

Avoiding problems, Solving problems, Asking for help

Jonathan Fine
LTS Strategic
Open University
Milton Keynes
J.Fine@open.ac.uk

20 October 2006

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching `comp.text.tex` and the internet

Creating a minimal examples

Asking for help

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Types of problem

First we discuss four common types of problems.

1. Installation

Types of problem

First we discuss four common types of problems.

1. Installation
2. Authoring

Types of problem

First we discuss four common types of problems.

1. Installation
2. Authoring
3. Compiling (typesetting)

Types of problem

First we discuss four common types of problems.

1. Installation
2. Authoring
3. Compiling (typesetting)
4. Doh! and other Gotchas

Types of problem

First we discuss four common types of problems.

1. Installation
2. Authoring
3. Compiling (typesetting)
4. Doh! and other Gotchas

Some other types of problems are including graphics, preview, printing and, finally, bugs in TEX the program (very rare).

Types of problem

First we discuss four common types of problems.

1. Installation
2. Authoring
3. Compiling (typesetting)
4. Doh! and other Gotchas

Some other types of problems are including graphics, preview, printing and, finally, bugs in T_EX the program (very rare).

Here are some recent examples of problems I've dealt with:

- ▶ T_EX cannot write to output dvi file
- ▶ T_EX hangs
- ▶ T_EX creates corrupt dvi file
- ▶ Unexplained change in pagination

Types of problem

First we discuss four common types of problems.

1. Installation
2. Authoring
3. Compiling (typesetting)
4. Doh! and other Gotchas

Some other types of problems are including graphics, preview, printing and, finally, bugs in T_EX the program (very rare).

Here are some recent examples of problems I've dealt with:

- ▶ T_EX cannot write to output dvi file
- ▶ T_EX hangs
- ▶ T_EX creates corrupt dvi file
- ▶ Unexplained change in pagination

Earlier run of T_EX not finished, network failure, out of disk space, change in math symbol font used in subscripts.

Installation problems

Something very basic is not working.

Installation problems

Something very basic is not working.

- ▶ T_EX the program is identical on all platforms

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform
- ▶ Don't change or update your installation without good reason

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform
- ▶ Don't change or update your installation without good reason
- ▶ There are distribution-specific newsgroups

Installation problems

Something very basic is not working.

- ▶ T_EX the program is identical on all platforms
- ▶ Most 'pure' T_EX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform
- ▶ Don't change or update your installation without good reason
- ▶ There are distribution-specific newsgroups
 - ▶ Mac OS/X: gmane.comp.tex.macosx
 - ▶ MikTeX: gmane.comp.tex.miktex
 - ▶ TeXLive: gmane.comp.tex.texlive

Is yours a problem with a particular distribution?

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform
- ▶ Don't change or update your installation without good reason
- ▶ There are distribution-specific newsgroups
 - ▶ Mac OS/X: gmane.comp.tex.macosx
 - ▶ MikTeX: gmane.comp.tex.miktex
 - ▶ TeXLive: gmane.comp.tex.texlive

Is yours a problem with a particular distribution?

- ▶ For each platform, there may be several 'front ends'

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform
- ▶ Don't change or update your installation without good reason
- ▶ There are distribution-specific newsgroups
 - ▶ Mac OS/X: gmane.comp.tex.macosx
 - ▶ MikTeX: gmane.comp.tex.miktex
 - ▶ TeXLive: gmane.comp.tex.texlive

Is yours a problem with a particular distribution?

- ▶ For each platform, there may be several 'front ends'
 - ▶ Windows: WinEdt
 - ▶ Windows: Texnic centre
 - ▶ Mac OS/X: TeXshop

Is yours a problem with a particular front end?

Installation problems

Something very basic is not working.

- ▶ \TeX the program is identical on all platforms
- ▶ Most 'pure' \TeX errors are (or should be) cross-platform
- ▶ Many preview, printing and graphics problems depend on the platform
- ▶ Don't change or update your installation without good reason
- ▶ There are distribution-specific newsgroups
 - ▶ Mac OS/X: gmane.comp.tex.macosx
 - ▶ MikTeX: gmane.comp.tex.miktex
 - ▶ TeXLive: gmane.comp.tex.texlive

Is yours a problem with a particular distribution?

