

prerex.sty: Prerequisite Charts in L^AT_EX

Version 6.6

R. D. Tennent
rdt@cs.queensu.ca

October 5, 2017

1 Description

`prerex.sty` defines macros that L^AT_EX processors can use to create charts consisting of labelled course boxes linked by arrows that represent pre- and co-requisite requirements; see the example chart in Figure 1 on page 2. If the source file satisfies the constraints of the `prerex(5)` format, a chart description may be edited using the `prerex(1)` interactive editor (as well as any conventional text editor).

2 Terminology

A prerequisite chart consists of a number of *course boxes*, linked by *arrows*. Courses are either *half* or *full*, and may be *required* or *optional* (or neither). Each course box can contain a *course code* (upper left corner), a *course title* (lower half), and *timetable* information (upper right corner). An arrow can be either a *prerequisite* (solid), a *corequisite* (dotted), or *recommended* (dashed). When a conventional arrow would be inappropriate, it is possible to use a *mini* course “box” (consisting of just a course code) just above a target box. Lines of text may be placed anywhere on the chart.

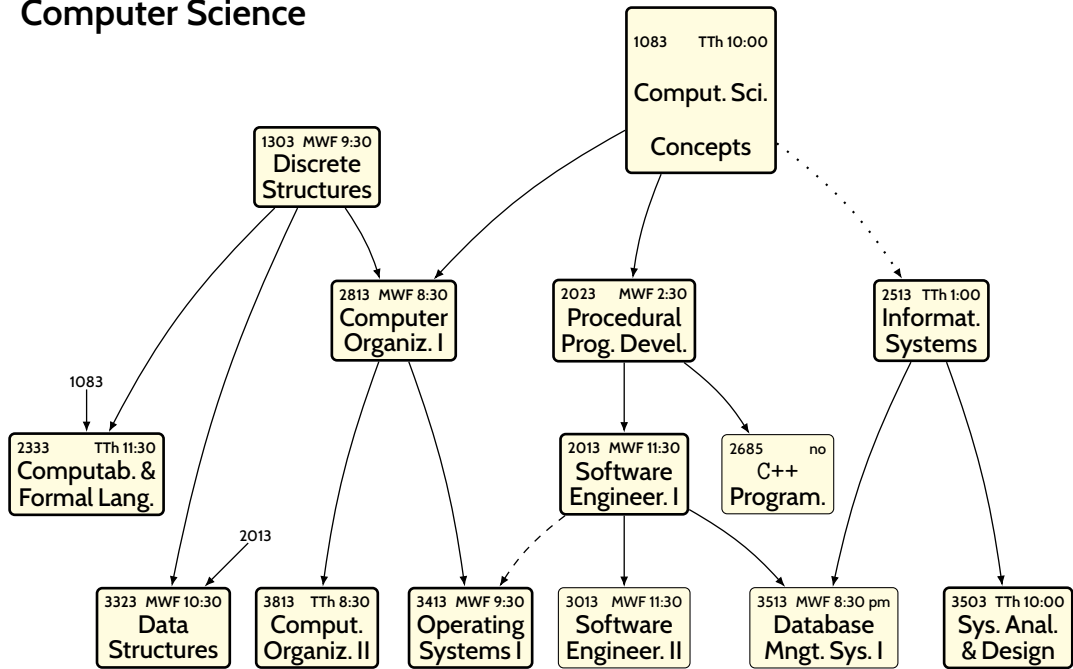
3 Coordinate System

A conventional two-dimensional Cartesian coordinate system is used to specify the locations of diagram elements. The origin (where $x = 0$ and $y = 0$) is at the lower-left corner of the diagram. The diagram height is configurable; the width is determined by the values of `\textwidth` and an adjustable `\unit` length, which defines the absolute value of a coordinate unit. It is possible to put a coordinate grid with labelled axes in the background of a chart. The coordinates of a *box*, *mini*, or *text* are those of its centre point. An *arrow* is described by the coordinates of the centre points of its source and target boxes/minis/texts.

4 Usage

The command `\usepackage{prerex}` should appear in the document preamble. This will define a L^AT_EX environment

Computer Science



Notes

- A solid arrow \longrightarrow indicates a required prerequisite, a dotted arrow $\cdots \rightarrow$ indicates a corequisite (to be taken before or concurrently), and a dashed arrow $--- \rightarrow$ indicates a recommended prerequisite. Core courses are in **bold** boxes; other courses (i.e., options or prerequisites) are in **light** boxes.
- Timetable abbreviations: M, T, W, Th, F=Mon, Tue, Wed, Thur, Fri, resp.; no=not offered.
- ...

