There Is No Largest Prime Number

Euclid of Alexandria
euclid@alexandria.edu

Hamburg University of Technology

27th International Symposium of Prime Numbers

TUHH
Technische Universität Hamburg-Harburg

Outline

Outline

The Pause Commands

The \pause command

The \pause command offers an easy, but not very flexible way of creating frames that are uncovered piecewise.

- Shown from first slide on.
- Shown from second slide on.
  - Shown from third slide on.
  - Shown from fourth slide on.
- Shown from fifth slide on.

The Pause Commands

The \pause command

The \pause command offers an easy, but not very flexible way of creating frames that are uncovered piecewise.

- Shown from first slide on.
- Shown from second slide on.
  - Shown from third slide on.
  - Shown from fourth slide on.
- Shown from fifth slide on.

The Pause Commands

The \pause command

The \pause command offers an easy, but not very flexible way of creating frames that are uncovered piecewise.

- Shown from first slide on.
- Shown from second slide on.
  - Shown from third slide on.
  - Shown from fourth slide on.
- Shown from fifth slide on.

The Pause Commands

The \pause command

The \pause command offers an easy, but not very flexible way of creating frames that are uncovered piecewise.

- Shown from first slide on.
- Shown from second slide on.
  - Shown from third slide on.
  - Shown from fourth slide on.
- Shown from fifth slide on.

The Pause Commands

The \pause command

The \pause command offers an easy, but not very flexible way of creating frames that are uncovered piecewise.

- Shown from first slide on.
- Shown from second slide on.
  - Shown from third slide on.
  - Shown from fourth slide on.
- Shown from fifth slide on.

Outline

The General Concept of Overlay Specifications

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification:

```
\tt,\it,\bf,\sl,\rm,\st,\color,\alert,\structure
```

This line is bold on all three slides.
This line is bold only on the second slide.
This line is bold only on the third slide.

The General Concept of Overlay Specifications

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification:

```
\tt,\it,\bf,\sl,\rm,\st,\color,\alert,\structure
```

This line is bold on all three slides.
This line is bold only on the second slide.
This line is bold only on the third slide.

The General Concept of Overlay Specifications

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification:

```
\tt,\it,\bf,\sl,\rm,\st,\color,\alert,\structure
```

This line is bold on all three slides.
This line is bold only on the second slide.
This line is bold only on the third slide.

The General Concept of Overlay Specifications

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification:

`\tt,\it,\bf,\sl,\rm,\st,\color,\alert,\structure`

This line is bold on all three slides.
This line is bold only on the second slide.
This line is bold only on the third slide.

or

This line is inserted only on slide **4**.

The General Concept of Overlay Specifications

For the following commands, adding an overlay specification causes the command to be simply ignored on slides that are not included in the specification:

```
\tt,\it,\bf,\sl,\rm,\st,\color,\alert,\structure
```

This line is bold on all three slides.
This line is bold only on the second slide.
This line is bold only on the third slide.

or

This line is inserted only on slide **5**.

Outline

Commands with Overlay Specifications

For the following commands, the effect of an overlay specification is special:

1. `\onslide`
2. `\uncover`
3. `\visible`
4. `\only`

The overlay specifications are given in pointed brackets.
The specification $<1->$ means "from slide 1 on."
Shown from slide **1** on
Ranges are specified like this: $<2-3>$
Shown on the slides **2 and 3**

- Still shown on the slides **2 and 3**
- Still shown on the slides **2 and 3**

Shown from slide **4** on.

Shown on **all** slides.

Commands with Overlay Specifications

For the following commands, the effect of an overlay specification is special:

1. `\onslide`
2. `\uncover`
3. `\visible`
4. `\only`

The overlay specifications are given in pointed brackets.
The specification $<1->$ means "from slide 1 on."
Shown from slide **1** on
Ranges are specified like this: $<2-3>$
Shown on the slides **2 and 3**

- Still shown on the slides **2 and 3**
- Still shown on the slides **2 and 3**

Shown from slide **4** on.

Shown on **all** slides.

Commands with Overlay Specifications

For the following commands, the effect of an overlay specification is special:

1. `\onslide`
2. `\uncover`
3. `\visible`
4. `\only`

The overlay specifications are given in pointed brackets.
The specification $<1->$ means "from slide 1 on."
Shown from slide **1** on
Ranges are specified like this: $<2-3>$
Shown on the slides **2 and 3**

- Still shown on the slides **2 and 3**
- Still shown on the slides **2 and 3**

Shown from slide **4** on.

Shown on **all** slides.

Commands with Overlay Specifications

For the following commands, the effect of an overlay specification is special:

1. `\onslide`
2. `\uncover`
3. `\visible`
4. `\only`

The overlay specifications are given in pointed brackets.
The specification $<1->$ means "from slide 1 on."
Shown from slide **1** on
Ranges are specified like this: $<2-3>$
Shown on the slides **2 and 3**

- Still shown on the slides **2 and 3**
- Still shown on the slides **2 and 3**

Shown from slide **4** on.

Shown on **all** slides.

Commands with Overlay Specifications

For the following commands, the effect of an overlay specification is special:

1. `\onslide`
2. `\uncover`
3. `\visible`
4. `\only`

The overlay specifications are given in pointed brackets.
The specification $<1->$ means "from slide 1 on."
Shown from slide **1** on
Ranges are specified like this: $<2-3>$
Shown on the slides **2 and 3**

- Still shown on the slides **2 and 3**
- Still shown on the slides **2 and 3**

Shown from slide **4** on.

Shown on **all** slides.
e.g. -3,5-6,8- means "on all slides, except for slides 4 and 7.

Commands with Overlay Specifications
\onslide

\onslide<*overlay specification*>{<*text*>}

The behaviour of this command depends on whether the optional argument <*text*> is given or not.
If present, the <*modifier*> can be either a + or a *.
If no <*text*> is given, the following happens:

All text following this command will only be shown (uncovered) on the specified slides. On non-specified slides, the text still occupies space. If no slides are specified, the following text is always shown.

If the <*modifier*> is +, hidden text will not be treated as covered, but as invisible. The difference is the same as the difference between \uncover and \visible.

The <*modifier*> * may not be given if no <*text*> argument is present.

Commands with Overlay Specifications

\uncover and \visible

\uncover
\uncover<*overlay specification*>{<*text*>}

If the <*overlay specification*> is present, the <*text*> is shown ("uncovered") only on the specified slides. On other slides, the text still occupies space and it is still typeset, but it is not shown or only shown as if transparent.

Example: `\uncover<3->{Text shown from slide 3 on.}`

\visible
\visible<*overlay specification*>{<*text*>}

This command does almost the same as \uncover. The only difference is that if the text is not shown,it is never shown in a transparent way, but rather it is not shown at all. Thus, for this command the transparency settings have no effect.

Example: `\visible<2->{Text shown from slide 2 on.}`

Commands with Overlay Specifications

If a <*text*> argument is present,

- `\onslide` (without a <*modifier*>) is mapped to `\uncover`,
- `\onslide+` is mapped to `\visible`,
- `\onslide*` is mapped to `\only`

Example

```
\begin{frame}
  \onslide<1>{Same effect as the following command.}
  \uncover<1>{Same effect as the previous command.}

  \onslide+<2>{Same effect as the following command.}
  \visible<2>{Same effect as the previous command.}

  \onslide*<3>{Same effect as the following command.}
  \only<3>{Same effect as the previous command.}
\end{frame}
```

Commands with Overlay Specifications

If a <*text*> argument is present,

- `\onslide` (without a <*modifier*>) is mapped to `\uncover`,
- `\onslide+` is mapped to `\visible`,
- `\onslide*` is mapped to `\only`

Example

```
\begin{frame}
  \onslide<1>{Same effect as the following command.}
  \uncover<1>{Same effect as the previous command.}

  \onslide+<2>{Same effect as the following command.}
  \visible<2>{Same effect as the previous command.}

  \onslide*<3>{Same effect as the following command.}
  \only<3>{Same effect as the previous command.}
\end{frame}
```

Commands with Overlay Specifications

If a <*text*> argument is present,

- `\onslide` (without a <*modifier*> is mapped to `\uncover`,
- `\onslide+` is mapped to `\visible`,
- `\onslide*` is mapped to `\only`

Example

```
\begin{frame}
  \onslide<1>{Same effect as the following command.}
  \uncover<1>{Same effect as the previous command.}

  \onslide+<2>{Same effect as the following command.}
  \visible<2>{Same effect as the previous command.}

  \onslide*<3>{Same effect as the following command.}
  \only<3>{Same effect as the previous command.}
\end{frame}
```

Outline

What Are Prime Numbers?

The BEAMER class predefines several environments.
The complete list is the following:

1. theorem, proof, lemma and corollary
2. definition and definitions
3. fact, example and examples

Definition

A prime number is a number that has exactly two divisors.

Example

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).
- 4 is not prime (three divisors: 1, 2, and 4).

What Are Prime Numbers?

The BEAMER class predefines several environments.
The complete list is the following:

1. theorem, proof, lemma and corollary
2. definition and definitions
3. fact, example and examples

Definition

A prime number is a number that has exactly two divisors.

Example

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).
- 4 is not prime (three divisors: 1, 2, and 4).

What Are Prime Numbers?

The BEAMER class predefines several environments.
The complete list is the following:

1. theorem, proof, lemma and corollary
2. definition and definitions
3. fact, example and examples

Definition

A prime number is a number that has exactly two divisors.

Example

- 2 is prime (two divisors: 1 and 2).
- 3 is prime (two divisors: 1 and 3).
- 4 is not prime (three divisors: 1, 2, and 4).

Theorem Environments

Uncovering an Enumeration Piecewise

Theorem (Euclid)

There is no largest prime number.

Proof.

1. Suppose p were the largest prime number.
2. Let q be the product of the first p numbers.
3. Then $q + 1$ is not divisible by any of them.
4. Thus $q + 1$ is also prime and greater than p. □

The proof used *reductio ad absurdum*.

Theorem Environments
Uncovering an Enumeration Piecewise

Theorem (Euclid)

There is no largest prime number.

Proof.

1. Suppose p were the largest prime number.
2. Let q be the product of the first p numbers.
3. Then $q + 1$ is not divisible by any of them.
4. Thus $q + 1$ is also prime and greater than p. □

The proof used *reductio ad absurdum*.

Theorem Environments

Uncovering an Enumeration Piecewise

Theorem (Euclid)

There is no largest prime number.

Proof.

1. Suppose p were the largest prime number.
2. Let q be the product of the first p numbers.
3. Then $q + 1$ is not divisible by any of them.
4. Thus $q + 1$ is also prime and greater than p. □

The proof used *reductio ad absurdum*.

Theorem Environments

Uncovering an Enumeration Piecewise

Theorem (Euclid)

There is no largest prime number.

Proof.

1. Suppose p were the largest prime number.
2. Let q be the product of the first p numbers.
3. Then $q + 1$ is not divisible by any of them.
4. Thus $q + 1$ is also prime and greater than p. □

The proof used *reductio ad absurdum*.

