various forms, such as the following:

```
\IF <condition> \THEN <stmt>
\IF <condition> \THEN <stmt> \ELSE <stmt>
\IF <condition> \THEN <stmt> \ELSEIF <stmt> \THEN <stmt>
```

Note that there is no limit placed on the number of that may be used in an if-then-else construct.

would produce the following output:

```
\IF some condition is true
\THEN
\BEGIN
    some statement\\
    another statement\\
    yet another statement
\END
\ELSEIF some other condition is true
\THEN
\BEGIN
    some statement\\
    another statement\\
    yet another statement
\END
\ELSE
do the default actions
```

For example, the commands:

if some condition is true

then $\begin{cases} somestatement \\ anotherstatement \\ yetanotherstatement \end{cases}$

else if some other condition is true

then $\begin{cases} somestatement \\ anotherstatement \\ yetanotherstatement \end{cases}$

else do the default actions

The for Loop

The for loop takes the following forms:

```
\FOR <var> \GETS <lower> \TO <upper> \DO <stmt>
\FOR <var> \GETS <upper> \DOWNTO <lower> \DO <stmt>
\FOREACH <condition> \DO <stmt>
```

For example,

```
\FOR i \GETS 0 \TO 10 \DO
    some processing
```

produces,

for i ← 0 **to** 10
  do some processing

and

```
\FOREACH x \in \mathcal{S} \DO
some processing
```

produces,

for each x ∈ 𝒮
  do some processing

The while Loop

The while loop takes the following form:

```
\WHILE <condition> \DO <stmt>
```

For example,

```
\WHILE some condition holds \DO
        some processing
```

produces,

while some condition holds
  do some processing

The repeat-until Loop

The repeat-until loop takes the following form:

```
\REPEAT <stmt> \UNTIL <condition>
```

For example,

```
\REPEAT
    some processing
\UNTIL some condition is met
```

produces,

repeat
 someprocessing
until some condition is met

Main Programs and Procedures

We can describe a main program that calls one (or more) procedures as follows:

```
\begin{pseudocode}{<Name>}{<Parameters>}
\PROCEDURE{<ProcedureName>}{<ProcedureParameters>}
some stuff
\ENDPROCEDURE
\MAIN
    some stuff\\
\CALL{<ProcedureName>}{<ActualParameters>}>\\
    more stuff
\ENDMAIN
\end{pseudocode}
```

Here is a simple example to illustrate the use of a main program calling a procedure. The commands

```
\begin{pseudocode}{TemperatureTable}{lower, upper}
\PROCEDURE{CelsiusToFahrenheit}{c}
    f \GETS {9c/5} + 32\\
\RETURN{f}
\ENDPROCEDURE
\MAIN
    x \GETS lower \\
\WHILE x \leq upper \DO
\BEGIN
\OUTPUT{x, \CALL{CelsiusToFahrenheit}{x}}\\
    x \GETS x+1
\END
\ENDMAIN
\end{pseudocode}
```

produces the following output:

Algorithm 0.4: TEMPERATURETABLE(*lower*, *upper*)

procedure CELSIUSTOFAHRENHEIT(*c*)
$f \leftarrow 9c/5 + 32$
return (f)

main
$x \leftarrow lower$
while $x \leq upper$
  do $\begin{cases} \textbf{output } (x, \text{CELSIUSTOFAHRENHEIT}(x)) \\ x \leftarrow x + 1 \end{cases}$

Other Predefined Keywords

Command	Output
local {list of variables}	**local** list of variables
global {list of variables}	**global** list of variables
external {list of procedures}	**external** list of procedures
return (list of values)	**return** (list of values)
output (list of values)	**output** (list of values)
exit	**exit**
and	**and**
or	**or**
not	**not**
true	**true**
false	**false**
GETS	$\leftarrow$

