

Prerex-Formatted Prerequisite Charts

An Introduction

prerex Version 5.3

Bob Tennent

School of Computing, Queen's University

rdt@cs.queensu.ca

Contents

1	Introduction	2
2	Frequently Asked Questions	2
3	Viewing and Printing a Prerequisite Chart	7
4	Installing a Basic LaTeX System	8
5	Additional LaTeX Software	9
6	Installing a LaTeX Development Environment	9
7	Fonts	10
8	Conversion to Portable Network Graphic (PNG) Format	11
9	Terminology and Coordinate System	12
10	The prerex Interactive Chart Editor	12
11	Source Files	13
11.1	Overall Structure	13
11.2	Chart Instructions	14
11.3	LaTeX Input Conventions	14
11.4	Configuration	17
	Bibliography	18

1 Introduction

This document is intended to help a user who might want to create or edit prerex-formatted prerequisite charts, but is finding the installation and usage instructions in the README and manual pages to be incomprehensible.

2 Frequently Asked Questions

What is a prerequisite chart?

A prerequisite chart gives an attractive graphical presentation of courses in a program (or set of related programs), organized by terms or years, linked by pre- and co-requisite arrows, and, when possible, supplemented by timetable information; Figure 1 is a small example.

Who would use a prerequisite chart?

Prerequisite charts are useful to students selecting courses and planning their programs, and to faculty and staff advising students and managing curriculum development.

The information is in calendars and on-line; why is a prerequisite chart needed?

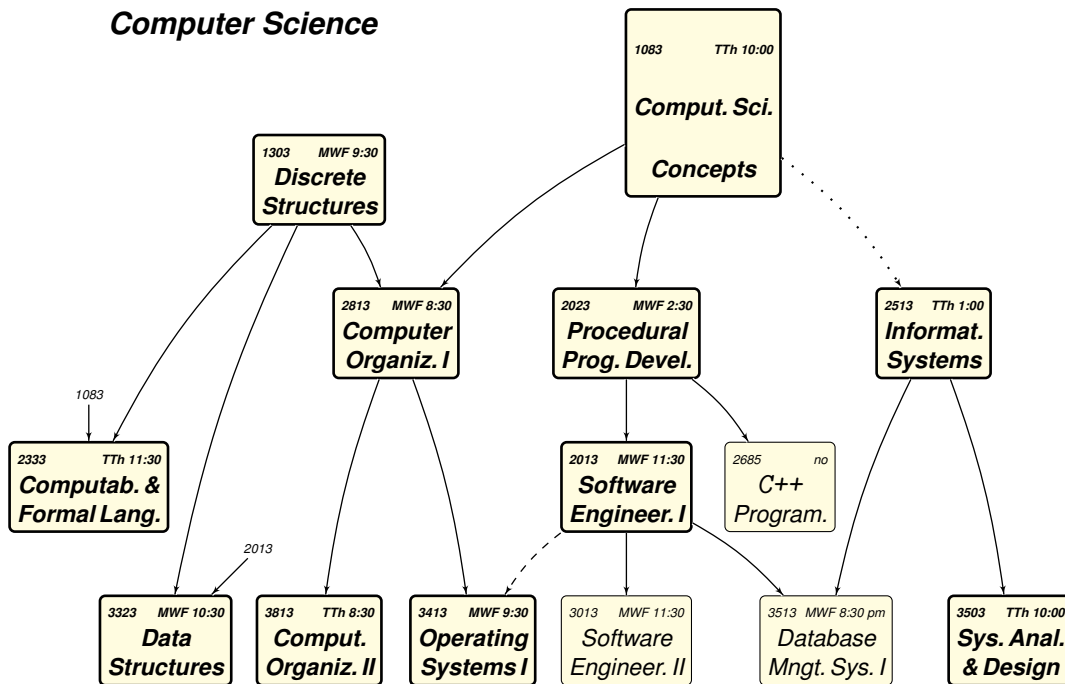
Conventional sources of course, program and timetable information are rarely in a user-friendly format. Faculty and staff advisors, being already fairly familiar with their courses and programs, often do not appreciate how inconvenient it is for students to access all the information they need to make course choices.

For example, program requirements are typically specified as lists of course codes for required and optional courses; one must look elsewhere to find the titles of the courses, the relations between them, their prerequisites, and the timetabling information.

As another example, a course description in a calendar typically specifies what the prerequisite courses are, but hardly ever specifies which course or courses the course is prerequisite to. This information may be very important to a student, but to discover it, he or she must search through all the subsequent course descriptions in the calendar. With a prerequisite chart, it is evident at a glance.

