Assignment 6

This NASA post, NASA Data Link Pollution to Rainy Summer Days in the Southeast, claims that it rains more in the southeastern US Tuesday through Thursday than it does Saturday through Monday. The presence of a seven-day cycle in the weather is "errie" evidence that human activity influences the weather.

Your mission in this assignment is to see if you can validate their claim using data from the instruments at RDU airport. Of course, the NASA researchers had access to much richer data, so we are not really equipped to confirm or refute their claim but we're in the southeast, and have some data, let's see what we can do.

I have collected 10 years of data into a single file, krdu-rain-2001-2010.csv, in a format suitable for use with `numpy.loadtxt()`. The data have 4 columns; year, month, day, and rainfall in inches.

There is no template for this assignment. You'll need to

1. Read the data.
2. Determine the days of the week from the dates.
3. Write a function to get the average daily rainfall during midweek (Tuesday through Thursday) and weekend (Saturday through Monday). Don't print them.
4. Report the mean rainfall respectively for all midweek days and for all weekend days, and their difference Δ= mean rainfall in midweek – mean rainfall in weekend.
5. Determine and report the p value (the likelihood that the effect is not real) by simulation. You'll need to first, compute the Δ in 4, which in our case is the difference between the means for midweek and weekend. Then you'll run the function many times, each time permuting the day labels (this is equivalent to permuting year, month, and date all together), and counting the number of times that the difference between the new means is greater than Δ. Now divide that count by the number of trials you ran. That will be the p value. If count is 0, that means you didn't find even a single permutation that produced a greater difference in means; the effect is very likely real. On the other hand, if count is huge, then there were many permutations that produced greater differences, so the difference is likely just random.
6. **(!!Bonus)** Starting from the first Tuesday in the data, save the rainfall means in the midweek (Tuesday to Thursday) in a 1D array. Then determine a list of the longest **increasing** subsequence in that array and print out the list. The list you print will show the longest increase of the midweek rainfall means in the data. Note: The increase does not need to be consecutive. **Hint**: dynamic programming would be helpful (make base condition, assume the n-1 case, take care of the transition from n-1 to n). **More direct hint**: how to connect this problem with the LCS problem.

Hints

1. You'll find these posts by Allen Downey very illuminating: "There is only one test!" and "More hypotheses, less trivia" (especially the paragraph **Permutation** under **Difference in means**). You shouldn't use his code; write your own.
2. The `weekday` method of the `date` class from the `datetime` module will be useful for getting the day of the week from the date. You can use it like this.

```
from datetime import date

# later in your code when you want to determine the day
# of the week
# we create a date object
do = date(year, month, day)
# and use its weekday method
day = do.weekday()

# or do it all in one step
day = date(year, month, day).weekday()
```

3. The date class knows nothing about numpy so you'll need to use a loop to process all the data.
4. The `np.random.shuffle` function will be useful for permuting the data.
5. You'll need to use a loop to run your simulation many times. I found that 1000 trials gives a fairly stable result (0.04 to 0.06) and doesn't require long to run (good for debugging). On my laptop 100,000 trials took only a couple of minutes.
6. I confirmed my intuition about how this should work by tweaking the data. For example, if I add 0.1 inch of rain to every Tuesday, the p value drops to zero but if I replace the rainfall with random numbers it rises to near 0.5.
7. I get about 0.02 inches of difference in rainfall between midweek and weekend.

Discussion is encouraged. However, what you hand in must be your own work. If you collaborate on the solution, please list your collaborators at the top of the code. **Copying or sharing code is prohibited.**