Outline

Block And Column Environments

What's Still To Do?

Answered Questions

How many primes are there?

Open Questions

Is every even number the sum of two primes? [1]

Block And Column Environments

What's Still To Do?

Answered Questions

How many primes are there?

Open Questions

Is every even number the sum of two primes? [1]

Outline

Framed and Boxed Text

Place me somewhere!

Place me somewhere!

Place me somewhere!

Theorem

$A = B$.
$C = D$.

Outline

An Algorithm For Finding Primes Numbers.

Add the \fragile option to the frame to list an algorithm.

```
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 `std::`.

An Algorithm For Finding Primes Numbers.

Add the `\fragile` option to the frame to list an algorithm.

```
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 `std::`.

An Algorithm For Finding Primes Numbers.

Hilighting the Current Item in an Enumeration

Use the \semiverbatim environment to uncover part of the algorithm stepwise and to add an emphasis on certain lines or parts of lines.

```
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;
}
```

An Algorithm For Finding Primes Numbers.

Hilighting the Current Item in an Enumeration

Use the \semiverbatim environment to uncover part of the algorithm stepwise and to add an emphasis on certain lines or parts of lines.

```
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;
}
```

An Algorithm For Finding Primes Numbers.

Hilighting the Current Item in an Enumeration

Use the \semiverbatim environment to uncover part of the algorithm stepwise and to add an emphasis on certain lines or parts of lines.

```
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;
}
```

An Algorithm For Finding Primes Numbers.

Hilighting the Current Item in an Enumeration

Use the \semiverbatim environment to uncover part of the algorithm stepwise and to add an emphasis on certain lines or parts of lines.

```
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 std::.

Outline

Ballot Environment

Changing Symbol Before an Enumeration

- ✔ First point.
- Second point.
- Third point.

Ballot Environment

Changing Symbol Before an Enumeration

- ✔ First point.
- ✔ Second point.
- ● Third point.

Ballot Environment

Changing Symbol Before an Enumeration

- ✔ First point.
- ✔ Second point.
- ✔ Third point.

Outline

Uncovering a Table Rowwise

Uncovering a Table Rowwise

Class	A	B	C	D
X	1	2	3	4
Y	3	4	5	6
Z	5	6	7	8

Uncovering a Table Rowwise

Uncovering a Table Rowwise

Class	A	B	C	D
X	1	2	3	4
Y	3	4	5	6
Z	5	6	7	8

Uncovering a Table Rowwise

Uncovering a Table Rowwise

Class	A	B	C	D
X	1	2	3	4
Y	3	4	5	6
Z	5	6	7	8

Uncovering Tagged Formulas Piecewise

Uncovering Tagged Formulas Piecewise

$$A = B \tag{1}$$
$$C = D \tag{2}$$
$$E = F \tag{3}$$

Uncovering Tagged Formulas Piecewise

Uncovering Tagged Formulas Piecewise

$$A = B \tag{1}$$
$$C = D \tag{2}$$
$$E = F \tag{3}$$

Uncovering Tagged Formulas Piecewise

Uncovering Tagged Formulas Piecewise

$$A = B \tag{1}$$
$$C = D \tag{2}$$
$$E = F \tag{3}$$

Outline

Mathematical Formulas

Hilighting the Current Item in an Enumeration

Let

$$v^1 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad v^2 := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad v^3 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

Decide whether the following statements are true or false :

- ☐ The intersection of the plain given by $x^T v^1 = 2$ and $x^T v^2 = 1$ is not empty.
- ☐ $v^1 \times v^3 = \frac{1}{2} v^2$.
- ☐ The mapping $T : \begin{cases} \mathbb{R}^3 & \longrightarrow \mathbb{R}^3 \\ x & \longmapsto x \times v^1 + v^2 \times x \end{cases}$ is injektive.
- ☐ $v^1 \left(v^1\right)^T + v^2 \left(v^2\right)^T = v^3 \left(v^3\right)^T$.

Mathematical Formulas

Hilighting the Current Item in an Enumeration

Let

$$v^1 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad v^2 := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad v^3 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

Decide whether the following statements are true or false :

W The intersection of the plain given by $x^T v^1 = 2$ and $x^T v^2 = 1$ is not empty.

☐ $v^1 \times v^3 = \frac{1}{2} v^2$.

☐ The mapping $T : \begin{cases} \mathbb{R}^3 & \longrightarrow \mathbb{R}^3 \\ x & \longmapsto x \times v^1 + v^2 \times x \end{cases}$ is injektive.

☐ $v^1 \left(v^1\right)^T + v^2 \left(v^2\right)^T = v^3 \left(v^3\right)^T$.

Mathematical Formulas

Hilighting the Current Item in an Enumeration

Let

$$v^1 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad v^2 := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad v^3 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

Decide whether the following statements are true or false :

W The intersection of the plain given by $x^T v^1 = 2$ and $x^T v^2 = 1$ is not empty.

F $v^1 \times v^3 = \frac{1}{2} v^2$.

☐ The mapping $T : \begin{cases} \mathbb{R}^3 & \longrightarrow \mathbb{R}^3 \\ x & \longmapsto x \times v^1 + v^2 \times x \end{cases}$ is injektive.

☐ $v^1 \left(v^1\right)^T + v^2 \left(v^2\right)^T = v^3 \left(v^3\right)^T$.

Mathematical Formulas

Hilighting the Current Item in an Enumeration

Let

$$v^1 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad v^2 := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad v^3 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

Decide whether the following statements are true or false :

W The intersection of the plain given by $x^T v^1 = 2$ and $x^T v^2 = 1$ is not empty.

F $v^1 \times v^3 = \frac{1}{2} v^2$.

F The mapping $T : \begin{cases} \mathbb{R}^3 & \longrightarrow \mathbb{R}^3 \\ x & \longmapsto x \times v^1 + v^2 \times x \end{cases}$ is injektive.

☐ $v^1 \left(v^1\right)^T + v^2 \left(v^2\right)^T = v^3 \left(v^3\right)^T$.

Mathematical Formulas

Hilighting the Current Item in an Enumeration

Let

$$v^1 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad v^2 := \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad v^3 := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

Decide whether the following statements are true or false :

W The intersection of the plain given by $x^T v^1 = 2$ and $x^T v^2 = 1$ is not empty.

F $v^1 \times v^3 = \frac{1}{2} v^2$.

F The mapping $T : \begin{cases} \mathbb{R}^3 & \longrightarrow \mathbb{R}^3 \\ x & \longmapsto x \times v^1 + v^2 \times x \end{cases}$ is injektive.

F $v^1 \left(v^1\right)^T + v^2 \left(v^2\right)^T = v^3 \left(v^3\right)^T$.

Outline

Adding Hyperlinks and Buttons.

- You specify a target using the command `\hypertarget` or (easier) the command `\label`.
- You render the button using `\beamerbutton` or a similar command. This will render the button, but clicking it will not yet have any effect..
- You put the button inside a `\hyperlink` command. Now clicking it will jump to the target of the link..

Jump to third slide

Adding Hyperlinks and Buttons.

- You specify a target using the command `\hypertarget` or (easier) the command `\label`.
- You render the button using `\beamerbutton` or a similar command. This will render the button, but clicking it will not yet have any effect..
- You put the button inside a `\hyperlink` command. Now clicking it will jump to the target of the link..

Jump to third slide

Adding Hyperlinks and Buttons.

- You specify a target using the command `\hypertarget` or (easier) the command `\label`.
- You render the button using `\beamerbutton` or a similar command. This will render the button, but clicking it will not yet have any effect..
- You put the button inside a `\hyperlink` command. Now clicking it will jump to the target of the link..

Jump to third slide

Adding Hyperlinks and Buttons.

The `\label` command creates a hypertarget as a side-effect and the

$$\texttt{label=}\langle\ \textit{name}\ \rangle$$

option of the `\frame` command creates a label named

$$\langle\ \textit{name}\ \rangle < \langle \textit{slide number} \rangle >$$

for each slide of the frame as a side-effect.

Thus the above example could be written more easily as:

- First item.
- Second item.
- Third item.

Jump to third slide

Adding Hyperlinks and Buttons.

The `\label` command creates a hypertarget as a side-effect and the

$$\texttt{label=}\langle\ \textit{name}\ \rangle$$

option of the `\frame` command creates a label named

$$\langle\ \textit{name}\ \rangle < \langle \textit{slide number} \rangle >$$

for each slide of the frame as a side-effect.

Thus the above example could be written more easily as:

- First item.
- Second item.
- Third item.

Jump to third slide

Adding Hyperlinks and Buttons.

The `\label` command creates a hypertarget as a side-effect and the

$$\texttt{label=}\langle\ \textit{name}\ \rangle$$

option of the `\frame` command creates a label named

$$\langle\ \textit{name}\ \rangle < \langle \textit{slide number} \rangle >$$

for each slide of the frame as a side-effect.

Thus the above example could be written more easily as:

- First item.
- Second item.
- Third item.

Jump to third slide

\beamerskipbutton.

Theorem

⋮

first theorem

Skip proof

\beamerskipbutton.

Theorem

⋮

first theorem

Proof.

⋮

proof to the first theorem □

\beamerreturnbutton.

Theorem

:

second theorem

\beamerreturnbutton.

Theorem

⋮

second theorem

Proof.

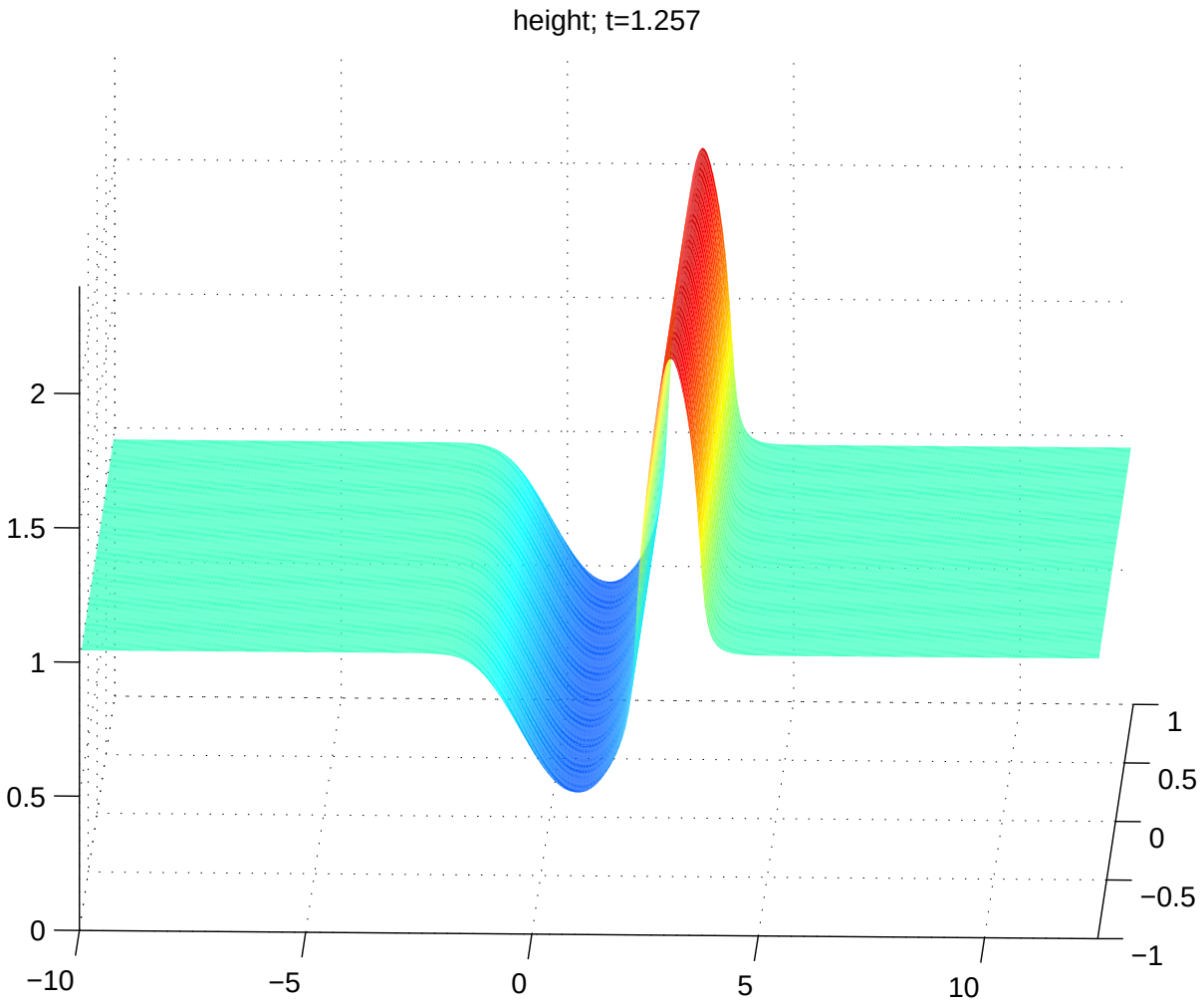
⋮

proof to the second theorem □

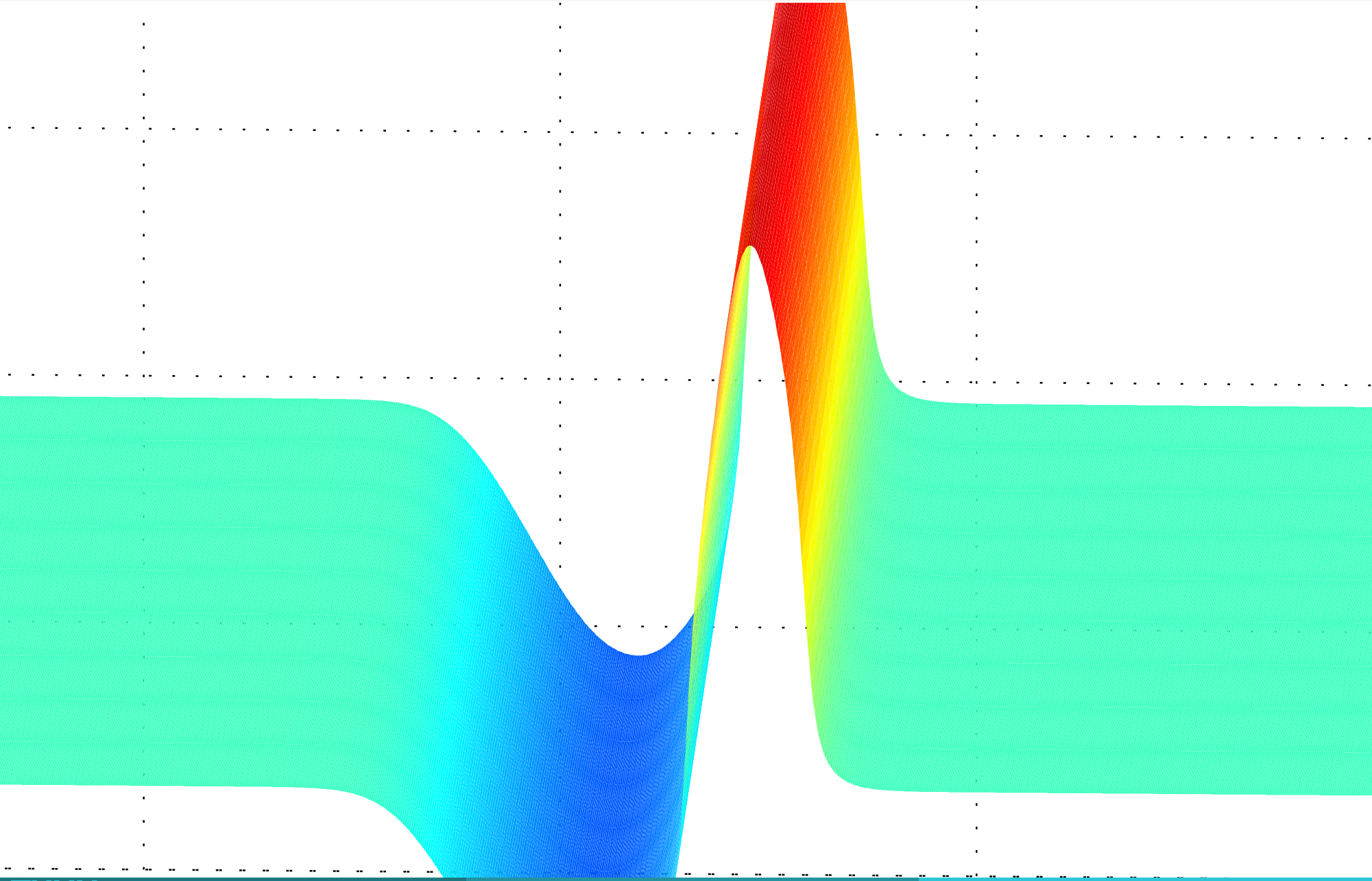
hide proof

Outline

Adding Anticipated Zooming



Adding Anticipated Zooming



Outline

Dynamically Changing Text

Some text for the first slide.
Possibly several lines long.

Dynamically Changing Text

Replacement on the second slide. Supressed for handout.

Outline

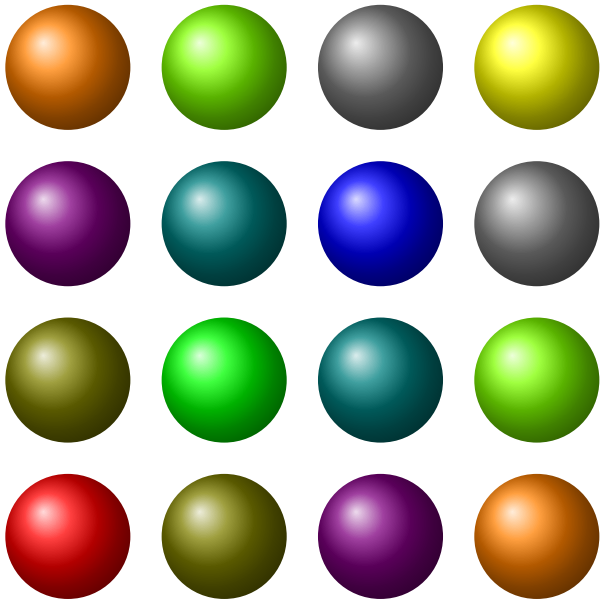
Including External Graphic Files

You can use all the standard LaTeX commands for inserting graphics, like `\includegraphics` or `\pgfimage`.

- The four formats .pdf, .jpg, .jpeg, and .png can be used only with `\pdflatex`.
- If you have a .eps graphic and wish to use `\pdflatex`, you can use the program eps2pdf to convert the graphic to a .pdf file.
- Alternatively you can directly 'draw' in LaTeX if you use the packages PGF and the frontend layer to PGF – TikZ with `\usepackage{tikz}` and necessary libraries, for example `\usetikzlibrary{snakes}` or `\usetikzlibrary{arrows}`

TikZ examples

First example

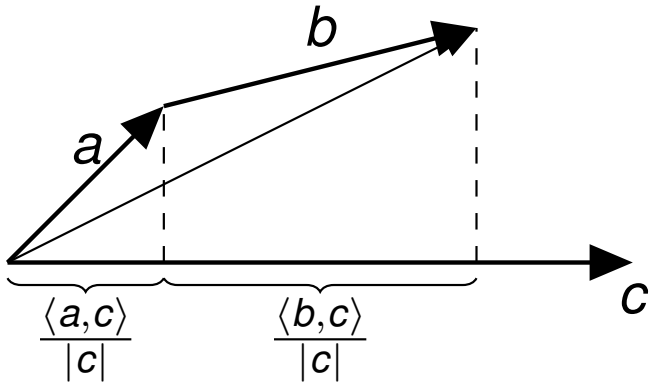


Source code for the first picture