- ▶ For each platform, there may be several 'front ends'
 - ▶ Windows: WinEdt
 - ▶ Windows: Texnic centre
 - ▶ Mac OS/X: TeXshop

Is yours a problem with a particular front end?

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in T_EX.

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in T_EX.

- ▶ Focus on your thesis/article/book

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in T_EX.

- ▶ Focus on your thesis/article/book
- ▶ Don't write your own complex macros

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in T_EX.

- ▶ Focus on your thesis/article/book
- ▶ Don't write your own complex macros
- ▶ Keep close to standard L^AT_EX

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in $\text{T}_{\text{E}}\text{X}$.

- ▶ Focus on your thesis/article/book
- ▶ Don't write your own complex macros
- ▶ Keep close to standard \LaTeX
- ▶ Use the AMS maths packages

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in $\text{T}_{\text{E}}\text{X}$.

- ▶ Focus on your thesis/article/book
- ▶ Don't write your own complex macros
- ▶ Keep close to standard \LaTeX
- ▶ Use the AMS maths packages
- ▶ Find a good source for \LaTeX coding in *your* area.

Authoring problems 1/2

You know what you want (we hope), but you don't know how to do it in T_EX.

- ▶ Focus on your thesis/article/book
- ▶ Don't write your own complex macros
- ▶ Keep close to standard L^AT_EX
- ▶ Use the AMS maths packages
- ▶ Find a good source for L^AT_EX coding in *your* area. Some examples:
 - ▶ *The T_EXbook*, by Don Knuth
 - ▶ *L^AT_EX user's guide*, by Leslie Lamport
 - ▶ *The L^AT_EX companion*, by Frank Mittlebach et al
 - ▶ L^AT_EX for logicians website, by Peter Smith
 - ▶ *Math into L^AT_EX*, by George Grätzer
 - ▶ *LaTeX line by line*, by Antoni Diller

Authoring problems 2/2

Keep it simple. Hegel said:

To do something great, one must learn to limit oneself

Authoring problems 2/2

Keep it simple. Hegel said:

To do something great, one must learn to limit oneself

- ▶ Practice writing simply
- ▶ Take care with choice of notation

Authoring problems 2/2

Keep it simple. Hegel said:

To do something great, one must learn to limit oneself

- ▶ Practice writing simply
- ▶ Take care with choice of notation
- ▶ Draw only simple diagrams and figures
- ▶ How much \LaTeX do I need to know?

Authoring problems 2/2

Keep it simple. Hegel said:

To do something great, one must learn to limit oneself

- ▶ Practice writing simply
- ▶ Take care with choice of notation
- ▶ Draw only simple diagrams and figures
- ▶ How much \LaTeX do I need to know?
- ▶ Learn from the example of your peers (ask your colleagues). . .
- ▶ . . . but you don't have to follow their example

Authoring problems 2/2

Keep it simple. Hegel said:

To do something great, one must learn to limit oneself

- ▶ Practice writing simply
- ▶ Take care with choice of notation
- ▶ Draw only simple diagrams and figures
- ▶ How much \LaTeX do I need to know?
- ▶ Learn from the example of your peers (ask your colleagues). . .
- ▶ . . . but you don't have to follow their example
- ▶ Use existing symbols whenever possible
- ▶ Use existing successful documents as templates

Authoring problems 2/2

Keep it simple. Hegel said:

To do something great, one must learn to limit oneself

- ▶ Practice writing simply
- ▶ Take care with choice of notation
- ▶ Draw only simple diagrams and figures
- ▶ How much \LaTeX do I need to know?
- ▶ Learn from the example of your peers (ask your colleagues). . .
- ▶ . . . but you don't have to follow their example
- ▶ Use existing symbols whenever possible
- ▶ Use existing successful documents as templates
- ▶ Use simple typography, e.g. remove rules from tables

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates
- ▶ Compile the document often (particularly if you are not so confident)

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates
- ▶ Compile the document often (particularly if you are not so confident)
- ▶ Insert { and } as matched pairs
- ▶ Insert environment commands as matched pairs

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates
- ▶ Compile the document often (particularly if you are not so confident)
- ▶ Insert { and } as matched pairs
- ▶ Insert environment commands as matched pairs
- ▶ Copy offending item to top of file, and `\end{document}` early

