Math 291: Lecture 1

Prof. Fagerstrom

Minnesota State University Moorhead
web.mnstate.edu/fagerstrom
fagerstrom@mnstate.edu

January 11, 2018

Table of Contents

Outline

Software Components

- MikTeX

Software Components

- MikTeX
- TeXnicCenter

Software Components

- MikTeX
- TeXnicCenter
- Ghostscript

Installation Instructions

See Handout

- Prepared in essence by Dr. James
- I'll try to help (tech support is not my strength, but I can usually muddle on through)
- Can also use the campus labs

The History and Development of LaTeX

- TeX was developed by Donald Knuth in the 1970's. He used it to help him typeset the classic volume *The Art of Computer Programming*.

The History and Development of LaTeX

- TeX was developed by Donald Knuth in the 1970's. He used it to help him typeset the classic volume *The Art of Computer Programming*.
- LaTeX was originally a large set of macros developed by Leslie Lamport in 1985.

The History and Development of LaTeX

- TeX was developed by Donald Knuth in the 1970's. He used it to help him typeset the classic volume *The Art of Computer Programming*.
- LaTeX was originally a large set of macros developed by Leslie Lamport in 1985.
- LaTeX has gone through many upgrades since then. Currently the American Mathematical Society has developed AMS-LaTeX with more symbols and macros.

The History and Development of LaTeX

- TeX was developed by Donald Knuth in the 1970's. He used it to help him typeset the classic volume *The Art of Computer Programming*.
- LaTeX was originally a large set of macros developed by Leslie Lamport in 1985.
- LaTeX has gone through many upgrades since then. Currently the American Mathematical Society has developed AMS-LaTeX with more symbols and macros.
- Today, in mathematics, it is almost unacceptable to use anything else. Graduate math professors often expect homework handed in via LaTeX.

The History and Development of LaTeX

- TeX was developed by Donald Knuth in the 1970's. He used it to help him typeset the classic volume *The Art of Computer Programming*.
- LaTeX was originally a large set of macros developed by Leslie Lamport in 1985.
- LaTeX has gone through many upgrades since then. Currently the American Mathematical Society has developed AMS-LaTeX with more symbols and macros.
- Today, in mathematics, it is almost unacceptable to use anything else. Graduate math professors often expect homework handed in via LaTeX.
- Also used in physics (widely), computer science, and other physical sciences.

Outline

The "Preamble"

- Open a blank document using TeXnicCenter.

The "Preamble"

- Open a blank document using TeXnicCenter.
- Type `\documentclass{article}`

The "Preamble"

- Open a blank document using TeXnicCenter.
- Type `\documentclass{article}`
- Leave some space and type

  `\begin{document}`.

  Leave some additional space and type

  `\end{document}`

The "Preamble"

- Open a blank document using TeXnicCenter.
- Type `\documentclass{article}`
- Leave some space and type

  `\begin{document}`.

  Leave some additional space and type

  `\end{document}`
- The portion of the file between the statements

  `\documentclass{article}` and `\begin{document}`

  is called the Preamble.

The "Preamble"

- Open a blank document using TeXnicCenter.
- Type `\documentclass{article}`
- Leave some space and type

  `\begin{document}`.

  Leave some additional space and type

  `\end{document}`
- The portion of the file between the statements

  `\documentclass{article}` and `\begin{document}`

  is called the Preamble.
- This is where overall control of your document takes place. We'll use this area to add one package before the end of class today.

The "Body"

- The text between the statements

  `\begin{document}` and `\end{document}`

  is the body of your document. This will contain (for the most part) all of the text that you want to see once your document has been compiled.

The "Body"

- The text between the statements

  `\begin{document}` and `\end{document}`

  is the body of your document. This will contain (for the most part) all of the text that you want to see once your document has been compiled.
- In the body, type

```
This is my first \LaTeX document.
```

The "Body"

- The text between the statements

  `\begin{document}` and `\end{document}`

  is the body of your document. This will contain (for the most part) all of the text that you want to see once your document has been compiled.
- In the body, type

  ```
  This is my first \LaTeX document.
  ```
- Save your file in "Documents" as "Math291Week1".

The "Body"

- The text between the statements

  `\begin{document}` and `\end{document}`

  is the body of your document. This will contain (for the most part) all of the text that you want to see once your document has been compiled.
- In the body, type

  `This is my first \LaTeX document.`
- Save your file in "Documents" as "Math291Week1".
- To compile this document, we use build command (after selecting the options we want). Set the build mode to LaTeX $\Longrightarrow$ PS and then build.

The "Body"

- The text between the statements

  `\begin{document}` and `\end{document}`

  is the body of your document. This will contain (for the most part) all of the text that you want to see once your document has been compiled.
- In the body, type

  `This is my first \LaTeX document.`
- Save your file in "Documents" as "Math291Week1".
- To compile this document, we use build command (after selecting the options we want). Set the build mode to LaTeX $\Longrightarrow$ PS and then build.
- View the file by using the preview icon.
- Open the Documents folder. What do you see?