```
\begin{tikzpicture}[shading=ball]
  \put(110,0) {
  \foreach \x / \cola in {0/red,1/green,2/blue,3/yellow}
  \foreach \y / \colb in {0/red,1/green,2/blue,3/yellow}
  \shade[ball color=\cola!50!\colb] (\x,\y) circle (0.4cm);
  }
\end{tikzpicture}
```

TikZ examples

Second example

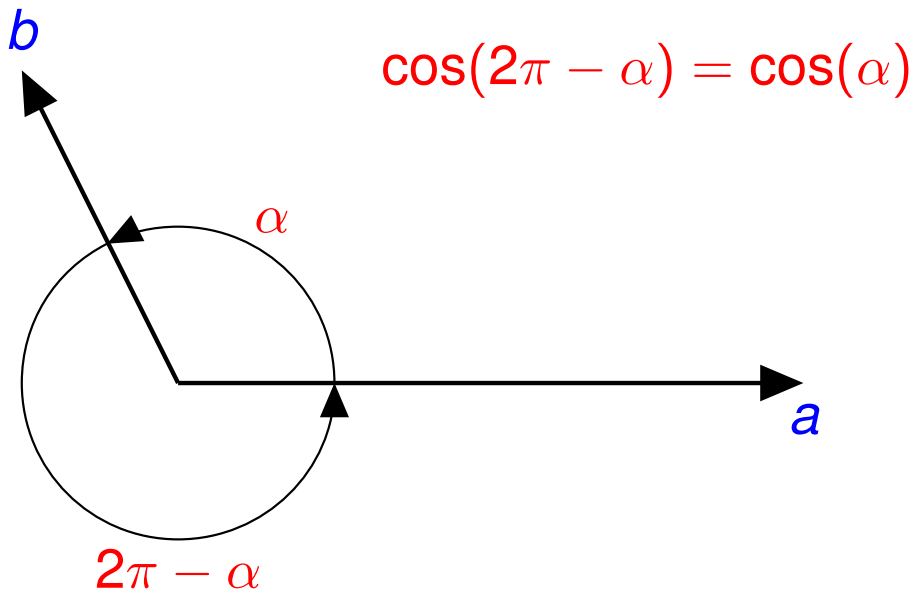


Source code for the second picture