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates
- ▶ Compile the document often (particularly if you are not so confident)
- ▶ Insert { and } as matched pairs
- ▶ Insert environment commands as matched pairs
- ▶ Copy offending item to top of file, and `\end{document}` early
- ▶ Confusion results when you forget to supply arguments to a command

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates
- ▶ Compile the document often (particularly if you are not so confident)
- ▶ Insert { and } as matched pairs
- ▶ Insert environment commands as matched pairs
- ▶ Copy offending item to top of file, and `\end{document}` early
- ▶ Confusion results when you forget to supply arguments to a command
- ▶ Sometimes deleting the generated (auxiliary) files clears a problem

Compilation errors

Most of these are T_EX errors, such as an undefined control sequence, a missing brace, or an extra brace.

- ▶ Use an editor that provides syntax highlighting
- ▶ Use a front end that provides templates
- ▶ Compile the document often (particularly if you are not so confident)
- ▶ Insert { and } as matched pairs
- ▶ Insert environment commands as matched pairs
- ▶ Copy offending item to top of file, and `\end{document}` early
- ▶ Confusion results when you forget to supply arguments to a command
- ▶ Sometimes deleting the generated (auxiliary) files clears a problem

We all make errors that are caught in compilation

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour
4. The package `hyperref` likes to be loaded last

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour
4. The package `hyperref` likes to be loaded last
5. Conversion to HTML is not always reliable

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour
4. The package `hyperref` likes to be loaded last
5. Conversion to HTML is not always reliable
6. Macro packages that make characters active (in addition to `~`) often cause problems in \LaTeX

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour
4. The package `hyperref` likes to be loaded last
5. Conversion to HTML is not always reliable
6. Macro packages that make characters active (in addition to `~`) often cause problems in \LaTeX
7. Float placement and complex makeup can be tricky

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour
4. The package `hyperref` likes to be loaded last
5. Conversion to HTML is not always reliable
6. Macro packages that make characters active (in addition to `~`) often cause problems in \LaTeX
7. Float placement and complex page makeup can be tricky

Here's an obscure gotcha: `$$\alpha\time\beta$` produces α . So where has the β gone?

Doh! and other Gotchas!

There are some things that that \LaTeX is not well equipped to handle.

1. Verbatim (e.g. typewriter computer code) does not work in macro arguments
2. You cannot wrap verbatim environments
3. \LaTeX does not provide perfect support for colour
4. The package `hyperref` likes to be loaded last
5. Conversion to HTML is not always reliable
6. Macro packages that make characters active (in addition to `~`) often cause problems in \LaTeX
7. Float placement and complex page makeup can be tricky

Here's an obscure gotcha: `\alpha\time\beta` produces α . So where has the β gone?

`\time` is a count parameter, and `\beta` is a `\mathchar`. Author should have written `\times`.

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros . . .

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros . . .
- ▶ . . . except for good purposes

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros ...
- ▶ ... except for good purposes
- ▶ OK: `\newcommand\sigmabar{\overline{\sigma}}`

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros ...
- ▶ ... except for good purposes
- ▶ OK: `\newcommand\sigmabar{\overline{\sigma}}`
- ▶ Good motives: Save on typing, consistency of notation and typography
- ▶ More examples to follow

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros ...
- ▶ ... except for good purposes
- ▶ OK: `\newcommand\sigmabar{\overline{\sigma}}`
- ▶ Good motives: Save on typing, consistency of notation and typography
- ▶ More examples to follow
- ▶ Don't change category codes

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros ...
- ▶ ... except for good purposes
- ▶ OK: `\newcommand\sigmabar{\overline{\sigma}}`
- ▶ Good motives: Save on typing, consistency of notation and typography
- ▶ More examples to follow
- ▶ Don't change category codes
- ▶ Don't try to wrap verbatim environments

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros ...
- ▶ ... except for good purposes
- ▶ OK: `\newcommand\sigmabar{\overline{\sigma}}`
- ▶ Good motives: Save on typing, consistency of notation and typography
- ▶ More examples to follow
- ▶ Don't change category codes
- ▶ Don't try to wrap verbatim environments
- ▶ Don't write macros with optional parameters

Author defined macros 1/2

This is an interesting and sometimes controversial subject.