As a third example, consider a student who wants to know whether they can defer a course till the following year without risk of timetable conflicts. With conventional sources of information, such a question can be answered only by doing an exhaustive search through both timetable and calendar for a possible conflict. A prerequisite chart incorporating timetable data provides the necessary information in a compact and usable form, and the question can be answered immediately.

Computer Science



- 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.
- Timetabling abbreviations: M, T, W, Th, F=Mon, Tue, Wed, Thur, Fri, resp.; eve=7:00–9:50 pm; no=not offered.

Figure 1: A Prereq-Formatted Prerequisite Chart

If prerequisite charts are so great, why doesn't every university department have them?

Few academic departments have the resources to produce prerequisite charts. If one searches on the web for "prerequisite chart" or "prerequisite diagram," virtually all of the matches are from departments that one would expect to have expertise in technical drawings. Here are some of these:

- Queen's University, Electrical and Computer Engineering¹
- University of Minnesota, Computer Science²
- University of Auckland Business School, Department of Information Systems³
- University of Waterloo, Computer Science⁴
- Georgia Tech, Electrical Engineering⁵
- Georgia State University, Computer Science⁶
- Langara College, Computer Science and Information Systems⁷
- Iowa State University, Civil, Construction, & Environmental Engineering⁸
- Kansas State University, Mathematics⁹
- Carleton University, Engineering Physics¹⁰
- Mississippi State, Civil Engineering¹¹

¹http://www.ece.queensu.ca/undergraduate/coursesummaries/Queens_ECE_prerequisite_graph_2007.pdf

²<http://www.cs.umn.edu/academics/courses/prereq.php>

³<http://www.code.auckland.ac.nz/comwebcontent/isom/diagrams/ispre-req.html>

⁴<http://www.cs.uwaterloo.ca/current/courses/charts/majorPrereq.shtml>

⁵http://www.ece.gatech.edu/academics/undergrad/ee_prereq.html

⁶<http://www.cs.gsu.edu/degrees/undergrad/prereq.pdf>

⁷<http://www.langara.bc.ca/science-technology/computing-science-systems/course-prerequisites.html>

⁸<http://www.ccee.iastate.edu/fileadmin/www.ccee.iastate.edu/academics/mechanical-prereq-0507.pdf>

⁹http://www.math.ksu.edu/main/course_info/courses/prereqcht.htm

¹⁰<http://www.eng.carleton.ca/academicssupport/prereq/EngineeringPhysics.pdf>

¹¹<http://www.cee.msstate.edu/documents/flow%20chart.pdf>

Where can I see prerex-formatted charts in use?

There are several charts for both specialized and multi-disciplinary programs on-line here: <http://www.cs.queensu.ca/undergraduate/prereq.html>.

What file formats are used for prerex-formatted charts?

The charts themselves are best distributed in the Portable Document Format (PDF) defined by Adobe; PDFs are readable on any platform and in most browsers using freely-available software (such as the Adobe Reader), printable on any high-resolution printer, relatively compact, and scalable to accommodate paper size, display resolution, or user preference.

Charts can also be converted to Portable Network Graphics (PNG) format for in-line use on a web page. PNG format is also relatively compact and can be rendered directly by most web browsers.

Prerex-formatted charts are produced by software that processes simple and convenient text-based descriptions. These descriptions can be edited in any text editor or in the prerex interactive editor discussed in Section 10. The format of chart descriptions is documented in Section 11.

What software is needed to produce prerex-formatted prerequisite charts?

To process the description of a prerex-formatted chart, the following software is essential:

- an up-to-date standard-conforming implementation of LaTeX, a powerful document processing system based on TeX, a professional-quality typesetting engine; see Section 4.
- a few additional LaTeX packages, such as pgf, that may not be included in a minimal LaTeX installation; see Section 5;
- a file prerex.sty of specialized LaTeX definitions, which is included in the prerex package available at any repository of the Comprehensive TeX Archive Network (CTAN)¹².

The following additional software is optional but recommended:

- a PDF viewer that
 - can be configured to “watch” the PDF file so that it re-loads automatically when this is updated, and
 - shows hyperlinks in a “tooltip” when the mouse hovers over it, which is a convenient way to get the coordinates of course boxes and arrows while editing the description;

see Section 3.

¹²<http://www.ctan.org/>

- suitable fonts; see Section 7.
- the prerex interactive editor for prerex-formatted charts; see Section 10.
- an integrated LaTeX development environment; see Section 6.