```
\begin{tikzpicture}
  \put(110,0){
  \draw[thick,-triangle 45](0,0)--(4,0) node[below]{$c$};
  \draw[thick,-triangle 455](0,0)--(1,1) node[above,pos=0.5]{$a$};
  \draw[-triangle 45](0,0)--(3,1.5);
  \draw[thick,-triangle 45](1,1)--(3,1.5) node[above,pos=0.5]{$b$};
  \draw[dashed](1,0)--(1,1); \draw[dashed](3,0)--(3,1.5);
  \draw[snake=brace](1,-0.1)--(0,-0.1)
    node[below,pos=0.5]{$\frac{\langle a,c\rangle }{|c|}$};
  \draw[snake=brace](3,-0.1)--(1,-0.1)
    node[below,pos=0.5]{$\frac{\langle b,c\rangle }{|c|}$};
\end{tikzpicture}
```

TikZ examples

Third example

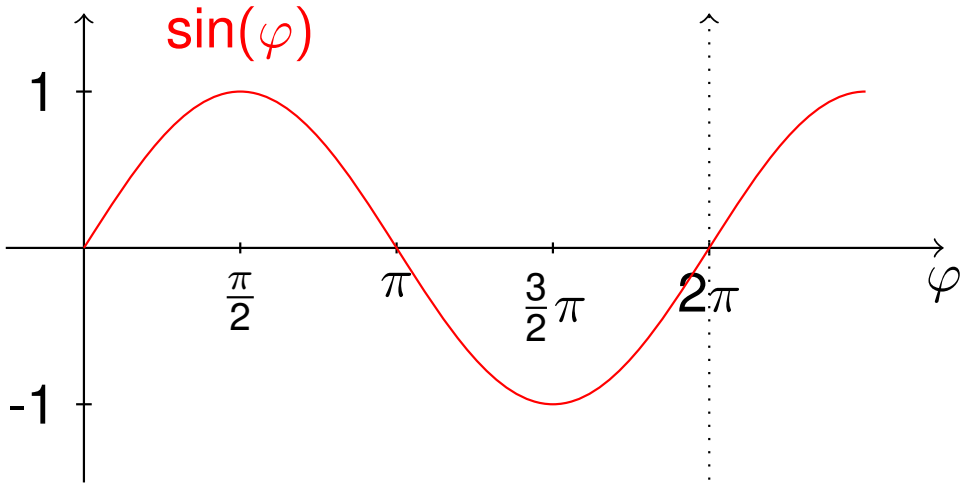


Source code for the third picture

```
\begin{tikzpicture}
  \put(90,0){
  \draw[thick,-triangle 45](0,0)--(-1,2) node[above]{$b$};
  \draw[thick,-triangle 45](0,0)--(4,0) node[below]{$a$};
  \draw[-triangle 45](1,0)arc(0:117:1cm);
  \draw(60:1.2cm) node{$\alpha$};
  \draw[triangle 45-](1,0)arc(0:-243:1cm);
  \draw(270:1.2cm) node{$2\pi - \alpha$};
  \draw[xshift=3cm,yshift=2cm] node{$\cos(2\pi -\alpha)= \cos(\alpha )$};}
\end{tikzpicture}
```

TikZ examples

Fourth example

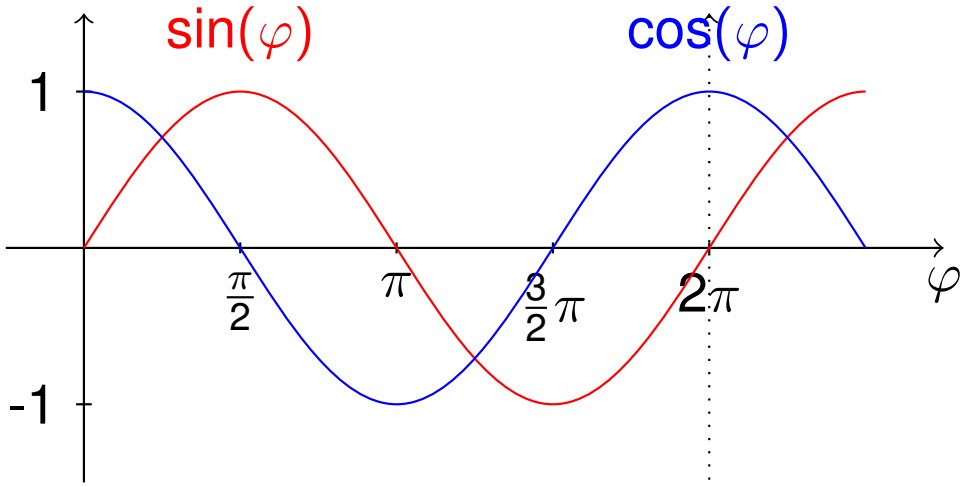


Source code for the fourth picture

```
\begin{tikzpicture}\put(70,0){
  \draw [->](-0.5,0) -- (5.5,0)node[below]{$\varphi$};
  \draw [->](0,-1.5) -- (0,1.5); \draw[dotted,<-](4,1.5)--(4,-1.5);
  \foreach \x/\xtext in {1/\frac{\pi}{2},2/\pi,3/\frac{3}{2}\pi,4/2\pi}
  \draw[xshift=\x cm] (0pt,1pt) -- (0pt,-1pt) node[below] {$\xtext$};
  \draw(0.05,1)--(-0.05,1) node[left]{1};
  \draw(0.05,-1)--(-0.05,-1) node[left]{-1};
  \draw[color=red](0,0)sin(1,1)node[above=2pt]{$\sin(\varphi)$}
    cos(2,0)sin(3,-1)cos(4,0)sin(5,1);
  \only<2-> {
  \draw[color=blue](0,1)cos(1,0) sin(2,-1)
    cos(3,0)sin(4,1) node[above=2pt]{$\cos(\varphi)$} cos(5,0);}}
\end{tikzpicture}
```

TikZ examples

Fourth example

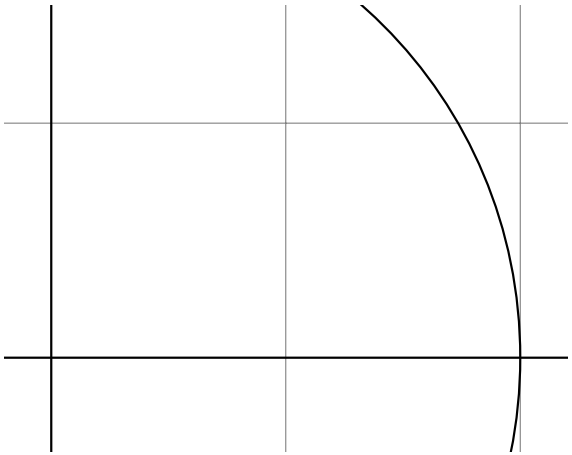


Source code for the fourth picture

```
\begin{tikzpicture}\put(70,0){
  \draw [->](-0.5,0) -- (5.5,0)node[below]{$\varphi$};
  \draw [->](0,-1.5) -- (0,1.5); \draw[dotted,<-](4,1.5)--(4,-1.5);
  \foreach \x/\xtext in {1/\frac{\pi}{2},2/\pi,3/\frac{3}{2}\pi,4/2\pi}
  \draw[xshift=\x cm] (0pt,1pt) -- (0pt,-1pt) node[below] {$\xtext$};
  \draw(0.05,1)--(-0.05,1) node[left]{1};
  \draw(0.05,-1)--(-0.05,-1) node[left]{-1};
  \draw[color=red](0,0)sin(1,1)node[above=2pt]{$\sin(\varphi)$}
    cos(2,0)sin(3,-1)cos(4,0)sin(5,1);
  \only<2-> {
  \draw[color=blue](0,1)cos(1,0) sin(2,-1)
    cos(3,0)sin(4,1) node[above=2pt]{$\cos(\varphi)$} cos(5,0);}}
\end{tikzpicture}
```

TikZ examples

Fifth example

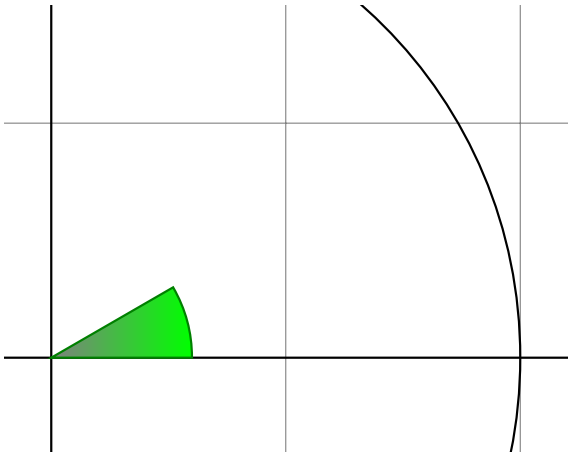


Source code for the fifth picture

```
\begin{tikzpicture}[scale=3]
  \put(100,0) {
  \clip (-0.1,-0.2) rectangle (1.1,0.75);
  \draw[step=.5cm,gray,very thin] (-1.4,-1.4) grid (1.4,1.4);
  \draw (-1.5,0) -- (1.5,0);
  \draw (0,-1.5) -- (0,1.5);
  \draw (0,0) circle (1cm);
  \shadedraw[left color=gray,right color=green, draw=green!50!black]
  (0,0) -- (3mm,0mm) arc (0:30:3mm) -- cycle;
  }
\end{tikzpicture}
```

TikZ examples

Fifth example

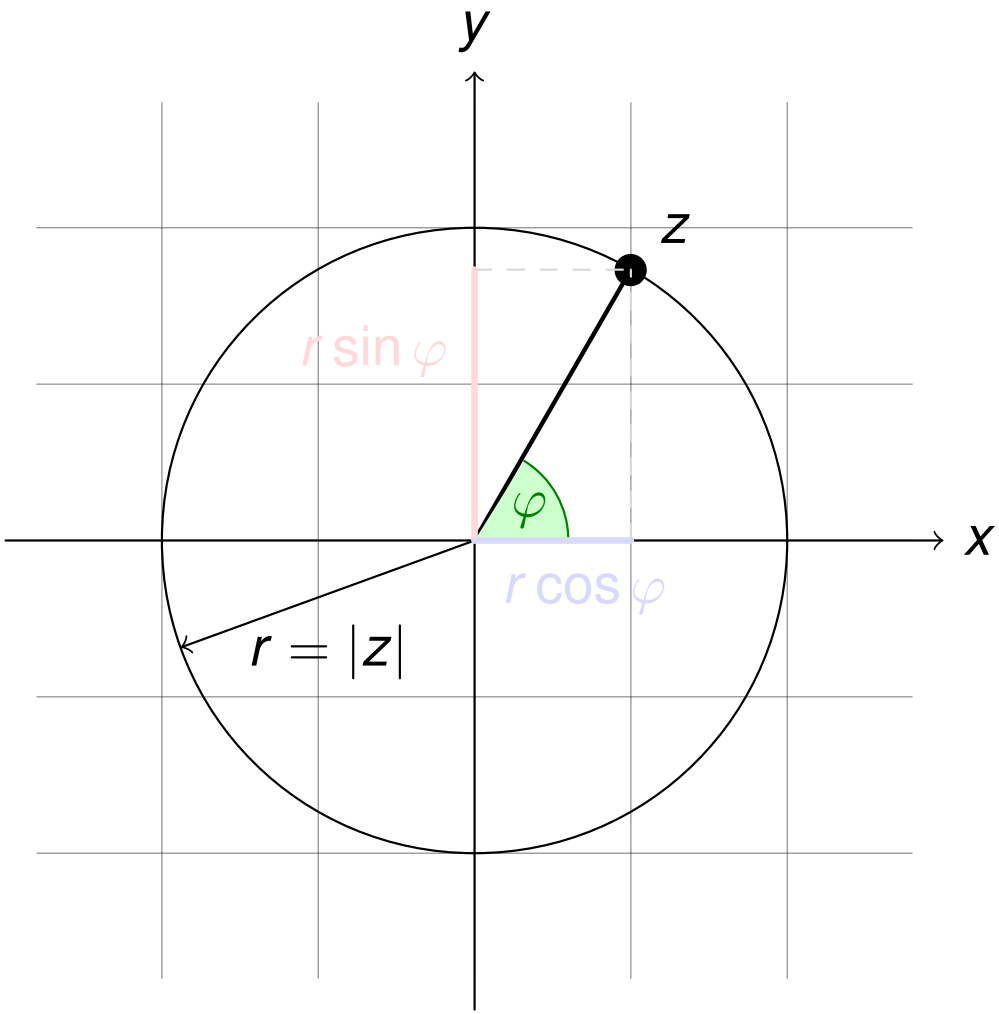


Source code for the fifth picture