- ▶ Don't define your own macros ...
- ▶ ... except for good purposes
- ▶ OK: `\newcommand\sigmabar{\overline{\sigma}}`
- ▶ Good motives: Save on typing, consistency of notation and typography
- ▶ More examples to follow
- ▶ Don't change category codes
- ▶ Don't try to wrap verbatim environments
- ▶ Don't write macros with optional parameters
- ▶ Don't load packages just so you can write a macro

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros . . .

- ▶ Don't redefine existing commands

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros . . .

- ▶ Don't redefine existing commands
- ▶ Don't invent a new syntax

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros . . .

- ▶ Don't redefine existing commands
- ▶ Don't invent a new syntax
- ▶ Don't change any category codes

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros ...

- ▶ Don't redefine existing commands
- ▶ Don't invent a new syntax
- ▶ Don't change any category codes
- ▶ Don't write macros that are 'delimited', e.g. by `\this`
e.g. `\def\wibble #1\this #2 { ...`

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros ...

- ▶ Don't redefine existing commands
- ▶ Don't invent a new syntax
- ▶ Don't change any category codes
- ▶ Don't write macros that are 'delimited', e.g. by `\this`
e.g. `\def\wibble #1\this #2 { ...`
- ▶ Take great care with stray white space, e.g. after #2 above

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros ...

- ▶ Don't redefine existing commands
- ▶ Don't invent a new syntax
- ▶ Don't change any category codes
- ▶ Don't write macros that are 'delimited', e.g. by `\this`
e.g. `\def\wibble #1\this #2 { ...`
- ▶ Take great care with stray white space, e.g. after #2 above
- ▶ Worth repeating — stray white space

Author defined macros 2/2

When you write complex macros, you are ceasing to be an author. Go outside the author area, and there's an awful lot you may need to know.

If you must define your own macros ...

- ▶ Don't redefine existing commands
- ▶ Don't invent a new syntax
- ▶ Don't change any category codes
- ▶ Don't write macros that are 'delimited', e.g. by `\this`
e.g. `\def\wibble #1\this #2 { ...`
- ▶ Take great care with stray white space, e.g. after #2 above
- ▶ Worth repeating — stray white space

...and don't expect help if you post a so-called minimal example that uses these complex macros you defined.

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Note that

- ▶ We have here 10 numbers

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Note that

- ▶ We have here 10 numbers
- ▶ Each number is the previous, plus 210

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Note that

- ▶ We have here 10 numbers
- ▶ Each number is the previous, plus 210
- ▶ Each of these numbers is prime

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Note that

- ▶ We have here 10 numbers
- ▶ Each number is the previous, plus 210
- ▶ Each of these numbers is prime

Tao and Green proved “The primes contain arbitrarily long arithmetic progressions”. This is a major result, and helped Tao win the Fields medal this year (roughly equivalent to Nobel prize).

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Note that

- ▶ We have here 10 numbers
- ▶ Each number is the previous, plus 210
- ▶ Each of these numbers is prime

Tao and Green proved “The primes contain arbitrarily long arithmetic progressions”. This is a major result, and helped Tao win the Fields medal this year (roughly equivalent to Nobel prize).

They wrote it, of course, in L^AT_EX.

Real life T_EX: Green and Tao on primes 1/2

In 2004 Ben Green and Terence Tao proved a major result in mathematics (number theory).

Consider the sequence 199, 409, 619, 829, 1039, 1249, 1459, 1669, 1879, 2089.

Note that

- ▶ We have here 10 numbers
- ▶ Each number is the previous, plus 210
- ▶ Each of these numbers is prime

Tao and Green proved “The primes contain arbitrarily long arithmetic progressions”. This is a major result, and helped Tao win the Fields medal this year (roughly equivalent to Nobel prize).

They wrote it, of course, in L^AT_EX.

Question: How much L^AT_EX did Green and Tao need to write this paper?