Computing the set of arcs for a given contour

Require: Contour (l_c), Structuring Elements (SE)
Ensure: Set of arcs, L_c

1: $m = $ Total number of data points in l_c
Initialize, $List = \{NULL\}$; $I_e = \{NULL\}$
2: Repeat the following for each element in SE
$I_e = [I_e \cup (V \ominus SE) \oplus SE]$
3: Repeat the following for $i = 1$ to m
$List = [List \cup I_e(i)]$
4: $l_d = diff(List)$
5: $l_1 = find(l_d == -1) + 1$
6: $l_2 = find(l_d == 1)$
return $(l_c = l_1 \cup l_2)$

Algorithm Pseudo Code in LaTeX

There are certain options for the algorithm2e package which control the algorithm style which need to be specified when you import the package:

```
\usepackage[ruled, lined, linesnumbered, commentsnumbered,
                          longend]{algorithm2e}
```

The meaning of these options are list below:

- ruled: display algorithm like three-line table.
- lined: display indentation line so that you can clearly see the indentation level.
- linenumbered: number the statement line.
- commentnumbered: number the comment line.
- longend: used end for to end for loop and use end if to end if condition.

Bibliographic References

Making references is one of the most important aspects of LaTeX

- References can be self-contained in the latex file; or
- (preferred) You can use a bibliographic database file.

The reference to an item (An already published paper or similar material) is made by

`\cite{citation-index in your .bib file}`.

Note: That you need to use the following two commands

- `\bibliographystyle{style form}` in the beginning of the body.
- `\bibliography{database file .bib}` at the end of the body.

Bibliographic References

A typical entry in the bibliography database is shown below:

```
@Book{abramo64,
 author    = "Milton {Abramowitz} and Irene A. {Stegun}",
 title     = "Handbook of Mathematical Functions with
               Formulas, Graphs, and Mathematical Tables",
 publisher = "Dover",
 address   = "New York",
 edition   = "ninth Dover printing, tenth GPO printing"
}
```

Note that:

- There are other types like article, inproceeding, manual, thesis etc.
- Abramo64 is the chosen unique keyword used as the citation index with \cite{ } command.

Bibliographic References

A typical bibliographic reference is listed at the end of the document as shown below.

```
References
1. Dawoud, A., Kamel, M.: Iterative model-based binariza-
tion algorithm for cheque images. IJDAR 5(1), 28{38 (2002)

2. Gatos, B., Ntirogiannis, K., Pratikakis, I.: Icdar 2009
document image binarization contest (dibco 2009). In: IC-DAR,
pp. 1375{1382 (2009)
```

Example Template

A specimen article producing LaTeX file could be as follows

```
\documentclass[12pt,twocolumn]{article}
\usepackage{graphicx}
\bibliographystle{ieee}
\begin{document}
\title{A (Very) Mechanical Introduction to \LaTeX}
\author{YOU}
\institute{CSE Dept., JGEC}

\date{}
\maketitle
```

Here the body will be

Example Template ... contd

```
\begin{abstract}
\end{abstract}
\section{section name e.g., Introduction}
\subsection{subsection name}
\subsubsection{subsubsection name}
.
.
.
\section{last section say, Conclusion}
\subsection{subsection name}
\subsubsection{subsubsection name}

\thebibliography{database .bib}
\end{document}
```

A small contribution to LaTeX community

The custom package for marking changes in manuscript

```
\definecolor{darkgreen}{rgb}{0,.4,0}
\definecolor{darkcyan}{rgb}{0,.4,.4}
\newcommand{\REMOVE}[1]%
         {{\color{blue}\sout{#1}}}
\newcommand{\ADD}[1]{{\color{red}{#1}}}
\newcommand{\REPLACE}[2]{\REMOVE{#1}{\color{red}{#2}}}
\newcommand{\COMMENT}[1]%
         {{\color{darkgreen}\textbf{{SKR: }} {#1}}}
```

Adding missing words in text using ADD

Today is Sunday. I'm wondering whether I should stay at home or go out. If I out, I will the lovely Sunday lunch at home. If I for lunch, I will the Sunday film showing at Theatre.