Why couldn't I use a program like Visio[®] or CorelDRAW[®] to produce a prerequisite chart?

You could, if you have such a program available, are very skilled in its use, and won't be embarrassed distributing a chart that looks amateurish. Look carefully at a prerex-formatted chart and notice the following properties:

- Each course box is sized to just enclose the text within it, with uniform standard margins.
- Each arrow between courses is oriented from box centre to box centre, rather than from/to standard "connection points" on the box edges.
- The arrows are "clipped" by the course boxes, but the arrow heads abut the target box exactly.

These desirable properties are not easily achieved using conventional "do-it-yourself" drawing software, no matter how "user-friendly" it purports to be. The prerex system provides complete flexibility as well as professional quality:

- Text within a course box may be partitioned into regions with varying characteristics. For example, the course code and the timetable information on the first line of course boxes are in a smaller font than the course name. The latter is centered and the former are left- and right-justified, respectively. Arbitrary LaTeX formatting can be used for the text.
- Any available Type 1 (Postscript) or TrueType format fonts may be used; in the chart in Figure 1, the typeface used is Optima, an elegant *sans serif* face designed by Hermann Zapf. The professional-quality typesetting engine takes advantage of kerns and ligatures in the fonts.
- Line thickness for boxes may be varied; in the example diagram, heavier boxes (and bold-face text) are used to indicate that a course is "required" in the program, rather than an option.
- Different styles of connectors can be used, for example to distinguish prerequisites, co-requisites, and recommended prerequisites.
- Various sizes or shapes of course boxes may be used, for example to distinguish between half and full courses.
- Graphic images such as logos can be imported.
- Colours and hyperlinks to on-line course descriptions or calendars are possible.

Wouldn't a commercial drafting service be able to produce a professional-looking prerequisite chart?

Possibly, but you will almost certainly have to do the global lay-out yourself. A more serious long-term problem is that it is unlikely you will be able to maintain the chart yourself using freely available software. You will be "locked-in" to use the drafting service for every future modification of the chart.

If you want the quality and maintainability of a prerex-formatted chart but don't think you can produce it yourself, you can outsource the production; check out the following: <http://www.prerex.com>.

How can a prerex-formatted chart be edited?

University programs and timetables change frequently: courses are added or dropped, pre- or co-requisites or timetabling of courses change, and so on. Use of a separate text-based description file and intelligent processing software make it easy to modify a prerex-formatted chart. If the text within a course box is changed, the size of the box, the length of connectors to and from that box, and the location of arrowheads into the box are all adjusted automatically by the processing software. If a course box must be moved, it is straightforward to edit the relevant instructions for arrows into or from the box because the coordinates specified for an arrow are those of the box centres, rather than those of the edges or corners of the boxes. Small changes can be made with any text editor.

For creating a prerex-formatted chart or for making substantial changes to an existing chart, the prerex interactive chart editor is available; see Section 10.

Editing a text file in either of these ways is far easier and faster than trying to fiddle with an image using a mouse.

3 Viewing and Printing a Prerequisite Chart

A prerequisite chart in Portable Document Format (PDF) can be "opened" for viewing or printing in any PDF viewer. The most popular of these is the Adobe Reader.¹³ However, the Adobe Reader is quite slow to load and somewhat inconvenient to use if one wants to re-load the document after changing it (one must first close the document and then re-open it).

There are faster and more convenient alternatives to the Adobe Reader. One that is available for Windows, Macs, and Linux systems is a ghostscript-based viewer called GSview.¹⁴ For Windows or Linux: download and install the most recent version of ghostscript for your platform,¹⁵ and then install the most recent version of GSview for your platform.¹⁶ For Macs: just get MacGSView.¹⁷

¹³<http://www.adobe.com/prodindex/acrobat/readstep2.html>

¹⁴<http://www.cs.wisc.edu/~ghost/>

¹⁵<http://www.cs.wisc.edu/~ghost/doc/AFPL/index.htm>

¹⁶<http://www.cs.wisc.edu/~ghost/gsview/index.htm>

¹⁷<http://www.cs.wisc.edu/~ghost/macOS/index.htm>

For Linux systems, other ghostscript-based viewers such as `gv`¹⁸ are available. Also, `xpdf`¹⁹ is suitable for any X-window based systems, such as any version of Linux. If Gnome or KDE libraries are installed, variants `gpdf`, `kghostview`, `evince`, and `kpdf` (now renamed `okular`) of these programs are available. You may print a prerequisite chart from within any of these viewing applications.