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions
3. Simple numbering scheme for theorem-like elements

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions
3. Simple numbering scheme for theorem-like elements
4. About 30 author's convenience macros

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions
3. Simple numbering scheme for theorem-like elements
4. About 30 author's convenience macros
5. Commands defined using `\def` (T_EX primitive) instead of `\newcommand` (tut, tut)

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions
3. Simple numbering scheme for theorem-like elements
4. About 30 author's convenience macros
5. Commands defined using `\def` (T_EX primitive) instead of `\newcommand` (tut, tut)
6. Some of the definitions could be improved

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions
3. Simple numbering scheme for theorem-like elements
4. About 30 author's convenience macros
5. Commands defined using `\def` (T_EX primitive) instead of `\newcommand` (tut, tut)
6. Some of the definitions could be improved

Conclusion: You don't have to be a L^AT_EX wizard to be successful (and win a major prize) in mathematics.

Real life T_EX: Green and Tao on primes 2/2

L^AT_EX source downloaded from <http://www.arXiv.org>, and used here with authors' permission. (See Appendix to handouts.)

1. Only standard packages used
2. Hand fiddling of dimensions
3. Simple numbering scheme for theorem-like elements
4. About 30 author's convenience macros
5. Commands defined using `\def` (T_EX primitive) instead of `\newcommand` (tut, tut)
6. Some of the definitions could be improved

Conclusion: You don't have to be a L^AT_EX wizard to be successful (and win a major prize) in mathematics. (And you can download L^AT_EX source for papers in your field.)

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Real life T_EX: Printing handouts from beamer 1/2

1. Search internet to find pdf for beamer manual

Real life T_EX: Printing handouts from beamer 1/2

1. Search internet to find pdf for beamer manual
2. Print out documentation (214 pages!) to protect my eyes

Real life T_EX: Printing handouts from beamer 1/2

1. Search internet to find pdf for beamer manual
2. Print out documentation (214 pages!) to protect my eyes
3. A piece of luck. Table of Contents: Creating Handouts

Real life T_EX: Printing handouts from beamer 1/2

1. Search internet to find pdf for beamer manual
2. Print out documentation (214 pages!) to protect my eyes
3. A piece of luck. Table of Contents: Creating Handouts
4. Beamer manual says use the command
`\pgfpage layout{2 on 1}{a4paper} % from beamer manual`

Real life T_EX: Printing handouts from beamer 1/2

1. Search internet to find pdf for beamer manual
2. Print out documentation (214 pages!) to protect my eyes
3. A piece of luck. Table of Contents: Creating Handouts
4. Beamer manual says use the command
`\pgfpage layout{2 on 1}{a4paper}` % from beamer manual
5. Get and read the log file error message

```
(C:\texmf\tex\latex\beamer\themes\outer\beamerouterthem
```

```
(C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

```
(C:\texmf\tex\latex\tools\calc.sty))
```

```
! Undefined control sequence.
```

```
1.4 \pgfpage layout
```

```
      {2 on 1}[a4paper] % from beamer manual
```

```
?
```

Real life T_EX: Printing handouts from beamer 1/2

1. Search internet to find pdf for beamer manual
2. Print out documentation (214 pages!) to protect my eyes
3. A piece of luck. Table of Contents: Creating Handouts
4. Beamer manual says use the command
5. Get and read the log file error message

```
\pgfpagelayout{2 on 1}{a4paper} % from beamer manual
```

```
(C:\texmf\tex\latex\beamer\themes\outer\beamerouterthem
```

```
(C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

```
(C:\texmf\tex\latex\tools\calc.sty))
```

```
! Undefined control sequence.
```

```
1.4 \pgfpagelayout
```

```
{2 on 1}[a4paper] % from beamer manual
```

```
?
```

6. Open the file

```
C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

Real life T_EX: Printing handouts from beamer 2/2

1. From last slide, open the file

```
C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

Real life T_EX: Printing handouts from beamer 2/2

1. From last slide, open the file

```
C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

2. Re-read the log file error message

```
! Undefined control sequence.
```

```
1.4 \pgfpagelayout
```

```
          {2 on 1}[a4paper] % from beamer manual
```

```
?
```

Real life T_EX: Printing handouts from beamer 2/2

1. From last slide, open the file

```
C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

2. Re-read the log file error message

```
! Undefined control sequence.
```

```
1.4 \pgfpagelayout
```

```
          {2 on 1}[a4paper] % from beamer manual
```

```
?
```

