



SKILLPILLS

Skill Pill: L^AT_EX

Lecture 4: Beamer, Tikz, and more!

Christina Nguyen

Okinawa Institute of Science and Technology
christina-nguyen@oist.jp

14 Nov 2018



- 1 OIST Templates
- 2 Introduction to Beamer & TikZ
- 3 Beamer
 - Exercise
- 4 TikZ
- 5 BONUS

- 1 OIST Templates
- 2 Introduction to Beamer & TikZ
- 3 Beamer
 - Exercise
- 4 TikZ
- 5 BONUS

Templates can be found at

<https://github.com/oist/LaTeX-templates>.

Templates can be found at

<https://github.com/oist/LaTeX-templates>.

Available Templates:

- Rotation Proposal and Report
- Thesis Proposal
- Thesis
- OIST Presentation

In general, you'd want to download everything in the GitHub for the template and save it in one folder.

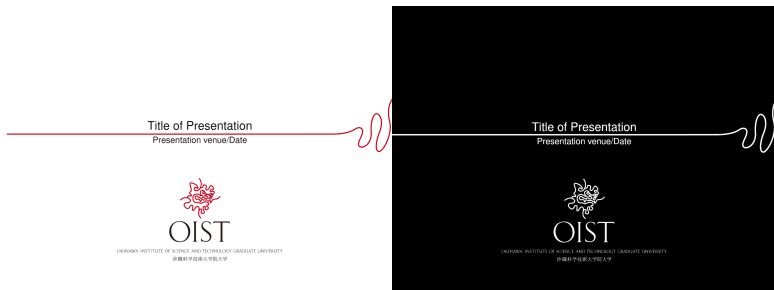
- Just plug and play with everything you learnt in the previous sections
- You can do minor formatting changes
- The one page limit isn't too strict (say if you need a little more than one page)

- A bit more complicated than the rotation proposal and report.
- The documents are pretty similar in terms of format.
- There are two files in the thesis template folder, temp thesis and final thesis.
 - Temp thesis is double spaced.
 - Also, you can include the PDFs of published articles in the temp thesis.

This is all for the Thesis templates

- In the main document you can change the preamble pages
- You can change which chapters are included
- mydefinitions
 - Where you can add the names of authors
 - Where you can change the bibliography style
 - Where you can put extra packages
 - Where you can put new commands

- With what you will learn today you can basically plug and play.
- There are light and dark versions available



- 1 OIST Templates
- 2 Introduction to Beamer & TikZ**
- 3 Beamer
 - Exercise
- 4 TikZ
- 5 BONUS

Beamer is a *documentclass* for \LaTeX that enables the creation of presentations from within \LaTeX . As an example this presentation and some of the previous presentations in this skill pill have been written with Beamer.

Beamer is a *documentclass* for \LaTeX that enables the creation of presentations from within \LaTeX . As an example this presentation and some of the previous presentations in this skill pill have been written with Beamer.

Why use beamer?

Beamer is a *documentclass* for \LaTeX that enables the creation of presentations from within \LaTeX . As an example this presentation and some of the previous presentations in this skill pill have been written with Beamer.

Why use beamer?

- Mathematics

Beamer is a *documentclass* for \LaTeX that enables the creation of presentations from within \LaTeX . As an example this presentation and some of the previous presentations in this skill pill have been written with Beamer.

Why use beamer?

- Mathematics
- Citations and references

Beamer is a *documentclass* for \LaTeX that enables the creation of presentations from within \LaTeX . As an example this presentation and some of the previous presentations in this skill pill have been written with Beamer.

Why use beamer?

- Mathematics
- Citations and references
- Consistency.

Beamer is a *documentclass* for \LaTeX that enables the creation of presentations from within \LaTeX . As an example this presentation and some of the previous presentations in this skill pill have been written with Beamer.

Why use beamer?

- Mathematics
- Citations and references
- Consistency.
- You're already using \LaTeX so why not?

