CS 553 Homework Template

Michelle Mills Strout

August 21, 2014

Latex is a type setting package that is extremely popular for writing computer science documents. Latex has the disadvantage that it is NOT WSYWYG, however it enables using style files to meet the formatting requirements for various conferences, enables easier revision control diffs since the source files are text, renders math equations specified with relatively simple text markups, and automates the processing of building a bibliography when used in coordination with latex.

This file provides information about using latex and gnuplot and is itself a sample latex file. The following sections describe how to compile a latex file into pdf, how to do tables in latex, how to create plots in gnuplot, how to include figures into the pdf file, how to write equations, and how to cite other people's work.

1 Using Latex

To compile the cs553-template.tex file into a pdf, you will also need to download the cs553.bib and plot.pdf files provided on the assignments page. To compile this file into a pdf use the following commands:

```
pdflatex cs553-template
bibtex cs553-template
pdflatex cs553-template
pdflatex cs553-template
```

I recommend using a Makefile. Here is the one our group usually uses.

```
LATEX = pdflatex
```

Processor Type	Procs x Cores	First Level Data Cache	Mid Level Cache	Last Level Cache	Memory
Intel Core i7 920	1x4	32 kB	256 kB	8 MB shared/4 cores	6 GB
Intel Xeon E5450	2x4	32 kB	256 kB	6 MB shared/core pair	16 GB
Intel Xeon E7-4860	4x10	32 kB	256 kB	24 MB shared/10 cores	256 GB

Table 1: Hardware Used in the Performance Evaluation

```
BIBTEX = bibtex

DOC = cs553-template

$(DOC): $(DOC).tex
        $(LATEX) $(DOC)
        $(BIBTEX) $(DOC)
        $(LATEX) $(DOC)
        $(LATEX) $(DOC)

clean:
        -rm *.bbl *.aux *.blg *.div *.log *.ps

all: $(DOC)
```

2 Tables in Latex

Table 1 shows an example of how to create a table in latex. You put in horizontal lines with the backslash followed by the word hline. The ampersands indicate breaks between columns within rows. The number of columns is specified with the string "`|l|p{.5in}|p{.5in}|p{.5in}|c|c|`". Notice that before this string in the latex source we used a double backslash to force a line break. The c's indicate center justification, the l indicates leftmost justification, the p with a size indicates a column of the specified width, and the vertical bars indicate a vertical line will separate the columns.

3 Graphs using gnuplot

Generating graphs with gnuplot is useful because you can create a script and reuse the same script for multiple data files.

Create the following data files and command files:

```
# xy.dat
1 10
2 20
3 30
4 50

# commands.txt
 set term post eps color
 set output "plot.eps"
 set size square
 set ylabel "ylabel"
 set xlabel "xlabel"
 set title "title"
 set pointsize 2.5              # increases symbol size
 plot "xy.dat" with linespoints # connects points with lines

# morelines.dat
1 10 12
2 20 15
3 30 25
4 50 55

# commands-morelines.txt
 set term post eps color
 set output "plot.eps"
 set size square
 set ylabel "ylabel"
 set xlabel "xlabel"
 set title "title"
 set pointsize 2.5
 plot "morelines.dat" using 1:2 with linespoints, \
      "morelines.dat" using 1:3 with linespoints
```

Note that the latex source for this file shows how you can have pre-formatted monospace text using the verbatim environment.

When you type

```
gnuplot commands.txt
```

or

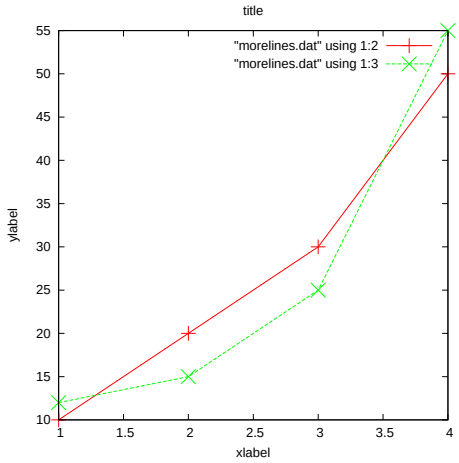

Figure 1: A graph showing how to plot two columns of data that share the same x coordinates.

```
gnuplot commands-morelines.txt
```

gnuplot will generate a plot.eps file.

Then use the command

```
epstopdf plot.eps
```

to generate a plot.pdf file that you will learn how to include in your latex file in the next section of this document.

4 Including Figures

The graphicx package included at the top of the file enables the inclusion of figures like Figure 1, which shows the figure we created with commands-morelines.txt.

5 Equations in latex

Equations are one of the main reasons to use latex. Here are some equations that you will find handy in this class.

$$P = \{(i_1, i_2, ..., i_d) \mid Q\vec{i} \geq (\vec{q} + B\vec{p})\}$$

$$A_{J_0 \to Z_0} = \{[j] \to [i] | i = l(j) \vee i = r(j)\}$$

$$\begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} i \\ j \end{bmatrix} \geq \begin{bmatrix} 1 \\ -6 \\ 1 \\ -5 \end{bmatrix}$$

$$r'() = \{[v] \to [w] \mid w = \sigma(r(v)) \wedge 0 \leq v, w \leq 7\}$$

$$\Sigma_{i=1}^{n} i^2$$

You can also have inline equations such as $x_2 = 5 + \gamma$ and summations such as $\Sigma_{i=1}^{n} i^2$.

6 Code Listings in latex

We are computer scientists so we usually write about code. Figure 2 has an example that Chris Krieger wrote for a paper we recently submitted. The code listing uses the listings package included above and the lstset command also at the top of this file.

7 Lists in latex

Sometimes it is

```
// constructor & destructor
TaskGraph();
virtual ~TaskGraph();

// building a TaskGraph
int addTask(Task* t);
int addPred(int nodeID, int predID);
int addSucc(int nodeID, int succID);

// using an existing TaskGraph
int numNodes();
int numLeafNodes();

Task* getTask(int index);

int numPreds(int nodeID);
int numSuccs(int nodeID);
const vector<int>& getPreds(int nodeID);
const vector<int>& getSuccs(int nodeID);
```

Figure 2: Public interface to the TaskGraph class (some utility methods omitted)

- useful
- to
- have
- a
- bulleted list

or a

1. numbered
2. list
3. in
4. latex.

We can also cite papers such as the "Twelve Ways to Fool the Masses" paper [1]. Other examples include a technical report [3] and website [2]. You should start using either Bibdesk

or JabRef to manage your bibtex references. Start putting all of the bibtex entries into Bibdesk or JabRef. There will be an assignment to turn in your bibtex entries for all of the reading assignments in the class so far. You can often find bibtex entries on the internet at places like ACM portal and Citeseer.

References

[1] D. H. Bailey. Twelve ways to fool the masses when giving performance results on parallel computers. *Supercomputing Review*, pages 54–55, August 1991.

[2] A. Stone, J. Dennis, and M. M. Strout. The cgpop miniapp. http://www.cs.colostate.edu/hpc/cgpop/, June 30 2011.

[3] A. Stone, J. Dennis, and M. M. Strout. The cgpop miniapp, version 1.0. Technical Report Technical Report CS-11-103, Colorado State University, July 1 2011.