```
\begin{tikzpicture}[scale=3]
  \put(100,0) {
  \clip (-0.1,-0.2) rectangle (1.1,0.75);
  \draw[step=.5cm,gray,very thin] (-1.4,-1.4) grid (1.4,1.4);
  \draw (-1.5,0) -- (1.5,0);
  \draw (0,-1.5) -- (0,1.5);
  \draw (0,0) circle (1cm);
  \shadedraw[left color=gray,right color=green, draw=green!50!black]
  (0,0) -- (3mm,0mm) arc (0:30:3mm) -- cycle;
  }
\end{tikzpicture}
```

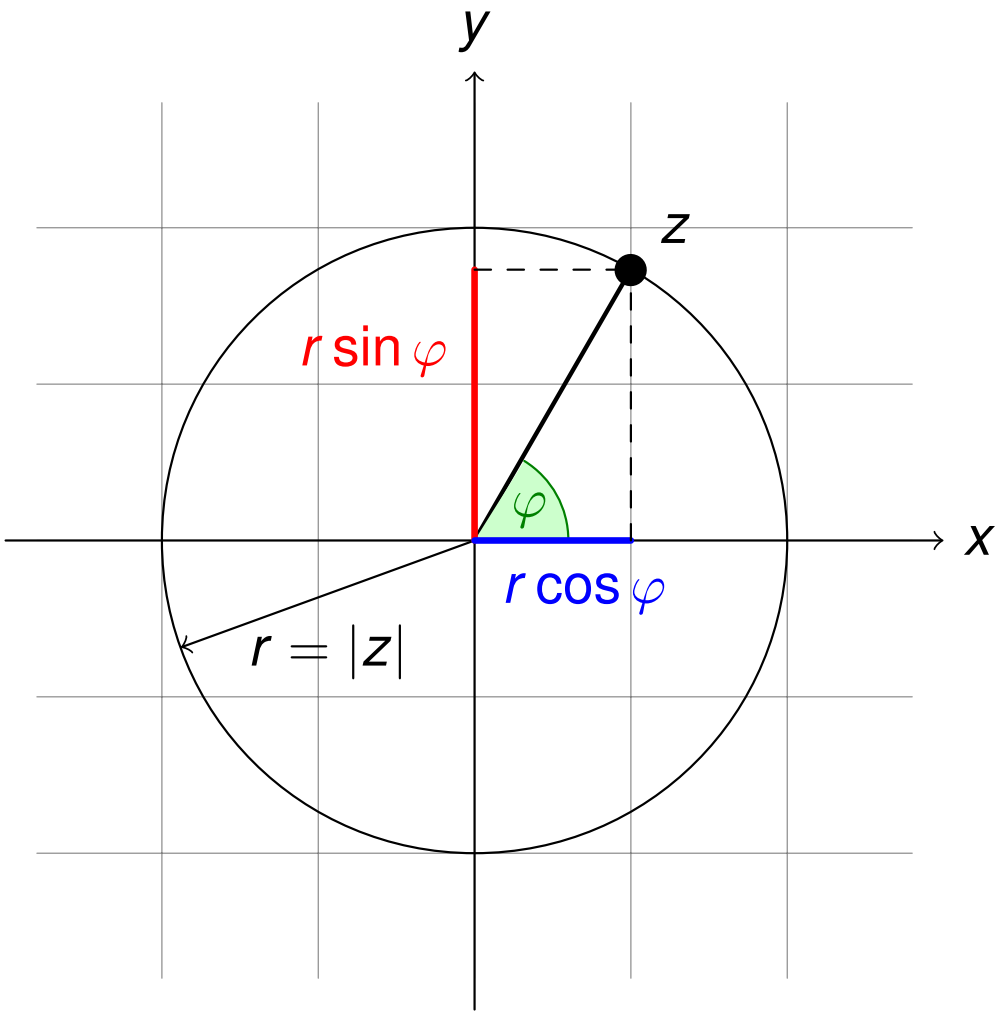
TikZ examples

Sixth example



TikZ examples

Sixth example



TikZ examples

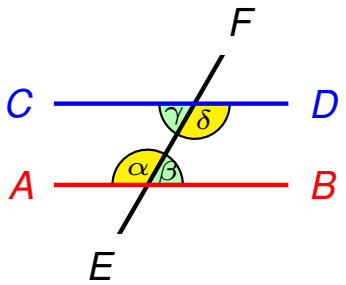
Sixth example

Source code for the sixth picture

```
\begin{tikzpicture}[scale=2,cap=round]
 \put(80,0){
  \def\costhirty{0.8660256}
  \colorlet{anglecolor}{green!50!black} \colorlet{sincolor}{red}
  \colorlet{tancolor}{orange!80!black} \colorlet{coscolor}{blue}
  \tikzstyle{axes}=[] \tikzstyle{important line}=[very thick]
  \draw[style=help lines,step=0.5cm] (-1.4,-1.4) grid (1.4,1.4);
  \draw (0,0) circle (1cm);
  \begin{scope}[style=axes]
    \draw[->] (-1.5,0) -- (1.5,0) node[right] {$x$} coordinate(x axis);
    \draw[->] (0,-1.5) -- (0,1.5) node[above] {$y$} coordinate(y axis);
    \draw[->](0,0)--(200:1cm) node[midway,below=2pt]{$r=|z|$};
  \end{scope}
  \draw[-*,thick](0,0) -- (60:1.05cm) node[above=5pt,right]{$z$};
  \filldraw[fill=green!20,draw=anglecolor] (0,0) -- (3mm,0pt) arc(0:58:3mm);
  \draw (29:2mm) node[anglecolor] {$\varphi$};
  \pause
  \draw[style=important line,sincolor]
    (y axis |- 60:1cm) -- node[pos=0.3,left=1pt] {$r\sin\varphi$} (0,0);
  \draw[dashed](y axis |- 60:1cm)--(60:1cm);
  \draw[style=important line,coscolor]
    (60:1cm |- x axis) -- node[below=2pt,pos=0.3] {$r\cos \varphi$} (0,0);
  \draw[dashed](60:1cm |- x axis)--(60:1cm);
  }
\end{tikzpicture}
```

TikZ examples

Seventh example



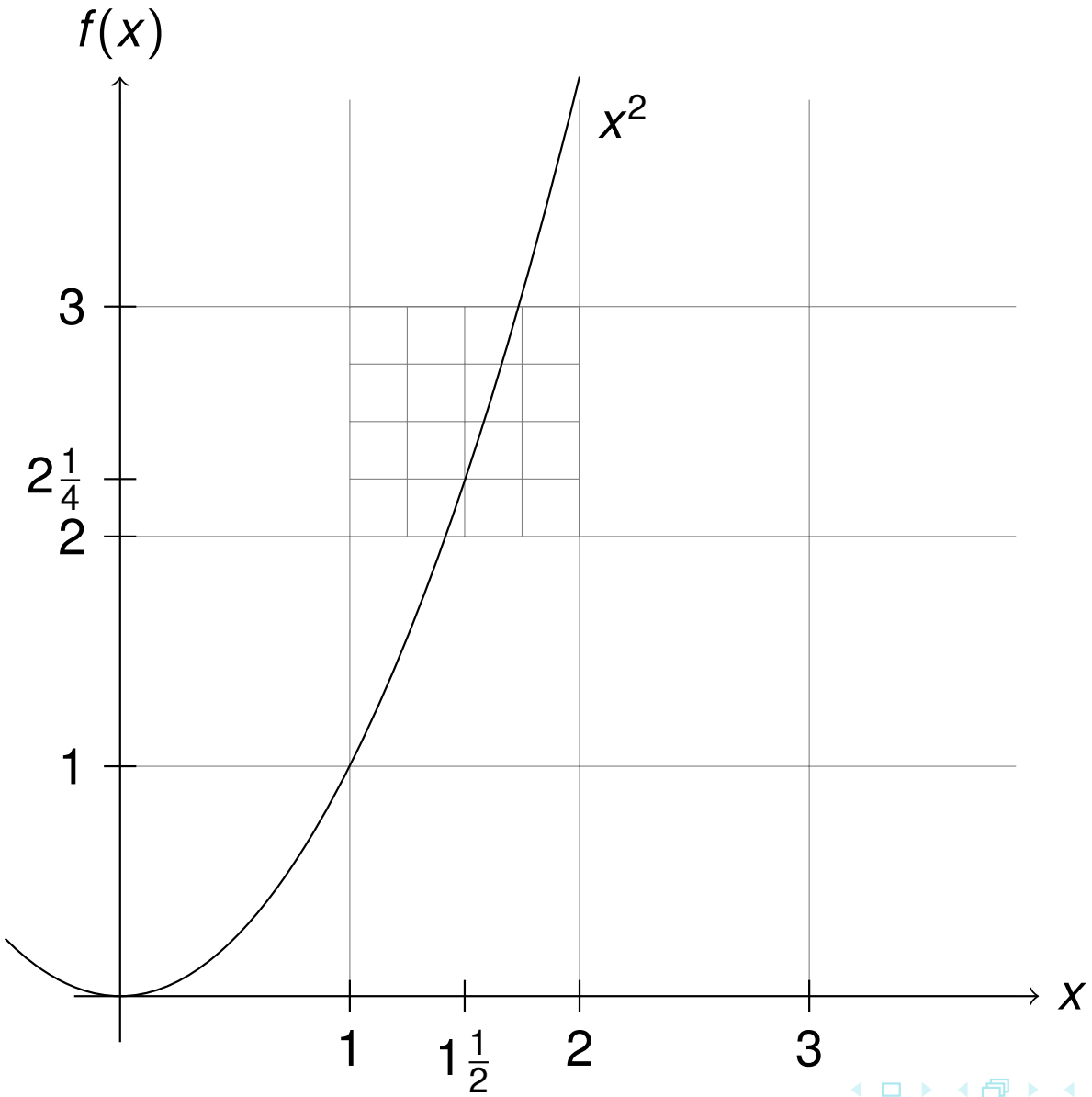
If *AB* and *CD* are parallel, i. e., *AB* || *CD*, then $\alpha = \delta$ and $\beta = \gamma$.

Source code for the seventh picture