Today is Sunday. I'm wondering whether I should stay at home or go out. If I out, I will the lovely Sunday lunch at home. If I for lunch, I will the Sunday film showing at Theatre.

Adding missing words in text using ADD

```
Today is Sunday. I'm wondering whether I should stay at home
or go out. If I out, I will the lovely Sunday lunch at home.
If I for lunch, I will the Sunday film showing at Theatre.
```

Today is Sunday. I'm wondering whether I should stay at home or go out. If I out, I will the lovely Sunday lunch at home. If I for lunch, I will the Sunday film showing at Theatre.

```
Today is Sunday. I'm wondering whether I should stay at home
or go out. If I \ADD{go} out, I will \ADD{miss} the lovely
Sunday lunch at home. If I  \ADD{stay} for lunch, I will
\ADD{miss} the Sunday film showing at Theatre.
```

Today is Sunday. I'm wondering whether I should stay at home or go out. If I go out, I will miss the lovely Sunday lunch at home. If I stay for lunch, I will miss the Sunday film showing at Theatre.

Removing words in text using <u>REMOVE</u>

```
He wandered down the stairs and into the basement. The damp,
musty smell of unease hung in the air. A single, small
window let in a glimmer of light, but this simply made
the shadows in the basement deeper.
```

He wandered down the stairs and into the basement. The damp, musty smell of unease hung in the air. A single, small window let in a glimmer of light, but this simply made the shadows in the basement deeper.

Removing words in text using REMOVE

```
He wandered down the stairs and into the basement. The damp,
musty smell of unease hung in the air. A single, small
window let in a glimmer of light, but this simply made
the shadows in the basement deeper.
```

He wandered down the stairs and into the basement. The damp, musty smell of unease hung in the air. A single, small window let in a glimmer of light, but this simply made the shadows in the basement deeper.

```
He wandered down the stairs and into the basement. The
\REMOVE{damp,} musty smell of unease hung in the air.
A \REMOVE{single,} small window let in a glimmer of
light, but this simply made the shadows in the basement
deeper.
```

He wandered down the stairs and into the basement. The ~~damp,~~ musty smell of unease hung in the air. A ~~single,~~ small window let in a glimmer of light, but this simply made the shadows in the basement deeper.

Replacing words in text using <u>REPLACE</u>

```
There was nothing to indicate Nancy was going to change the
world. She looked like an average girl going to an average
high school.
```

There was nothing to indicate Nancy was going to change the world. She
looked like an average girl going to an average high school.

Replacing words in text using REPLACE

```
There was nothing to indicate Nancy was going to change the
world. She looked like an average girl going to an average
high school.
```

There was nothing to indicate Nancy was going to change the world. She looked like an average girl going to an average high school.

```
There was nothing to indicate \REPLACE{Nancy}{Susan} was
going to change the world. She looked like an average girl
going to an average \REPLACE{high school}{college}.
```

There was nothing to indicate ~~Nancy~~Susan was going to change the world. She looked like an average girl going to an average ~~high school~~college.

Commenting in text using COMMENTS

```
He had done everything right. There had been no mistakes
throughout the entire process. It had been perfection and
he knew it without a doubt.
```

He had done everything right. There had been no mistakes throughout the entire process. It had been perfection and he knew it without a doubt.

Commenting in text using COMMENTS

> He had done everything right. There had been no mistakes throughout the entire process. It had been perfection and he knew it without a doubt.

He had done everything right. There had been no mistakes throughout the entire process. It had been perfection and he knew it without a doubt.

> He had done everything right.\COMMENTS{There had been no mistakes}{Was he sure?} throughout the entire process. It had been perfection and he knew it without a doubt.

He had done everything right.**SKR:** There had been no mistakes Was he sure? throughout the entire process. It had been perfection and he knew it without a doubt.

THANK YOU