Figure 1: A Prerequisite Chart

```
\begin{chart} ... \end{chart}
```

within which the commands described in the following sub-sections may be used to produce course boxes, minis, texts, and arrows. The order of commands is not significant except that the commands for the source and target boxes of an arrow should precede the command for the arrow, and the `\grid` command, if present, should precede other chart commands.

4.1 Course Boxes

The commands producing course boxes have the following forms:

```
\halfcourse x,y :{ code }{ title }{ timetable }  
\reqhalfcourse x,y :{ code }{ title }{ timetable }  
\opthalfcourse x,y :{ code }{ title }{ timetable }  
\fullcourse x,y :{ code }{ title }{ timetable }  
\reqfullcourse x,y :{ code }{ title }{ timetable }  
\optfullcourse x,y :{ code }{ title }{ timetable }
```

where x,y are the coordinates of the centerpoint of the box, relative to the origin of the coordinate system at the lower-left corner of the diagram. The *code*, *title*, and *timetable* arguments are arbitrary (well-bracketed) text, possibly with L^AT_EX markup.

The following are similar but take an additional argument to specify the (non-default) background color of the course box:

```
\halfcoursec x,y :{ code }{ title }{ timetable } { color }  
\reqhalfcoursec x,y :{ code }{ title }{ timetable } { color }  
\opthalfcoursec x,y :{ code }{ title }{ timetable } { color }  
\fullcoursec x,y :{ code }{ title }{ timetable } { color }  
\reqfullcoursec x,y :{ code }{ title }{ timetable } { color }  
\optfullcoursec x,y :{ code }{ title }{ timetable } { color }
```

4.2 Mini Course Boxes and Texts

A mini course “box” is produced by a command of the form

```
\mini x,y :{ code }
```

A paragraph of text is produced by a command of the form

```
\text x,y :{ text }
```

There are several differences between minis and texts. The maximum length allowed for a course-code may be less than that for texts. A sans-serif font and a reduced font size is used for a course-code. A course-code may be hyperlinked but a text is not (except when the coordinate grid is on).

4.3 Arrows

Arrows between course boxes (or minis or texts) are produced by commands of the form

```
\prereq  $x_0, y_0, x_1, y_1$ :
```

```
\coreq  $x_0, y_0, x_1, y_1$ :
```

```
\recomm  $x_0, y_0, x_1, y_1$ :
```

These produce, respectively, solid, dotted, and dashed arrows from the course box (or mini or text) centered at coordinates x_0, y_0 to the entity centered at coordinates x_1, y_1 .

Arrows by default have a non-zero curvature (unless they are very short). It is possible to override the default curvature for a particular arrow by using the instructions

```
\prereqc  $x_0, y_0, x_1, y_1; c$ :
```

```
\coreqc  $x_0, y_0, x_1, y_1; c$ :
```

```
\recommc  $x_0, y_0, x_1, y_1; c$ :
```

where c is an integer in the range 0–100 specifying the desired curvature; for example, $c = 0$ will produce a straight arrow. To change the default curvature for all arrows, the user program may redefine the `\DefaultCurvature` command.

4.4 Web Links

Course boxes and minis may be hyperlinked. The default URL for course boxes is `course-code.html` where `course-code` is the `code` argument of the course box. The URL scheme may be changed by the user by re-defining the `\CourseURL` command; see Section 7.

4.5 Coordinate Grid

The following command produces a coordinate grid with labelled axes, which is useful while editing a diagram:

```
\grid
```

This is normally placed before all the other chart commands so that grid lines are in the background, covered by course boxes, minis, texts, and arrows, as in Figure 2 on page 5.

The `\grid` command also redefines the `CourseURL` command so that, in suitable PDF viewers, if the mouse cursor is placed over a box, mini, or midpoint of an arrow, the *coordinates* of the box/mini/text (or the coordinates of the source *and* target of the arrow) will be displayed either in the status bar or in a “tooltip.” This coordinate display make it easier to edit chart descriptions. When the grid is on, the mid-point of every arrow is highlighted by a “bullet”.

The `prerex` package at <http://mirror.ctan.org/graphics/prerex/> contains source code for a graphical frontend `vprerex` to the `prerex` editor. It incorporates a minimalistic `prerex`-enabled PDF viewer which allows coordinates of course boxes, arrows, and background points to be captured in the X selection clipboard (for pasting into a command being composed at the `prerex` prompt).