PGF (portable graphics format)

- Enables the creation of graphics directly within \LaTeX .
- Three layers: System, Basic, and Frontend

PGF (portable graphics format)

- Enables the creation of graphics directly within \LaTeX .
- Three layers: System, Basic, and Frontend

TikZ (TikZ ist kein Zeichenprogramm)

- Frontend layer for PGF
- High level user interface

PGF (portable graphics format)

- Enables the creation of graphics directly within \LaTeX .
- Three layers: System, Basic, and Frontend

TikZ (TikZ ist kein Zeichenprogramm)

- Frontend layer for PGF
- High level user interface

Why use TikZ?

PGF (portable graphics format)

- Enables the creation of graphics directly within \LaTeX .
- Three layers: System, Basic, and Frontend

TikZ (TikZ ist kein Zeichenprogramm)

- Frontend layer for PGF
- High level user interface

Why use TikZ?

- Convenient.

PGF (portable graphics format)

- Enables the creation of graphics directly within \LaTeX .
- Three layers: System, Basic, and Frontend

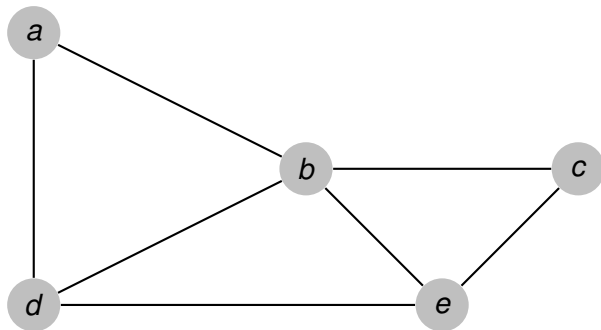
TikZ (TikZ ist kein Zeichenprogramm)

- Frontend layer for PGF
- High level user interface

Why use TikZ?

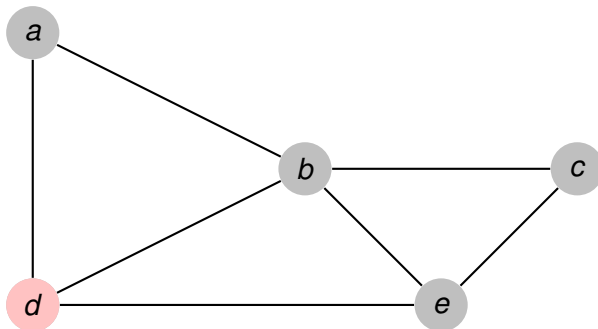
- Convenient.
- You're already using \LaTeX so why not?

Consider a graph $G = (V, E)$ with v vertices and e edges. The degree of a vertex v is the number of edges connected to it and will be denoted by $\deg(v)$.



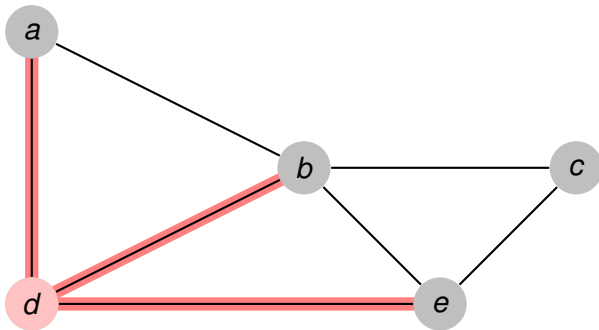
Consider a graph $G = (V, E)$ with v vertices and e edges. The degree of a vertex v is the number of edges connected to it and will be denoted by $\deg(v)$.

Consider the vertex d .



Consider a graph $G = (V, E)$ with v vertices and e edges. The degree of a vertex v is the number of edges connected to it and will be denoted by $\deg(v)$.

Consider the vertex d . Then $\deg(d) = 3$.



Beamer

```
\ documentclass{beamer}  
  
\ begin{document}  
  
  \ begin{frame}{Title}  
    Frame content  
  \ end{frame}  
  
\ end{document}
```

Similarly to Latex you can also automatically create table of contents. In order to do so add sections and subsections outside your frames.

```
\ begin { frame }  
\ titlepage  
\ end { frame }
```

```
\ begin { frame }{ Table of Contents }  
\ tableofcontents  
\ end { frame }
```

```
\ section { Introduction }  
\ begin { frame }{ Introduction }  
\ end { frame }
```

```
\ section { Beamer }  
\ subsection { Examples }
```

Effects can help draw attentions to talking points.