If you are *editing* a prerequisite chart (using either a text editor on the chart description or the prerex editor), you will want to use a PDF viewer that not only can be configured to “watch” the PDF file (so that it re-loads automatically when this is updated), but also shows hyperlinks in a “tooltip” when the mouse hovers over it; this is a convenient way to get the coordinates of course boxes and arrows while editing the description. Recent versions of some of these viewing programs have the latter capability.

`kpdf` (a.k.a. `okular`) is an open-source PDF viewer for KDE. The prerex distribution (from version 3.2) contains a “patch” file that can be used to modify the source files for `kpdf/okular` so it can display URI tooltips and capture box, connector, or background-point coordinates while the coordinate grid is enabled. See the README for instructions on obtaining the sources, patching them, and building and installing a prerex-enabled version of `kpdf/okular`.

4 Installing a Basic LaTeX System

To “process” a chart description, you will need to install an application called LaTeX, together with several auxiliary applications to convert the output into Postscript and then PDF.

LaTeX is a powerful document processor based on a professional-quality typesetting program called TeX. The following web sites will point you to as much information as you might want on LaTeX and TeX:

- LaTeX Project home page²⁰
- TeX Users Group home page²¹
- TeX FAQ²²

If you use Windows, we suggest you install the “basic” version of MiKTeX.²³ If you use Mac OS X, go to the *TeX on Mac OS X* site.²⁴ Unix and Linux users should install `tetex` packages, or get the sources from the *teTeX Homepage*.²⁵

The resulting installation will allow you to process simple `.tex` files. However, you won’t be able to process the description of a prerex-formatted prerequisite chart until some necessary additional software is installed.

¹⁸<http://wino.physik.uni-mainz.de/~plass/gv/>

¹⁹<http://www.foolabs.com/xpdf/home.html>

²⁰<http://www.latex-project.org>

²¹<http://www.tug.org/>

²²<http://www.tex.ac.uk/cgi-bin/texfaq2html>

²³<http://www.miktex.org/setup.html>

²⁴<http://www.rna.nl/tex.html>

²⁵<http://www.tug.org/teTeX/>

5 Additional LaTeX Software

To process descriptions of prerex-formatted charts, some additional packages are needed. The file `prerex.sty` is essential; it should be stored in the same folder as the `.tex` file to be processed (or in the local `texmf` tree) so the LaTeX system can find it. Other add-ons that may be necessary are as follows: `pgf`, `tikz`, `multido`, `textcomp`, and `xcolor`. If you are using MikTeX, these can be installed using the package manager, or, if you prefer, will be installed automatically the first time a chart description is processed. If you are not using MikTeX, all of the packages are available for download from any CTAN²⁶ repository.

You are now ready to process the description for a chart. Get a command-line prompt, move to the folder with the chart description and issue the following instruction:

```
pdflatex chart
```

This operation will overwrite any existing `chart.pdf`, so you should first re-name that file if you want to preserve it.

If the `pdflatex` command isn't recognized, you will have to edit the system executable-command path. On a Unix-like system, adjust the value of `PATH`; on Windows, click on *Control Panel* → *System* → *Advanced* → *Environment Variables*, select *Path* and click on *Edit*; then append the appropriate paths, separated by semi-colons. For MikTeX, the default path for commands is `C:\texmf\miktex\bin`.

6 Installing a LaTeX Development Environment

Most beginning users will prefer to use a graphical “development environment” which supports editing, processing and previewing of documents, rather than the command-line interface. There are *many* such applications.

For Windows, we suggest you go the Texmaker download site²⁷ and click on the link for the self-installing archive `texmakerwin32_install.exe`. For Mac OS X, go to the same site, get the `texmakerbinmacosx` archive and follow the instructions. For Linux, install a texmaker package or get the source files at the Texmaker site.

When Texmaker has been installed, start it up and then click on *Options* → *Configure Texmaker* to do any necessary configuration, such as the *Quick Build* operation.

You should now be able to open the source file for your chart, process it, and then view the result using the items in the *Tools* menu, or the *Quick Build* button. If there are any processing errors, error messages will appear in the bottom window pane.

