

Prerex-Formatted Prerequisite Charts

An Introduction

Bob Tennent
School of Computing, Queen's University
rdt@cs.queensu.ca

April 22, 2014

Contents

1	Introduction	2
2	Frequently Asked Questions	2
2.1	Prerequisite Charts	2
2.2	Prerex-Formatted Prerequisite Charts	4
3	Viewing and Printing a Prerequisite Chart	6
4	Installing a Basic LaTeX System	7
5	Additional LaTeX Software	8
6	Installing a LaTeX Development Environment	8
7	Fonts	8
8	Conversion to Scalable Vector Graphics (SVG) Format	10
9	Terminology and Coordinate System	10
10	Chart Editing	11
11	Source Files	12
11.1	Overall Structure	12
11.2	Chart Instructions	15
11.3	LaTeX Input Conventions	15
11.4	Configuration	16
	Bibliography	17

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

2.1 Prerequisite Charts

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 on page 3 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.

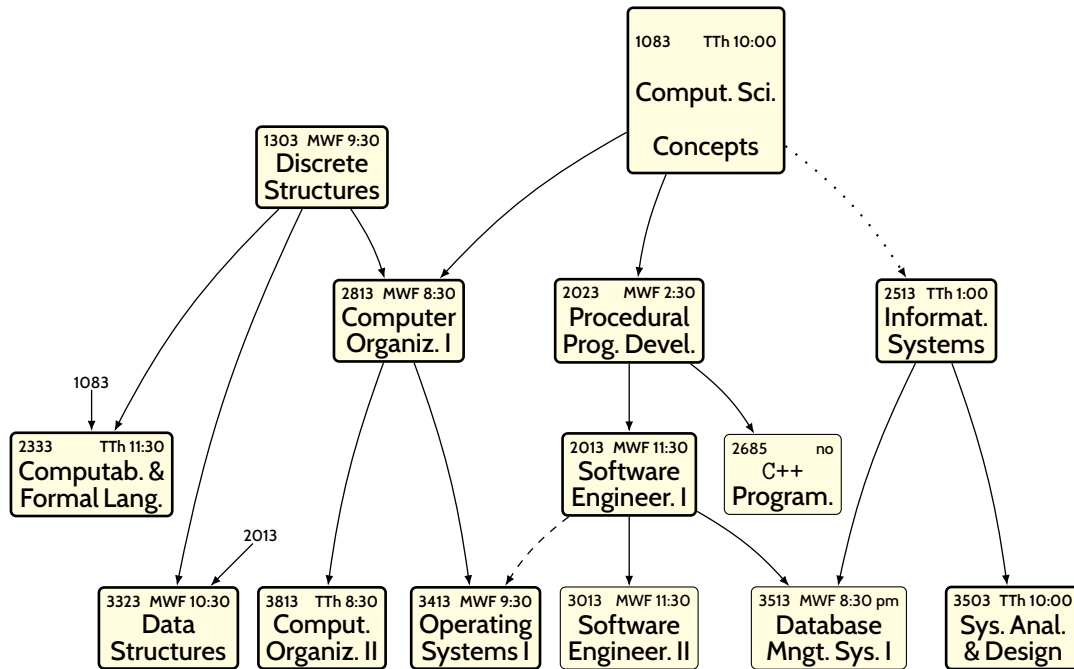
As another example, a course description in a calendar typically specifies what the prerequisite courses are, but hardly ever specifies which course or courses the course is prerequisite to. This information may be very important to a student, but to discover it, he or she must search through all the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

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

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 Prerex-Formatted Prerequisite Chart

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 Waterloo, Computer Science²
- Georgia Tech, Electrical Engineering³
- Georgia State University, Computer Science⁴
- Kansas State University, Mathematics⁵
- Penn State, Computer Science⁶

2.2 Prerex-Formatted Prerequisite Charts

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/prerequisites>.

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.

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.

Charts can be converted to Scalable Vector Graphic (SVG) format for posting on the web. SVG images can be rendered directly by all modern web browsers and are comparable to PDF in quality. When compressed (.svgz), the size is also comparable to PDF. It is fairly easy to convert charts to SVG images.

