

# 121COM: Introduction to Computing

Academic Year 2015/16

## LabSheet 6

For use in labs the week beginning Mon 2nd November 2015

Author: Dr Matthew England (Coventry University)



### Lab Exercises 1 - Using Dictionaries and Files

- Q1 Find the string method to split a string into words. Create a function which takes a string and outputs a dictionary in which the words appearing are the key and the number of times they occur the values.
- Q2 Write a function `codeOnly` which takes a file name and prints the file contents excluding any lines which start with the hash symbol (`#`): the comment character of Python. Test this on your Python scripts.
- Q3 Write a function to compare two text files. If they are different, give the line and column numbers in the files where the first difference occurs.
- Q4 Revisit your code from Worksheet 5 Q1 to create Fibonacci Numbers. Write an alternative function which uses a dictionary to remember previously computed Fibonacci numbers (Hint: make the dictionary a second argument of the function).

Compare the two functions - what are the largest  $F_n$  they can compute easily (i.e. in a few seconds)?



**Optional:** Repeat this for the Ackermann Function (Week 5 Slides A).



### Lab Exercises 2 - Word Fun

These exercises use the file `words.txt` available on Moodle. It contains 113809 English words, one per line.

- Q5 Create a list of all the words in the file. You will need this for the other questions so pickle it!
- Q6 Write a function `isThisAWord(S)` which takes a string `S` and returns the appropriate Boolean.
- Q7 An anagram of a word is another formed from the same letters. For example, `parse` and `spare` are anagrams. Create a dictionary where the keys are tuples of letters and the values anagrams they create. In Scrabble a player receives a bonus if they play all 7 of their tiles at once, creating an 8 letter word (combining with one tile on the board). Which group of 8 letters can create the most words?



**Extension:** Which group of 7 letters offers the best chance of a bonus? Assume all letters equally likely to be on the board; or if you like, look up their frequencies in a Scrabble set!

Q8 Two words form a **metathesis pair** if you can transform one into the other by swapping two letters; for example, *converse* and *conserve*. Write a program that finds all of the metathesis pairs in the file.



**Extension:** Do this without testing all pairs and without trying all swaps.



## Extended Task

This task concerns UK Crime Statistics as provided by the API at: <https://data.police.uk/docs/>. We can make a request for data to an API by modifying a url call. For example, the following url asks for all reported crimes that occurred in February 2012 at the location with latitude 52.629729 and longitude -1.131592 (which is the centre of Leicester):

```
https://data.police.uk/api/crimes-at-location?date=2012-02&lat=52.629729&lng=-1.131592
```

Your task is to build functions which make similar calls. Initial tasks:

- Write a function which takes a date, latitude and longitude and outputs the url to call as a string.
- Write another function which takes a date, latitude and longitude, forms the url, opens the url (which provides data in json format) and converts the data to a Python dictionary. Look at the functions provided for the Week 3 Extended Task for help here.
- Write a function to analyse the data and print out how many crimes there were of each type.
- Generate data for a number of different locations and store them in a file so that the data can be retrieved without making new API calls.

Some other latitudes and longitudes to practice with are: Coventry Cathedral (52.408166, -1.508528); Earlsdon (52.398413, -1.530350); University of Birmingham (52.450817, -1.930513). You can generate more coordinates from a map or address at <http://www.latlong.net/convert-address-to-lat-long.html>



### Extensions:

- Most people in the UK do not know the latitude and longitude of their location, but do know the postcode. A different API can be called to find the latitude and longitude for any UK postcode using calls like the following:

```
http://api.postcodes.io/postcodes/CV15FB
```

Write a function which takes a postcode and outputs latitude and longitude coordinates using this API. Write another which accepts postcodes and produces crime statistics.

- The call above gives crimes as a specific coordinate. There is an alternative API call which provides all crimes in a square determined by coordinates:

```
https://data.police.uk/api/crimes-street/all-crime?poly=COORDS&date=2012-12
```

where COORDS is `tLat,lLon:tLat,rLon:bLat,rLon,bLat,lLon` and `tLat` is the top latitude coordinate, `bLon` the bottom Longitude, `lLat` the left Longitude and `rLon` the right Longitude etc. Write a new function which generates data from such coordinates. Write another which given coordinates for a central point generates the coordinates for a square 1 miles long on each side. Use the approximation that 1 degree latitude is 70 miles and 1 degree longitude is  $70 * \cos(\text{Central Latitude})$  miles.

- Use the module `matplotlib` to visualise some of the data.
- Read the documentation on the crime data API and experiment with other calls.