Many other LaTeX editors are available if Texmaker doesn't satisfy you. Two of the most popular of these for Windows are WinEdt²⁸ and TeXnicCenter;²⁹ the former is “shareware” and the latter is free. Mac users may want to check out the list of “front-

²⁶<http://www.ctan.org/>

²⁷<http://www.xmlmath.net/texmaker/download.html>

²⁸<http://www.winedt.com/>

²⁹<http://sourceforge.net/projects/texniccenter>

ends” at the *TeX on Mac OS X* site.³⁰ Linux users also have Kile³¹ and Klat.³²

7 Fonts

Look carefully at the samples in Figure 2. Times and Palatino are *serifed* typefaces; the



Times
Palatino
Computer Modern Sans
Helvetica
Avant Garde
Futura
Gill Sans
Frutiger
Optima

Figure 2: Typeface Samples

serifs are the small projections at the baselines, and sometimes at the tops, of characters. Serifs improve the readability of *long* texts. The others are *sans serif* typefaces; this class of typefaces is more appropriate for use in prerequisite charts where *legibility* of fairly small and short textual fragments is of primary importance.

To specify that the default text font in a LaTeX document is to be *sans serif*, the following command is used in the preamble of the source file:

```
\renewcommand{\rmdefault}{\sfdefault}
```

Computer Modern Sans is the default *sans serif* font in LaTeX, and Helvetica and Avant Garde should also be available in every LaTeX installation. But we recommend that you consider Optima. If you look carefully at the samples above, you’ll see that, unlike the boringly uniform strokes of the Computer Modern, Helvetica, Avant Garde, Futura, and Gill Sans characters, the strokes of the Optima characters are *tapered* (i.e., shaped). This gives Optima text an elegance and liveliness lacking in most other *sans serif* typefaces. Palatino (used in this document) and Optima were both designed by Hermann Zapf, considered the foremost type designer of the modern era.

So how does one install Optima (or any other non-standard typeface) for use with LaTeX? Authentic Optima fonts in Windows Postscript (Adobe Type 1) format are available commercially from font vendors such as myfonts.com. The necessary LaTeX support files and instructions for installation into a standard LaTeX implementation may be found at CTAN³³; the file `pop.zip` is the package of support files and `pop.txt` gives the installation instructions.

³⁰<http://www.rna.nl/tex.html>

³¹<http://kile.sourceforge.net/>

³²<http://research.edm.luc.ac.be/jori/klat/klat.html>

³³<http://www.ctan.org/tex-archive/fonts/psfonts/w-a-schmidt/>

But there are less expensive alternatives. The name “Optima” is trademarked by Linotype, but other font foundries have produced good-quality clones with other names. For example, the family of Zapf-Humanist 601 fonts from Bitstream are available from commercial distributors such as `myfonts.com`. They are actually more expensive than authentic Optima; however, some of these fonts have been bundled with Corel applications such as WordPerfect, and so you may already have them. The support files for LaTeX (`bop.zip`) and instructions (`bop.txt`) are also available at the site referenced above.

Another usable clone of Optima is the URW Classico family of fonts. Although they are only slightly less expensive than authentic Optima at commercial vendors, four Classico fonts are available in a package of 79 fonts in Type 1 format that may be legally downloaded for non-commercial use from Artifex³⁴. The four Classico font files, renamed for use in LaTeX, with the necessary support files and installation instructions are available from CTAN³⁵. Note that the font files cannot be distributed commercially without a commercial license; please read the license in the archive before re-distributing them.

When any of these Optima or Optima-like fonts and corresponding support files have been installed, it will be possible to request their use in your charts by using `pop`, `bop`, or `uop` in a

```
\renewcommand{\sfdefault}{...}
```

command in the preamble of the source file. To get Helvetica (scaled down 20%) or Avant Garde, use

```
\usepackage[scaled=0.8]{helvet}
```

or

```
\renewcommand{\sfdefault}{pag}
```

respectively. If none of these is used, (or the command is commented out by preceding it with a `%` character), Computer Modern Sans will be used.

8 Conversion to Portable Network Graphic (PNG) Format

Although PDF is currently the best available format for distributing and printing charts, you may want to produce a raster-image version for use on a web site. There are several conversion programs and services available to do this. For example, the `convert` tool of the multi-platform ImageMagick³⁶ package may be used as follows:

```
convert -density 125 -quality 100 chart.pdf chart.png
```

The density parameter specifies the number of pixels per inch; if this is too small, the image is of poor quality, and if it is too large, the image file is too big. The quality parameter ensures maximal (loss-free) compression. From version 5.0, `prerex.sty` generates an auxiliary file containing image-map data for the hyperlinked nodes in the chart; consult a reference on the HTML language for an explanation of (client-side) image maps.

³⁴<http://www.artifex.com/downloads/>

³⁵<ftp://tug.ctan.org/pub/tex-archive/nonfree/fonts/urw/classico.zip>

³⁶<http://www.imagemagick.org/>

9 Terminology and Coordinate System

Before we discuss the prerex editor and the format of “source files” (i.e., chart descriptions), we shall summarize the necessary terminology and coordinate conventions.