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.

¹http://www.ece.queensu.ca/Current-Students/Undergraduate/files/Queens_ECE_prerequisite_graph_2011.pdf

²<http://www.cs.uwaterloo.ca/current/courses/charts/majorPrereq.shtml>

³http://www.ece.gatech.edu/academics/undergrad/ee_prereq.html

⁴<http://www.cs.gsu.edu/sites/default/files/prereq.pdf>

⁵<http://www.math.ksu.edu/courses/catalog/prereqcht.pdf>

⁶<http://cs.hbg.psu.edu/cs/csprereq.html>

- 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)⁷ and in many TeX distributions.

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.
- 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. The professional-quality typesetting engine takes advantage of kerns and ligatures in the fonts.

⁷<http://www.ctan.org/>

- 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 and the graphical “wrapper” vprerex are 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.⁸ However, the Adobe Reader

⁸<http://www.adobe.com/prodindex/acrobat/readstep2.html>

is quite slow to load and somewhat inconvenient to use if one wants to re-load the document after changing it (normally 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.⁹ For Windows or Linux: download and install the most recent version of ghostscript for your platform,¹⁰ and then install the most recent version of GSview for your platform.¹¹ For Macs: just get MacGSView.¹²

For Linux systems, other ghostscript-based viewers such as gv¹³ are available. Also, xpdf¹⁴ 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. The vprerex “wrapper” for the prerex editor includes a built-in prerex-enabled PDF viewer that makes it extremely easy to transfer coordinates from the chart to the editor.

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 PDF or other formats.

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 page¹⁵
- TeX Users Group home page¹⁶
- TeX FAQ¹⁷

If you use Windows, we suggest you install the “basic” version of MiKTeX.¹⁸ If you use

⁹<http://www.cs.wisc.edu/~ghost/>

¹⁰<http://www.cs.wisc.edu/~ghost/doc/AFPL/index.htm>

¹¹<http://www.cs.wisc.edu/~ghost/gsview/index.htm>

¹²<http://www.cs.wisc.edu/~ghost/macOS/index.htm>

¹³<http://wino.physik.uni-mainz.de/~plass/gv/>

¹⁴<http://www.foolabs.com/xpdf/home.html>

¹⁵<http://www.latex-project.org>

¹⁶<http://www.tug.org/>

¹⁷<http://www.tex.ac.uk/cgi-bin/texfaq2html>

¹⁸<http://www.miktex.org/setup.html>

Mac OS X, go to the *TeX on Mac OS X* site.¹⁹ UNIX and Linux users should install at least the “basic” scheme of TeXLive.²⁰

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.

5 Additional LaTeX Software

To process descriptions of prerex-formatted charts, some additional packages are needed. The file `prerex.sty` from the prerex package is essential; it should be stored in the same folder as the .tex file to be processed (or in a `texmf` tree so the LaTeX system can find it). Other add-ons that may be necessary are as follows: `pgf`, `tikz`, `multido`, `hyperref`, `textcomp`, and `xcolor`. If you are using MikTeX or TeXLive, these packages can be installed using their package managers. All of the packages are also available for download from any CTAN²¹ 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.

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; one called TeXWorks is supported in both MikTeX and TeXLive.

7 Fonts

Look carefully at the samples in Figure 2 on page 9. Times and Palatino are *serifed* typefaces; the 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.

There are several *sans serif* options available for LaTeX. Computer Modern Sans is the default *sans serif* font, and Helvetica should also be available in every LaTeX installation. To get Helvetica (scaled down 20%), use

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

¹⁹<http://www.rna.nl/tex.html>

²⁰<http://www.tug.org/texlive/>

²¹<http://www.ctan.org/>

Times
Palatino
Computer Modern Sans
Helvetica
Open Sans
Cabin
Quattrocento Sans
Linux Biolinum
Optima

Figure 2: Typeface Samples

The Open Sans family²² of fonts, designed by Steve Matteson of Ascender, is supported for use in LaTeX by the `opensans` package, available from CTAN²³ and in most distributions. The Cabin²⁴ and Quattrocento Sans²⁵ families of fonts, designed by Pablo Impallari, are supported for use in LaTeX by the `cabin` and `quattrocento` packages. The chart in Figure 1 on page 3 uses Cabin. The Linux Biolinum fonts are the sans-serif companions to the Linux Libertine family²⁶, designed by Philipp H. Poll, and are supported for use in LaTeX by the `libertine` package.