3. Search for 'layout' in pgfpages.sty

Real life T_EX: Printing handouts from beamer 2/2

1. From last slide, open the file

```
C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

2. Re-read the log file error message

```
! Undefined control sequence.
```

```
1.4 \pgfpagelayout
```

```
          {2 on 1}[a4paper] % from beamer manual
```

```
?
```

3. Search for 'layout' in pgfpages.sty

4. Find the following, and follow its advice

```
% Use a layout
```

```
% #1 = layout name
```

```
% #2 = options
```

```
% Example:
```

```
% \pgfpagesuselayout{resize to}[a4paper]
```

Real life T_EX: Printing handouts from beamer 2/2

1. From last slide, open the file

```
C:\texmf\tex\latex\pgf\utilities\pgfpages.sty
```

2. Re-read the log file error message

```
! Undefined control sequence.
```

```
1.4 \pgfpagelayout
```

```
          {2 on 1}[a4paper] % from beamer manual
```

```
?
```

3. Search for 'layout' in pgfpages.sty

4. Find the following, and follow its advice

```
% Use a layout
```

```
% #1 = layout name
```

```
% #2 = options
```

```
% Example:
```

```
% \pgfpagesuselayout{resize to}[a4paper]
```

5. Be a careful good citizen, if you can (Sourceforge bug id 1490542)

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Problem solving tools

- ▶ Syntax highlighting in your editor

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)
- ▶ `\tracingpages = 1` (and expect a large log file)

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)
- ▶ `\tracingpages = 1` (and expect a large log file)
- ▶ The `dvitype` program

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)
- ▶ `\tracingpages = 1` (and expect a large log file)
- ▶ The `dvitype` program
- ▶ The `cmp` and `diff` programs

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)
- ▶ `\tracingpages = 1` (and expect a large log file)
- ▶ The `dvitype` program
- ▶ The `cmp` and `diff` programs
- ▶ For font problems, `tex testfont` to make font table

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)
- ▶ `\tracingpages = 1` (and expect a large log file)
- ▶ The `dvitype` program
- ▶ The `cmp` and `diff` programs
- ▶ For font problems, `tex testfont` to make font table
- ▶ For graphics problems, include graphics only test file

Problem solving tools

- ▶ Syntax highlighting in your editor
- ▶ Version control (commit successes and problems)
- ▶ The log file
- ▶ To skip (include) text, `\iffalse` (`\iftrue`) and `\fi`
- ▶ (Advanced alternative, `\ifcase 1` and `\ifcase 0`)
- ▶ `\tracingall` (and expect a large log file)
- ▶ `\tracingpages = 1` (and expect a large log file)
- ▶ The `dvitype` program
- ▶ The `cmp` and `diff` programs
- ▶ For font problems, `tex testfont` to make font table
- ▶ For graphics problems, include graphics only test file
- ▶ Preamble only test file, as template for minimal example

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching `comp.text.tex` and the internet

Creating a minimal examples

Asking for help

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate
2. I find the internet (web pages) better for problems related to a specific program (such as dvips)
3. I find comp.text.tex better for T_EX macro problems

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate
2. I find the internet (web pages) better for problems related to a specific program (such as dvips)
3. I find comp.text.tex better for T_EX macro problems
4. Use error messages as search keys
e.g. ! LaTeX Error: File 'pgfcore.sty' not found.

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate
2. I find the internet (web pages) better for problems related to a specific program (such as dvips)
3. I find comp.text.tex better for T_EX macro problems
4. Use error messages as search keys
e.g. ! LaTeX Error: File 'pgfcore.sty' not found.
5. You don't need to understand an error message to copy and paste it

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate
2. I find the internet (web pages) better for problems related to a specific program (such as dvips)
3. I find comp.text.tex better for T_EX macro problems
4. Use error messages as search keys
e.g. ! LaTeX Error: File 'pgfcore.sty' not found.
5. You don't need to understand an error message to copy and paste it
6. You may need to exercise some ingenuity in your search
7. Think carefully about how you would describe the problem

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate
2. I find the internet (web pages) better for problems related to a specific program (such as dvips)
3. I find comp.text.tex better for T_EX macro problems
4. Use error messages as search keys
e.g. ! LaTeX Error: File 'pgfcore.sty' not found.
5. You don't need to understand an error message to copy and paste it
6. You may need to exercise some ingenuity in your search
7. Think carefully about how you would describe the problem
8. Use key words from the description of the problem

Searching comp.text.tex and the internet

1. Search internet and newsgroups as appropriate
2. I find the internet (web pages) better for problems related to a specific program (such as dvips)
3. I find comp.text.tex better for T_EX macro problems
4. Use error messages as search keys
e.g. ! LaTeX Error: File 'pgfcore.sty' not found.
5. You don't need to understand an error message to copy and paste it
6. You may need to exercise some ingenuity in your search
7. Think carefully about how you would describe the problem
8. Use key words from the description of the problem
9. Take care *not* to include irrelevant words in your search -
e.g. if a T_EX macro problem, don't include MikTeX

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Creating a minimal example

If you present a complicated problem to solve to the experts on comp.text.tex, and there is nothing clearly wrong with your input, they will ask for *a minimal example*.

Creating a minimal example

If you present a complicated problem to solve to the experts on `comp.text.tex`, and there is nothing clearly wrong with your input, they will ask for *a minimal example*. A small one will be enough though.

Creating a minimal example

If you present a complicated problem to solve to the experts on `comp.text.tex`, and there is nothing clearly wrong with your input, they will ask for *a minimal example*. A small one will be enough though.