A prerequisite chart consists of a number of *course boxes*, linked by *arrows*. Courses are either *half* or *full*, and may be *required*. 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 *co-requisite* (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. A line of text may be placed anywhere on the chart.

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 the LaTeX length `\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-line are those of its centre point. An *arrow* is described by the coordinates of the centre points of its source and target boxes/minis/text-lines.

10 The prerex Interactive Chart Editor

The source code of prerex, an interactive prerequisite-chart editor, is available at CTAN repositories.³⁷ It should build on any UNIX-like system with suitable tools and libraries installed; for example, it can be compiled on GNU/Linux, Solaris, the Cygwin³⁸ platform on Windows, and the Fink³⁹ platform on OS X, if the GNU `readline` library is installed.

The editor supports add, remove, cut-and-paste, and edit operations on course boxes, minis, text-lines, and arrows, and shifts of specified diagram elements or an entire diagram. A summary of all commands supported by prerex is given in Table 1 on page 13. Note that if a course box/mini/text-line is “cut” and then “pasted” elsewhere, the target or source coordinates of arrows into or out of the source and target are adjusted accordingly, and similarly if nodes are shifted or raised. To allow permutations of two or more chart elements, the editor maintains a “stack” of elements that have been cut but not yet pasted. The effects of editing operations may be observed in a PDF viewer. See the manual page `prerex(1)` for more information.

For initial creation of a prerex-formatted chart or making substantial changes to the boxes and arrows of an existing prerex-formatted chart, it is more convenient to use this program than to edit the source file directly; however, the chart editor has no access to other elements of a source file, such as preamble declarations, headings, notes, logos, and

³⁷<http://www.ctan.org/tex-archive/graphics/prerex/>

³⁸<http://www.cygwin.com/>

³⁹<http://fink.sourceforge.net/>

Command	Effect
file.tex> box x,y	[create and] edit course box at x,y
file.tex> mini x,y	[create and] edit mini course at x,y
file.tex> text x,y	[create and] edit text centered at x,y
file.tex> arrow x0,y0,x1,y1	[create and] edit arrow from $x0,y0$ to $x1,y1$
file.tex> cut x,y	(temporarily) remove box, mini, or text at x,y
file.tex> paste x,y	re-insert removed box, mini, or text at x,y
file.tex> delete x,y	remove box, mini, or text at x,y
file.tex> delete x0,y0,x1,y1	remove arrow from $x0,y0$ to $x1,y1$
file.tex> undelete	undelete deleted box, mini, text, or arrow
file.tex> write !	save to file.tex and process (i.e., make file.pdf)
file.tex> quit exit x ^D	turn off grid, save to file.tex, process, and exit
file.tex> shift [-]x [xi,yi ...]	move specified/all diagram elements x units right [left]
file.tex> raise [-]y [xi,yi ...]	move specified/all diagram elements y units up [down]
file.tex> !cmd	execute shell command cmd, then reload
file.tex> Backup	copy file.tex to .file.tex
file.tex> Restore	restore from .file.tex
file.tex> grid [y/n]	turn on/off coordinate grid background
file.tex> help ?	print this summary

Table 1: prerex Command Summary (version 5.3)

so on. A conventional text editor or LaTeX development environment should be used for these.

11 Source Files

11.1 Overall Structure

A .tex file is essentially text with interspersed LaTeX commands, much as a web-page description has text with Hypertext Mark-up Language (HTML) tags interspersed. Commands for LaTeX can be recognized by the initial backslash character “\”.

The first several lines make up the “preamble” of the source file for a chart:

```
\documentclass{article}
\usepackage{prerex}
:
```

This describes the kind of document to be processed (article, as opposed to book or report) and indicates a file of definitions to be used (prerex.sty). The remaining lines of the preamble specify such things as the page layout and the fonts to be used.

The structure

```
\begin{document}
:
\end{document}
```

contains the content of the document. Typically, the first element of the content is a heading, contained within a formatting structure of the form

```
\begin{center}  
:  
\end{center}
```

which specifies that each line of the material contained should be centered on the page. The double-backslash command `\\` indicates a line break.

After the heading comes the description of the chart itself:

```
\begin{chart}  
:  
\end{chart}
```

Finally, there will typically be some notes to explain the abbreviations and conventions used in the chart.

11.2 Chart Instructions

Table 2 on page 15 summarizes the formats of the instructions defined by `prerex.sty`.

