CS150 - Lab Prep 11
Due: Friday May 2, at the beginning of class

For our last (yay!) lab prep we're going to be revisiting our data that we collected from lab 9. If you'd like to work with a pair on the lab and the lab prep, you're welcome (and encouraged) to do so.

1 The data

This week, we're going to be extending our weather aggregator program from Lab 9. Recall that the end result of the program was a file of the form:

```
<date>  <hour>  <temp>
<date>  <hour>  <temp>
...
```

where each entry is separated by a tab. For example, if we collected data for a few hours we might get:

```
11-9-2011 14 65
11-9-2011 15 66
11-9-2011 16 70
```

```
11-9-2011 17 68
11-9-2011 18 59
```

If you setup this program to run every hour for a particular zip code, you would eventually end up with a file spanning multiple days, with one entry per hour containing the temperature for that date/hour. I've posted two examples of files like this online for you to use for the lab at:

http://www.cs.middlebury.edu/~dkauchak/classes/cs150/assignments/assign11/

At this location there are two raw temperature files in the above format:

- `test.raw.txt`: A short file with just a few days of data that will be useful for checking your answers on.

- `wisc.raw.txt`: Real weather data aggregated for roughly 1 year from Wisconsin Rapids, WI obtained from:

    http://www.soils.wisc.edu/asigServlets/asos/SelectHourlyAsos.jsp

  (with a bit of post-processing by myself)

Take a quick look at them just to make sure you understand what is in these files.

2 Matlab friendly data

Matlab is not very good at processing text files, so often when working with real data you will manipulate the data with another language (e.g. Python) to get it into a format that's easier for Matlab to handle.

The goal of this week's lab will be to plot the temperature data as well as generate some statistics characterizing the weather. Before we can do this, though, we need to convert our data into something more Matlab-friendly.

I created a Python program that reads through the data in a `raw` file output from our Lab 9 data and creates a file where each line in the file represents a days worth of data.

For example, for `test.raw.txt` the output would be:

```
40 37 35 35 34 36 37 42 45 47 50 52 53 53 52 54 51 50 45 43 41 37 36 35
33 30 29 28 30 30 32 40 41 42 43 43 43 44 45 45 44 42 40 39 40 40 41 41
41 42 42 44 45 50 51 55 65 72 80 86 93 101 102 100 95 80 77 70 67 63 60 60
```

There are three days worth of data. Each line/row is a day and each column is an hour (0-23).

For both of the `raw` files I have included a corresponding `matrix` file that is much easier for Matlab to read.

In the `wisc.matrix.txt` file, what does the 7th entry on the 5th line represent? Write your answer on a piece of paper and bring it to class on Friday. Be specific!.

3 Due diligence

I've posted the lab for this week. Read through the entire handout (yep, starting at the beginning comic all the way to that lovely grading table at the end). This will make sure your time gets best utilized for the lab on Friday.

If you haven't done so yet, make sure you've worked through the examples in section 5 of the matlab basics handout that we looked at on Monday. These examples will be similar to the types of things you'll do for the lab.

4 *(Optional)* The conversion

If you want more practice with Python (and in particular Python data processing and file manipulation) you can try and write a program that does the data conversion described above. The program should read the data from a `raw` file and then **write** a file in the `matrix` format with one day per line.

If you get stuck (or if you're just curious), I've put a copy of a program called `process_weather.py` online in the same directory as the data files that does this conversion.