Effects can help draw attentions to talking points.

- Text visible on slide 2
- Text visible only on slide 2

Effects can help draw attentions to talking points.

- Text visible on slide 2
- Text visible on slide 3
- 1

Effects can help draw attentions to talking points.

- Text visible on slide 2
- Text visible on slide 3
- 1
- 2

```
\alert<1-2>{Effects} can help draw
    attentions to talking points.
\pause

\begin{itemize}
    \item<1-> Text visible on slide 2
    \item<2> Text visible only on slide 2
    \item<3-> Text visible on slide 3
\end{itemize}

% Short version
\begin{itemize}[<+ - >]
    \item 1
    \item 2
\end{itemize}
```


- Combining effects.

- This is quite fancy?
- How does this work?

- Combining effects.
- Really now?
- This is quite fancy?
- How does this work?

- Combining effects.
- Really now?

Yes, really.

- This is quite fancy?
- How does this work?

- Combining effects.
- Really now?
- This is quite fancy?
- How does this work?

- Combining effects.
- Really now?
- This is quite fancy?
- How does this work?

```
\begin{itemize}[<+-|alert@+>]
  \item Combining effects.
  \item Really now?
\end{itemize}

\pause

\only<3>{Yes, really.}

\setbeamercovered{transparent}

\pause

\begin{itemize}[<+->]
  \item This is quite fancy?
  \item How does this work?
\end{itemize}
```

This is a block

You've seen a lot of them so far.

```
\begin{block}{This is a block}  
You've seen a lot of them so far.  
\end{block}
```

In this slide, some important text will be **highlighted** because it's important. Please, don't abuse it.

In this slide, some important text will be **highlighted** because it's important. Please, don't abuse it.

Remark

Sample text

Important theorem

Sample text in alert block

In this slide, some important text will be **highlighted** because it's important. Please, don't abuse it.

Remark

Sample text

Important theorem

Sample text in alert block

Examples

Sample text in example block. "Examples" is fixed as block title.

In this slide, some important text will be
 `\alert{highlighted}` because it's important.
Please, don't abuse it.
`\pause`
`\begin{block}{Remark}`
Sample text
`\end{block}`
`\begin{alertblock}{Important theorem}`
Sample text in alert block
`\end{alertblock}`
`\pause`
`\begin{examples}`
Sample text in example block.
"Examples" is fixed as block title.
`\end{examples}`

There is a plethora of built-in themes available:

`https://www.hartwork.org/beamer-theme-matrix`

It is also possible to write your own.

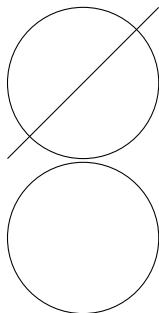
- <https://www.sharelatex.com/learn/Beamer>
- <http://mirrors.ctan.org/macros/latex/contrib/beamer/doc/beameruserguide.pdf>
- <https://en.wikibooks.org/wiki/LaTeX/Presentations>

Exercise time!

Exercise time!

- Create a simple beamer presentation with a few slides, that has a title page and a table of contents.
- Create a slide with a list of talking points, that appear after each other.

- 1 OIST Templates
- 2 Introduction to Beamer & TikZ
- 3 Beamer
 - Exercise
- 4 TikZ
- 5 BONUS

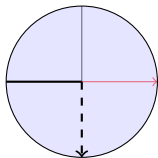


```
\begin{tikzpicture}
  \draw (0,0) circle (1);
  \draw (-1,-1) -- (1,1);
\end{tikzpicture}

\tikz \draw (0,0) circle (1);
```

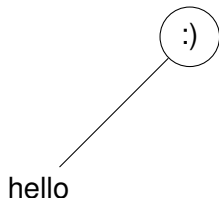
You can draw using paths which we can define using straight and curved line segments.

```
\begin{tikzpicture}
  \draw (0,0) -- (0,1);
  \draw [->, color=red] (0,0) -- (1,0);
  \draw [fill=blue!20!white, fill opacity=0.5]
    (0,0) circle (1.0);
  \draw [thick] (0,0) -- (-1,0);
  \draw [thick, dashed, ->] (0,0) -- (0,-1);
\end{tikzpicture}
```



