CLP-1 Lecture Slides

The slides in this folder are written to correspond to CLP-1 by Feldman, Rechnitzer, and Yeager. The slides and their source files can be found online at https://github.com/ecyeager/CLP1Slides.

VERSIONS

Most slides have three versions.

beamer The beamer mode is used to generate slides that will be used in class. These have all the "clicks" (pauses inside frames) and space to write solutions to questions. Solutions to most questions are also included.

handout The handout mode is used to generate slides that students can access before lecture and use to take notes. Most clicks are suppressed. Solutions to questions are also suppressed, so students can work in class without spoilers.

prep There are occasional notes with scripts for lecture, notes on common student difficulties, and explanations of otherwise-confusing slides. You wouldn't want to display them in class, but they might be helpful to look at when you're preparing.

Organization

PDFs can be found in the `pdfs` folder, while the folder `src` has their source files.

Slides for a single section are compiled in files named with the section, a short description, and the version type. (For example, `1_3LimitOfFunction_Handout.pdf` is the handout version of Section 1.3, The Limit of a Function.)

The slides are also arranged into batches corresponding to roughly one week of lecture each, for a 12-week class. The filenames give the week number, a description, the range of associated sections, and the version. (For example, `W1Limits_1_1-1_4_Prep` is the prep version of Week 1, which includes sections 1.1-1.4, covering limits.)

IRREGULARITIES

Sections 1.7 and 1.8 are marked as optional. They do have section PDFs, but are not included in any of the weekly batches. In the slides, Section 1.8 is meant to be covered before Section 1.7.

Sections 2.4-2.6 are closely related. They are all included in the section marked 2.4. That is why there are no sections marked 2.5 or 2.6.

Miscellaneous Files

The file `StudentWorkbook.pdf` has slides for the whole course, with answers removed, formatted for printing. Students who wish to write on paper (rather than annotating PDFs on a tablet) can print out the file and have a workbook to follow for the entire semester.

The file `Review.pdf` has a list of questions from Chapters 1-3.

Compiling Options

There are many files, so we included options for compiling them efficiently. The files can be compiled using pdflatex (in the usual way you probably compile LaTeX); using a Python script to compile many files in one go; and using a Python script with parallel processing and latexmk to quickly compile many files in one go.

Compiling Without Python

In the folder `src`, you'll find a list of .tex files corresponding to each section, each weekly batch, StudentWorkbook, ReadMe, and Review.

To change the version of the file (beamer, handout, prep), change which headers are commented out in the .tex file.

Individual .tex files are compiled with pdflatex. From the `src` folder, `pdflatex -output-directory=../pdfs <filename>` will put the PDF into the folder `pdfs`.

Compiling With Python

Running `buildPDFs_parallel.py` or `buildPDFs_slow.py` will compile the slides and sort the PDFs into their directory. By default, they will compile all versions of all files. This may take a long time: each (non-optional) section is included in beamer, handout, and prep versions of single sections and weeks, as well as the student handout file, so most sections are compiled 7 times.

Both start with options to specify which files should be compiled. Set each variable to True or False in the .py file.

variable	**value**	**behaviour**
`print_beamer`	True	print beamer versions of slides
	False	do not print beamer versions
`print_handout`	True	print handout versions of slides
	False	do not print handout versions
`print_prep`	True	print prep versions of slides
	False	do not print prep versions

variable	value	behaviour
`print_sections`	False	print no single sections
	True	print all single sections from the list
`print_weeks`	False	print no weekly batches
	True	print all weekly batches from the list
`print_readme`, `print_review`, `print_studentworkbook`	False	do not print the corresponding file (ReadMe, Review, or StudentWorkbook)
	True	print the corresponding file
`delete_aux` (in parallel file only)	False	all auxiliary files (.aux, .log, .toc, .nav, etc.) are kept
	True	all files that are generated when the program runs will be deleted (except the resulting .pdf).

Below the compilation choices are lists of sections and weeks. These are lists of exactly which sections and which weeks to compile. If you would like to omit some (but not all) of these files, comment them out of the list.

If you have already set `print_sections=False` or `print_weeks=False`, there is no need to comment out the sections or weeks. None of them will be compiled.

There are no further customization options after the lists of sections and weeks.

BUILDPDFS_SLOW.PY VS. BUILDPDFS_PARALLEL.PY

`buildPDFs_parallel.py` uses parallel processing to speed things up. It also uses latexmk, which requires Perl to be installed. The parallel file has the option to delete auxiliary files. You only need to run `buildPDFs_parallel.py` once.

`buildPDFs_slow.py` compiles files one-at-a-time using pdflatex. It is slower, but might work out-of-the-box on more machines – for example, Windows machines without Perl installed. The repository does not track auxiliary files, so using this option you will initially have to compile twice to generate the navigation bars in the headers and to make the internal references work.

