ECMM455 Python Worksheet 15: Putting it all together

Prof Hywel Williams

September 12, 2019

1 Aims

- Practice solving problems using all of the Python skills you have learned

2 Recap: What have we learned?

You have covered a lot of Python very quickly. There are still many parts of the language that we have not looked at in detail. Some of these will turn up later in the course, others you will learn yourself when needed, outside of this course. Very few people ever completely master a language - most learn the bits they need and pick up additional knowledge through experience. Learning about new functions and methods to solve a particular problem is part of the programming process.

So far we have covered:

- Syntax
- Variables
- String handling
- Lists
- Algorithms & control
- Loops (for and while)
- Logical operators and expressions
- Conditional statements: if, if-else, if-elif, if-elif-else, while.
- File input/output
- Functions
- Modules

This worksheet contains a set of programming problems, all of which can be solved using only the skills you have learned so far. There are many right answers to each problem. The challenge is to find a solution which is effective and follows good programming principles; look for simple, clear code which does the job.

3 Consolidation exercises

These exercises are adapted from those by Torbjorn Lager (http://www.ling.gu.se/~lager/python_exercises.html).

1. "99 Bottles of Beer" is a traditional song in the United States and Canada. It is popular to sing on long road trips, as it has a very repetitive format which is easy to memorize, and can take a long time to sing. The song's simple lyrics are as follows:

   > 99 bottles of beer on the wall, 99 bottles of beer.
   > Take one down, pass it around, 98 bottles of beer on the wall.

   The same verse is repeated, each time with one fewer bottle. The song is completed when the singer or singers reach zero. Your task here is write a Python program to generate and display all the verses of the song. The program should also write all the verses to a text file called "beer.txt".

2. Write a program that accepts a file name from the user, builds a frequency listing of the characters contained in the file, and prints a sorted and nicely formatted character frequency table to the screen. That is, if the file contains the text:

   > The cat sat on the mat.

   then the program should display something like:

CHARACTER	FREQUENCY
	5
.	1
T	1
a	3
c	1
e	2
h	2
m	1
n	1
o	1
s	1
t	4

   Create a small plain text file in to debug and test your code. Then count the characters in *war_and_peace.txt* (on ELE). Which character is most common? Which character is least common?

3. Write a program able to play the "Guess the number" game, where the number to be guessed is randomly chosen between 1 and 20. This is how it should work:

```
ECMM429$ python guess_number.py

    Hello! What is your name?
    Ptolemy
    Well, Ptolemy, I am thinking of a number between 1 and 20. Take a guess.
    10
    Your guess is too low. Take a guess.
    15
    Your guess is too low. Take a guess.
    18
    Good job, Ptolemy! You guessed my number in 3 guesses!

ECMM429$
```