The `grid` command creates a light gray “coordinate grid” in the background of the chart. Figure 3 on page 16 shows the example chart of Figure 1 with the coordinate grid enabled. Note that the “origin” of the coordinate system (the point with x -coordinate 0 and y -coordinate 0) is at the lower-left corner of the chart.

The arrows produced by `prereq`, `coreq`, and `recomm` instructions by default are slightly curved (except very short ones). It is possible to increase or decrease the degree of curvature of a particular arrow by using instructions `prereqc`, `coreqc`, or `recommc`; the additional argument specifies the degree of curvature: 0 means no curvature, i.e., straight, and to increase the curvature, use a larger number (up to 100). To change the default curvature for *all* arrows, see Section 11.4 on page 17 below.

The “in-line” instructions create small examples of course boxes and arrows for use in explanatory notes outside of the chart environment:

```
\solidarrow  →  
\dottedarrow  ····→  
\dashedarrow  - - - →  
\lightbox    light  
\boldbox     bold
```

See the manual pages `prerex(5)` and `prerex.sty(7)` and the documentation for `prerex.sty` for more information.

11.3 LaTeX Input Conventions

The following are some of the input conventions for LaTeX that you might want to know about while editing the source file for a `prerex`-formatted prerequisite chart.

Course Boxes and Text-Lines	
<code>\halfcourse</code>	$x, y: \{course\ code\} \{course\ title\} \{timetable\ code\}$
<code>\fullcourse</code>	$x, y: \{course\ code\} \{course\ title\} \{timetable\ code\}$
<code>\reqhalfcourse</code>	$x, y: \{course\ code\} \{course\ title\} \{timetable\ code\}$
<code>\reqfullcourse</code>	$x, y: \{course\ code\} \{course\ title\} \{timetable\ code\}$
<code>\mini</code>	$x, y : \{course\ code\}$
<code>\text</code>	$x, y : \{line\ of\ text\}$
Arrows	
<code>\prereq</code>	$x_0, y_0, x_1, y_1: \quad \quad \quad \backslashprereqc \ x_0, y_0, x_1, y_1; C:$
<code>\coreq</code>	$x_0, y_0, x_1, y_1: \quad \quad \quad \backslashcoreqc \ x_0, y_0, x_1, y_1; C:$
<code>\recomm</code>	$x_0, y_0, x_1, y_1: \quad \quad \quad \backslashrecommc \ x_0, y_0, x_1, y_1; C:$
Coordinate Grid	
<code>\grid</code>	
In-Line Instructions	
<code>\solidarrow</code>	
<code>\dottedarrow</code>	
<code>\dashedarrow</code>	
<code>\lightbox</code>	
<code>\boldbox</code>	