A node is a simple shape with some text on it. A coordinate is a node without text. Nodes can be named and reference later on.

```
\begin{tikzpicture}
  \coordinate (origin) at (0,0);
  \node (A) at (origin) {hello};
  \node [shape=circle, draw] (C) at (2,2) {:)};
  \draw (A) -> (C);
\end{tikzpicture}
```



- **Absolute coordinates**

```
\tikz \draw [thick, red]  
    (0, 0) -- (2mm, 0) -- (2mm, 5pt);
```



- **Absolute coordinates**

```
\tikz \draw [thick, red]
      (0, 0) -- (2mm, 0) -- (2mm, 5pt);
```



- **Relative coordinates**

```
\tikz \draw [thick, red]
      (0, 0) -- +(2mm, 0) -- +(2mm, 5pt);
```



- **Absolute coordinates**

```
\tikz \draw [thick, red]
      (0, 0) -- (2mm, 0) -- (2mm, 5pt);
```



- **Relative coordinates**

```
\tikz \draw [thick, red]
      (0, 0) -- +(2mm, 0) -- +(2mm, 5pt);
```

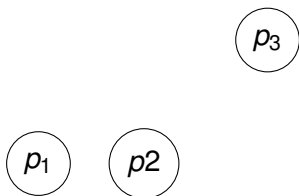


- **Shifting**

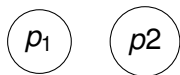
```
\tikz \draw [thick, red]
      (0, 0) -- ++(2mm, 0) -- ++(2mm, 5pt);
```



```
\begin{tikzpicture}
  \node[circle, draw] (p1) at (0, 0.5) {$p_1$};
  \node[circle, draw, right=0.5 of p1] (p2) {$p_2$};
  \node[circle, draw, above right=of p2]
    (p3) {$p_3$};
\end{tikzpicture}
```



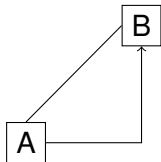
```
\begin{tikzpicture}
  \node[circle, draw] (p1) at (0, 0.5) {$p_1$};
  \node[circle, draw, right=0.5 of p1] (p2) {$p_2$};
  \node[circle, draw, above right=of p2]
    (p3) {$p_3$};
\end{tikzpicture}
```



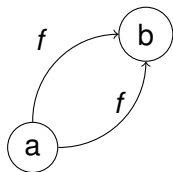
- Node (p2) should be right of (p1) no matter where you put (p1)
- Don't forget to include the TikZ library:
`\usetikzlibrary{positioning}`
- Available commands: right=of, below=of, above=of, ect

Anchors can be used to specify from which part of the node a path should start.

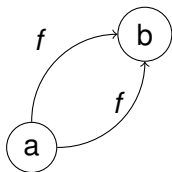
```
\begin{tikzpicture}
  \node [draw] (A) at (0,0) {A};
  \node [draw, above right=of A] (B) {B};
  \draw (A.north) -- (B.west);
  \draw [->] (A) -| (B);
\end{tikzpicture}
```



```
\begin{tikzpicture}  
  \node [circle, draw] (a) {a} ;  
  \node [circle, draw] (b) [above right=of a] {b} ;  
  \draw [->] (a) to [bend left=45] node [auto]  
    {$f$} (b) ;  
  \draw [->] (a) to [bend right=45] node [above]  
    {$f$} (b) ;  
\end{tikzpicture}
```



```
\begin{tikzpicture}
  \node [circle, draw] (a) {a} ;
  \node [circle, draw] (b) [above right=of a] {b} ;
  \draw [->] (a) to [bend left=45] node [auto]
    {$f$} (b) ;
  \draw [->] (a) to [bend right=45] node [above]
    {$f$} (b) ;
\end{tikzpicture}
```