```
\begin{tikzpicture}
  \draw[fill=yellow] (0,0) -- (60:.75cm) arc (60:180:.75cm);
  \draw (120:0.4cm) node {$\alpha$};
  \draw[fill=green!30] (0,0) -- (right:.75cm) arc (0:60:.75cm);
  \draw (30:0.5cm) node {$\beta$};
  \begin{scope}[shift={(60:2cm)}]
    \draw[fill=green!30] (0,0) -- (180:.75cm) arc (180:240:.75cm);
    \draw (30:-0.5cm) node {$\gamma$};
    \draw[fill=yellow] (0,0) -- (240:.75cm) arc (240:360:.75cm);
    \draw (-60:0.4cm) node {$\delta$};
  \end{scope}
  \begin{scope}[thick]
    \draw (60:-1cm) node[fill=white] {$E$} -- (60:3cm) node[fill=white] {$F$};
    \draw[red] (-2,0) node[left] {$A$} -- (3,0) node[right]{$B$};
    \draw[blue,shift={(60:2cm)}] (-3,0) node[left] {$C$} -- (2,0) node[right]{$D$};
    \draw[shift={(60:1cm)},xshift=3.5cm]
    node [right,text width=5cm,rounded corners,fill=red!10,inner sep=1ex] {
    If ${\color{red}AB} \mathbin{\|} \color{blue}CD$, then $\alpha = \delta$
    and $\beta = \gamma$.};
  \end{scope}
\end{tikzpicture}
```

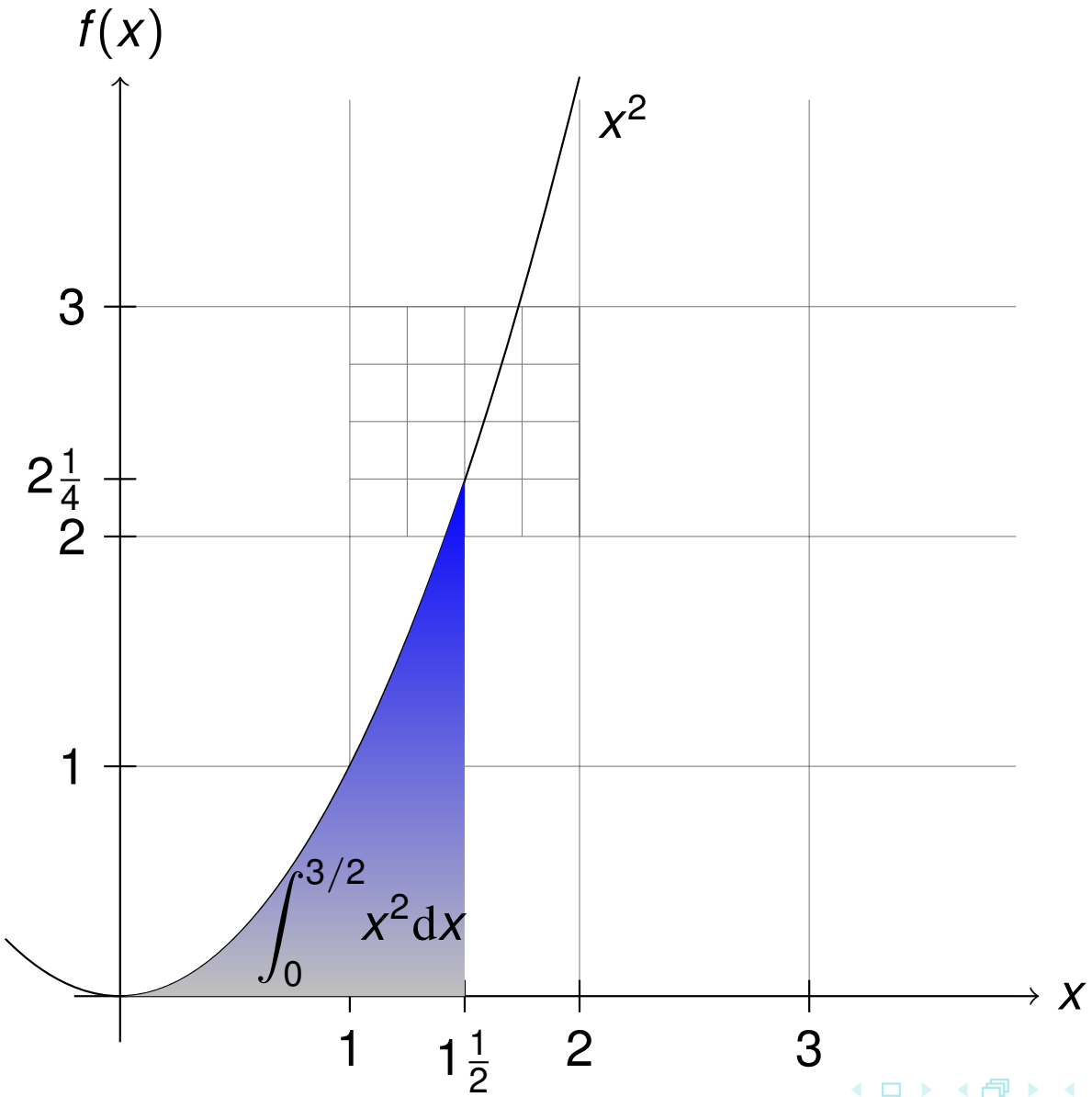
TikZ examples

Eighth example



TikZ examples

Eighth example



TikZ examples

Eighth example

Source code for the picture on the previous page

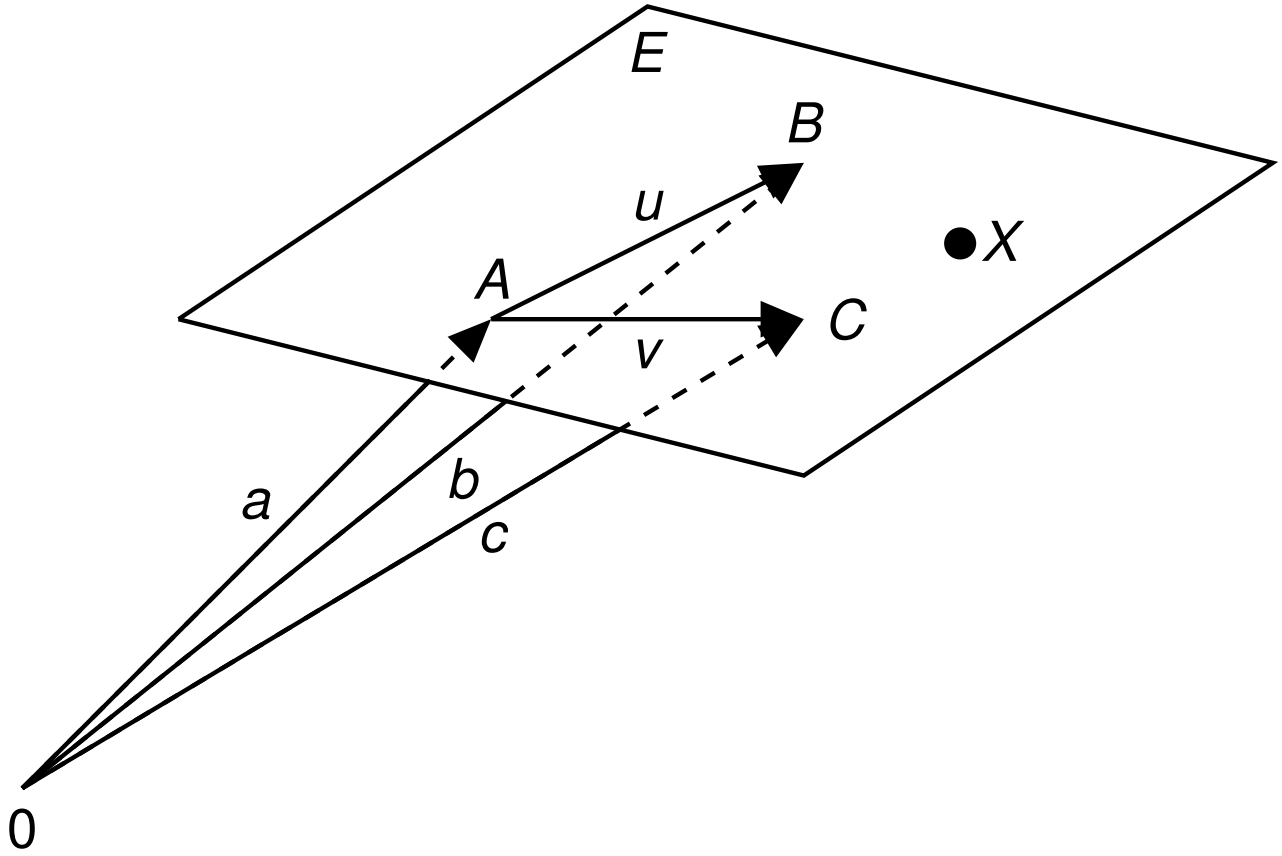
Source code for the eighth picture

```
\begin{tikzpicture}[scale=1.6]
\put(60,0) {
  \draw[style=help lines] (0,0) grid (3.9,3.9)
  [step=0.25cm] (1,2) grid +(1,1);
  \draw[->] (-0.2,0) -- (4,0) node[right] {$x$};
  \draw[->] (0,-0.2) -- (0,4) node[above] {$f(x)$};
  \foreach \x/\xtext in {1/1, 1.5/1\frac{1}{2}, 2/2, 3/3}
  \draw[shift={(\x,0)}] (0pt,2pt) -- (0pt,-2pt) node[below] {$\xtext$};
  \foreach \y/\ytext in {1/1, 2/2, 2.25/2\frac{1}{4}, 3/3}
  \draw[shift={(0,\y)}] (2pt,0pt) -- (-2pt,0pt) node[left] {$\ytext$};
  \draw (-.5,.25) parabola bend (0,0) (2,4) node[below right] {$x^2$};