Table 2: Instructions for prerex-Formatted Prerequisite Charts

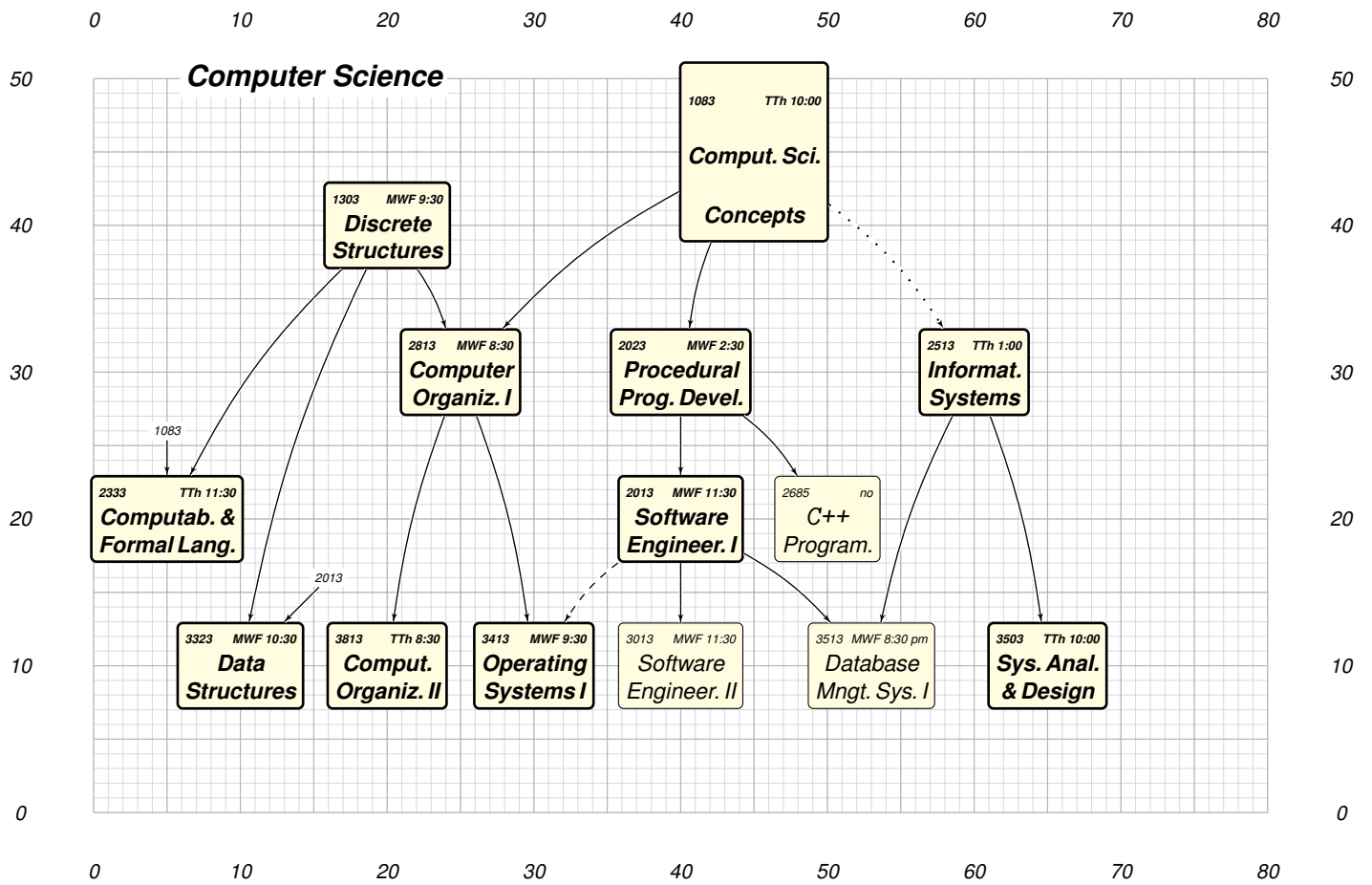


Figure 3: A Prerequisite Chart with a Coordinate Grid

- The characters # \$ % & _ { } ~ ^ \ are interpreted in special ways by LaTeX; to produce any of the first seven of these characters, you will have to “escape” them with a backslash, as in, for example, \&. The \\ combination produces a line break; to produce a backslash, use \backslash. The remaining two special characters, ~ and ^, may also be escaped with a backslash but produce accents; for example, \u produces û.
- To produce a space where LaTeX would normally not do so, use \ . For example, LaTeX normally interprets a period immediately after a word as the end of a sentence and leaves a larger-than-usual space; to prevent the larger-than-usual space after an abbreviation that finishes with a period, use \ instead of a space, as in Diff.\ Equations.
- An unescaped % character signals the beginning of a comment; any characters following the % to the end of that line are ignored by LaTeX.
- To prevent a line break between two words, use a ~ instead of a space between them, as in MATH~223.
- To leave a smaller-than-usual space between two words, use \, instead of a space between them, as in MATH\, 223.
- To prevent hyphenation of some word, use the word as the argument to an \mbox command, as in \mbox{Robert}.

11.4 Configuration

The following LaTeX definitions are made in `prerex.sty` but may be re-defined by the user in the chart source file:

```
\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}{LightYellow}
```

where `LightYellow` is defined by

```
\definecolor{LightYellow}{rgb}{1.0, 1.0, 0.878431}
```

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

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

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

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

These three commands have to re-defined together if the image-map data are to be correct.

```
\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 $\text{\textwidth} \div \text{\unit} + 10$.

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}{0.8pt}
```

```
\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{\smallersize}{\relsize{-3}}
```

Bibliography

The following are three recommended books on LaTeX. Many other books and tutorial materials are available; check out the web sites referenced in Section 4.

- [1] Helmut Kopka and Patrick W. Daly. *Guide to LaTeX*⁴⁰. Addison Wesley Professional, 4th edition, 2003.
- [2] Leslie Lamport. *LaTeX: A Document Preparation System*⁴¹. Addison Wesley Professional, 2nd edition, 1994.
- [3] Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley. *The LaTeX Companion*⁴². Addison Wesley Professional, 2nd edition, 2004.

⁴⁰<http://www.awprofessional.com/titles/0321173856>

⁴¹<http://www.awprofessional.com/titles/0201529831>

⁴²<http://www.awprofessional.com/titles/0201362996>