Four URW-Classico fonts, a fairly good clone of Hermann Zapf's Optima, are available in a package of 79 fonts in Type 1 format that may be legally downloaded for non-commercial use from Artifex²⁷. The four Classico font files, renamed for use in LaTeX, with the necessary support files and installation instructions are available from CTAN²⁸. Note that the font files cannot be distributed commercially without a commercial license; please read the license in the archive before re-distributing them. This restriction does not prevent you from distributing documents that *use* the fonts.

²²<http://www.google.com/webfonts/specimen/Open+Sans>

²³<http://www.ctan.org/pkg/opensans>

²⁴<http://www.google.com/webfonts/specimen/Cabin>

²⁵<http://www.google.com/webfonts/specimen/Quattrocento+Sans>

²⁶<http://www.linuxlibertine.org/index.php?id=1&L=1>

²⁷<http://www.artifex.com/downloads/>

²⁸<http://www.ctan.org/pkg/classico>

8 Conversion to Scalable Vector Graphics (SVG) Format

Scalable Vector Graphics (SVG) format is an emerging standard for high-quality graphics. Recent versions of popular browsers support SVG graphics natively; i.e., without requiring a plug-in or external application, making it an ideal format for posting charts on the web. The quality and size of (compressed) SVG images are comparable to PDF. The `inkview` application in the open-source `inkscape`²⁹ package is a convenient SVG image viewer.

One approach is to use `dvisvgm`³⁰ to convert the output of `latex` processing directly into a (compressed) SVG image. Versions since 1.3 generate links in the image from `hyperref` links in the source file. Use `dvisvgm` as a global class option and `hypertex` as a package option to `hyperref` to ensure that the appropriate drivers are used by `latex`, and invoke `dvisvgm` as follows:

```
dvisvgm -n -b dvi -L none ... .
```

Add the `-z` option to obtain a compressed image. If `dvisvgm` reports that it can't find Ghostscript, see the `dvisvgm` documentation for configuration options.

On Unix-like platforms such as Linux, a small program called `pdf2svg` is available³¹; it uses standard libraries (`poppler` and `cairo`) to implement the conversion. From version 0.2.3, `pdf2svg` adds clickable links to the image and supports a scaling option.

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 (or paragraph) 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* 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.

²⁹<http://www.inkscape.org/en/>

³⁰<http://dvisvgm.sourceforge.net/>; also available from [CTAN](#) and in [TeXLive](#).

³¹<http://www.cityinthesky.co.uk/opensource/pdf2svg/>

Command	Effect
file> box x,y	[create and] edit course box at x,y
file> mini x,y	[create and] edit mini course at x,y
file> text x,y	[create and] edit text centered at x,y
file> arrow x_0,y_0,x_1,y_1	[create and] edit arrow from x_0,y_0 to x_1,y_1
file> cut x,y ...	(temporarily) remove box, mini, or text at x,y
file> paste x,y	re-insert removed box, mini, or text at x,y
file> xchange $x_0,y_0 x_1,y_1$	exchange elements at x_0,y_0 and x_1,y_1
file> delete [x,y x_0,y_0,x_1,y_1 $x_0,y_0:x_1,y_1$] ...	remove specified elements/arrows
file> shift [-]x $x_0,y_0[:x_1,y_1]$...	move [specified] elements x units right [left]
file> raise [-]y $x_0,y_0[:x_1,y_1]$...	move [specified] elements y units up [down]
file> undo	undo the effect of the previous editing command
file> write !	save to file and process (with pdflatex)
file> quit exit x ^D	turn off grid, save to file, process, and exit
file> !cmd	execute shell command cmd, then reload
file> Backup	copy file to .file
file> Restore	restore from .file
file> grid [y/n]	turn on/off coordinate grid background
file> help ?	print this summary

Table 1: prerex Command Summary (since version 6.4)

10 Chart Editing

The source code of prerex, an interactive (command-line) prerequisite-chart editor, is available at CTAN repositories.³² 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 Cygwin³³ platform on Windows, and the Fink³⁴ platform on OS X.