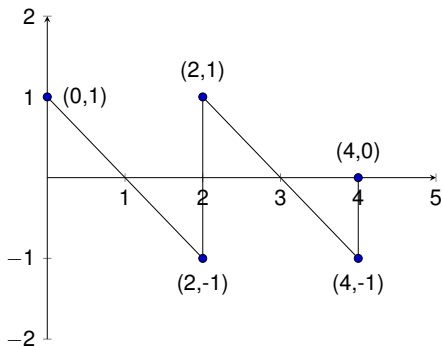
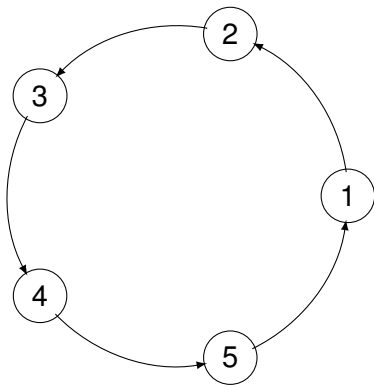


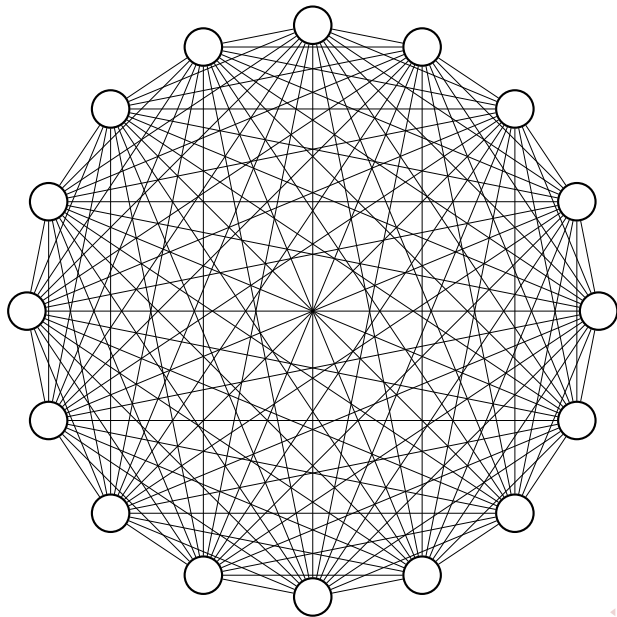
- The auto option places the label automatically
- You can also place them manually using: *left, right, above, below*

```
\begin{tikzpicture}
  \foreach \y in {1,2,3} {
    \draw [blue, ultra thick] (0,\y)
      [radius=0.3] ;
  }
\end{tikzpicture}
```

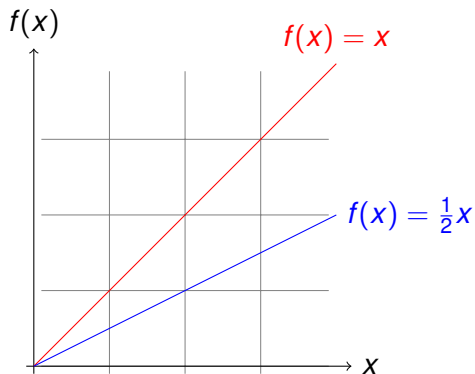
You can also create standalone graphics using TikZ.

```
\documentclass[tikz]{standalone}
\begin{document}
\begin{tikzpicture}
\end{tikzpicture}
\end{document}
```