How to create a minimal example:

- ▶ This is an art, rather than a science. However, the binary search (or 'divide-and-conquer') is a useful tactic.

Creating a minimal example

If you present a complicated problem to solve to the experts on `comp.text.tex`, and there is nothing clearly wrong with your input, they will ask for *a minimal example*. A small one will be enough though.

How to create a minimal example:

- ▶ This is an art, rather than a science. However, the binary search (or 'divide-and-conquer') is a useful tactic.
- ▶ Getting the minimal example can be hard, e.g. when page breaks are unexpected.

Creating a minimal example

If you present a complicated problem to solve to the experts on `comp.text.tex`, and there is nothing clearly wrong with your input, they will ask for a *minimal example*. A small one will be enough though.

How to create a minimal example:

- ▶ This is an art, rather than a science. However, the binary search (or 'divide-and-conquer') is a useful tactic.
- ▶ Getting the minimal example can be hard, e.g. when page breaks are unexpected.
- ▶ Do look at $\text{T}_{\text{E}}\text{X}$'s log file. It contains many clues.
- ▶ Read error messages from $\text{T}_{\text{E}}\text{X}$ very carefully — $\text{T}_{\text{E}}\text{X}$ does not know what is supposed to be happening, but it's always right about what is happening.

Creating a minimal example

If you present a complicated problem to solve to the experts on `comp.text.tex`, and there is nothing clearly wrong with your input, they will ask for a *minimal example*. A small one will be enough though.

How to create a minimal example:

- ▶ This is an art, rather than a science. However, the binary search (or 'divide-and-conquer') is a useful tactic.
- ▶ Getting the minimal example can be hard, e.g. when page breaks are unexpected.
- ▶ Do look at $\text{T}_{\text{E}}\text{X}$'s log file. It contains many clues.
- ▶ Read error messages from $\text{T}_{\text{E}}\text{X}$ very carefully — $\text{T}_{\text{E}}\text{X}$ does not know what is supposed to be happening, but it's always right about what is happening.
- ▶ I find using `\tracingall` and then jumping to the failure in the log file very useful.

Creating a minimal example

If you present a complicated problem to solve to the experts on `comp.text.tex`, and there is nothing clearly wrong with your input, they will ask for a *minimal example*. A small one will be enough though.

How to create a minimal example:

- ▶ This is an art, rather than a science. However, the binary search (or ‘divide-and-conquer’) is a useful tactic.
- ▶ Getting the minimal example can be hard, e.g. when page breaks are unexpected.
- ▶ Do look at $\text{T}_{\text{E}}\text{X}$ ’s log file. It contains many clues.
- ▶ Read error messages from $\text{T}_{\text{E}}\text{X}$ very carefully — $\text{T}_{\text{E}}\text{X}$ does not know what is supposed to be happening, but it’s always right about what is happening.
- ▶ I find using `\tracingall` and then jumping to the failure in the log file very useful.

Once you get to a minimal (or small) example, the problem is already half-solved. (But half-proved is not-proved.)

Outline

Types of problem

Author defined macros

Real life T_EX— primes

Real life T_EX— Printing handouts from beamer

Problem solving tools

Searching comp.text.tex and the internet

Creating a minimal examples

Asking for help

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences. If you are asking for help on a newsgroup or email discussion forum:

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

If you are asking for help on a newsgroup or email discussion forum:

- ▶ Try to choose the appropriate forum. What sort of problem do you have?

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

If you are asking for help on a newsgroup or email discussion forum:

- ▶ Try to choose the appropriate forum. What sort of problem do you have?
- ▶ Have you searched for the answer yourself?

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

If you are asking for help on a newsgroup or email discussion forum:

- ▶ Try to choose the appropriate forum. What sort of problem do you have?
- ▶ Have you searched for the answer yourself?
- ▶ Have you produced a minimal example?

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

If you are asking for help on a newsgroup or email discussion forum:

- ▶ Try to choose the appropriate forum. What sort of problem do you have?
- ▶ Have you searched for the answer yourself?
- ▶ Have you produced a minimal example?
- ▶ Choose a good title. Here are some recent poor titles (with better title in parentheses):

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

If you are asking for help on a newsgroup or email discussion forum:

- ▶ Try to choose the appropriate forum. What sort of problem do you have?
- ▶ Have you searched for the answer yourself?
- ▶ Have you produced a minimal example?
- ▶ Choose a good title. Here are some recent poor titles (with better title in parentheses):
 - ▶ Metafont (dvips does not find home-made Metafont font)
 - ▶ dvips (Error: dvips: ! out of string space in dospecial)
 - ▶ Tiff image (Can I include TIF images?)

Asking for help

If you are asking someone in person, they may appreciate your providing them with *all* the relevant information. What to do will depend on your relationship with the person, and their preferences.

If you are asking for help on a newsgroup or email discussion forum:

- ▶ Try to choose the appropriate forum. What sort of problem do you have?
- ▶ Have you searched for the answer yourself?
- ▶ Have you produced a minimal example?
- ▶ Choose a good title. Here are some recent poor titles (with better title in parentheses):
 - ▶ Metafont (dvips does not find home-made Metafont font)
 - ▶ dvips (Error: dvips: ! out of string space in dospecial)
 - ▶ Tiff image (Can I include TIF images?)
- ▶ Thank people for their help and efforts

A minimal `\tracingrestores` example

When debugging some \LaTeX macros, I wanted to know what values were restored (global assignment) at the end of the group.

A minimal `\tracingrestores` example

When debugging some \LaTeX macros, I wanted to know what values were restored (global assignment) at the end of the group. What output do you expect from this input?