Compiling Errors

- Go back to your document file in TeXnicCenter and change the cap X on the end of

  `\LaTeX`

  to a lowercase x.

Compiling Errors

- Go back to your document file in TeXnicCenter and change the cap X on the end of

  `\LaTeX`

  to a lowercase x.
- Build your document again. At the bottom it should tell you that you have 1 error.

Compiling Errors

- Go back to your document file in TeXnicCenter and change the cap X on the end of

  `\LaTeX`

  to a lowercase x.
- Build your document again. At the bottom it should tell you that you have 1 error.
- Press F9. This will take you to a description of the error.

Compiling Errors

- Go back to your document file in TeXnicCenter and change the cap X on the end of

  `\LaTeX`

  to a lowercase x.
- Build your document again. At the bottom it should tell you that you have 1 error.
- Press F9. This will take you to a description of the error.
- It should say "undefined control sequence"

  `\LaTex`

  Notice the break.

Compiling Errors

- Go back to your document file in TeXnicCenter and change the cap X on the end of

  `\LaTeX`

  to a lowercase x.
- Build your document again. At the bottom it should tell you that you have 1 error.
- Press F9. This will take you to a description of the error.
- It should say "undefined control sequence"

  `\LaTex`

  Notice the break.
- Fix the mistake. (Remember that the LaTeX command is case sensitive).

Reserved Symbols in LaTeX

Some "Reserved Symbols" in LaTeX are:
\, ~, $, %, &, #, _, {, }, and ∧.

Reserved Symbols in LaTeX

Some "Reserved Symbols" in LaTeX are:
\, ~, $, %, &, #, _, {, }, and ∧.

The reason these symbols are reserved is that they each perform a special function within LaTeX.

Reserved Symbols in LaTeX

Some "Reserved Symbols" in LaTeX are:
\, ~, $, %, &, #, _, {, }, and ∧.

The reason these symbols are reserved is that they each perform a special function within LaTeX.
The \ indicates to LaTeX that there is a command rather than text.

Reserved Symbols in LaTeX

Some "Reserved Symbols" in LaTeX are:
\, ~, $, %, &, #, _, {, }, and ^.

The reason these symbols are reserved is that they each perform a special function within LaTeX.
The \ indicates to LaTeX that there is a command rather than text.
The ~ is a "hard space". Put one after the \LaTeX command to fix the spacing issue in your document.

Reserved Symbols in LaTeX

Some "Reserved Symbols" in LaTeX are:
\, ~, $, %, &, #, _, {, }, and ^.

The reason these symbols are reserved is that they each perform a special function within LaTeX.
The \ indicates to LaTeX that there is a command rather than text.
The ~ is a "hard space". Put one after the \LaTeX command to fix the spacing issue in your document.
We will discuss the specific function of most of the rest of these later.

Outline

Environments

- Documents created using LaTeX are best thought of as being divided into different environments.

Environments

- Documents created using LaTeX are best thought of as being divided into different environments.
- We are going to change the statement that we originally wrote to make it a title.

Environments

- Documents created using LaTeX are best thought of as being divided into different environments.
- We are going to change the statement that we originally wrote to make it a title.
- Before the statement write

  `\begin{center}`

  After the statement write

  `\end{center}`

Environments

- Documents created using LaTeX are best thought of as being divided into different environments.
- We are going to change the statement that we originally wrote to make it a title.
- Before the statement write

  `\begin{center}`

  After the statement write

  `\end{center}`
- Build your document and see what happens.

Environments

- Documents created using LaTeX are best thought of as being divided into different environments.
- We are going to change the statement that we originally wrote to make it a title.
- Before the statement write
  `\begin{center}`
  After the statement write
  `\end{center}`
- Build your document and see what happens.
- What if we want the text boldfaced? We could type the commands necessary by hand, but let's use TeXnicCenter to help us. Highlight the text and click the Boldfaced F on the top of the screen.

Environments

- Documents created using LaTeX are best thought of as being divided into different environments.
- We are going to change the statement that we originally wrote to make it a title.
- Before the statement write
  `\begin{center}`
  After the statement write
  `\end{center}`
- Build your document and see what happens.
- What if we want the text boldfaced? We could type the commands necessary by hand, but let's use TeXnicCenter to help us. Highlight the text and click the Boldfaced F on the top of the screen.
- Build, and see how the output has changed.

Making a Title Page

- Let's put our name before the title of this document.

Making a Title Page

- Let's put our name before the title of this document.
- Above the title type

```
<your name> \\ \today \\
```

Making a Title Page

- Let's put our name before the title of this document.
- Above the title type

  `<your name> \\ \today \\`
- Next, since we want this text to be in the upper right hand corner, we highlight and click on the flushright icon.