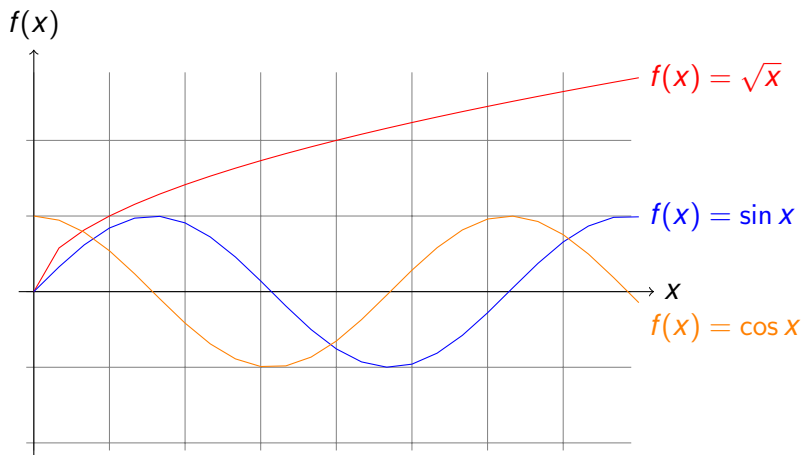


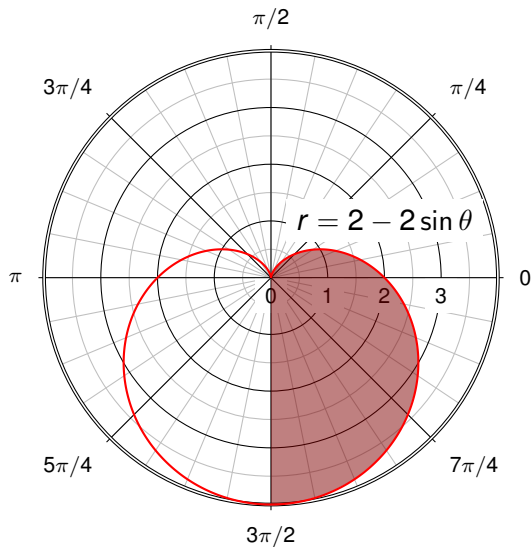


You can also plot functions in TikZ.




```
\begin{tikzpicture} [domain=0:4]
  \draw[very thin, color=gray]
    (0.1,-0.1) grid (3.9,3.9);
  \draw[->] (-0.1,0) -- (4.2,0)
    node[right] {$x$};
  \draw[->] (0,-0.1) -- (0,4.2)
    node[above] {$f(x)$};
  \draw[color=red] plot[id=x] function{x}
    node[above] {$f(x) = x$ };
  \draw[color=blue] plot[id=x] function{0.5*x}
    node[right] {$f(x) = \frac{1}{2} x$}
\end{tikzpicture}
```





- <https://en.wikibooks.org/wiki/LaTeX/PGF/TikZ>
- <http://www.texample.net/tikz/examples/>
- <http://www.gnuplot.info/>

Here's a classic theorem from discrete geometry.

Here's a classic theorem from discrete geometry.

Theorem (Radon)

Any set of $n + 2$ points in \mathbb{R}^n can always be partitioned in two subsets v_1 and v_2 such that the convex hulls of v_1 and v_2 always intersect. [1]

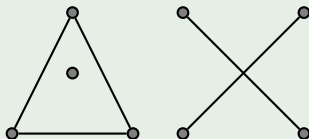
Here's a classic theorem from discrete geometry.

Theorem (Radon)

Any set of $n + 2$ points in \mathbb{R}^n can always be partitioned in two subsets v_1 and v_2 such that the convex hulls of v_1 and v_2 always intersect. [1]

Example

In the plane \mathbb{R}^2 we would need a set with 4 elements in order to guarantee the intersection of the convex hulls.











J. Matouek.

Lectures on Discrete Geometry (Graduate Texts in Mathematics).
Springer, 2002.

- 1 OIST Templates
- 2 Introduction to Beamer & TikZ
- 3 Beamer
 - Exercise
- 4 TikZ
- 5 **BONUS**

You can use tex markup and codeblocks in Facebook messenger on desktop. (They need to get it on the app. Yell at Mark Zuckerberg until it happens.)

To format text in Facebook chat messages on a computer:

Format	Looks like	Example	Instructions
Italic		<code>_Text_</code>	Type an underscore before and after the text.
Bold		<code>*Text*</code>	Type an asterisk before and after the text.
Strikethrough		<code>~Text~</code>	Type a tilde before and after the text.
Monospace		<code>`Text`</code>	Type a backtick before and after the text.
Codeblock		<pre>... Text ...</pre>	<ol style="list-style-type: none">1 Type three backticks2 Type one line break (press shift + return)3 Type your text4 Type another line break (press shift + return)5 Type three more backticks
LaTeX typesetting		<code>\(Text\)</code>	Type a backslash and then a parenthesis before and after the text.