The editor supports add, remove, cut-and-paste, and edit operations on course boxes, minis, texts, 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. Note that if a course box/mini/text 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 notation $x_0,y_0 : x_1,y_1$ denotes all the nodes (course boxes, minis, texts) in the rectangle whose northwest and southeast corners are at coordinates x_0,y_0 and x_1,y_1 , respectively.

The effects of editing operations may be observed in a PDF viewer. See the manual page prerex(1) for more information. The prerex package at CTAN also has the source files for vprerex (visual prerex), a graphical “wrapper” for prerex which incorporates a prerex-enabled PDF viewer.

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 one of these programs

³²<http://mirror.ctan.org/graphics/prerex/>

³³<http://www.cygwin.com/>

³⁴<http://fink.sourceforge.net/>

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 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.

Course Boxes and Text-Lines	
<code>\halfcourse x,y:{course code}{course title}{timetable code}</code>	
<code>\fullcourse x,y:{course code}{course title}{timetable code}</code>	
<code>\reqhalfcourse x,y:{course code}{course title}{timetable code}</code>	
<code>\opthalfcourse x,y:{course code}{course title}{timetable code}</code>	
<code>\reqfullcourse x,y:{course code}{course title}{timetable code}</code>	
<code>\optfullcourse x,y:{course code}{course title}{timetable code}</code>	
<code>\halfcoursec x,y:{course code}{course title}{timetable code}{color}</code>	
<code>\reqhalfcoursec x,y:{course code}{course title}{timetable code}{color}</code>	
<code>\opthalfcoursec x,y:{course code}{course title}{timetable code}{color}</code>	
<code>\fullcoursec x,y:{course code}{course title}{timetable code}{color}</code>	
<code>\reqfullcoursec x,y:{course code}{course title}{timetable code}{color}</code>	
<code>\optfullcoursec x,y:{course code}{course title}{timetable code}{color}</code>	
<code>\mini x,y:{course code}</code>	
<code>\text x,y:{line of text}</code>	
Arrows	
<code>\prereq x_0,y_0,x_1,y_1:</code>	<code>\prereqc x_0,y_0,x_1,y_1; c:</code>
<code>\coreq x_0,y_0,x_1,y_1:</code>	<code>\coreqc x_0,y_0,x_1,y_1; c:</code>
<code>\recomm x_0,y_0,x_1,y_1:</code>	<code>\recommc x_0,y_0,x_1,y_1; c:</code>
Coordinate Grid	
<code>\grid</code>	
In-Line Instructions	
<code>\solidarrow</code>	
<code>\dottedarrow</code>	
<code>\dashedarrow</code>	
<code>\lightbox</code>	
<code>\boldbox</code>	
<code>\dottedbox</code>	