A Note on Repetition

I have observed in my 100-level classes the usefulness of repeated concrete examples. At this stage in their development, many students have a hard time applying theorems in the way mathematicians usually want to state them, but are adept at noticing patterns.

This observation lies in the background of many of the decisions I made in putting together these slides. In this document, you'll see that I often help myself keep track of where I am in a repetitive sequence. It may seem silly or boring, but in my own experience, I have found the repetition to be useful.

Active Learning

Another technique I often employ in introductory classes is to pepper my lectures with opportunities for students to engage on their own (and with their neighbours) with the material.

There will sometimes be several questions visible on the same slide. In class, I encourage students to work on the first one or two questions, and to move on to the others only if they are "super speedy" and finish the target question(s) early. This balances out different student needs. Those who need more time can have it, because those who need less time are still able to stay engaged.

Posting Notes

What exactly to give students access to is a very personal decision. I do not know the best solution. Students who do not have access to lecture notes may be more motivated to synthesize content in their own words. Students who do have access to lecture notes may feel less pressure during class.

My personal practice in 100-level courses is to give students a handout without solutions to write on in class, and post the full slide version with solutions online after class. Providing the handout means that students don't spend time and energy copying down things like definitions that are easily found in the textbook. Withholding solutions during class enables the active learning described on the previous slide. Letting students know that solutions will be posted later encourages students to copy down only what seems important to them in the moment, rather than trying to reproduce every line exactly.

Common Mistakes

When addressing common misconceptions, rather than saying "it's common to make mistake *X*...," I'll give students a multiple-choice question where *X* is an option. I think the students who go out on a limb and select *X* are more likely to remember that it's incorrect when they've been primed by that gentle bit of chagrin. I like to follow these with a second round of questions, which allows students to claim the new knowledge for themselves, rather than sitting with their embarrassment. This is also an opportunity to promote growth mindset.

ATTRIBUTION

Attribution information is generally found at the end of each file, if the file includes work from other sources.

Lecturer Cues

The lower right-hand corner of the slides holds icons that will be useful for the lecturer but that are not necessary for students to pay attention to. These are meant to be relatively unobtrusive. They do not appear in handout mode.

This slide has a box letting you know that the current question has its answer written in. If you click too far, you may give it away.

No answer

This slide has the opposite cue: the answer to the current question does *not* appear on the slides.

More space

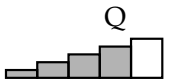
The notebook icon in the corner tells you that there is more room on the next slide. This is used when a slide has a question with little room after it to work.

Usually, the question is sharing the slides with a theorem or a similar example. You can see the problem together with the other content in order to talk about the problem before you write anything down, or to give students a chance to start working on their own. When it's time to work the problem on a tablet, you can advance to the next slide, hiding the extra information to open up space.

Question Set

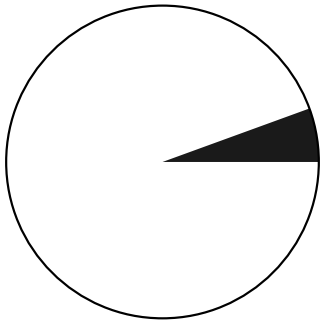
The staircase status bar is used for sets of related questions. The status bar on this slide shows the 4th of 5 questions in a set.

I overload my slides with questions to make sure I don't run out of content. I do not generally present every single problem in a file. When there's a large set of similar problems, I choose the ones I have time for.



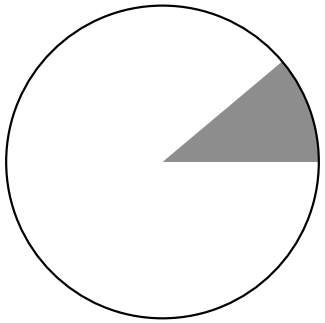
Status Bar

These frames have a flipbook-style animation. The status bar in the corner tells you where you are in that flipbook, so you know when to stop clicking.



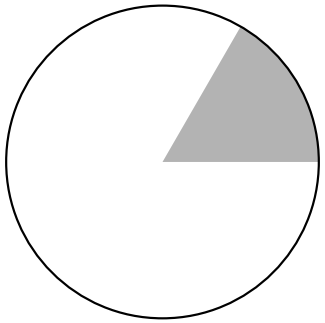
Status Bar

These frames have a flipbook-style animation. The status bar in the corner tells you where you are in that flipbook, so you know when to stop clicking.



Status Bar

These frames have a flipbook-style animation. The status bar in the corner tells you where you are in that flipbook, so you know when to stop clicking.



Now You

Now You  The brain-training icon indicates a question for students to work on alone or in groups. (Since this is information for students as well as instructors, the icon is not stashed in the lower right-hand corner.)

