

# Beamer Theme

Zhibo Wang

November 26th, 2018

# Outline

1. Introduction
2. Beamer Basic
  - Hightlight
  - Other Environments
3. Beamer More
  - Split Screen
  - Table
  - Math
4. Conclusion

# Outline

## 1. Introduction

## 2. Beamer Basic

- Highlight
- Other Environments

## 3. Beamer More

- Split Screen
- Table
- Math

## 4. Conclusion

# Latex and Beamer

LaTeX is a high-quality typesetting system; it includes features designed for the production of technical and scientific documentation.

# Latex and Beamer

LaTeX is a high-quality typesetting system; it includes features designed for the production of technical and scientific documentation.

Beamer is a LaTeX class to create powerful, flexible and nice-looking presentations and slides.

The beamer class is focussed on producing (on-screen) presentations, along with support material such as handouts and speaker notes.

# Outline

1. Introduction
2. Beamer Basic
  - Highlight
  - Other Environments
3. Beamer More
  - Split Screen
  - Table
  - Math
4. Conclusion

# Block and Alert

## Pythagorean theorem

$$a^2 + b^2 = c^2$$

where  $c$  represents the length of the hypotenuse and  $a$  and  $b$  the lengths of the triangle's other two sides.

## Remark

- the environment above is **block**
- the environment here is **alertblock**

# Proof

## Pythagorean theorem

$$a^2 + b^2 = c^2$$

## Proof.

$$3^2 + 4^2 = 5^2$$

$$5^2 + 12^2 = 13^2$$





# Algorithm

**Data:** this text

**Result:** how to write algorithm with  $\text{\LaTeX}2\text{e}$  initialization;

**while** *not at end of this document* **do**

    read current;

**if** *understand* **then**

        go to next section;

        current section becomes this one;

**else**

        go back to the beginning of current section;

**end**

**end**

**Algorithm 1:** How to write algorithms (copied from [here](#))

# An Algorithm For Finding Primes Numbers.

```
int main (void)
{
    std::vector<bool> is_prime (100, true);
    for (int i = 2; i < 100; i++)
        if (is_prime[i])
        {
            std::cout << i << " ";
            for (int j = i; j < 100; is_prime [j] = false, j+=i);
        }
    return 0;
}
```

Note the use of \alert.

# More

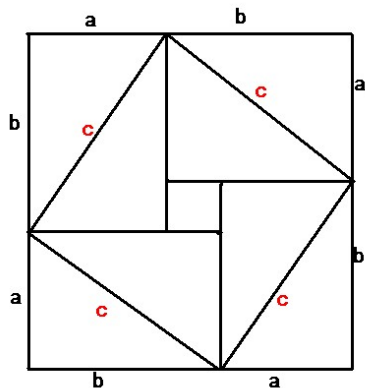
More environments such as

- Definition
- lemma
- corollary
- example

# Outline

1. Introduction
2. Beamer Basic
  - Highlight
  - Other Environments
3. Beamer More
  - Split Screen
  - Table
  - Math
4. Conclusion

# Minipage



- 1 item
- 2 another
- 3 more
  - first
  - second
  - third

# Columns

This is a text in first column.

$$E = mc^2$$

- First item
- Second item

first block

columns achieves splitting the screen

second block

stack block in columns

# Create Tables

first	second	third
1	2	3
4	5	6
7	8	9

# Equation1

A matrix in text must be set smaller:  $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$  to not increase leading in a portion of text.

$$f(n) = \begin{cases} n/2 & \text{if } n \text{ is even} \\ -(n+1)/2 & \text{if } n \text{ is odd} \end{cases}$$

$$50apples \times 100apples = lotsofapples^2$$



## Equation2

$$\sum_{\substack{0 \leq i < m \\ 0 \leq j < n}} P(i, j) = \int_a^b \prod P(i, j)$$

$$P\left(A = 2 \left| \frac{A^2}{B} > 4 \right.\right)$$

$$(a), [b], \{c\}, |d|, \|e\|, \langle f \rangle, \lfloor g \rfloor, \lceil h \rceil, \lceil i \rceil$$

## Equation3

$$Q(\alpha) = \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

$$Q(\alpha) = \alpha^i \alpha^j y^{(i)} y^{(j)} (x^i \cdot x^j)$$

$$\Gamma = \beta + \alpha + \gamma + \rho$$

# Outline

1. Introduction
2. Beamer Basic
  - Hightlight
  - Other Environments
3. Beamer More
  - Split Screen
  - Table
  - Math
4. Conclusion

# End

The last page.