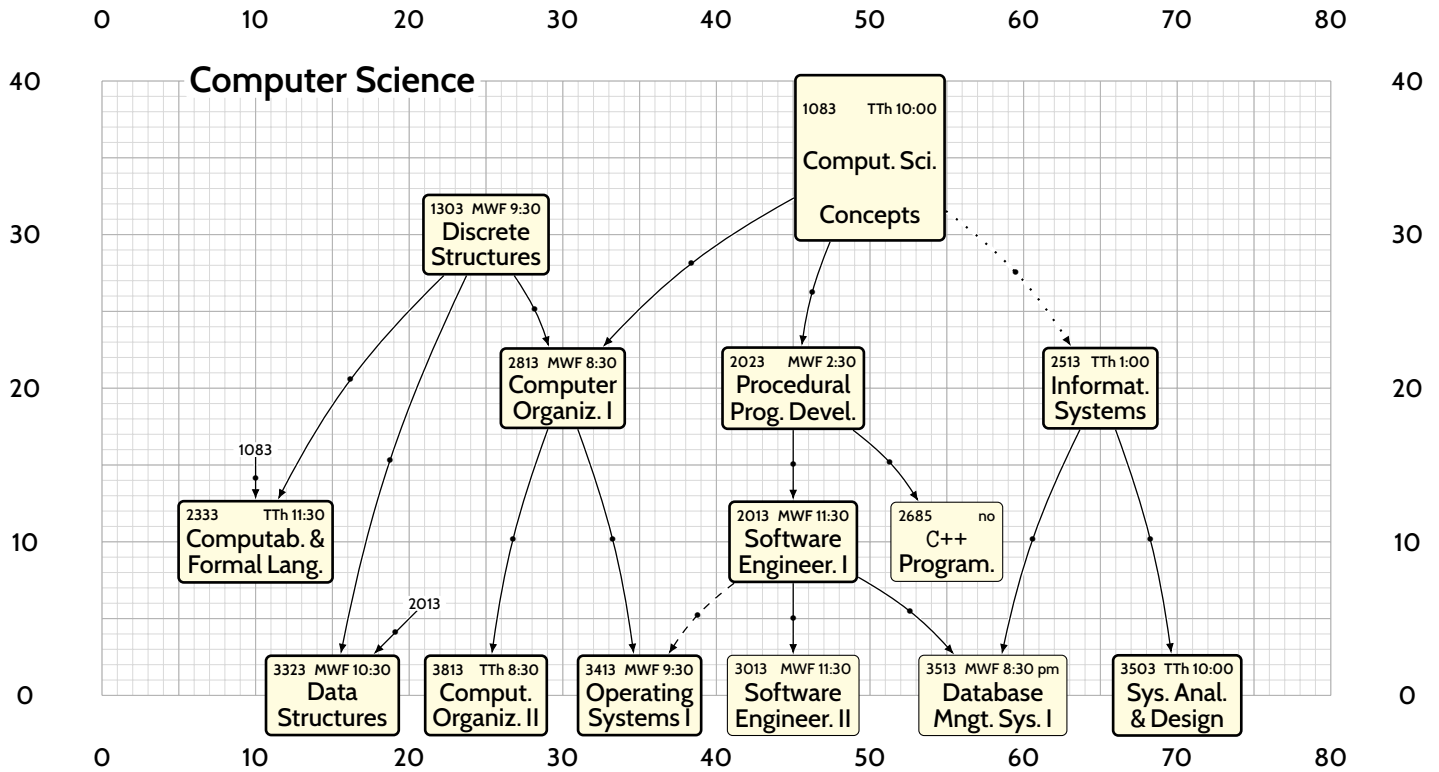


Figure 2: A Prerequisite Chart with a Coordinate Grid

4.6 Stand-Alone Commands

The following commands may be used outside a chart environment, such as in the explanatory notes for a chart:

<code>\solidarrow</code>	\longrightarrow
<code>\dottedarrow</code>	$\cdots\rightarrow$
<code>\dashedarrow</code>	\dashrightarrow
<code>\lightbox</code>	light
<code>\boldbox</code>	bold
<code>\dottedbox</code>	dotted

5 Dependencies

This implementation of `prerex.sty` uses `pgf` (version 3.0.0 or later) with add-on package `tikz`. The following additional packages are used: `multido`, `relsize`, `calc`, `textcomp`, `xcolor`, and `hyperref`. *Either* `pdflatex` or `xelatex` or `latex`, followed by `dvips` and, possibly, `ps2pdf` or `latex` (with suitable driver options), followed by either `dvipdfmx` or `dvisvgm` may be used.