Table 2: Instructions for prerex-Formatted Prerequisite Charts

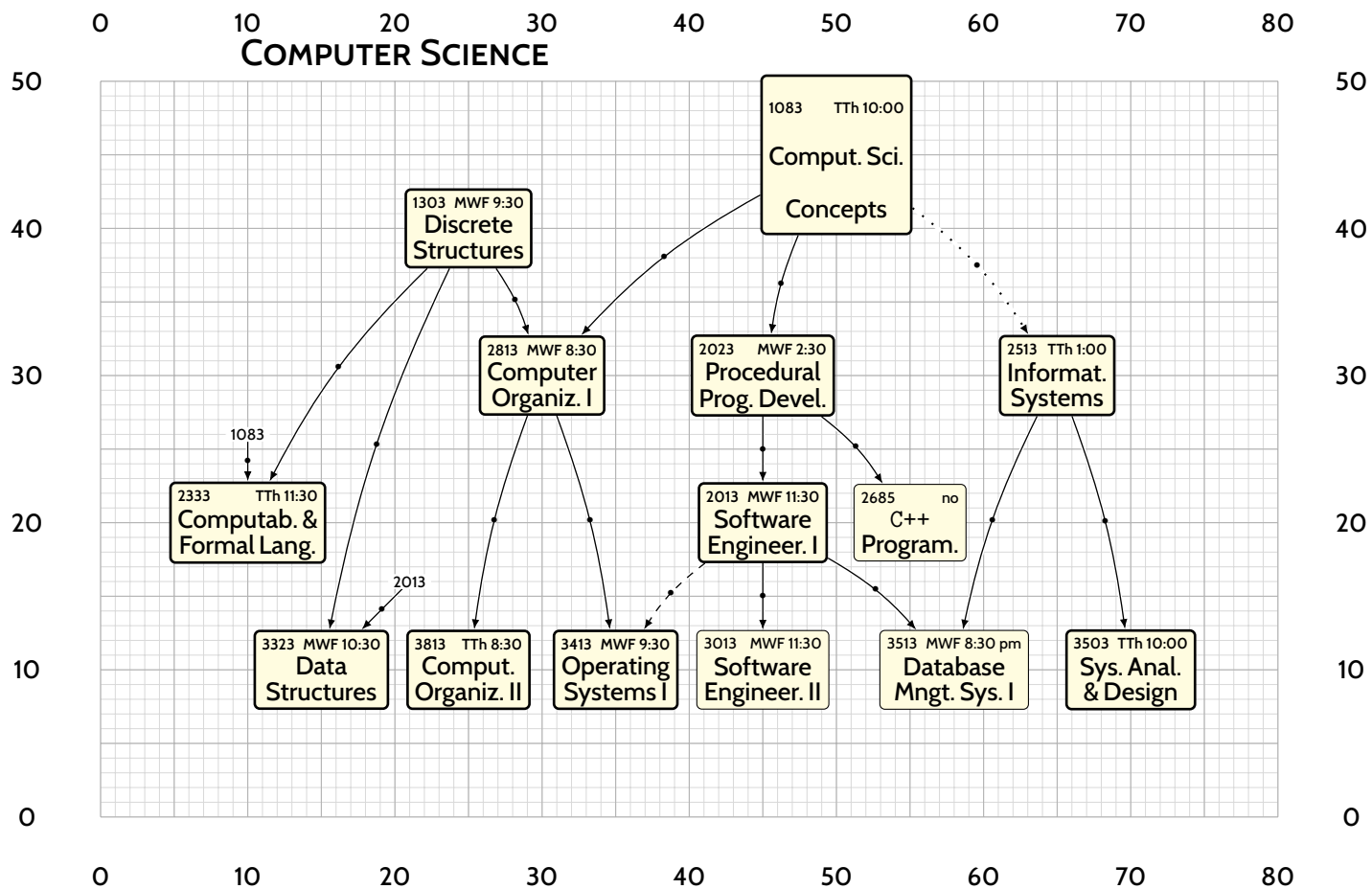


Figure 3: A Prerequisite Chart with a Coordinate Grid







11.2 Chart Instructions

Table 2 on page 13 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 14 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 16 below.

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

<code>\solidarrow</code>	
<code>\dottedarrow</code>	
<code>\dashedarrow</code>	
<code>\lightbox</code>	
<code>\boldbox</code>	
<code>\dottedbox</code>	

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.

- 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 \hat{u} .
- 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}{yellow!15}
```

which defines the default background color for course nodes.

```
\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 be 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 `\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}{0.8pt}
```

```
\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}}
```

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

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


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*³⁵. Addison Wesley Professional, 4th edition, 2003.
- [2] Leslie Lamport. *LaTeX: A Document Preparation System*³⁶. Addison Wesley Professional, 2nd edition, 1994.
- [3] Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, and Chris Rowley. *The LaTeX Companion*³⁷. Addison Wesley Professional, 2nd edition, 2004.

³⁵<http://www.awprofessional.com/titles/0321173856>

³⁶<http://www.awprofessional.com/titles/0201529831>

³⁷<http://www.awprofessional.com/titles/0201362996>