Making a Title Page

- Let's put our name before the title of this document.
- Above the title type

  `<your name> \\ \today \\`
- Next, since we want this text to be in the upper right hand corner, we highlight and click on the flushright icon.
- Build and see how the output has been modified.

The Math Environment

- The main power of LaTeX is its ability to typeset mathematical formulas quickly and easily.

The Math Environment

- The main power of LaTeX is its ability to typeset mathematical formulas quickly and easily.
- The math environment is invoked by using the symbol: $.

The Math Environment

- The main power of LaTeX is its ability to typeset mathematical formulas quickly and easily.
- The math environment is invoked by using the symbol: $.
- To see this, first type

```
$x^n$
```

- Build your document and see the output.

The Math Environment

- The main power of LaTeX is its ability to typeset mathematical formulas quickly and easily.
- The math environment is invoked by using the symbol: $.
- To see this, first type

  ```
  $x^n$
  ```

- Build your document and see the output.
- Next, add

  ```
  $x_n$
  ```

The Math Environment

- The main power of LaTeX is its ability to typeset mathematical formulas quickly and easily.
- The math environment is invoked by using the symbol: $.
- To see this, first type

  ```
  $x^n$
  ```

- Build your document and see the output.
- Next, add

  ```
  $x_n$
  ```

- Then build again.

A Polynomial

Let's type a polynomial. Use what you now know to add following polynomial to your document.

$$2x^{10} - x^8 + 5x^3$$

A Polynomial

Let's type a polynomial. Use what you now know to add following polynomial to your document.

$$2x^{10} - x^8 + 5x^3$$

How many of you got:

$$2x^10 - x^8 + 5x^3?$$

Using Curly Brackets

- When you want to apply a command to an object with more than one character within it, you need to use { and }.

Using Curly Brackets

- When you want to apply a command to an object with more than one character within it, you need to use { and }.
- To get the correct polynomial from the previous slide we type

```
2x^{10}-x^8+5x^3
```

Using Curly Brackets

- When you want to apply a command to an object with more than one character within it, you need to use { and }.
- To get the correct polynomial from the previous slide we type

```
2x^{10}-x^8+5x^3
```

- Try this and build your document.

Typesetting Fractions

- To create the fraction $\frac{1}{2}$, we type

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

Typesetting Fractions

- To create the fraction $\frac{1}{2}$, we type

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

- Try it and build.

Typesetting Fractions

- To create the fraction $\frac{1}{2}$, we type

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

- Try it and build.
- To create the fraction $\frac{\frac{1}{3}x^3}{\frac{1}{2}x^2}$, we type

  ```
  $\frac{\frac{1}{3}x^3}{\frac{1}{2}x^2}$
  ```

Typesetting Fractions

- To create the fraction $\frac{1}{2}$, we type

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

- Try it and build.
- To create the fraction $\frac{\frac{1}{3}x^3}{\frac{1}{2}x^2}$, we type

  ```
  $\frac{\frac{1}{3}x^3}{\frac{1}{2}x^2}$
  ```

- Try it and build.

Outline

A Cool Example

- Suppose you are explaining polynomial long division to a group of people and you don't want to go through the trouble of manually typing out the solution to the example that you are using to illustrate it.

A Cool Example

- Suppose you are explaining polynomial long division to a group of people and you don't want to go through the trouble of manually typing out the solution to the example that you are using to illustrate it.
- In your preamble (that is, between the documentclass and begin commands) type

```
\usepackage{polynom}
```

A Cool Example

- Suppose you are explaining polynomial long division to a group of people and you don't want to go through the trouble of manually typing out the solution to the example that you are using to illustrate it.
- In your preamble (that is, between the documentclass and begin commands) type

```
\usepackage{polynom}
```

- Now in your document somewhere type

```
$\polylongdiv{x^3+x^2-1}{x-1}$
```

A Cool Example

- Suppose you are explaining polynomial long division to a group of people and you don't want to go through the trouble of manually typing out the solution to the example that you are using to illustrate it.
- In your preamble (that is, between the documentclass and begin commands) type

```
\usepackage{polynom}
```

- Now in your document somewhere type

```
$\polylongdiv{x^3+x^2-1}{x-1}$
```

- Now build your document, and see what happens.

A Cool Example

- Suppose you are explaining polynomial long division to a group of people and you don't want to go through the trouble of manually typing out the solution to the example that you are using to illustrate it.
- In your preamble (that is, between the documentclass and begin commands) type

```
\usepackage{polynom}
```

- Now in your document somewhere type

```
$\polylongdiv{x^3+x^2-1}{x-1}$
```

- Now build your document, and see what happens.
- This is the power of LaTeX.