6 Installation

1. install the most recent version of pgf from <http://sourceforge.net/projects/pgf/>.
2. copy `prerex.sty` to *your-texmf-tree*/tex/latex/misc
3. copy `doc/prerex.{tex,pdf}` to *your-texmf-tree*/doc/latex/misc
4. if necessary, refresh the T_EX database (e.g., `texhash`)
5. on a UNIX-like system, copy `doc/prerex.5` and `doc/prerex.sty.7` to sub-directories of a `man` directory on your `MANPATH`

7 Configuration

The following are defined by the style file but may be re-defined by the user:

```
\newcommand{\DefaultCurvature}{20}
```

where the argument should be in the range 0–100; 0 means no curvature.

```
\newcommand{\CourseURL}[3]{./#3.html}
```

where the arguments supplied at the call are the x and y coordinates of the box or mini and the course code.

```
\newcommand{\background}{yellow!15}
```

which specifies the default background color of course boxes.

```
\setlength{\unit}{5.7816pt}
```

which defines the absolute size of a coordinate unit; this yields 10 pixels per coordinate unit if the chart is converted to an image at 125 dpi.

```
\newcommand{\dpi}{125}
```

```
\newcommand{\PixelsPerUnit}{10}
```

```
\setcounter{diagheight}{75}
```

which is the y -coordinate of the top of the diagram (reduced to 65 in landscape mode); the nominal diagram width is `\textwidth ÷ \unit + 10`.

The following define the thicknesses of the three kinds of arrows, and the thickness of bold boxes:

```
\newcommand{\solidwidth}{0.5pt}
```

```
\newcommand{\dottedwidth}{0.8pt}
```

```
\newcommand{\dashedwidth}{0.5pt}
```

```
\newcommand{\boldwidth}{1.0pt}
```

The following command is used to change to a smaller size for course codes, timetable information, and mini-course codes:

```
\newcommand{\smallsize}{\relsize{-3}}
```

The following command defines the baseline adjustment for the stand-alone arrows and boxes; this parameter is font-dependent.

```
\newcommand{\baselineAdj}{-0.5ex}
```

8 Conversion to Scalable Vector Graphic Format

Scalable Vector Graphics (SVG) is an emerging standard for high-quality graphics. It is supported by all modern browsers natively, making it ideal for posting charts on the web. An SVG image is comparable in quality to a chart in PDF and, when compressed, is comparable in size. The `dvisvgm`¹ program may be used to produce a (compressed) SVG image directly from latex output, provided suitable class options are used.

Another approach is to use `pdf2svg`², a small program that uses poppler and cairo libraries to convert a PDF page to a scalable vector-graphic image.

Unfortunately, many web *servers* are not properly configured for `svgz` (or even `svg`) content. If you encounter this problem, try renaming the file to have extension `.svg.gz`; otherwise, check out the information appropriate to your server software below:

- [apache](#)³
- [Microsoft Internet Information Services \(IIS\)](#)⁴
- [nginx](#)⁵
- [thttpd](#)⁶
- [lighttpd](#)⁷
- [cherokee](#)⁸
- [hiawatha](#)⁹

¹<http://dvisvgm.sourceforge.net/>; also available from CTAN and in TeXLive.

²<http://www.cityinthesky.co.uk/opensource/pdf2svg/>

³<https://htaccess.wordpress.com/2008/07/16/sending-correct-content-type-headers-with-htaccess/>

⁴<http://forums.iis.net/t/1175276.aspx>

⁵<http://trac.nginx.org/nginx/ticket/372>

⁶<https://lists.w3.org/Archives/Public/www-svg/2014Mar/0092.html>

⁷<http://redmine.lighttpd.net/issues/2579>

⁸<https://github.com/cherokee/webserver/issues/101>

⁹<https://www.hiawatha-webserver.org/forum/topic/1628>

9 Implementation Details

9.1 Boxes

To improve the appearance, boxes are drawn with slightly rounded corners. The half-course boxes are assigned a minimum height to give a more uniform appearance to horizontal rows of such boxes. The mini and text boxes have white frames and their fill colour is white, which is assumed to be the background colour of the chart. A text box is not normally hyperlinked; however, if the coordinate grid is on, it is hyperlinked and its chart coordinates are encoded in the URI.

9.2 Arrows

Arrows with a small height and arrows whose source and target points have equal x or y coordinates are always drawn straight (using a specialized and simpler macro) unless an explicit non-zero curvature argument is provided.

If the coordinate grid is on, a hyperlink with the source and target coordinates is placed at the mid-point of the arrow.

A wider background white edge is drawn below an arrow to improve the appearance of crossing arrows; the background also ensures visibility of the arrow when the coordinate grid is on.

9.3 Coordinate Grid and Associated Changes

When the coordinate grid is drawn, the `CourseURL` command is re-defined to pass node coordinates in URIs. Also, two “anchor” hyperlinks are inserted at the southwest and northeast corners of the diagram to allow prerex-enabled PDF viewers to compute chart coordinates.

10 License

`prerex.sty`, `prerex.sty.7`, `README.prerex` and this document may be distributed and/or modified under the conditions of the L^AT_EX Project Public License.