  \only<2-> {
    \shade[top color=blue,bottom color=gray!50] (0,0)
      parabola (1.5,2.25) |- (0,0);
    \draw (1.05cm,-1pt) node[above]
      {$\displaystyle\int_0^{3/2} \!\!x^2\mathrm{d}x$};
    }
  }
\end{tikzpicture}
```

TikZ examples

Ninth example



TikZ examples

Ninth example

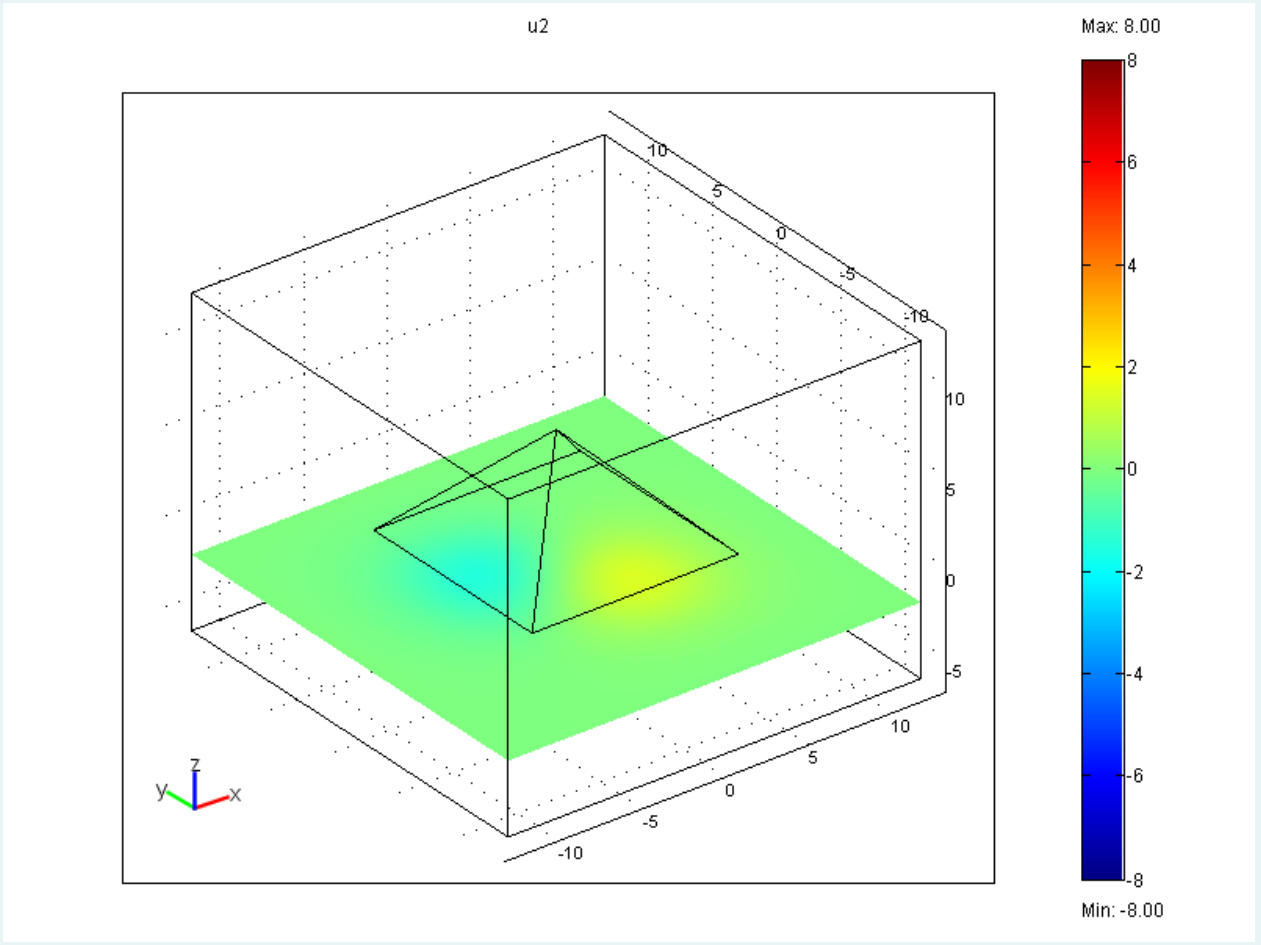
Source code for the picture on the previous page

Source code to the picture for the ninth page