```
\immediate\write16{Wish to see end of group restores}  
\begingroup  
{\let \aaa \relax} % don't want to see this  
\let \xxx \relax   % do want to see this  
\tracingrestores = 1  
\let \yyy \relax   % and want to see this  
\endgroup
```

A minimal `\tracingrestores` example

When debugging some \LaTeX macros, I wanted to know what values were restored (global assignment) at the end of the group. What output do you expect from this input?

```
\immediate\write16{Wish to see end of group restores}  
\begingroup  
{\let \aaa \relax} % don't want to see this  
\let \xxx \relax % do want to see this  
\tracingrestores = 1  
\let \yyy \relax % and want to see this  
\endgroup
```

This is what \TeX produces

```
Wish to see end of group restores  
{restoring \yyy=undefined}
```

A minimal `\tracingrestores` example

When debugging some \LaTeX macros, I wanted to know what values were restored (global assignment) at the end of the group. What output do you expect from this input?

```
\immediate\write16{Wish to see end of group restores}  
\begingroup  
{\let \aaa \relax} % don't want to see this  
\let \xxx \relax % do want to see this  
\tracingrestores = 1  
\let \yyy \relax % and want to see this  
\endgroup
```

This is what \TeX produces

```
Wish to see end of group restores  
{restoring \yyy=undefined}
```

Why don't we see the restoration of `\xxx`? Is this a bug in \TeX ?