In a first-year class, it may take time to establish a culture where students work earnestly on in-class problems. I like to emphasize that students are building skills. Building a skill requires not just understanding but practice.

Unobtrusive Notes

References to the text are sometimes put next to the page number. Content on the slides is similar, but usualy not identical, to the reference. The references can help you judge which part of the text corresponds to the current slide.

Keen students can investigate the references in the text if they want a longer written explanation, or a similar example to practice on. For that reason, these show up the handouts as well as in beamer mode.

If you just want to use the slides as they are, you can stop here.

The remaining sections describe how to use the custom macros, and are only necessary if you intend to edit the slides.

Who Does a Question

NOW YOU 

The command `\NowYou` places a visual cue that the written exercise is meant to be done by students, rather than the lecturer. There is no other formatting.

The command includes a citation for the brain training icon, which will show up when you run `\LastPage` to include the attribution page.

Printing Answers

You can set the toggle printsolutions to true or false in the preamble.

```
\settoggle{printsolutions}{true}
\settoggle{printsolutions}{false}
```

If you set the toggle to false, the answers will not be printed. The toggle is automatically set to false in handout mode, and to true in beamer mode.

The syntax to use it directly is:

```
\iftoggle{printsolutions}{<A>}{<B>}
```

where <A> will show if the toggle is set to true, and <B> will show otherwise.

You can use `\answer{<A>}` as a shortcut for `\iftoggle{printsolutions}{<A>}{}`. That macro does nothing else.

The printsolutions toggle also affects the commands: `\SetQuestion{}`, `\SetAnswer{}`, `\AnswerYes`, `\AnswerNo`, and `\AnswerSpace`

Sets of Questions

Sometimes there are sets of questions on a particular topic. You can put these inside a `QuestionSet` environment, then give questions inside `\SetQuestion{}` and answers inside `\SetAnswer{}`. The answers will be shown or hidden according to the printsolutions toggle. Since handout mode sets the printsolutions toggle to false, answers are automatically hidden in handouts.

Each question (in both modes) and each answer (in beamer mode) will show up on one frame. Question and answer staircase status bars will be printed accordingly. If the set has more than about 7 questions, these bars might overrun the page.

More Flexible Sets of Questions

The `QuestionSet` environment takes care of a lot of things for you (like making the items show up on different slides in a handout, and counting how many there are total). The trade-off is that it might be too restrictive. It can only be in one frame environment, for example, and each question will be on a new overlay.

A more general way of getting the staircase Q&A status bar is to use `\QuestionBar{<a>}{<b>}` and `\AnswerBar{<a>}{<b>}`. This will put a staircase status bar showing <a> of <b> total. You'll probably want to wrap these in an `\only` overlay specification to control where they show up. Unlike `QuestionSet`, the total size of the status bar is fixed, so you don't have to worry about overrunning margins.

```
\QuestionBar{2}{7}
```

`\AnswerYes` and `\AnswerNo` place answer boxes in the lower right-hand corner. To have them show up on only certain slides, you must use `\only`, rather than `\onslide`.

The answer boxes can affect the alignment of a slide. `\AnswerSpace` avoids jarring text displacement between overlays by making an invisible box of the same size. (It makes `\only` behave more like `\onslide`.)

Lecturer cues in the corner won't be placed correctly if you have `\centering` on the page, so use `\begin{center} \end{center}` instead, and put the commands outside.

The macros in this category are:

- `\MoreSpace`
- `\StatusBar`
- `QuestionSet` environment with `\SetQuestion` and `\SetAnswer`
- `\QuestionBar, \AnswerBar`
- `\AnswerYes, \AnswerNo, \AnswerSpace`

Slide Clicks

The command `\StatusBar{<a>}{<b>}` will place a small status bar in the lower-right hand corner of the frame, starting at slide number <a> and ending at slide number <b>.

Map of Contents

The mindmap tables of contents aren't automatically generated from the sections in a document. Rather, they are saved in the file `header_maps.sty`.

Each chapter has its own map. You can highlight any number of sections in a chapter. Since numbers don't work in TeX as variable names, sections are labeled with letters: 1 is a or A, 2 is b or B, etc. There is one exception: to avoid re-defining `\bf`, in Chapter 2, sections 6 and up are off by one in their labels. So section 2.5 is `\be`, and section 2.6 is `\bg`.

The command `\mapofcontentsA{\ac,\ab}` will show the map of contents for Chapter 1 (that's the "A" in the command itself) with sections 1.3 and 1.2 highlighted (that's the list in the argument). The command `\mapofcontentsC{}` will show the map of contents for Chapter 3 (that's the "C" in the command itself) with no sections highlighted (since the argument is empty).