```
\begin{tikzpicture}
  \put(60,0){
  \draw[thick](1,3)--(4,5)--(8,4)--(5,2)--(1,3)
    node[above=2.2cm,pos=0.25]{$E$};
  \draw[thick,-triangle 45,dashed](0,0)--(3,3)coordinate(a)
    node[above]{$A$}  node[below,pos=0]{$0$};
  \draw[thick](0,0)--(intersection of 1,3--5,2 and 0,0--3,3)
    node[left=3pt,pos=0.7]{$a$};
  \draw[thick,-triangle 45,dashed](0,0)--(5,3) coordinate(c)
    node[right]{$C$};
  \draw[thick](0,0)--(intersection of 1,3--5,2 and 0,0--5,3)
    node[right=3pt,pos=0.7]{$c$};
  \draw[thick,-triangle 45,dashed](0,0)--(5,4)coordinate(b)
    node[above]{$B$};
  \draw[thick](0,0)--(intersection of 1,3--5,2 and 0,0--5,4)
    node[right=3pt,pos=0.8]{$b$};
  \draw[thick,-triangle 45](a)--(c)
    node[below,pos=0.5]{$v$};
  \draw[thick,-triangle 45](a)--(b)
    node[above,pos=0.5]{$u$};
  \draw[thick,-*](6,3.5)--(6,3.5)
    node[right]{$X$};
  }
\end{tikzpicture}
```

Dynamically Changing Images

Pictures

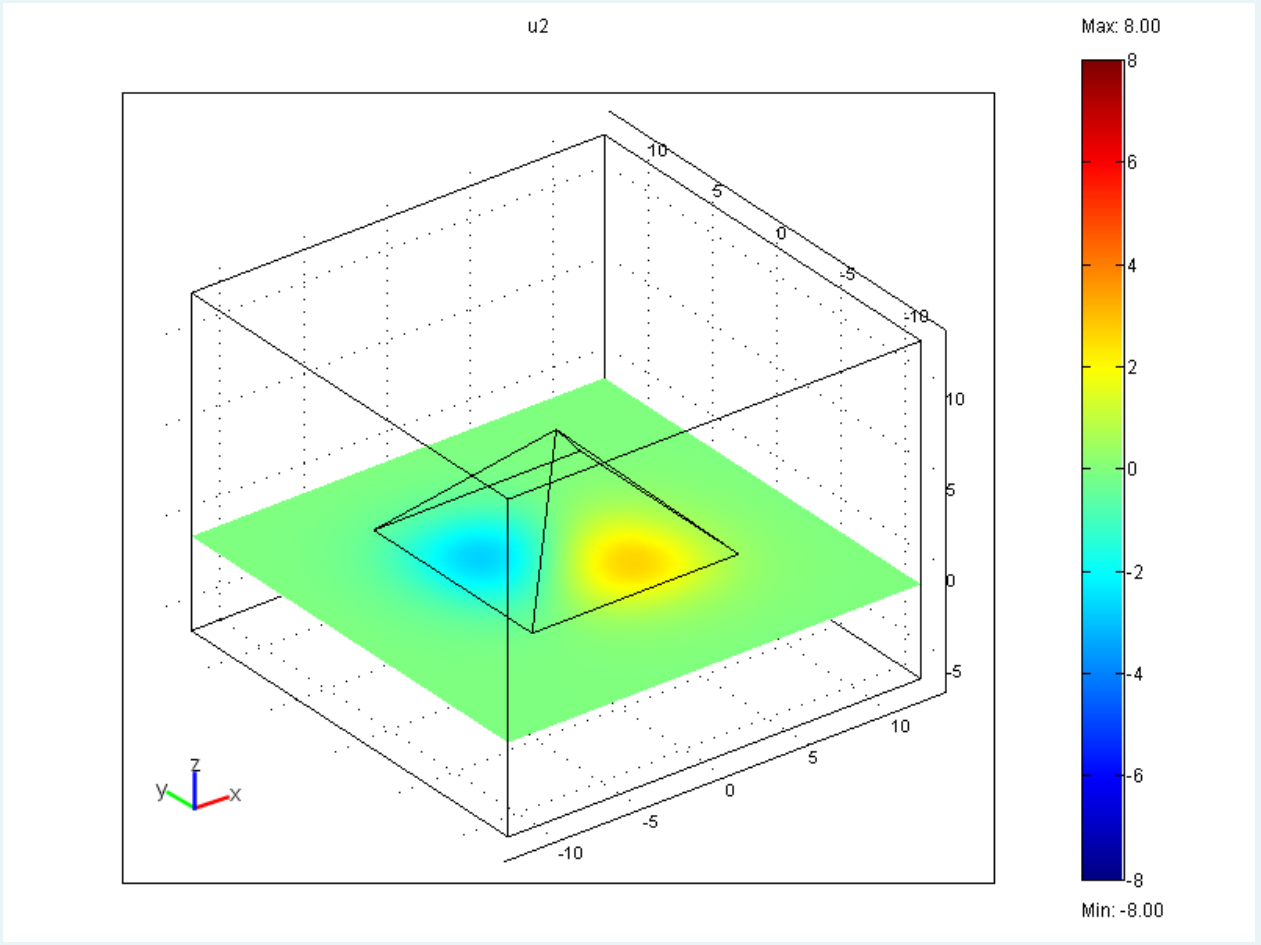


Legend

- picture one
- picture second
- picture third
- picture forth
- picture fifth

Dynamically Changing Images

Pictures

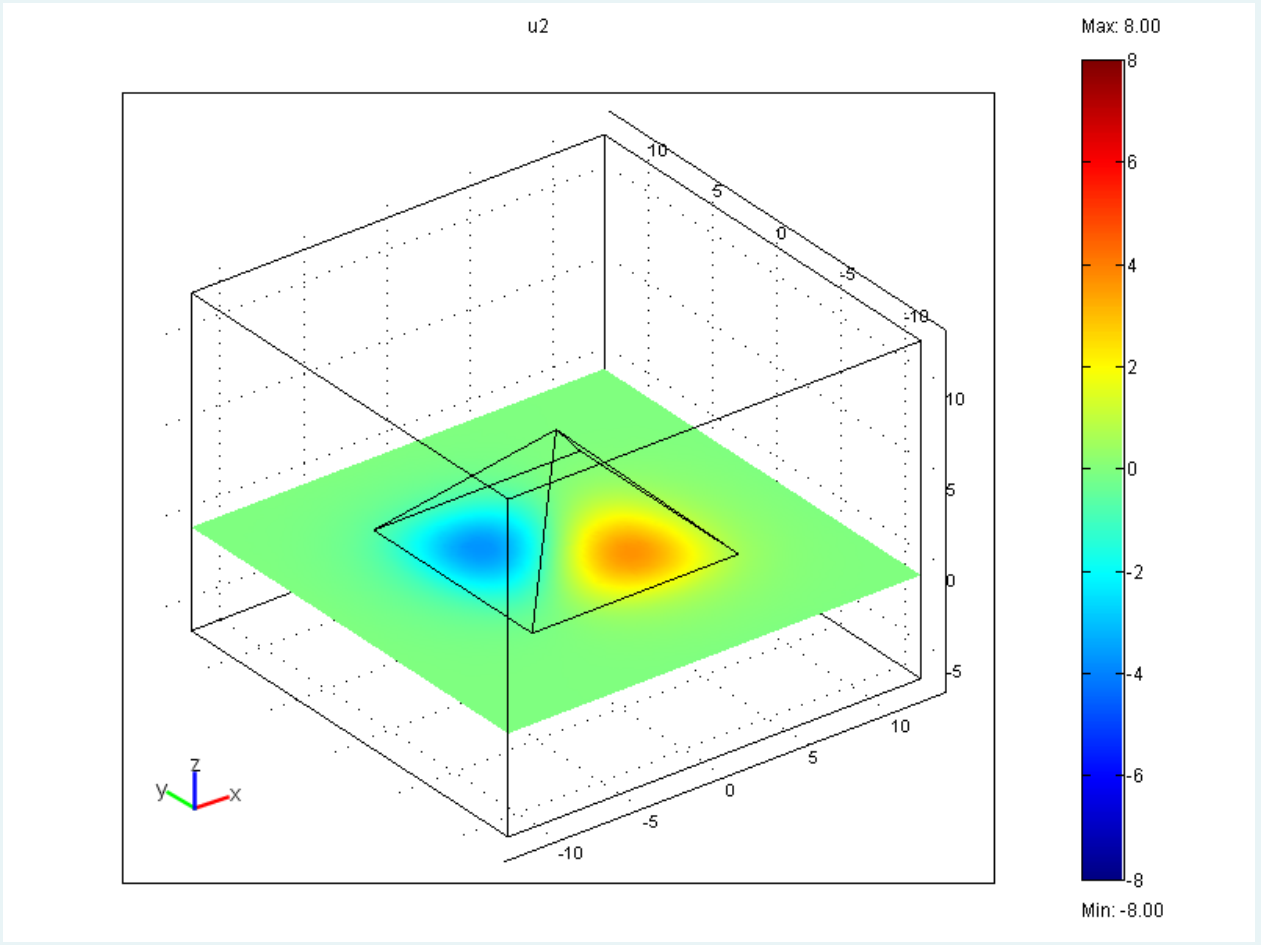


Legend

- picture one
- picture second
- picture third
- picture forth
- picture fifth

Dynamically Changing Images

Pictures

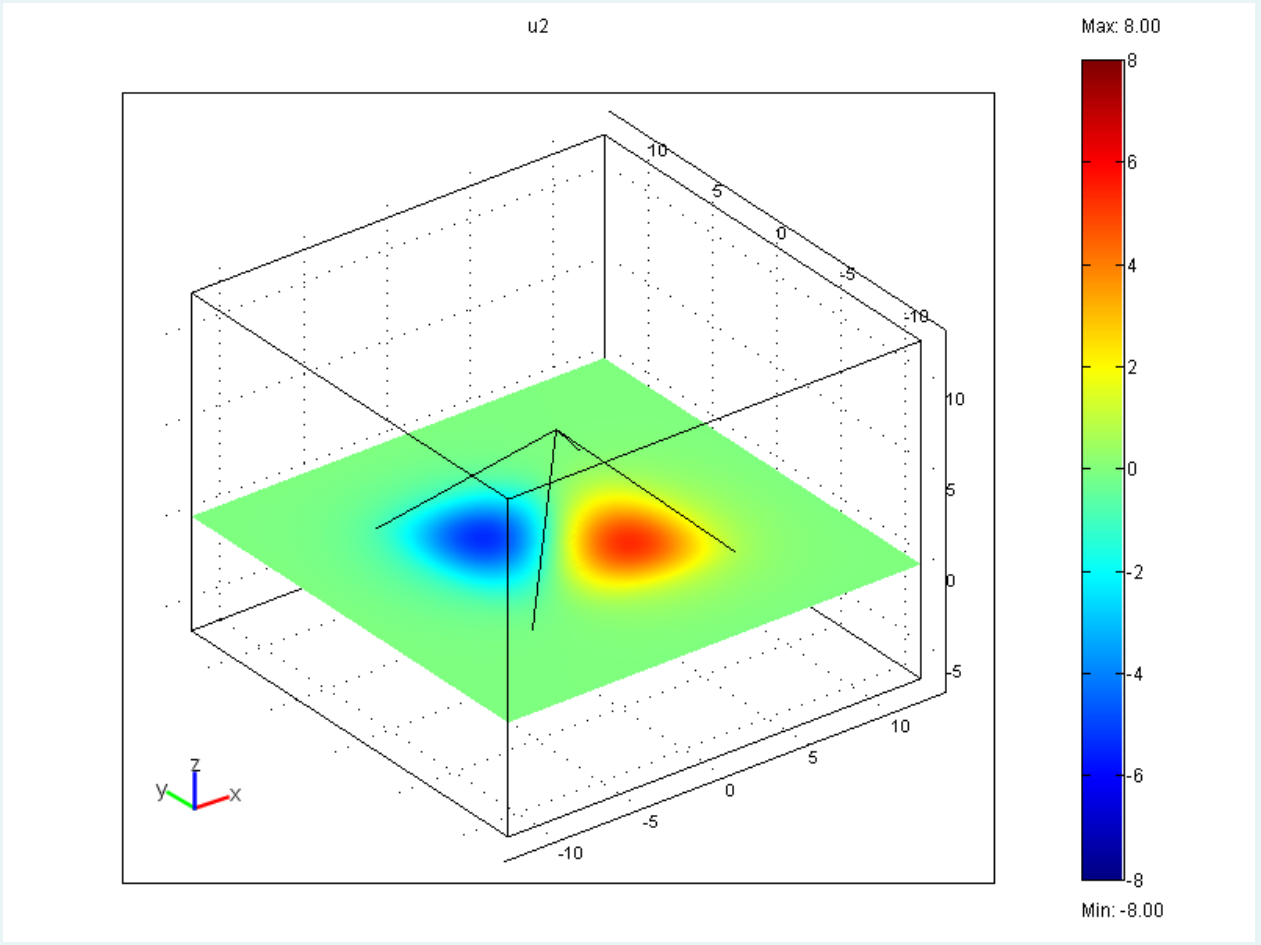


Legend

- picture one
- picture second
- picture third
- picture forth
- picture fifth

Dynamically Changing Images

Pictures

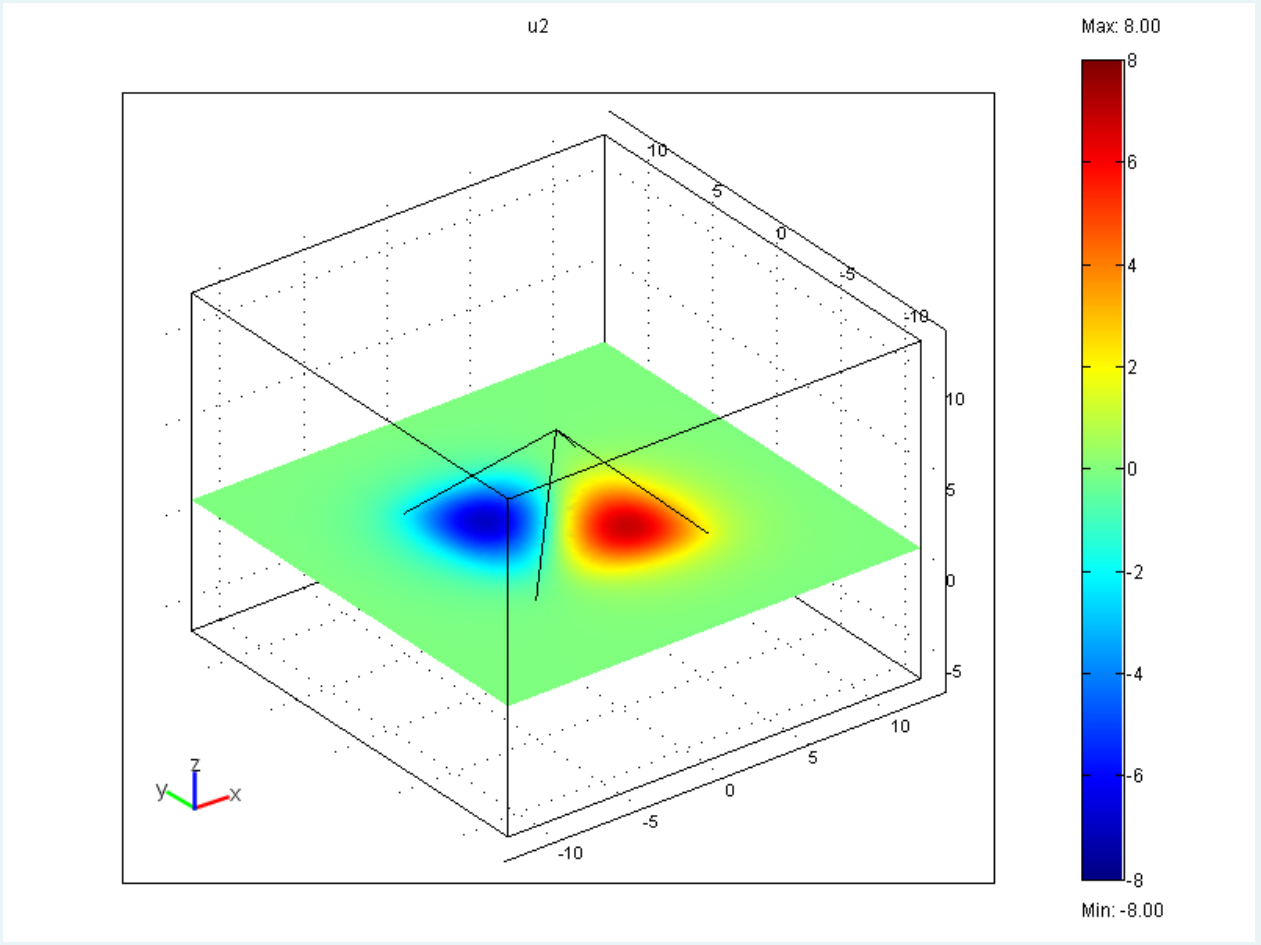


Legend

- picture one
- picture second
- picture third
- picture forth
- picture fifth

Dynamically Changing Images

Pictures



Legend

- picture one
- picture second
- picture third
- picture forth
- picture fifth

Outline

Including External Animation Files

Including External Animation Files

play video

Summary

- The first main message of your talk in one or two lines.
- The second main message of your talk in one or two lines.
- Perhaps a third message, but not more than that.

- Outlook
  - Something you haven't solved.
  - Something else you haven't solved.

A. Author.
Handbook of Everything.
Some Press, 300 BC.

S. Someone.
On this and that.
Journal of This and That, pp.50–100, 350 BC.