Referencing the Textbook

`\eref{text}{<label>}` and `\eref{prob}{<label>}` allow you to reference labelled items in the textbook and problem book, respectively. The folder XR has the aux files from these texts. When the texts are updated, copying the new aux files to that folder will make all the references in the slides update when they are re-compiled.

`\unote{<content>}` will put <content> next to the footline page number. I have used this exclusively to reference the textbook and problem book.

ATTRIBUTION

The index is used to cite copyrighted content. The standard format for setting up a citation of an image is:

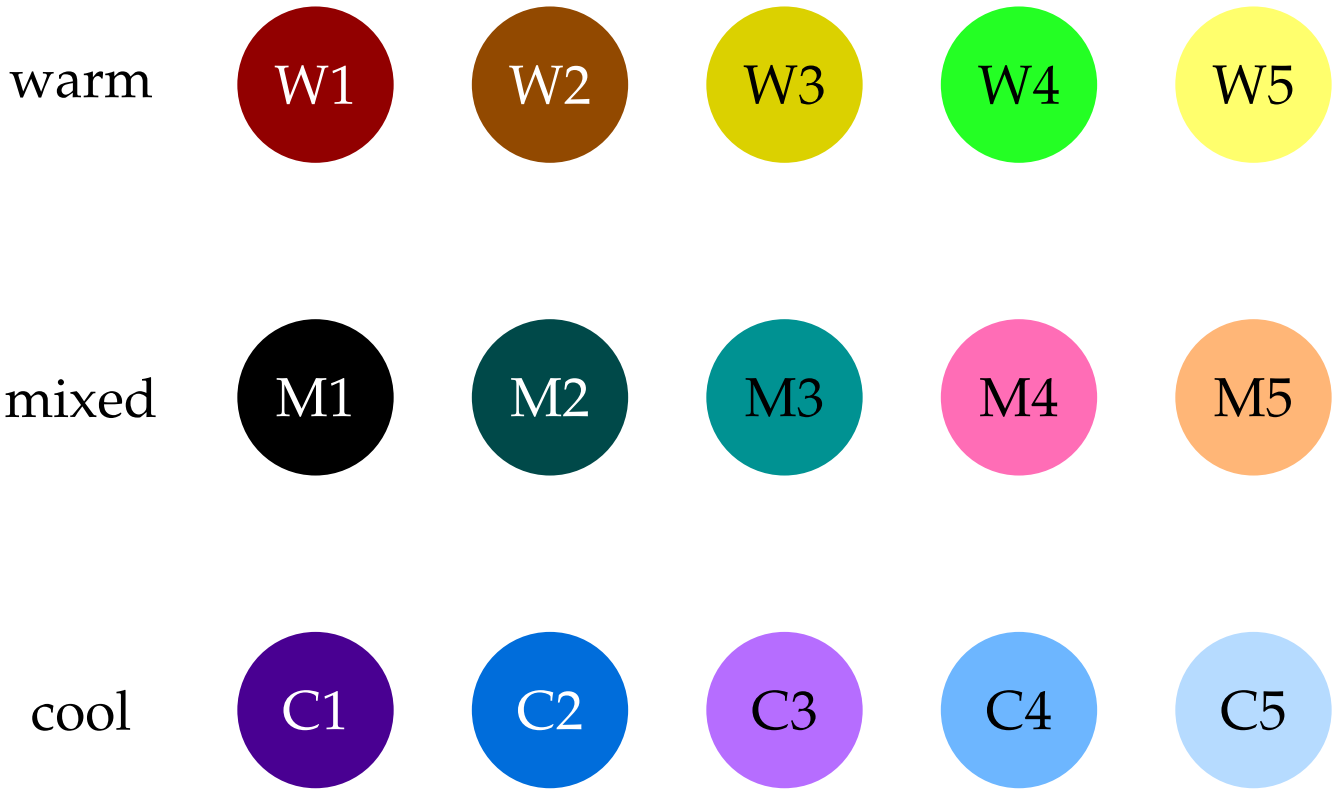
```
\index{\includegraphics[height=5mm]{<file.png>} <picture title (link)> by <artist (link)> is licensed under <license information (link)>}
```

The command `\LastPage` prints an index of copyrighted works included in the file so far. If none have been cited with the `\index` command, it will print nothing.

`\CCBYtwo` , `\CCBYthree` `\CCBYNCNDfour`, and `\CCBYNCSAtwo` display and link to their respective licenses.

Colour Scheme

The following named colours are used in graphics.



Included Work

 ‘Brain’ by Eucalyp is licensed under CC BY 3.0 (accessed 8 June 2021), 28, 32

 ‘Notebook’ by Iconic is licensed under CC BY 3.0 (accessed 9 June 2021), 23